

## BAB II

### TINJAUAN PUSTAKA

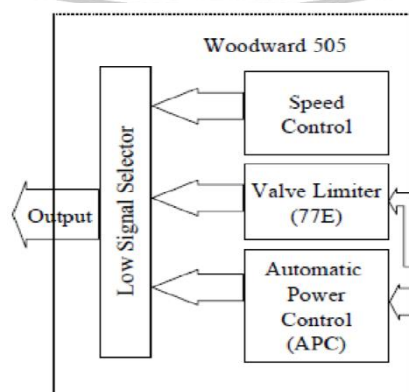
#### 2.1 Governor

*Governor* adalah mekanisme mekanis yang mengontrol berapa banyak cairan, seperti gas, uap, atau cairan, yang masuk ke turbin atau ruang bakar untuk mengontrol putaran mesin (turbin, mesin diesel).

Intinya, *governor* bekerja secara langsung dengan hanya mengandalkan kecepatan putaran mesin itu sendiri. Sebuah poros yang berputar dipasang pada sebuah *governor*. Poros tersebut dilekatkan pada sepasang bandul, yang berputar berbarengan dengan poros tersebut. Gaya sentrifugal yang diinduksi oleh putaran menyebabkan bandul terlempar. Kerah pada poros, tempat pendulum dipasang, naik turun dengan pendulum sesuai dengan pergerakan pusat gravitasi bandul. Saat mengoperasikan atau menyetel tuas bahan bakar (pada mesin diesel) atau aliran fluida, collar digerakkan (pada turbin gas atau uap).

Contoh governor yang digunakan untuk mengatur tekanan keturbin seperti yang digunakan di PLTP Wayang Windu Nusantara yang dilengkapi dengan Woodward 505 Governor, yang berfungsi sebagai pengontrol turbin dan pelindung keselamatan untuk turbin, yang dikendalikan oleh mikrokontroler 32-bit yang telah dikonfigurasi.

Dengan menggunakan tiga jenis sinyal yang masuk yaitu *speed*, *load*, dan *valve limiter*, perintah *governor* diputuskan. *Low Signal Selector* (LSS), yang menerima tiga sinyal, menggunakannya untuk memutuskan instruksi apa yang akan dikirim ke control valve. Ketiga sinyal tersebut dipilih dengan menggunakan LSS karena governor hanya dapat mengeluarkan satu macam instruksi. Seleksi sinyal adalah proses mengidentifikasi sinyal terendah yang mencapai *governor* dan memberi tahu *actuator* (*control valve*) bagaimana merespons.



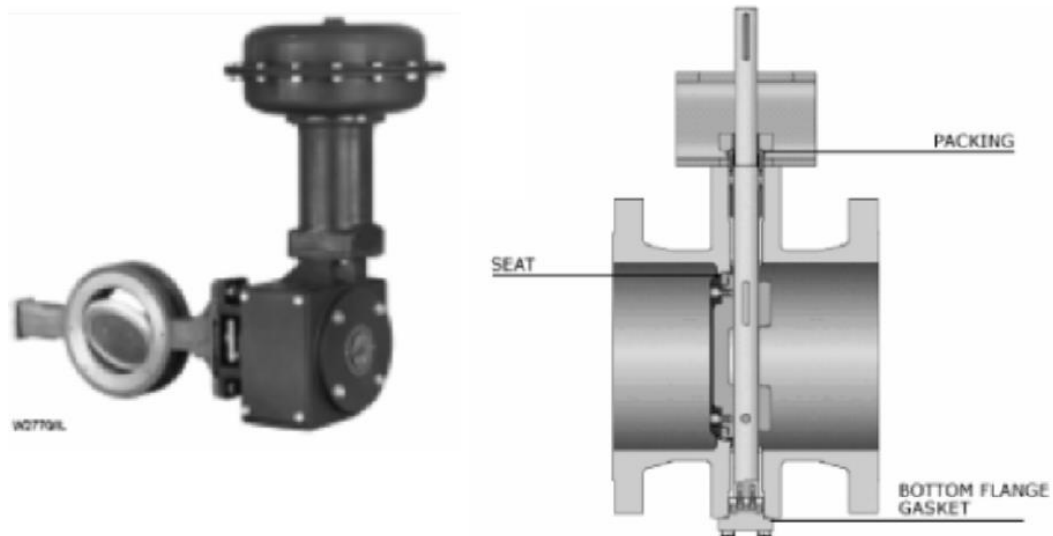
Gambar 2.1 Diagram Blok Governor Woodward 505

