

## APLIKASI FUZZY NEURAL NETWORK (FNN) PADA SISTEM KONTROL DENGAN WAKTU TUNDA

**Mukhtar Hanafi**

Program Studi Teknik Informatika Universitas Muhammadiyah Magelang  
Jl. Mayjend. Bambang Soeeng Km.5 Mertoyudan Magelang 56172 Telp.293-327945  
E-mail: [muhtarhanafi@yahoo.com](mailto:muhtarhanafi@yahoo.com)

### Abstrak

Waktu tunda pada sistem kontrol terjadi karena adanya jeda waktu bagi sensor untuk mendeteksi terjadinya error, sehingga mempengaruhi kecepatan controller untuk meresponnya. Akibat keidaklinieran dan ketidakpastian ini akan berdampak pada respon sistem secara keseluruhan. Jenis controller yang sering digunakan pada sistem kontrol ini adalah PID. Untuk mendapatkan respon sistem yang baik, harus dilakukan lebih dahulu tuning terhadap parameter  $P, I$  dan  $D$  pada PID controller pada nilai waktu tunda tertentu. Jika terjadi perubahan nilai waktu tunda diluar toleransi yang diijinkan, maka harus dilakukan tuning ulang untuk memperbaiki lagi respon sistemnya, dan ini membutuhkan waktu dan keahlian tersendiri. Sementara itu, Fuzzy Logic Controller (FLC) menawarkan metode pengontrolan yang lebih robust dalam menyesuaikan perubahan parameter akibat keidaklinieran dan ketidakpastian tersebut. Tetapi FLC sangat tergantung pada basis aturan (rule-based) yang digunakan dalam proses pengendalian sistem, dan rule-based tersebut sangat dipengaruhi oleh pengetahuan dan pengalaman perancangannya. Aturan yang salah dapat mengakibatkan ketidakstabilan sistem. Fuzzy Neural Network (FNN), sistem hybrid dari fuzzy dan jaringan syaraf, dapat memberikan solusi bagi sistem kontrol yang dikembangkan dengan konsep logika fuzzy ini. Hasil simulasi dengan matlab menunjukkan bahwa kemampuan pembelajaran yang dimiliki oleh unsur jaringan syaraf pada FNN dapat membantu memberikan respon sistem yang baik meskipun pada rule-based aturan yang digunakan terbatas.

**Kata Kunci:** waktu-tunda, respon sistem, rule-base, FNN

### 1. PENDAHULUAN

Untuk mengendalikan sistem yang memiliki atau dipengaruhi waktu tunda banyak digunakan controller konvensional seperti PID. Namun kendalanya adalah sistem kendali konvensional ini membutuhkan pengetahuan tentang parameter-parameter sistem terlebih dahulu. Selain itu, untuk sistem yang kompleks akan terdapat kendala dalam menentukan parameter-parameter yang sesuai agar mendapatkan respon keluaran yang stabil. Ada tiga parameter dalam pengontrol PID tersebut, yaitu *proporsional gain* ( $K_p$ ), *integral time* ( $T_i$ ) dan *derevative time* ( $T_d$ ). Ketiga parameter tersebut mempunyai pengaruh terhadap hasil respon sistem. Untuk memperoleh respon sistem yang baik (setabil), harus dilakukan proses penalaan atau *tuning* pada ketiga parameter tersebut.

Seiring dengan perkembangan *softcomputing*, sistem control berbasis logika fuzzy mulai digunakan untuk mengatasi permasalahan yang muncul pada sistem control konvensional. Selain hasil respon sistem yang lebih baik, kelebihan dari sistem kontrol berbasis logika fuzzy untuk sistem nonlinier adalah berkurangnya penggunaan model matematik yang rumit dalam proses perancangannya. Tapi kinerjanya akan sangat dipengaruhi oleh basis aturan (*rule base*) sebagai proses inferensi dalam mendapatkan output kontrol yang tepat. Sedangkan untuk menentukan *rule base* yang sesuai diperlukan pengalaman dari perancangannya, sehingga jika ini tidak tersedia maka tujuan pengontrolan akan sulit dicapai (Yan dkk, 1994).

Melalui penggabungan (*hybrid*) antara fuzzy dan jaringan syaraf tiruan atau *Fuzzy-Neural* suatu model dapat dilatih menggunakan jaringan syaraf untuk perbaikan fungsi keanggotaan dari sistem fuzzy yang digunakan sebagai sistem pengambil keputusan untuk mengendalikan peralatan, salah satunya adalah sistem kontrol dengan waktu tunda yang mempunyai sifat tidak linier (*nonlinier*) dan tidak pasti (*uncertain*).

