

# **APLIKASI LOGIKA FUZZY DAN JARINGAN SYARAF TIRUAN SEBAGAI METODE ALTERNATIF PREDIKSI BEBAN LISTRIK JANGKA PENDEK**

**Yadi Mulyadi<sup>1</sup>, Ade Gafar Abdullah<sup>2</sup>, Dadang Lukman Hakim<sup>3</sup>**

Laboratorium Komputasi dan Otomasi Industri  
Jurusan Pendidikan Teknik Elektro FPTK UPI  
Jl. Dr. Setiabudhi No.207 Bandung

## **ABSTRAK**

*Perkembangan teknologi komputasi yang sudah mengarah kepada teknologi soft computing (istilah lainnya komputer cerdas) mendorong peneliti untuk mencoba mencari suatu metode alternatif prediksi beban listrik jangka pendek berbasis kecerdasan buatan (yang populer dan banyak dipakai para ilmuwan : berbasis Fuzzy Logic/ Logika Fuzzy, Adaptive Neural Network/ Jaringan Syaraf Tiruan). Kemudahan konsep logika fuzzy dan jaringan syaraf tiruan mendorong peneliti untuk membuat prediksi pemakaian energi listrik jangka pendek. Sumber data yang diperlukan adalah data pengeluaran beban listrik dari Pusat Pembagi Beban PT. PLN (Persero) setiap jam mulai pukul 00.00 sampai dengan pukul 24.00 mulai hari senin sampai dengan minggu pada hari normal dan hari-hari libur nasional, yang kemudian data tersebut akan dibelajarkan pada sistem perangkat lunak yang sudah dirancang berbasis logika fuzzy dan jaringan syaraf tiruan. Software pendukung untuk merancang program digunakan Matlab ver.7.0 dari Mathwork Corp. Melalui perhitungan dari hasil simulasi didapatkan rata-rata error keseluruhan untuk ketiga metode, yaitu 3.93 % untuk PLN dan 1.08 % untuk fuzzy. Dengan demikian dapat disimpulkan bahwa prakiraan beban listrik jangka pendek dengan menggunakan fuzzy subtractive clustering lebih baik dibandingkan dengan prakiraan beban listrik dengan menggunakan metode koefisien yang dilakukan oleh PLN, akan tetapi hasil tersebut masih jauh dari prakiraan JST (prakiraan oleh JST 0.26 % .*

## **1. PENDAHULUAN**

Akhir-akhir ini kita banyak mendengar berita bahwa Perusahaan Listrik Negara (PLN) mengalami kerugian yang sangat besar setiap tahunnya yang disebabkan faktor-faktor yang sangat kompleks mulai dari dampak kenaikan BBM terhadap harga batu-bara sebagai bahan bakar utama PLTU, pencurian listrik, efisiensi unit pembangkitan yang sangat rendah akibat umur unit pembangkit yang sudah tua dan masalah lainnya, sehingga dijadikan alasan PLN untuk mendesak pemerintah menaikkan Tarif Dasar Listrik (TDL) yang tentunya akan semakin membebani pertumbuhan ekonomi masyarakat.

Tenaga listrik tidak dapat disimpan dalam skala besar, karenanya tenaga ini harus disediakan pada saat dibutuhkan. Akibatnya timbul persoalan dalam menghadapi kebutuhan daya listrik yang tidak tetap dari waktu ke waktu, bagaimana mengoperasikan suatu sistem tenaga listrik yang selalu dapat memenuhi permintaan daya pada setiap saat, dengan kualitas baik dan harga yang murah. Apabila daya yang dikirim dari bus-bus pembangkit jauh lebih besar daripada permintaan daya pada bus-bus beban, maka akan timbul persoalan pemborosan energi pada perusahaan listrik, terutama untuk pembangkit termal. Sedangkan apabila daya yang dibangkitkan dan dikirimkan lebih rendah atau tidak memenuhi kebutuhan beban konsumen maka akan terjadi pemadaman lokal pada bus-bus beban, yang akibatnya merugikan pihak konsumen. Oleh karena itu diperlukan penyesuaian antara pembangkitan dengan permintaan daya.

Untuk mempertahankan unjuk kerja (*performance*) sistem tenaga listrik maka sistem harus terus dikembangkan. Pengembangan sistem yang terlambat memberikan risiko terjadinya pemadaman/pemutusan dalam penyediaan tenaga listrik bagi pelanggan sebagai akibat terjadinya beban yang lebih besar daripada kemampuan instalasi. Analisis hasil-hasil operasi diperlukan sebagai masukan untuk rencana pemeliharaan instalasi. Pemeliharaan instalasi yang sebaik mungkin sangat diperlukan untuk mengurangi gangguan yang berarti menaikkan keandalan operasi sistem. Masalah yang unik dalam operasi sistem adalah bahwa : “Daya yang dibangkitkan/diproduksi harus selalu sama dengan daya yang dikonsumsi oleh para pemakai tenaga listrik yang secara teknis umumnya dikatakan sebagai beban sistem” (Djiteng Marsudi, 1990:13).