## 2.2 Control Valve

*Control valve* menggunakan listrik untuk mengubah laju aliran fluida atau tekanan dalam sistem proses.

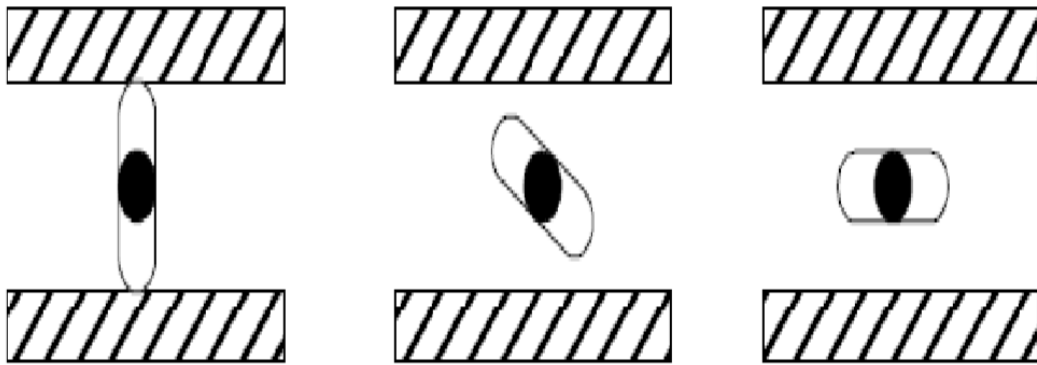
*Control valve* pada sistem PLTP digunakan untuk mengubah jumlah uap di dalam pipa sebelum mencapai turbin uap. *Governor* mengeluarkan perintah ke *Control valve*, yang menggunakannya untuk menentukan ukuran bukaan katup. Untuk mengatur dan mempertahankan kecepatan putaran turbin, *governor* membaca parameter yang menentukan kapan *Control valve* harus dibuka. Daya AC generator dipertahankan pada frekuensi tertentu, atau 50Hz, dengan mempertahankan putaran turbin pada nilai *setpoint* yang diperlukan (3000rpm).

Sebagai contoh PLTP Wayang Windu Unit I menggunakan *control valve* dengan jenis *Butterfly Valve*. *Butterfly Valves* terbuat dari cakram yang berputar di jalur aliran yang berguna mengontrol laju aliran. Sumbu pipa berfungsi sebagai pusat dari sumbu tersebut. Bagian saluran yang ramping dinaikkan oleh cakram. Lubang-lubang ini mengurangi gesekan dan keausan. Karena torsi diperlukan untuk keluar guna menarik katup keluar dari dudukannya saat dalam posisi tertutup, mengendalikan katup bisa jadi sulit. Fungsinya mirip dengan sayap kupu-kupu di mana cakram yang berputar digunakan untuk mengubah hambatan aliran.



Gambar 2.2 Butterfly Valve

*Butterfly valve* jenis ini memiliki beberapa keunggulan, antara lain kemampuannya untuk digunakan dengan slurry, kapasitas besar, dan kehilangan tekanan yang rendah. Torsi yang cukup besar memiliki kelemahan hanya berdampak pada aliran dalam kisaran terbatas (0–60%), yang mengharuskan penggunaan *seat* tertentu untuk menutup rapat.