### 2. TINJAUAN PUSTAKA

Logika fuzzy dirancang untuk dapat menerjemahkan pengetahuan pakar secara langsung melalui aturan-aturan dengan label-label *linguistic*, tapi umumnya membutuhkan waktu yang lama untuk mendesain dan menyesuaikan fungsi keanggotaan yang dapat memberikan definisi secara kuantitatif label-label *linguistic* ini. Teknik-teknik pembelajaran yang dimiliki jaringan syaraf dapat mengotomatisasi proses ini dan secara substansial mereduksi waktu perancangan dan biaya (Fuller, 1995).

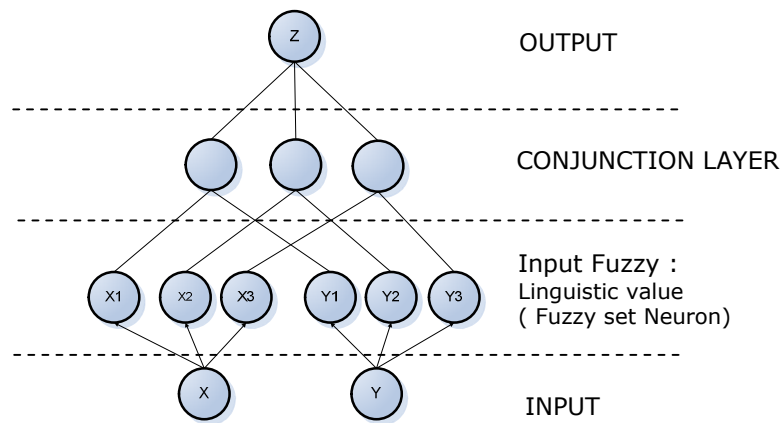
Kemampuan belajar dari jaringan syaraf tiruan pada *hybrid fuzzy neural network* dapat dilihat dari hasil penelitian yang dilakukan sebelumnya. Pada penelitian ini FNN diaplikasikan pada sistem control suspensi aktif. Dengan *rule base* dan fungsi keanggotaan yang sama dengan FLC, setelah dilakukan proses pembelajaran, hasil simulasi menunjukkan bahwa FNN mampu memperbaiki respon sistem dari FLC yang diujikan (Hanafi, 2011).

## 2.1 Fuzzy Neural Network

Fuzzy neural network merupakan jaringan syaraf tiruan dengan basis aturan fuzzy (*fuzzy rule base*), sehingga memungkinkan adanya pemetaan antara anteseden dan konsekuen dalam bentuk IF-THEN ke dalam jaringan syaraf. Untuk mengimplementasikan aturan fuzzy kedalam jaringan syaraf tiruan diperlukan level tambahan yang berupa level aturan-aturan fuzzy atau *fuzzy rules level* sehingga fuzzy neural network terdiri dari lima lapisan, yaitu lapisan input, lapisan input fuzzy, lapisan penghubung (*conjunction*), lapisan output fuzzy dan lapisan output (Fu, 1994).

Fungsi aktivasi dari unit input adalah nilai suatu variabel masukan yang sesuai dengan nilai yang diberikan. Nilai input dilewatkan dalam unit-unit himpunan fuzzy, yang akan merubah nilai kedalam suatu derajat keanggotaan sebagai fungsi aktifasi dari unit himpunan fuzzy (*fuzzy set unit*). Unit penghubung (*conjunction unit*) akan mengambil 'min' dari input (derajat keanggotaan) yang diterima dari masukan unit *fuzzy set* sebelumnya. Unit output fuzzy menampung informasi dari satu atau lebih unit *conjunction* (masing-masing berhubungan dengan sebuah aturan fuzzy). Ada variasi pada titik ini. Unit output himpunan fuzzy bisa menggunakan 'max' atau 'sum' dari masukan-masukan sebelumnya. Kemudian, unit output membangkitkan hasil akhir dengan mengintegrasikan informasi dari unit-unit output himpunan fuzzy.

Model lebih sederhana dari FNN ditunjukkan pada gambar 2.1. Pada FNN ini tidak digunakan lapisan 'ouput\_fuzzy'. Dari *fuzzy rule* pada *conjunction layer* yang digunakan fungsi keanggotaan keluarannya adalah *singletons* yang menyertai bobot koneksi pada lapisan terakhir, sehingga FNN ini hanya terdiri dari empat lapisan. (Kasabov, 1998) .



**Gambar 2.1** Fuzzy Naural Network dengan empat lapisan

Rumus yang digunakan untuk menentukan besarnya keluaran atau output z sebagai berikut,

$$z = \sum_{i=1}^m \mu_i \omega_i / \sum_{i=1}^m \mu_i \dots\dots\dots (2.1)$$

dimana  $z$  adalah level aktivasi dari unit output,  $\mu_i$  dan  $\omega_i$  berturut-turut adalah fungsi keanggotaan dan jalur terbobot unit ke  $i$  pada lapisan antara output fuzzy ke output.