Syarat mutlak yang pertama harus dilaksanakan untuk mencapai tujuan itu adalah pihak perusahaan listrik mengetahui beban atau permintaan daya listrik dimasa depan. Karena itu prakiraan beban jangka pendek, menengah dan panjang merupakan tugas yang penting dalam perencanaan dan pengoperasian sistem daya. Prakiraan beban jangka pendek, yaitu beban setiap jam atau tiap hari digunakan untuk penjadwalan dan pengontrolan sistem daya atau alokasi pembangkit cadangan berputar, juga digunakan untuk masukan dalam studi aliran daya.

Satu hal yang luput dari analisis kerugian PLN adalah masalah estimasi (prediksi) pengeluaran beban listrik. Padahal bahwa kemampuan pihak P2B (Pusat Pembagi Beban) PT. PLN (Persero) untuk memprediksi berapa besar beban listrik yang harus dikeluarkan setiap waktunya sangat diandalkan. Metoda Koefisien Beban yang sudah lama digunakan PLN ternyata masih memberikan error prediksi yang cukup besar yaitu rata-rata berkisar antara 4%-5%. Sehingga menimbulkan kerugian daya yang cukup besar bagi PLN untuk setiap satuan waktunya. Oleh sebab itu menjadi suatu tantangan bagi peneliti untuk mencari suatu model prediksi beban listrik sehingga menghasilkan error prediksi yang lebih baik dari model prediksi yang selama ini dipakai PLN.

Perkembangan teknologi komputasi yang sudah mengarah kepada teknologi *soft computing* (istilah lainnya komputer cerdas) mendorong peneliti untuk mencoba mencari suatu metode alternatif prediksi beban listrik jangka pendek berbasis kecerdasan buatan (yang populer dan banyak dipakai para ilmuwan : berbasis *Fuzzy Logic/ Logika Fuzzy, Adaptive Neural Network/ Jaringan Syaraf Tiruan*). Kemudahan konsep logika fuzzy dan jaringan syaraf tiruan mendorong peneliti untuk membuat prediksi pemakaian energi listrik jangka pendek. Sumber data yang diperlukan adalah data pengeluaran beban listrik dari Pusat Pembagi Beban PT. PLN (Persero) setiap jam mulai pukul 00.00 sampai dengan pukul 24.00 mulai hari senin sampai dengan minggu pada hari normal dan hari-hari libur nasional, yang kemudian data tersebut akan dibelajarkan pada sistem perangkat lunak yang sudah dirancang berbasis logika fuzzy dan jaringan syaraf tiruan. Software pendukung untuk merancang program digunakan Matlab ver.7.0 dari Mathwork Corp.

## **2. METODE PENELITIAN**

Dalam penelitian ini menggunakan 2 algoritma untuk dibandingkan keandalannya dalam memprediksi beban, yaitu menggunakan algoritma fuzzy subtractive clustering dan algoritma backpropagation untuk jaringan syaraf tiruan.

### **2.1. Algoritma Fuzzy Subtractive Clustering**

Algoritma fuzzy *subtractive clustering* adalah sebagai berikut :