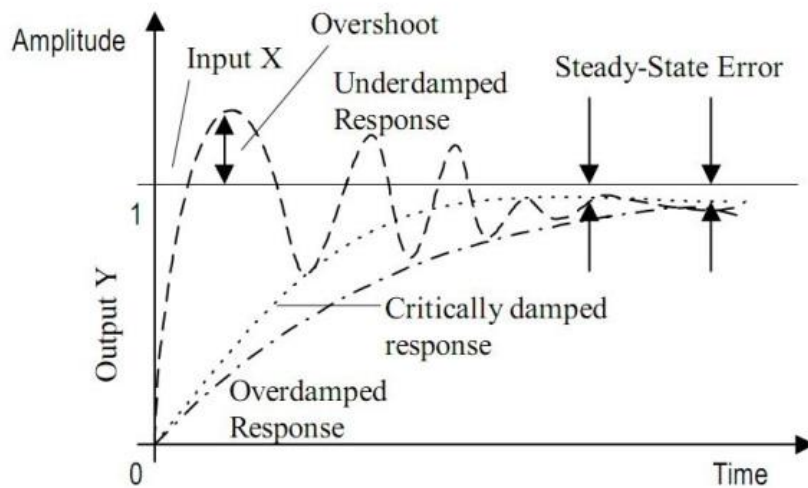


Gambar 2.3 Penampang Butterfly Valve Tampak Atas

### 2.3 Sistem Kontrol PID (*Proportional-Integral-Derrivative*)

Diperlukan pemahaman tentang karakteristik sistem untuk membangun dan menganalisis sistem kontrol. Simulasi sistem yang sedang berjalan memungkinkan dilakukannya analisis terhadap desain kendali yang telah dibuat.

Dengan mengamati bagaimana sistem bereaksi terhadap sinyal uji (sinyal input), kinerja sistem kontrol dapat ditentukan. Sinyal masukan datang dalam berbagai bentuk, termasuk sinyal *impuls*, *step*, *ramp*, *parabolic*, dan *sinusoidal*. Namun, *step*, *ramp*, dan sinyal *parabolic* adalah yang paling sering diuji dalam simulasi.



Gambar 2.4 Jenis – Jenis Respon Sistem

Ketika reaksi sistem mendekati nilai set-point secara bertahap dari waktu ke waktu, sistem dapat dikatakan stabil. Dalam kasus respon *underdamped*, respon melonjak di atas nilai set-point, berosilasi, dan akhirnya menetap ke dalam keadaan *steady state*. Meskipun reaksi *overdamped* tidak pernah melebihi set-point, dibutuhkan waktu lebih lama untuk mencapai kondisi *steady state*.

Sebaliknya, reaksi *critically damped* tidak pernah melebihi set-point dan dapat dengan cepat mencapai kondisi steady state.

Overshoot, peak time, rise time, error steady state, settling time, dan delay time adalah beberapa parameter yang muncul selama perancangan dan pengujian sistem kontrol.

### 2.3.1 Karakteristik respon waktu (Time Response)

Ciri-ciri respons yang persyaratannya didasarkan pada pengamatan cara *output system* berubah seiring berjalannya waktu. Penjelasan berikut memberikan gambaran umum tentang spesifikasi kinerja waktu reaksi:

- Spesifikasi respon transient

Segera setelah ada perubahan pada sinyal input, gangguan, atau beban, spesifikasi respons sistem yang diamati dimulai, dan berlanjut hingga respons mencapai kondisi steady state. rise time, delay time, peak time, settling time, dan overshoot digunakan untuk mengukur responsivitas reaksi transien ini.