## 2.2 Algoritma Backpropagation

Bobot-bobot pada setiap lapisan dari suatu jaringan syaraf antara lapisan input, lapisan tersembunyi dan lapisan output dilatih menggunakan algoritma pembelajaran terawasi (*supervised learning algorithm*), salah satunya adalah dengan algoritma backpropagation yang merupakan algoritma yang digeneralkan untuk meminimasi *mean-square error* diantara output jaringan yang diharapkan dengan output yang sebenarnya. Algoritma ini memiliki dasar matematis yang kuat, obyektif dan algoritma ini mendapatkan bentuk persamaan dan nilai koefisien dalam formula dengan meminimalkan jumlah kuadrat galat error melalui model yang dikembangkan (*training set*). Pelatihan suatu jaringan dengan algoritma *backpropagation* meliputi dua tahap, yaitu perambatan maju dan perambatan mundur (Fausset, 1994)

Proses pembelajaran dapat dipercepat dengan dimodifikasi backpropagation standar yaitu dengan menambahkan momentum. Penambahan momentum bertujuan untuk menghindari perubahan bobot yang mencolok akibat adanya data yang sangat berbeda dengan yang lain (*outlier*). Dengan penambahan momentum, bobot baru pada waktu ke  $(t+1)$  didasarkan atas bobot pada waktu  $t$  dan  $(t-1)$ . Jika  $\beta$  adalah konstanta

(( $0 \leq \beta \leq 1$ )) yang menyatakan parameter momentum, maka bobot baru dapat dihitung dengan persamaan (Fausset, 1994):

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk} + \beta(w_{jk}(t) - w_{jk}(t-1)) \dots\dots\dots(2.2)$$

dengan  $\Delta w_{jk} = \alpha \delta_k z_j$ , maka persamaan (2.10) dapat dituliskan kembali dengan

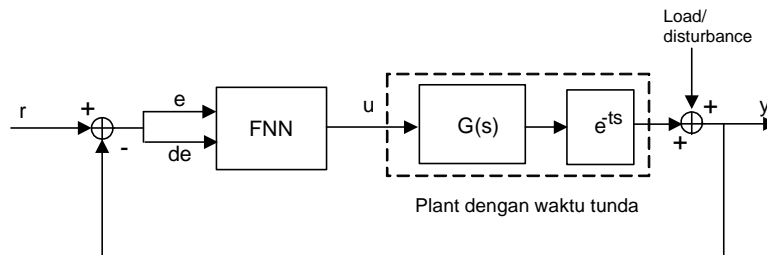
$$w_{jk}(t+1) = w_{jk}(t) + \alpha \delta_k z_j + \beta(w_{jk}(t) - w_{jk}(t-1)) \dots\dots\dots(2.3)$$

dimana  $\alpha$  adalah *learning factor* dan  $\beta$  adalah *momentum factor*.

### 3. METODE PENELITIAN

#### 3.1 Pembuatan Model Sistem Kontrol

Struktur sistem kontrol dengan waktu tunda ditunjukkan pada gambar 3.1. Sinyal kontrol  $u$  merupakan sinyal dihasilkan oleh pengontrol FNN yang digunakan untuk mengontrol  $G(s)$  dengan waktu tunda ( $e^{-ts}$ ).



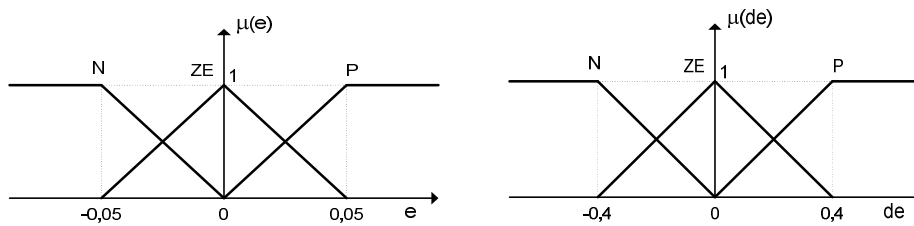
**Gambar 3.1** Struktur sistem kontrol FNN pada *plant* dengan waktu tunda

$G(s)$  adalah *transfer function* dari *plant* dan  $e^{-ts}$  merupakan unsur waktu tunda yang terjadi pada *plant*. Sedangkan masukan  $r$  merupakan *setpoint* atau input sistem. Sinyal informasi yang diterima oleh pengontrol FNN merupakan sinyal *error* ( $e$ ) dan perubahan *error* ( $de$ ) yang terjadi antara output ( $y$ ) sistem dengan input sistem.