1. Input data yang akan dicluster :  $X_{ij}$  dengan  $i = 1,2,\dots,n$ ; dan  $j = 1,2,\dots,m$ .

2. Tetapkan nilai :
  - a.  $r_j$  (*influence range*)
  - b.  $q$  (*squash factor*)
  - c. *Accept\_ratio*
  - d. *Reject\_ratio*
  - e.  $X_{min}$
  - f.  $X_{max}$
3. Normalisasi :  $X_{ij} = \frac{X_{ij} - X_{min_j}}{X_{max_j} - X_{min_j}}, i = 1, 2, \dots, n; j = 1, 2, \dots, m$
4. Tentukan potensi awal tiap-titik data
  - a.  $i = 1$
  - b. kerjakan hingga  $i = n$ 
    - $T_j = X_{ij}; j = 1, 2, \dots, m$
    - Hitung :  $D_{kj} = \left( \frac{T_j - X_{kj}}{r} \right); j = 1, 2, \dots, m; k = 1, 2, \dots, n$
    - Potensi awal : jika  $m = 1$ , maka  $D_i = \sum_{k=1}^n e^{-4(D_{k1}^2)}$   
 jika  $m > 1$ , maka  $D_i = \sum_{k=1}^n c^{-4 \left( \sum_{j=1}^m D_{kj}^2 \right)}$
5. Cari titik dengan D tertinggi
  - a.  $m = \max[D_i; i = 1, 2, \dots, n]$
  - b.  $h = i$ ; sedemikian hingga  $D_i = m$
6. Tentukan pusat cluster dan kurangi potensinya terhadap titik-titik disekitarnya :
  - a.  $Center = []$
  - b.  $D'_k = D_k - D_{c1} * e^{-\left( \frac{\|x_k - x_{c1}\|}{(r_b / 2)^2} \right)}$
7. Kembalikan pusat cluster dari bentuk ternormalisasi ke bentuk semula.  
 $Center_{ij} = Center_{ij} * (X_{max_j} - X_{min_j}) + X_{min_j}$
8. Hitung nilai sigma cluster :  $\sigma_j = r_j \left( \frac{X_{max_j} - X_{min_j}}{\sqrt{8}} \right)$
9. Hitung derajat keanggotaan :  $\mu_{ki} = e^{-\sum_{j=1}^m \frac{(x_{ij} - c_{ij})^2}{2\sigma_j^2}}$

## 2.2. Algoritma Backpropagation

Adapun algoritma backpropagation yang disusun untuk membuat model prakiraan dengan jaringan syaraf tiruan adalah :

- Inisialisasi bobot
- Menetapkan :
  - a. Maksimum epoh = 9000
  - b. Target Error = 0.001
  - c. Learning Rate ( $\alpha$ ) = 1.5
- Inisialisasi: Epoh = 0, MSE = 1

- Mengerjakan langkah-langkah berikut selama (Epoh < Maksimum Epoh, dan Mean Square Error / MSE > Target Error):
  1. Epoh = Epoh + 1
  2. Untuk setiap pasangan elemen yang akan dilakukan pembelajaran, dikerjakan:

**Feedforward:**

- a. Tiap-tiap unit neuron input ( $X_i, i=1,2,3,\dots,5$ ) menerima sinyal  $x_i$  dan meneruskan sinyal tersebut ke semua unit pada lapisan yang di atasnya (lapisan kedua/lapisan tersembunyi pertama,  $[Z_{aj}, j=1,2,3,\dots,10]$ ).
- b. Tiap-tiap unit neuron pada lapisan kedua/lapisan tersembunyi pertama ( $Z_{aj}, j=1,2,3,\dots,10$ ), menjumlahkan sinyal-sinyal input terbobot dengan rumus:

$$z_{a\_in\ j} = b_{1j} + \sum_{i=1}^5 X_i V_{ij}$$

dimana,

- $z_{a\_in\ j}$  = hasil penjumlahan sinyal-sinyal input terbobot dari lapisan pertama yang dikerjakan oleh neuron ke-j pada lapisan kedua.
- $b_{1j}$  = bobot bias yang menghubungkan lapisan pertama (input) dengan neuron ke-j pada lapisan kedua/lapisan tersembunyi pertama.
- $V_{ij}$  = Bobot input yang menghubungkan lapisan pertama (input) dengan neuron ke-j pada lapisan kedua.

Dengan menggunakan fungsi aktivasi *tansig*, sinyal output pada lapisan kedua akan dihitung dengan rumus:

$$Z_{aj} = f(z_{a\_in\ j})$$

dan sinyal  $z_{aj}$  ini kemudian akan dikirimkan pada lapisan yang ada di atasnya (yaitu:: lapisan ketiga/lapisan tersembunyi kedua).

- c. Tiap-tiap unit neuron pada lapisan ketiga/lapisan tersembunyi kedua ( $Z_{bk}, k=1,2,\dots,4$ ), menjumlahkan sinyal-sinyal input terbobot  $z_{aj}$  dengan rumus:

$$z_{b\_in\ k} = b_{2k} + \sum_{j=1}^{10} Z_{aj} U_{jk}$$

dimana,

- $z_{b\_in\ k}$  = hasil penjumlahan sinyal-sinyal input terbobot dari lapisan kedua yang dikerjakan oleh neuron ke-k pada lapisan ketiga.
- $b_{2k}$  = bobot bias yang menghubungkan lapisan kedua dengan neuron ke-k pada lapisan ketiga/lapisan tersembunyi kedua.
- $U_{ij}$  = Bobot input yang menghubungkan lapisan kedua dengan neuron ke-k pada lapisan ketiga.