- Spesifikasi respon steady state

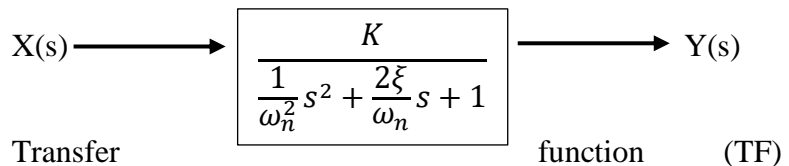
Ketika respons mencapai keadaan steady state, spesifikasi respons sistem yang diamati dimulai dan berlangsung tanpa batas waktu (pada praktik periode observasi begitu  $T_s \leq t \leq 5T_s$ ). Tolok ukur error steady state seperti untuk respons posisi, error kecepatan dan error percepatan digunakan untuk mengukur seberapa baik respons kondisi steady state ini bekerja.

### 2.3.2 Karakteristik respon step (step respon)

Sifat sistem yang dihasilkan dari menentukan respon output ke masukan step.

- Respon system orde satu

Sebuah system orde satu dapat digambarkan sebagai berikut:



system dapat dituliskan sebagai berikut:

$$\frac{Y(s)}{X(s)} = \frac{K}{\frac{1}{\omega_n^2} s^2 + \frac{2\xi}{\omega_n} s + 1} \quad (2.1)$$

- Spesifikasi respon transient sistem orde dua

Terdapat beberapa macam ukuran kualitas respon transient yg telah banyak digunakan antara lain;

1. Maximum Overshoot ( $M_p$ )

adalah nilai puncak kurva respon diukur dari satuan. Apabila nilai akhir keadaan tunak responnya jauh dari satu, maka biasa digunakan persen overshoot maksimum, dan didefinisikan oleh

$$\text{Maksimum (Persen) Overshoot} = \frac{e(t_p) - e(\infty)}{e(\infty)}, \quad (2.2)$$

atau

Nilai relatif yang menyatakan perbandingan harga maksimum respon yang melampaui harga steady state disbanding dengan nilai steady state.

$$\%M_p = \exp\left(-\frac{\pi\xi}{\sqrt{1-\xi^2}}\right) \quad (2.3)$$

2. Time Peak ( $T_p$ )

waktu puncak adalah waktu yang diperlukan respon untuk mencapai puncak pertama overshoot. Kriteria 2 % atau 5 %.

$$T_p = \frac{\pi}{\omega_n \sqrt{1-\xi^2}} \quad (2.4)$$

3. Rise Time ( $T_R$ )

waktu naik adalah waktu yang dibutuhkan oleh respon untuk naik dari 5% ke 95% atau 10% ke 90% dari nilai steady state.

$$T_R = \frac{1}{\omega_n \sqrt{1-\xi^2}} \left( \pi - \tan^{-1} \frac{\sqrt{1-\xi^2}}{\xi} \right) \quad (2.5)$$

4. Time Constan ( $\tau$ )

Ukuran waktu yang menyatakan kecepatan respon, yang diukur mulai t=0 s/d respon mencapai 63,2% ( $e^{-1} \times 100\%$ ) dari respon steady state.

$$\tau = \frac{1}{\xi \omega_n} \quad (2.6)$$

5. Settling Time ( $T_s$ )

Ukuran waktu yang menyatakan respon telah masuk  $\pm 5\%$  atau  $\pm 2\%$  atau  $\pm 0,5\%$  dari respon steady state.

$$T_s(5\%) = \frac{3}{\xi \omega_n} \quad (2.7)$$

atau

$$T_s(\pm 2\%) = \frac{4}{\xi \omega_n} \quad (2.8)$$

atau

$$T_s(\pm 0,5\%) = \frac{5}{\xi \omega_n} \quad (2.9)$$

#### 6. Delay Time ( $T_D$ )

Ukuran waktu yang menyatakan faktor keterlambatan respon *output* terhadap *input*, diukur mulai  $t=0$  s/d respon mencapai 50% dari respon steady state.