#### 3.2 Menentukan Fungsi Keanggotaan Fuzzy

Untuk mendapatkan besaran linguistik dari sinyal masukan yang berupa sinyal *script* dilakukan dengan proses fuzzifikasi. Proses fuzzifikasi digunakan untuk memetakan nilai input yang dilewatkan ( $e$  dan  $de$ ) kedalam suatu derajat keanggotaan fuzzy. Fungsi keanggotaan yang digunakan pada penelitian ini adalah fungsi segitiga yang memiliki tiga variabel linguistik yaitu negatif (N), *about zero* (Z) dan positif (P).

Berdasar hasil analisa respon sistem dengan waktu tunda yang kemudian dilakukan penyesuaian dengan coba-coba (*trial and error*) sampai diperoleh pengontrol yang baik, didapatkan fungsi keanggotaan segitiga untuk masukan  $e$  dan  $de$  seperti terlihat pada Gambar 3. 2

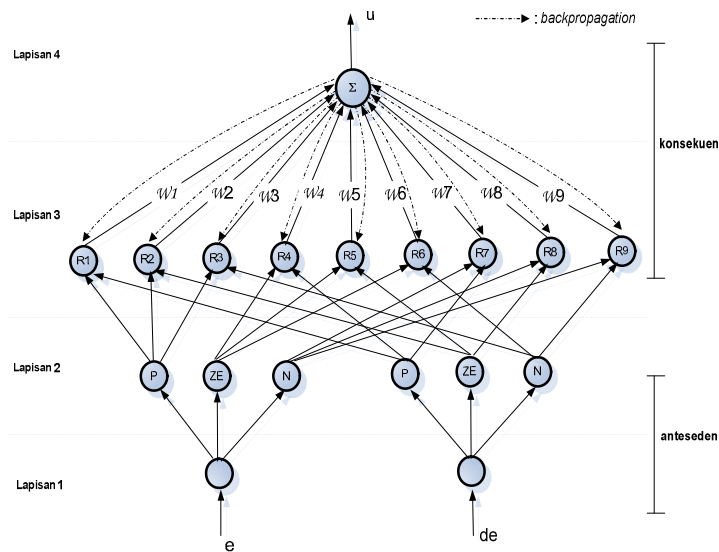


**Gambar 3.2** Fungsi keanggotaan dan variabel linguistik untuk  $e$  dan  $de$

Selanjutnya dari tiga variabel linguistik fuzzy yang masing-masing untuk  $e$  dan  $de$ , disusun aturan-aturan fuzzy (*fuzzy rule base*). Aturan-aturan fuzzy disusun untuk mengakomodasi semua kemungkinan masukan nilai linguistik  $e$  dan  $de$ , dan dengan *conjungtion AND* semua pasangan variabel linguistik antara masukan  $e$  dan  $de$  dikombinasikan sehingga diperoleh 9 aturan fuzzy.

#### 3.3 Perancangan Struktur Jaringan FNN

Struktur jaringan syaraf berbasis fuzzy atau fuzzy neural network sebagai *controller* yang dirancang pada penelitian ini ditunjukkan pada gambar 3.3.



**Gambar 3.3** Struktur FNN

Lapisan pertama dari struktur FNN merupakan lapisan input yang berfungsi melewati variabel masukan. Output dari lapisan pertama ini adalah:  $y_{1,i}^{(1)} = e$  dan  $y_{2,j}^{(1)} = de$ . Kemudian pada lapisan kedua dilakukan proses fuzzifikasi dengan memetakan nilai input yang dilewatkan ( $e$  dan  $de$ ) kedalam suatu derajat keanggotaan fuzzy seperti terlihat pada Gambar 3.2. Keluaran dari lapisan kedua ini adalah:  $y_{1,i}^{(2)} = \mu(y_{1,i}^{(1)}) \dots (3.1)$  dan  $y_{2,j}^{(2)} = \mu(y_{2,j}^{(1)}) \dots (3.2)$ .

Pada lapisan ketiga yang merupakan lapisan *conjunction*, digunakan 9 neuron untuk merealisasikan 9 aturan (*rule*) fuzzy. Dengan penghubung **AND** untuk besaran fuzzy dari masukan  $e$  dan  $de$ , maka fungsi aktivasi yang digunakan pada lapisan ini adalah  $\mu_{i,j} = \min(\mu(e_i), \mu(de_j))$  atau  $\mu_{i,j} = \min(\mu(y_{1,i}^{(2)}), \mu(y_{2,j}^{(2)}))$ . Keluarannya dapat dirumuskan dengan:  $y_{i,j}^{(3)} = \mu_{i,j} \dots (3.3)$ .