Dengan menggunakan fungsi aktivasi *logsig*, sinyal output pada lapisan ketiga akan dihitung dengan rumus:

$$Z_{bk} = f(z_{b\_in\ k})$$

dan sinyal  $z_{bk}$  ini kemudian akan dikirimkan pada lapisan yang ada di atasnya (yaitu:: lapisan keempat/lapisan output).

- d. Tiap-tiap unit neuron pada lapisan keempat/lapisan output ( $Y_m, m=1$ ), menjumlahkan sinyal-sinyal input terbobot  $Z_{bk}$  dengan rumus:

$$y_{in_m} = b_{3m} + \sum_{k=1}^4 Z_{b_{km}} W_{km}$$

dimana,

$y_{in_m}$  = hasil penjumlahan sinyal-sinyal input terbobot dari lapisan ketiga yang dikerjakan oleh unit neuron pada lapisan keempat/lapisan output.

$b_{3m}$  = bobot bias yang menghubungkan lapisan ketiga dengan unit neuron ke-m pada lapisan keempat.

$W_{km}$  = Bobot input yang menghubungkan lapisan ketiga dengan unit neuron ke-m pada lapisan keempat.

Dengan menggunakan fungsi aktivasi *purelin*, sinyal output pada lapisan keempat akan dihitung dengan rumus:

$$Y_m = f(y_{in_m})$$

### Backpropagation :

- e. Tiap-tiap unit output ( $Y_m$ ) pada lapisan keempat menerima target ( $T_m$ ) pola yang berhubungan dengan pola input pembelajaran, maka dapat dihitung informasi error dengan rumus:

$$\delta 4_m = (T_m - Y_m) f'(y_{in_m})$$

$$\phi 4_m = \delta 4_m Z_{bk}$$

$$\beta 4_m = \delta 4_m$$

dimana,

$(T_m - Y_m)$  = error antara target dengan output pada unit ke-m.

$f'(y_{in_m})$  = fungsi turunan dari  $y_{in_m}$

$\delta 4_m$  = delta input pada unit ke-m dalam lapisan keempat.

$\phi 4_m$  = besarnya bobot pada unit ke-m dalam lapisan keempat.

$\beta 4_m$  = besarnya bobot bias pada unit ke-m dalam lapisan keempat.

Kemudian menghitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $w_{km}$ ):  $\Delta W_{km} = \alpha \phi 4_m$

Juga menghitung koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $b_{3m}$ ):  $\Delta b_{3m} = \alpha \beta 4_m$

- f. Tiap-tiap unit output ( $Z_{bk}$ ) pada lapisan ketiga menerima target ( $T_k$ ) pola yang berhubungan dengan pola input pembelajaran, maka dapat dihitung informasi error dengan rumus:

$$\delta 3_k = (T_k - Z_{bk}) f'(z_{b\_in_k})$$

$$\phi 3_k = \delta 3_k Z_{bk}$$

$$\beta 3_k = \delta 3_k$$

dimana,

$(T_k - Z_{bk})$  = error antara target dengan output pada unit ke-k.

$f'(z_{b\_in_k})$  = fungsi turunan dari  $z_{a\_in_k}$

$\delta 3_k$  = besarnya delta input pada unit ke-k dalam lapisan ketiga.

$\phi 3_k$  = besarnya bobot input pada unit ke-k dalam lapisan ketiga.

$\beta 3_k$  = besarnya bobot bias pada unit ke-k dalam lapisan ketiga.

Kemudian menghitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $U_{jk}$ ):  $\Delta U_{jk} = \alpha \phi 3_k$

Juga menghitung koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $b_{2k}$ ):  $\Delta b_{2k} = \alpha \beta_{3k}$

- g. Tiap-tiap unit output ( $Z_{aj}$ ) pada lapisan kedua menerima target ( $T_j$ ) pola yang berhubungan dengan pola input pembelajaran, maka dapat dihitung informasi error dengan rumus:

$$\delta_{2j} = (T_j - Z_{aj}) f'(z_{a\_in_j})$$

$$\phi_{2j} = \delta_{2j} Z_{aj}$$

$$\beta_{2j} = \delta_{2j}$$

dimana,

$(T_j - Z_{aj})$  = error antara target dengan output pada unit ke-j.

$f'(z_{a\_in_j})$  = fungsi turunan dari  $z_{a\_in_j}$

$\delta_{2j}$  = besarnya delta input pada unit ke-j dalam lapisan kedua.

$\phi_{2j}$  = besarnya bobot input pada unit ke-j dalam lapisan kedua.

$\beta_{2j}$  = besarnya bobot bias pada unit ke-j dalam lapisan kedua.