$$T_D = \frac{0,742}{\xi \omega_n} \quad (2.10)$$

- Spesifikasi respon steady state system orde Satu  
Speasifikasi respon steady state diukur melalui %error posisi pada keadaan konstan :

$$\% \varepsilon = \frac{x_{ss} - Y_{ss}}{x_{ss}} \times 100\% \quad (2.11)$$

atau

$$\% \varepsilon = (1 - K) \times 100\% \quad (2.12)$$

Dalam sistem kontrol proses industri, kontroler *digital* PID (*Proportional Integral Differential*) adalah jenis kontroler yang umum. Meski memiliki bentuk yang sederhana, *controller* ini memiliki banyak kegunaan. Kontroler PI (*proportional integral*) dan PD (*proportional differential*) dapat digunakan untuk menggantikan kontroler ini. Interval waktu yang disebut waktu sampling digunakan oleh pengontrol digital untuk mengukur variabel proses yang harus dikendalikan.

Salah satu jenis kontroler yang paling populer digunakan saat ini adalah kontroler PID (*Proportional Integral Derivative*). Komponen Proporsional (P), komponen Integral (I), dan komponen Derivatif (D), membentuk pengontrol PID. Kelemahan masing-masing dari ketiga komponen ini dapat dikurangi dengan kekuatan yang lain. Komponen I dan D harus selalu dikopel dengan komponen P untuk membentuk kontroler PI atau PID karena tidak bisa digunakan secara terpisah. Dengan membandingkan kesalahan atau error yang merupakan selisih antara variabel proses dengan setpoint yang akan digunakan sebagai masukan kontroler untuk memberikan sinyal kontrol ( $u(t)$ ), maka kontroler PID akan mengeluarkan aksi kontrol.

Biasanya persamaan kontroler PID dapat dinyatakan dalam bentuk persamaan.

$$mv(t) = K_p(e(t) + \frac{1}{T_i} \int e(t)dt + T_d \frac{\partial e(t)}{\partial t}) \quad (2.13)$$

mv(t)	=	manipulated	variable
Kp	=	penguatan	proporsional
Ti	=	waktu	integral
Td	=	waktu	derivatif
e(t) = error = setpoint – keluaran			

## 2.4 Fuzzy Logic

Logika *Fuzzy* dapat menangani sebuah sistem non-linier yang membutuhkan sedikit ketentuan mengenai input dan deskripsi matematis dari sistem. Logika *fuzzy* dapat membaca perilaku sistem dalam bentuk logika akal sehat dan dinyatakan dalam bentuk serangkaian aturan. Aturan-aturan ini diharapkan dapat memastikan bahwa status pengoperasian perangkat tetap dalam kisaran operasi yang diinginkan. Turbin angin dapat dikategorikan sebagai sistem tersebut, menjadi sangat tidak pasti dan kurangnya penyediaan sistem yang tepat. Fitur pada logika *fuzzy* telah dieksploitasi oleh beberapa penelitian untuk menyelesaikan berbagai permasalahan di dalam turbin angin.

Logika *fuzzy* dapat mengontrol output daya pada keadaan kecepatan angin melebihi kemampuan turbin angin dan meredam fluktuasi daya dibawah kecepatan *angin* yang diinginkan. Kontrol *fuzzy* diimplementasikan untuk mengoptimalkan penerimaan daya pada kecepatan angin yang rendah dan batas kecepatan *angin* yang paling tinggi, sehingga berakibat muncul perluasan penyempurnaan ke ketiga kategori kecepatan angin.. Logika *fuzzy* memiliki *fuzzy set* yang meliputi *inclusion, union, intersection, complement, relation* dan *convexity* (Hamid Mughal, 2015).

### 2.4.1 Himpunan *Fuzzy*

Ide dari himpunan *fuzzy* adalah perluasan dari konsep matematika himpunan. Ada beberapa nilai dalam himpunan *fuzzy*. Masing-masing memiliki derajat keanggotaan (*membership*) mulai dari 0 hingga 1. Ekspresi dalam logika *boolean* menentukan apakah suatu nilai adalah "benar" atau "salah". Logika *fuzzy* mengungkapkan tingkat intensitas menggunakan istilah seperti "sangat lambat", "agak sedang", dan "sangat cepat", di antara yang lainnya.