Lapisan keempat merupakan lapisan *output*. Pada lapisan ini akan dilakukan defuzzifikasi atas hasil keluaran dari neuron-neuron pada lapisan sebelumnya. Pada lapisan ini dilakukan pembobotan dengan  $w_i$  pada setiap masukan. Keluaran dari lapisan ini dihitung dengan persamaan:

$$y^{(4)} = \frac{\sum_{i,j=1}^n y_{i,j}^{(3)} \cdot w_i}{\sum_{i,j=1}^n y_{i,j}^{(3)}} = \frac{\sum_{i,j=1}^n \mu_{i,j} \cdot w_i}{\sum_{i,j=1}^n \mu_{i,j}} \dots (3.4), \text{ dengan } n = 9 \text{ rules, maka: } u = \frac{\sum_{i,j=1}^9 \mu_{i,j} \cdot w_i}{\sum_{i,j=1}^9 \mu_{i,j}} \dots (3.5)$$

### 3.3 Pelatihan FNN

Proses pelatihan yang dilakukan pada FNN ini menggunakan algoritma *backpropagation*. Proses pelatihan ini hanya untuk melatih bobot koneksi lapisan ke 4, untuk lapisan yang lain bobot koneksi diset dengan nilai satu. Jika *Error* antara keluaran FNN yang sebenarnya ( $u(t)$ ) dengan keluaran yang diinginkan ( $ud(t)$ ), ditentukan dengan rumusan:  $E = \frac{1}{2}(ud(t) - u(t))^2 \dots (3.6)$  dan untuk modifikasi bobot koneksi jaringan digunakan persamaan 2.3, dimana  $\delta_k = (t_k - y_k) f'(y_{in_k})$ , sehingga modifikasi bobot koneksi pada lapisan ke 4 pada jaringan FNN yang dirancang ini dilakukan dengan persamaan:

$$w_{i4}(t+1) = w_{i4}(t) + \alpha \left( -\frac{\partial E}{\partial w_{i4}} \right) + \beta (w_{i4}(t) - w_{i4}(t-1)) \dots (3.7), \text{ dengan}$$

$$\frac{\partial E}{\partial w_{i4}} = \frac{\partial E}{\partial u} \frac{\partial u}{\partial w_{i4}} = (ud(t) - u(t)) \frac{\mu_i}{\sum_{i=1}^9 \mu_i} \dots (3.8), \text{ dengan } \alpha = \text{learning factor} \text{ dan } \beta = \text{momentum factor}.$$

### 3.4 Pengujian Sistem

Pengujian sistem dilakukan dengan simulasi menggunakan *software* Matlab. Model proses atau *plant* yang digunakan pada pengujian ini adalah *Heat Exchanger*. *Heat Exchanger* merupakan contoh proses atau *plant* yang mengandung *deadtime*. Fungsi alih *Heat Exchanger* dapat dinyatakan dalam bentuk sistem orde satu (Santoso, 2003):

$$\frac{T(s)}{F_h(s)} = \frac{K_{pex}}{\tau_{ex}s + 1} = G_{ex}(s) \dots\dots\dots(3.9),$$

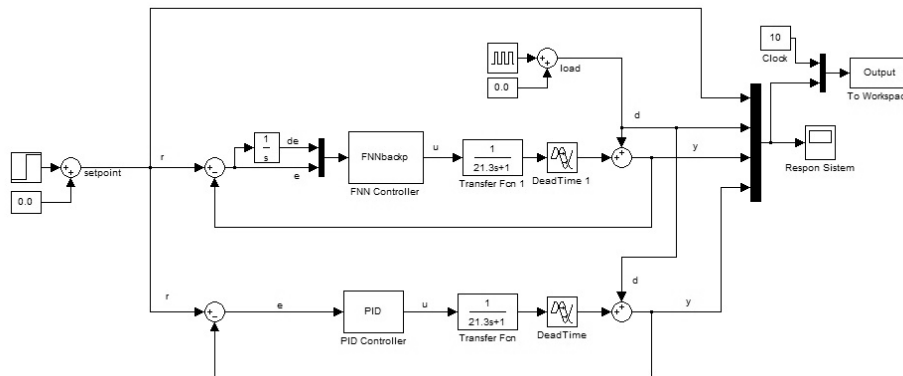
Karena ada waktu tunda / *death time* pada *plant* maka fungsi alih *plant* menjadi:

$$\frac{T(s)}{F_h(s)} = \frac{K_{pex}}{\tau_{ex}s + 1} e^{-ts} = G_{ex}(s) \dots\dots\dots(3.10),$$

dengan *t* merupakan *death time process*. Pada pengujian ini fungsi alih *plant* dinyatakan dengan:

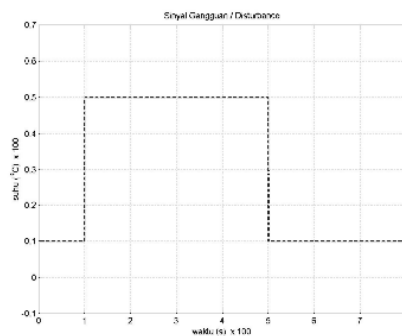
$$G(s) = \frac{1}{21,3s + 1} e^{-10} \dots\dots\dots(3.11).$$

Berdasarkan struktur kontrol dengan fungsi alih *plant* pada persamaan (3.11), dibuat rangkaian simulasi untuk pengujian sistem kontrol menggunakan simulink matlab seperti pada gambar 3.4



**Gambar 3.4.** Rangkaian simulasi sistem kontrol PID dan FNN

Pengujian dilakukan dengan *step respon system* dan pengaruh adanya gangguan (*disturbance*) atau *load*, seperti yang terlihat pada gambar 3.5. Untuk input *step respon system* ini diuji dengan lama waktu tunda yang bervariasi, yaitu 10s, 20s, 30s dan 45s.



**Gambar 3.5** Sinyal gangguan atau *load*

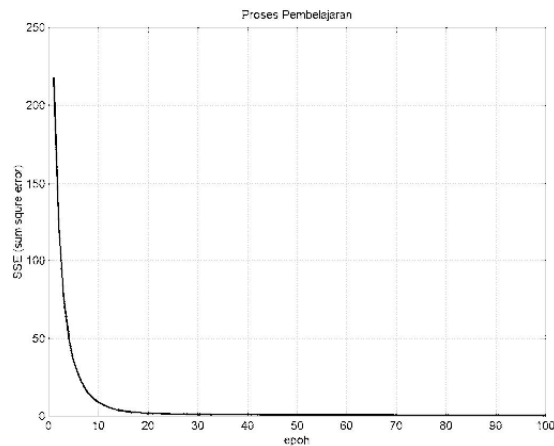
Pengujian berikutnya adalah pengujian karena pengaruh adanya gangguan (*disturbance*) atau *load*. Pada pengujian ini input sistem diset sebesar 10 °C, kemudian diberi *disturbance* atau *load* dengan kenaikan suhu menjadi 50 °C dalam 400 s dan turun kembali ke posisi set poinnya, seperti yang terlihat pada Gambar 3.5b. Pengujian ini juga dilakukan dengan lama waktu tunda yang bervariasi, yaitu 10s, 20s, 30s dan 45s.

## 4. HASIL DAN PEMBAHASAN

### 4.1 Hasil pelatihan FNN

Pada proses pelatihan FNN ini digunakan nilai bobot koneksi awal = 0 untuk semua bobot koneksi. Nilai *learning factor*( $\alpha$ ) yang digunakan 0,001 dan *momentum factor*( $\beta$ ) nya adalah 0,015. Kombinasi nilai  $\alpha$  dan  $\beta$  ini

berpengaruh terhadap kecepatan proses pelatihan dan ketepatan nilai bobot yang diperoleh. Nilai  $\alpha$  dan  $\beta$  disini didapatkan secara *trial and error*, semakin kecil nilai  $\alpha$  proses pembelajar akan semakin lama tetapi tingkat keakuratan bobot yang diperoleh akan semakin tinggi. Proses pelatihan dibatasi sampai 100 epoh. Hasil proses pembelajarannya seperti yang terlihat pada gambar 4.1.