Kemudian menghitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $V_{ij}$ ):  $\Delta V_{ij} = \alpha \phi_{2j}$

Juga menghitung koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $b_{1j}$ ):  $\Delta b_{1j} = \alpha \beta_{2j}$

- h. Tiap-tiap unit pada lapisan tersembunyi kedua ( $Z_{bk}$ ,  $k = 1, 2, \dots, 4$ ) menjumlahkan delta inputnya (dari unit-unit yang berada pada lapisan di atasnya):

$$\delta_{1\_in_k} = \sum_{m=1}^1 \delta_{4m} W_{km}$$

kemudian nilai ini dikalikan dengan turunan dari fungsi aktivasi *purelin* untuk menghitung informasi error:

$$\delta_{2k} = \delta_{1\_in_k} f'(z_{b\_in_k})$$

$$\phi_{2jk} = \delta_{2k} Z_{bk}$$

$$\beta_{2k} = \delta_{2k}$$

- i. Tiap-tiap unit pada lapisan tersembunyi pertama ( $Z_{aj}$ ,  $j = 1, 2, \dots, 10$ ) menjumlahkan delta inputnya (dari unit-unit yang berada pada lapisan di atasnya):

$$\delta_{in_j} = \sum_{k=1}^4 \delta_{2k} U_{jk}$$

kemudian nilai ini dikalikan dengan turunan dari fungsi aktivasi *logsig* untuk menghitung informasi error:

$$\delta_{1j} = \delta_{in_j} f'(z_{a\_in_j})$$

$$\phi_{1ij} = \delta_{1j} Z_{aj}$$

$$\beta_{1j} = \delta_{1j}$$

- j. Tiap-tiap unit output ( $Y_m$ ,  $m = 1$ ), memperbaiki bias dan bobotnya ( $m = 1$ ):

$$W_{km}(\text{baru}) = W_{km}(\text{lama}) + \Delta W_{km}$$

$$b_{3m}(\text{baru}) = b_{3m}(\text{lama}) + \Delta b_{3m}$$

- k. Tiap-tiap unit pada lapisan tersembunyi kedua ( $Z_{bk}$ ,  $k = 1, 2, \dots, 4$ ) memperbaiki bias dan bobotnya ( $j = 1, 2, \dots, 10$ ):

$$U_{jk}(\text{baru}) = U_{jk}(\text{lama}) + \Delta U_{jk}$$

$$b_{2k}(\text{baru}) = b_{2k}(\text{lama}) + \Delta b_{2k}$$

- l. Tiap-tiap unit pada lapisan tersembunyi pertama ( $Z_{aj}$ ,  $j = 1, 2, \dots, 10$ ) memperbaiki bias dan bobotnya ( $i = 1, 2, \dots, 5$ ):

$$V_{ij}(\text{baru}) = V_{ij}(\text{lama}) + \Delta V_{ij}$$

$$b_{1j}(\text{baru}) = b_{1j}(\text{lama}) + \Delta b_{1j}$$

Dari algoritma backpropagation di atas maka secara umum diperoleh rumusan output untuk tiap-tiap lapisan, yaitu:

1. Rumus output untuk lapisan kedua (lapisan tersembunyi pertama) adalah:

$$Z_{aj} = b_{1j} + \sum_{i=1}^n X_i V_{ij}$$

$Z_{aj}$  = nilai output unit neuron ke-j pada lapisan kedua (lapisan tersembunyi pertama).

$b_{1j}$  = bobot bias yang menghubungkan lapisan pertama dengan unit neuron ke-j pada lapisan kedua (lapisan tersembunyi pertama).

$I$  = data ke-i.

$n$  = panjang data.

$X_i$  = input data ke-i dari lapisan pertama (lapisan input).

$V_{ij}$  = bobot input data ke-i yang menghubungkan lapisan pertama dengan unit neuron ke-j pada lapisan kedua (lapisan tersembunyi pertama).

2. Rumus output untuk lapisan ketiga (lapisan tersembunyi kedua) adalah:

$$Z_{bk} = b_{2k} + \sum_{j=1}^p Z_{aj} U_{jk}$$

$Z_{bk}$  = nilai output unit neuron ke-k pada lapisan ketiga (lapisan tersembunyi kedua).

$b_{2k}$  = bobot bias yang menghubungkan lapisan kedua dengan unit neuron ke-k pada lapisan ketiga (lapisan tersembunyi kedua).

$j$  = data ke-j.

$p$  = panjang data.

$Z_{aj}$  = input data ke-j dari lapisan kedua (lapisan tersembunyi pertama).