### 2.4.2 *Fuzzifikasi*

Proses *fuzzifikasi* mengubah variabel yang *non-fuzzy* (variabel numerik) menjadi variabel *fuzzy* (variabel linguistik). Sebelum diproses

oleh kontroler *fuzzy*, nilai masukan yang masih berupa variabel numerik terkuantisasi terlebih dahulu harus ditransformasikan ke dalam variabel *fuzzy*. Nilai input diubah menjadi informasi *fuzzy* oleh fungsi keanggotaan, yang kemudian dapat digunakan untuk pemrosesan *fuzzy* selanjutnya. Fuzzifikasi adalah proses yang terlibat di dalamnya.

#### 2.4.3 Inferencing (Rule Base)

Pada umumnya, aturan-aturan *fuzzy* dinyatakan dalam bentuk ‘IF...THEN’ yang merupakan inti dari relasi *fuzzy*. Relasi *fuzzy*, dinyatakan dengan R, juga disebut implikasi *fuzzy*. Ada dua cara utama untuk mendapatkan aturan ‘IF...THEN’ yaitu:

1. Menggunakan ‘human expert’ adalah istilah untuk operator manusia yang memiliki kemampuan untuk mengoperasikan sistem secara manual.
2. Memanfaatkan algoritma untuk pelatihan yang didasarkan pada data input dan output.

#### 2.4.4 Defuzzifikasi

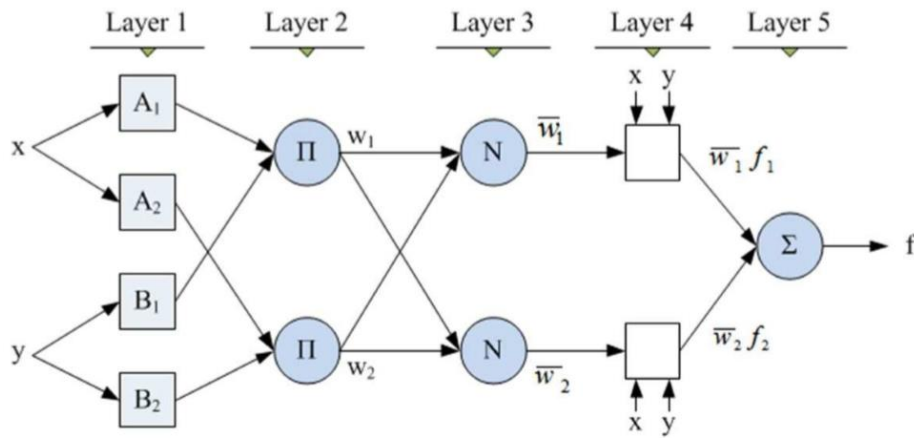
Keputusan yang dibuat sebagai hasil dari proses penalaran masih memiliki kualitas *fuzzy*, khususnya dalam bentuk derajat keanggotaan keluaran. Hasil ini perlu dilakukan *defuzzifikasi* agar dapat ditransformasikan kembali menjadi variabel numerik *non-fuzzy*.

### 2.5 Adaptive Neuro-Fuzzy Inference System (ANFIS)

Jaringan syaraf tiruan dan dua sistem logika fuzzy digabungkan untuk membentuk Neuro-Fuzzy. Sistem inferensi fuzzy yang dididik menggunakan algoritma pembelajaran yang dihasilkan dari sistem jaringan saraf tiruan berfungsi sebagai dasar dari sistem neuro-fuzzy. Oleh karena itu, sistem neuro-



fuzzy memiliki semua manfaat yang dimiliki sistem inferensi fuzzy dan sistem jaringan saraf tiruan.