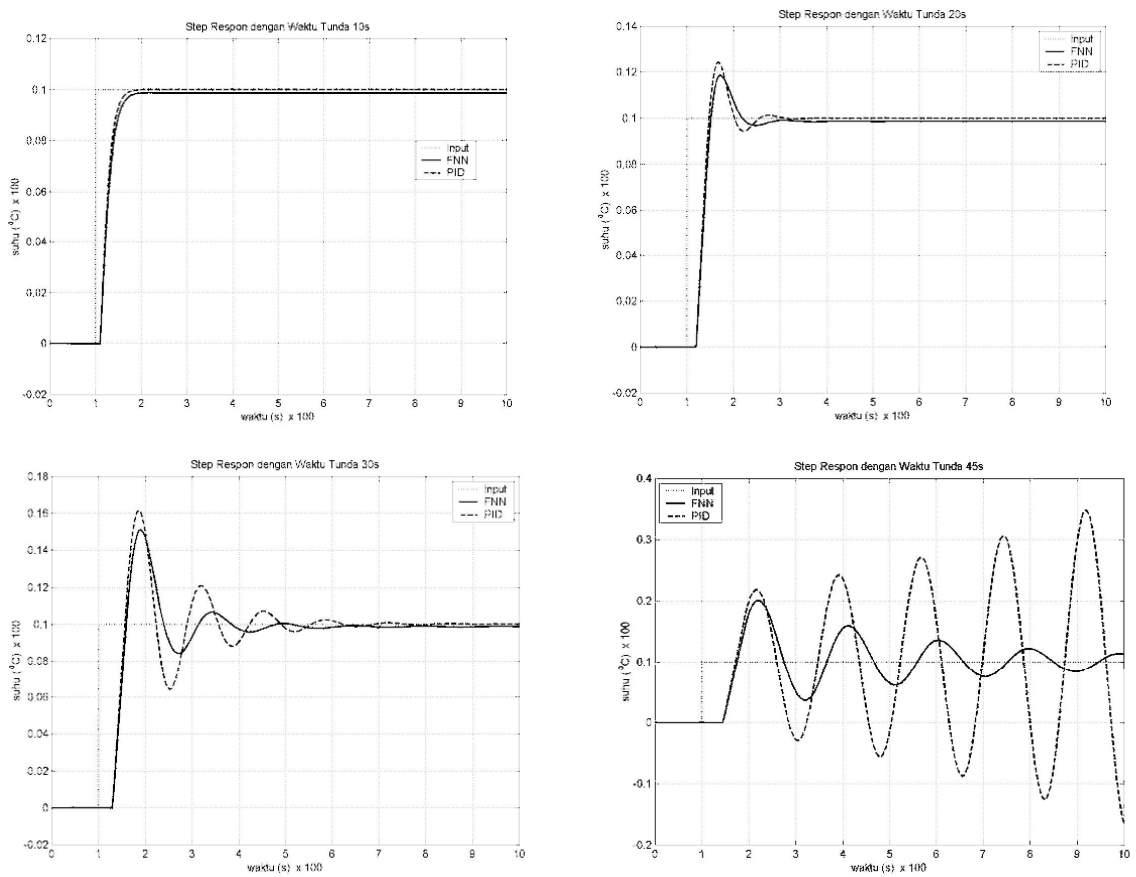


Gambar 4.1 Proses pelatihan FNN

Proses pembelajaran FNN pada gambar 4.1 diatas menunjukkan bahwa nilai SSE (sum square error) semakin kecil. Hal ini berarti jaringan syaraf pada FNN dapat melakukan proses pembelajaran dengan baik terhadap data yang dilatihkan.

#### 4.2 Respon Sistem

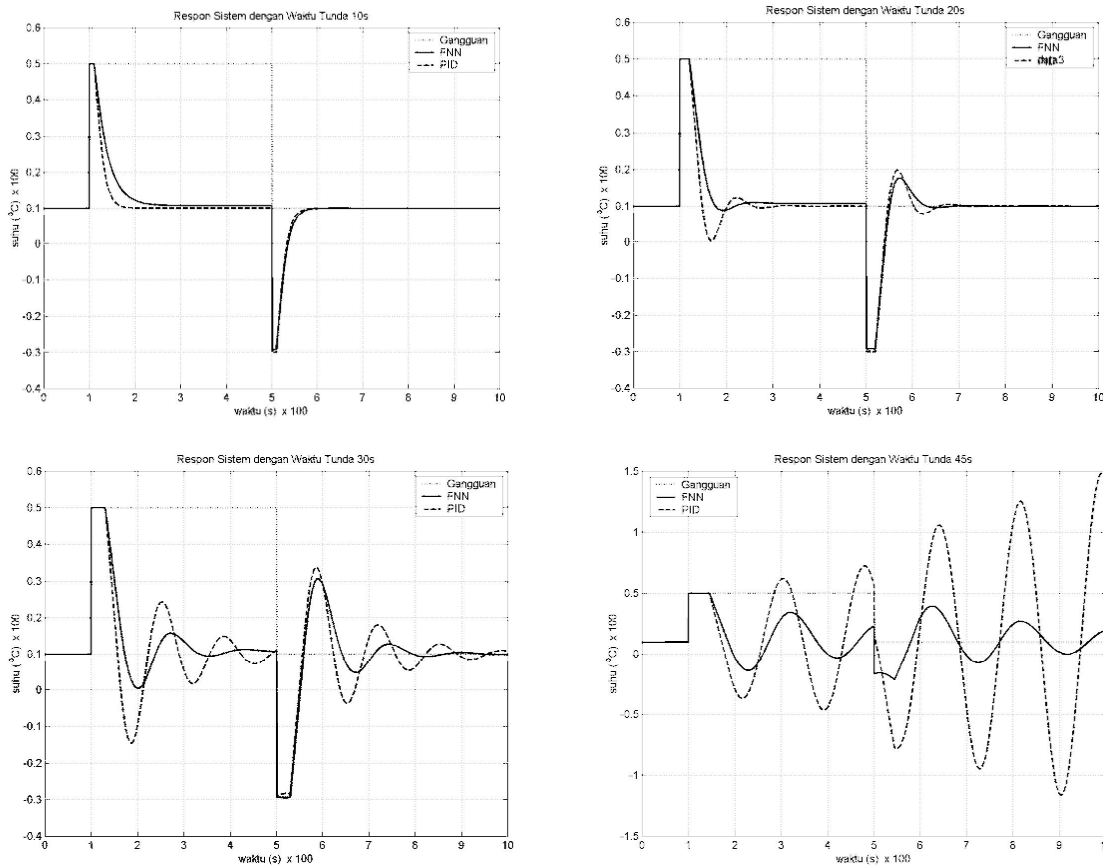
Hasil *step respon system* dari kedua sistem kontrol dengan lama waktu tunda yang bervariasi, yaitu 10s, 20s, 30s dan 45s, ditunjukkan pada gambar 4.2. Untuk *step respon system* dengan waktu tunda 10s, seperti yang ditunjukkan pada Gambar 14a, pada *controller* PID sudah dilakukan *tuning* parameter kontrol PID yaitu  $K_p$ ,  $K_i$  dan  $K_d$  sehingga diperoleh respon sistem yang baik.