$U_{jk}$  = bobot input data ke-j yang menghubungkan lapisan kedua (lapisan tersembunyi pertama) dengan unit neuron ke-k pada lapisan ketiga (lapisan tersembunyi kedua).

3. Rumus output untuk lapisan keempat (lapisan output) adalah:

$$Y_m = b_{3m} + \sum_{k=1}^q Z_{bk} W_{km}$$

$Y_m$  = nilai output unit neuron ke-m pada lapisan keempat (lapisan output).

$b_{3m}$  = bobot bias yang menghubungkan lapisan ketiga dengan unit neuron ke-m pada lapisan keempat (lapisan output).

$k$  = data ke-k.

$q$  = panjang data.

$Z_{bk}$  = input data ke-k dari lapisan ketiga (lapisan tersembunyi kedua).

$W_{km}$  = bobot input data ke-k yang menghubungkan lapisan ketiga (lapisan tersembunyi kedua) dengan unit neuron ke-m pada lapisan keempat (lapisan output).

Berdasarkan rumusan di atas dengan pengambilan data beban masa lalu 5 minggu sebelumnya, maka diperoleh suatu model prakiraan beban dengan algoritma backpropagation sebagai berikut :

- $Y_{th} = f(y_{in})$ ,  $Y_{th}$  adalah prakiraan pada jam t hari h.

$$y_{in} = b_3 + Z_{b1}W_1 + Z_{b2}W_2 + Z_{b3}W_3 + Z_{b4}W_4 = b_3 + \sum_{k=1}^4 Z_{bk}W_k$$

- $Z_{bk} = f(z_{b\_in_k}) ; (k = 1,2,\dots,4)$

$$z_{b\_in_k} = b_2 + Z_{a1}U_{1k} + Z_{a2}U_{2k} + Z_{a3}U_{3k} + Z_{a4}U_{4k} + Z_{a5}U_{5k} + Z_{a6}U_{6k} + Z_{a7}U_{7k} + Z_{a8}U_{8k} + Z_{a9}U_{9k} + Z_{a10}U_{10k} = b_2 + \sum_{j=1}^{10} Z_{aj}U_{jk}$$

- $Z_{aj} = f(z_{a\_in_j}) ; (j = 1,2,\dots,10)$

$$z_{a\_in_j} = b_1 + X_1V_{1j} + X_2V_{2j} + X_3V_{3j} + X_4V_{4j} + X_5V_{5j} = b_1 + \sum_{i=1}^5 X_{i(h-n)}V_{ij}$$

$$(i = 1,2,\dots,5)$$

Keterangan:

$Y_{th}$  = Prakiraan beban JST untuk jam t hari h.

$X_{i(h-n)}$  = Beban masa lalu ke-i pada 1 minggu sebelumnya, 2 minggu sebelumnya, dan seterusnya.

### 3. HASIL DAN PEMBAHASAN

Data beban yang akan dianalisis adalah data beban aktual 5 minggu sebelumnya yaitu untuk hari Senin mulai dari tanggal 11 Pebruari 2002 sampai tanggal 11 Maret 2002, dan akan membuat prakiraan untuk hari Senin tanggal 18 Maret 2002. Untuk hari-hari yang lain (selasa sampai minggu), data beban yang diambil mengikuti hari Senin tersebut di atas.

Selanjutnya hasil prakiraan beban yang diperoleh dari rumusan model tadi, akan dibandingkan dengan data beban aktualnya sehingga akan muncul error untuk setiap jam dan hari tertentu. Hasil perbandingan hasil prediksi dapat dilihat pada tabel di bawah ini :

**Tabel 1.** Perbandingan Error Model Prakiraan PLN dan Fuzzy

HARI	RATA-RATA ERROR (MW)		RATA-RATA ERROR (%)	
	PLN	Fuzzy	PLN	Fuzzy
SENIN	41.49	-4.17	2.45	1.07
SELASA	39.06	0.18	2.11	0.63
RABU	27.04	0.22	1.54	1.17
KAMIS	40.00	-0.06	2.43	0.86
JUM'AT	178.00	-0.16	11.65	0.57
SABTU	55.00	-0.85	3.47	0.78
MINGGU	65.00	0.06	3.88	2.45
<b>RATA-RATA</b>	<b>63.66</b>	<b>0.68</b>	<b>3.93</b>	<b>1.08</b>