Gambar 2.5 Struktur ANFIS dengan 2 Input

Tabel 2.1 Proses Pembelajaran Hibrida pada ANFIS

	Tahap Maju (Forward)	Tahap Mundur (Backward)
Parameter Premis	Tetap	Gradient descent- EBP
Parameter Konsekuen	LSE	Tetap
Sinyal	Keluaran Simpul	Sinyal Kesalahan

Penjelasan lebih lanjut tentang mekanisme pembelajaran ANFIS diberikan di bawah ini:

Dari lapisan input ke lapisan output, jaringan pertama kali merambat. Neuron/simpul ke-i pada lapisan ke-1 menghasilkan sinyal berikut:

### Lapisan 1

Simpul lapisan i ini adalah semua simpul adaptif dengan fungsi aktivasi simpul yang dijelaskan di bawah ini:

$$O_{1,i} = \mu_{A_i}(x) \quad (2.14)$$

untuk  $i = 1, 2$  atau

$$O_{1,i} = \mu_{A_{i-2}}(y) \quad (2.15)$$

untuk  $i = 3, 4$

Dengan x dan y adalah input pada simpul ke i, dan A<sub>i</sub> adalah label linguistik seperti baik, buruk, dsb. Dengan kata lain O<sub>1,i</sub> adalah fungsi keanggotaan dari A<sub>i</sub> dan menspesifikasikan derajat keanggotaan x dan y terhadap A<sub>i</sub>. Fungsi keanggotaan didasarkan pada persamaan bell dengan nilai maksimum 1 dan nilai minimum 0.

$$\mu_{A_i}(x) = bell(x; a, b, c) = \frac{1}{(1 + |\frac{x-c}{a}|^{2b})} \quad (2.16)$$

Jika kumpulan parameternya adalah "a, b, c". Pusat dan lebar kurva bell dapat diatur menggunakan parameter ci dan ai, namun kemiringan kurva diatur menggunakan sandi dan harus positif untuk mencegah kurva berbalik arah. Kurva fungsi bell akan berubah jika nilai parameter ini berubah. Menurut model fuzzy Sugeno, beberapa fungsi keanggotaan akan dibuat untuk himpunan linguistik Ai,....., I. Parameter-parameter lapisan ini dikenal sebagai parameter premis.

### Lapisan 2

Simpul lapisan ini semuanya adalah simpul non-adaptif. Setiap input yang masuk ke lapisan ini dikalikan dengan outputnya.

$$O_{2,i} = w_i = \mu_{A_i-2}(y), \quad (2.17)$$

$$i = 1, 2, \dots, n$$

Derajat pengaktifan (firing strength) dari setiap aturan fuzzy direpresentasikan oleh keluaran dari setiap simpul. Jumlah simpul di lapisan ini mewakili jumlah aturan yang dihasilkan.

### Lapisan 3

Setiap simpul pada lapisan ini adalah simpul non-adaptif yang menunjukkan fungsi derajat pengaktifan ternormalisasi (normalized firing strength), yang merupakan rasio dari semua output dari lapisan sebelumnya dengan output dari node ke-i pada lapisan sebelumnya.

$$O_{3,i} = \bar{w} = \frac{w_i}{w_1 + w_2}, \quad (2.18)$$

$$i = 1, 2$$

### Lapisan 4

Setiap simpul pada lapisan ini adalah simpul adaptif dengan fungsi simpul:

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \quad (2.19)$$

Dengan adalah derajat pengaktifan ternormalisasi dari lapisan ke 3 dan  $\{p_i, q_i, r_i\}$  menyatakan parameter konsekuen yang adaptif.

### Lapisan 5

Pada lapisan ini hanya ada satu simpul tetap yang fungsinya untuk menjumlahkan semua masukan.

$$O_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (2.20)$$

Jaringan adaptif dengan lima lapisan tersebut ekivalen dengan system inferensi fuzzy Sugeno.