Gambar 4.2 Step respon system dengan waktu tunda: 10s, 20s, 30s dan 45s

Hasil pengujian pada gambar 4.2 diatas, menunjukkan bahwa FNN mampu memberikan *step respon system* yang baik untuk seluruh *waktu tunda* yang diujikan.

Untuk pengujian sistem dengan menggunakan *disturbance* atau *load* terlihat pada gambar 4.3. Akibat adanya gangguan (*disturbance*), nilai *controlled variable* dalam hal ini temperatur naik selama 400s kemudian kembali normal, sehingga controller meningkatkan sinyal kontrol *u*, untuk membawa sistem kembali ke nilai *set point*-nya.



**Gambar 4.3** Respon sistem akibat *disturbance* atau *load* untuk waktu tunda: 10s, 20s, 30s dan 45s

Secara umum hasil simulasi pengujian, baik untuk *step respon* maupun *respon sistem* akibat adanya *disturbance* atau *load*, menunjukkan bahwa *controller* FNN dapat memberikan hasil pengontrolan yang baik untuk seluruh waktu tunda yang diujikan. Disini FNN mampu memberikan respon sistem yang baik meskipun pada unsur fuzzy yang digunakan hanya memakai sembilan *rule* dan kesebelas *rule* tersebut diperoleh dengan cara mengkombinasikan variabel-variabel linguistik dari masukan *e* dan *de*.

## 5. KESIMPULAN

Hasil simulasi pengendalian sistem dengan waktu tunda menggunakan *Fuzzy Neural Network* (FNN) controller menunjukkan bahwa, meskipun dengan menggunakan *rule base* yang terbatas pada unsur logika fuzzy, kemampuan belajar yang dimiliki jaringan syaraf tiruan pada FNN mampu mengoptimalkan unsur-unsur logika fuzzynya, sehingga respon sistem yang baik mampu ditunjukkan oleh FNN controller untuk seluruh waktu tunda yang diujikan.

## DAFTAR PUSTAKA

- Fausset, L., 1994, *Fundamental of Neural Network*, Prentice-Hall.
- Fuller, R. , 1995, *Neural Fuzzy Systems*, Abo Akademi University, Abo, Turki.
- Fu, L.M., 1994, *Neural Network in Computer Intelligence*, International Edition, McGraw Hill.

- Hanafi, M, 2011, Implementasi Sistem Hybrid Menggunakan Fuzzy Neural Network untuk Memperbaiki Kinerja Sistem Kontrol Suspensi Aktif Berbasis Logika Fuzzy, *Prossiding Seminar Teknik Informatika 2011 (STI 2011)*, edisi 1 Juli 2011, Universitas Ahmad Dahlan , Yogyakarta
- Kasabov, N.K., 1998, *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering*, Massachusetts Institute of Technology, England.
- Santoso F, 2003, Perbandingan Kinerja Sistem Kontrol Berumpan Balik (Feedback) Dengan Sistem Kontrol Berumpan Maju (Feedfoward) Pada Jaringan Penukar Panas (Heat Exchanger), *Jurnal Teknik Mesin Universitas Kristen Petra*, Vol. 5, No.1 April 2003, 36 – 42
- Yan, J., Ryan, M., Power, J., 1994, *Using Fuzzy Logic*, Prentice Hall.