**Tabel 2.** Perbandingan Error Model Prakiraan PLN dan JST

HARI	RATA-RATA ERROR (MW)		RATA-RATA ERROR (%)	
	PLN	JST	PLN	JST
SENIN	41.49	3.59	2.45	0.22
SELASA	39.06	3.20	2.11	0.19
RABU	27.04	4.60	1.54	0.27
KAMIS	40.00	3.29	2.43	0.24
JUM'AT	178.00	4.03	11.65	0.29
SABTU	55.00	4.49	3.47	0.29
MINGGU	65.00	5.42	3.88	0.35
<b>RATA-RATA</b>	<b>63.66</b>	<b>4.09</b>	<b>3.93</b>	<b>0.26</b>

Hasil prakiraan beban listrik dengan menggunakan *fuzzy subtractive clustering* telah dihasilkan dimana prakiraan dengan logika fuzzy lebih mendekati data aktualnya dan memberikan arti yang signifikan dibandingkan dengan metode koefisien PLN. Melalui perhitungan secara statistik didapatkan rata-rata error yang telah buat fuzzy sebesar 1.08 %. Tetapi hasil prediksi yang diolah melalui jaringan syaraf tiruan memberikan hasil yang lebih baik dengan error prediksi sebesar 0,26%.

#### 4. KESIMPULAN DAN SARAN

1. Dalam memprakirakan beban listrik jangka pendek selain metode yang telah ada, ternyata dapat pula digunakan model prakiraan dengan menggunakan model komputasi berbasis kecerdasan buatan, prediksi dengan menggunakan JST memberikan hasil terbaik jika dibandingkan dengan model Fuzzy ataupun PLN. Setelah dilakukan beberapa kali pengujian terhadap formula lapisan dan neuron yang berbeda-beda maka formula yang dianggap paling baik menurut penulis adalah 5 – 10 – 4 – 1.
2. Kelebihan Logika Fuzzy dan Jaringan Syaraf Tiruan terletak pada kemampuan belajar yang dimilikinya. Dengan kemampuan tersebut pengguna tidak perlu merumuskan kaidah atau fungsinya. Dengan demikian logika fuzzy ataupun JST mampu digunakan untuk menyelesaikan masalah yang rumit dan atau masalah yang terdapat kaidah atau fungsi yang tidak diketahui (seperti prakiraan beban listrik).
3. Melalui perhitungan secara simulasi didapatkan rata-rata error keseluruhan untuk ketiga metode metode, yaitu 3.93 % untuk PLN dan 1.08 % untuk fuzzy. Dengan demikian dapat disimpulkan bahwa prakiraan beban listrik jangka pendek dengan menggunakan *fuzzy subtractive clustering* lebih baik dibandingkan dengan prakiraan beban listrik dengan menggunakan metode koefisien yang dilakukan oleh PLN, akan tetapi hasil tersebut masih jauh dari prakiraan JST (prakiraan oleh JST 0.26 %).

#### DAFTAR PUSTAKA

Arismunandar, A. (1990). *Teknik Tenaga Listrik (Jilid III)*. Jakarta: Pradnya Paramita.

- A. Bakirtzis, S. Kiartzis, V. Petridis and Ath. Kehagias (1997) :”Short Term Load Forecasting using a Bayesian Combination Method”, *International Journal of Electrical Power and Energy Systems*, Vol. 19, pp.171-177.
- A.D.P.Lotufo dan C. R. Minussi (1999) “Electric Power Systems Load Forecasting: A Survey”, Paper BPT99–028–25 accepted for presentation at the IEEE Power Tech ’99 Conference, Budapest, Hungary.
- A. Sfetsos(2003), “Short-term Load Forecasting with a Hybrid clustering algorithm” *IEEE Proc.- Gener, Transm. Dsitrib.*, Greece, vol.150, pp. 257-262.
- Azzam Ul ASAR, Syed Riaz ul HASSNAIN dan Amjad Ullah KHATTACK (2005), “A Multi-Agent Approach To Short Term Load Forecasting Problem, *International Journal of Intellegent Control and Systems*”, Vol 10, No 1, March 2005, p 55-59
- Box, G.G. Jenkins, G.M., Holde n-day (1976), “*Time series analysis- forecasting and control*”, San Francisco.
- Demuth, H. dan Mark B. (1994). *Neural Network TOOLBOX For Use with MATLAB*. USA: Mathwork, Inc.
- Drezga, S. Rehman (1999), “STLF with Local ANN Predictors”, *IEEE Transactions on Power Systems*, Vol 14, No 3.
- Dipti Srinivasan, Swee Sien Ten and C S.Chang (1999), “Parallel Neural Network – Fuzzy expert system strategy for STLF: System implementation and performance evaluation”, *IEEE Transactions on Power Systems*, vol 14, pp. 1100–1106.
- Du Tao and Wang Xiuli (2002), “Combined model of wavelet and NN for STLF”, *Proceedings of IEEE International Conference on Power System Technology*, vol. 4, pp. 2331-2335.
- Hiroyuki Mori and Noriyuki Kosemura (2001), “Optimal Regression Tree Based Rule Discovery for Short Term Load Forecasting”, *PES IEEE Winter Meeting*, Columbus, OH, vol.2 pp. 421-426.
- Ho, K.L. and Hsu, Y.Y. (1992, Pebruari).*Short Term Load Forecasting using a Multilayer Neural Network with an Adaptive Learning Algorithm*. [IEEE Tr. On Power System], Vol. 7, No. 1. Halaman 9.
- Kusumadewi, S. (2003). *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.
- Kusumadewi, S. (2004). *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab dan Excel Link*. Yogyakarta: Graha Ilmu.
- Kusumadewi, Sri. *Analisis dan Desain Sistem Fuzzy Menggunakan Toolbox Matlab*. Yogyakarta: Graha Ilmu. 2002.
- Kun-Long Ho, Yuan-Yih Hsu, Chien –Cluen Yang (1992), “Short term load forecasting using multiplayer neural network with an adaptive learning algorithm”., *IEEE Trans. On power systems*, Vol.7,No.1.
- Kab Ju Hwang (2001), “A STLF Expert System”, *Proceedings of 5<sup>th</sup> Russian-Korean IEEE International Symposium on Science and Technology*, Tomsk, Russia, vol.1, pp. 112-116.

- Kwang –Ho Kim , Hyoung Sun Youn and Yong Cheol kang (2000), “ STLF for Special Days in Anomalous Load Conditions Using Neural Networks and Fuzzy Inference Method”. *IEEE Transactions on Power Systems*, vol. 15, pp. 559-565.
- Lee, Y. K. and Cha, Y.T. (1992, Pebruari). *Short Term Load Forecasting Using An Artificial Neural Network*. [IEEE Tr. On Power System], Vol. 7, No. 1. Halaman 7.
- L. C. Jain and N. M. Martin (1999), “*Fusion of neural networks, fuzzy sets, and genetic algorithms: Industrial applications*”, CRC Press.
- L. Rutkowski (2004), “*Flexible neuro-Fuzzy systems*”, Kluwer Academic Publishers.
- Marsudi, D. (1990). *Operasi Sistem Tenaga Listrik*. Jakarta: Balai Penerbit dan Humas ISTN.
- Muhammad Riaz Khan dan Ajith Abraham (2002) “Short Term Load Forecasting Models in Czech Republic Using Soft Computing Paradigms” ,Department of Computer Science, Oklahoma State University, Tulsa.
- Ma-WenXiao, Bai-XiaoMin and Mu-LianShun (2002), “Short-term Load Forecasting With Artificial Neural Network and Fuzzy Logic”, *Proceedings of the IEEE international Conference on Power System Technology*, vol.2, pp.1101-1104, Oct.
- M.A. Abu El Magd and R.D. Findlay(2003), “New approach using ANN and Time Series Models for STLF”, *IEEE Canadian Conference on elect. and Comp. Engg.*, Canada, vol.3, pp. 1723-1726,
- P. Stavroulakis (2004), “*Neuro-Fuzzy and Fuzzy Neural Applicaiions in telecommunications*”, Springer-Verlag.
- P.K Dash , S. Dash dan S.Rahman (1993) “A Fuzzy Adaptive Correction Scheme For Short Term Load Forecasting Using Fuzzy Layered Neural Network”, IEEE, pp 132-147.
- P.K.Dash , H.P.Satpathy dan S.Rahman (1995) “Short Term Daily Average and Peak Load Predications Using A Hybrid Intelligent Approach”, IEEE, pp 565-570
- R. Khosla and T. Dillon (1997), “*Engineering Intelligent Hybrid Multi-agent Systems*”, Kluwer Academic Publishers.
- Russell, S., and Norvig P(2003),”*Artificial Intelligence – A modern Approach*, Printice Hall, New Jersey, USA.
- Stefan E Karman and Evileno Jgonzalus(1997), “*STLF using Fuzzy Artmap N N*”. University of Central Florida.
- S.H. Ling, H.K. Lam, F.H.F. Leung and P.K.S.Tam (2002), “A Novel GA- Based Neural Network For Short-term Load Forecasting”, *Proceedings of the IEEE Joint international Conference on Neural Networks*, Honolulu, HI, USA, vol.3, pp. 2761-2766.
- S.H. Ling, H.K. Lam, F,H.F. Leung and P.K.S. Tam (2001), “Neural Fuzzy network with optimal Number of rules for STLF in Intelligent Home” *IEEE International Conference on Fuzzy Systems*.