

PENJADWALAN PREVENTIVE MAINTENANCE AIR CONDITIONING (AC) DENGAN MENGGUNAKAN METODE BACKPROPAGATION PADA KOMPLEKS PERUMAHAN (STUDI KASUS DI PERUMAHAN PT. CHEVRON PACIFIC INDONESIA)

Zulfi Azhar

Program Studi Sistem Informasi, STMIK Royal Kisaran

Jl. Prof. M. Yamin 173 Kisaran, Sumatera Utara 21222

Telp : (0623) 41079

E-mail : zulfi_azhar@yahoo.co.id

Abstrak

Perusahaan PT. Chevron Pacific Indonesia (CPI) mempunyai karyawan yang sebagian besar pegawainya tinggal di perumahan. Perusahaan membuat suatu program fasilitas pelayanan pada perumahan untuk memberikan pelayanan yang terbaik bagi para karyawan dan keluarganya agar mereka dapat menikmati dan tinggal di perumahan dengan nyaman. Penggunaan AC yang baru akan berjalan dengan baik dan lancar disertai dengan program pemeliharaan rutin dan pemeliharaan secara berkala (preventive schedule). Sementara pemeliharaan AC secara tidak berkala (preventive unschedule) juga dilakukan apabila ada kerusakan. Dalam melakukan pemeliharaan secara tidak berkala tersebut membutuhkan beberapa kriteria yang diambil dari beberapa laporan kerusakan yang ditemukan di lapangan. Kriteria kerusakan tersebut merupakan sebagai data dalam penjadwalan secara tidak berkala. Penjadwalan AC preventive schedule dan preventive unschedule digunakan sebagai target, yang salah satu diantaranya merupakan hasil akhir yang akan digunakan untuk menentukan penjadwalan yang selanjutnya. Untuk menentukan pemeliharaan AC penjadwalan secara berkala dan secara tidak berkala membutuhkan penggunaan metode backpropagation untuk menganalisa dan menguji dari data laporan dan kriteria kerusakan.

Kata Kunci : JST, data penjadwalan pemeliharaan secara berkala, data pemeliharaan bukan secara berkala dan Algoritma Backpropagation

1. PENDAHULUAN

PT. Chevron Pacific Indonesia (CPI) merupakan Perusahaan minyak yang beroperasi di Riau yang mempunyai karyawan lebih dari 1000 orang yang sebagian besar pegawainya tinggal di perumahan yang disediakan oleh Perusahaan. Perusahaan CPI membuat suatu program fasilitas pelayanan pada perumahan untuk memberikan pelayanan yang terbaik bagi para pekerjanya agar karyawan dan keluarganya dapat menikmati dan tinggal di perumahan dengan nyaman. Kebutuhan untuk mendapatkan pelayanan yang lebih baik kepada para pegawai PT. CPI maka pihak Management melakukan proyek penyediaan AC yang optimal dengan memasang AC yang baru pada kompleks perumahan. Proyek penyediaan AC baru diprogramkan sebagai fasilitas pelayanan kepada para pegawai dan keluarganya yang tinggal di kompleks perumahan. Penggunaan AC yang baru akan berjalan dengan baik dan lancar disertai dengan program pemeliharaan rutin dan secara berkala (*preventive schedule*). Pemeliharaan AC secara rutin dan berkala akan dapat menjaga pemakaian AC yang optimal dan mengurangi biaya kerusakan dan pemborosan. Penggunaan AC yang salah dan pemeliharaan AC yang tidak dilakukan

dengan benar akan membuat biaya yang lebih tinggi dalam perbaikan. Dengan melakukan pencegahan dalam pemeliharaan AC akan dapat mencegah kerusakan dan dapat mempertahankan kenyamanan yang optimal pada penghuni perumahan. Dan pelaksanaan pemeliharaan AC merupakan salah satu pencegahan dari kerusakan dan penggantian *spare part* yang tidak semestinya terjadi. Dari pengertian permasalahan tersebut maka dapat disimpulkan bahwa ada pemeliharaan AC secara berkala dan ada pemeliharaan AC secara tidak berkala berdasarkan kriteria yang diambil dalam menentukan penjadwalan pemeliharaan (*maintenance*) AC. Penggunaan metode *backpropagation* sangat tepat dalam mengetahui hasil berdasarkan target dan kriteria kerusakan yang diperoleh dalam pelaksanaan di lapangan.

Berdasarkan latar belakang yang telah dijelaskan diatas, maka dapat disimpulkan ada beberapa rumusan permasalahan yaitu :

1. Bagaimana hubungan antara penjadwalan AC secara berkala dan secara tidak berkala dalam menentukan penjadwalan ?
2. Bagaimana mendapatkan hasil (output) dan target yang menentukan penjadwalan secara berkala atau pemeliharaan secara tidak berkala ?

3. Bagaimana membangun model jaringan syaraf tiruan untuk mendapatkan target akhir dari data kerusakan dan pemeliharaan secara berkala ?

Program pelaksanaan untuk pemeliharaan AC (*preventive schedule*) secara berkala telah dilakukan oleh Pihak Manajemen CPI setiap dalam 3 bulan. Dalam setiap 3 bulan selalu diadakan pemeliharaan sedangkan untuk pemeliharaan secara tidak berkala dilakukan apabila ada kerusakan. Jadwal tersebut sudah dilakukan setiap tahunnya tanpa adanya perubahan. Dalam hal ini masalah yang ditinjau adalah berdasarkan laporan yang diambil dari :

1. Pemeliharaan AC yang berdasarkan dari laporan kerusakan AC dari pihak penghuni rumah.
2. Pemeliharaan AC berdasarkan temuan yang sering terjadi dari pihak pengawas teknisi dilapangan.
3. Pemeliharaan AC ini dilakukan pada perumahan (*housing*) yang ada penghuninya di dalam Perumahan CPI.

Adapun tujuan dari penelitian ini adalah :

1. Untuk mengetahui kriteria apa saja yang mempengaruhi dalam penggunaan dan pemeliharaan AC tersebut.
2. Menganalisa jadwal pemeliharaan AC yang sudah dilaksanakan untuk mengetahui penentuan penjadwalan pemeliharaan AC secara berkala (*preventive schedule*) atau bukan secara berkala (*preventive unscheduled*).
3. Membangun jaringan syaraf tiruan dengan menggunakan metode *Backpropagation* untuk mendapatkan target akhir dari penjadwalan pemeliharaan AC.
4. Menguji Jaringan syaraf tiruan dengan menggunakan *software Matlab*.

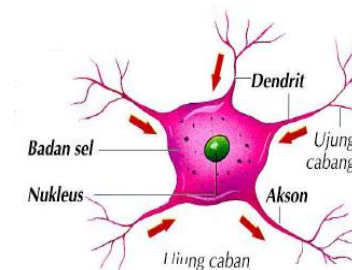
Hasil penelitian ini akan bermanfaat untuk memberikan kemudahan bagi pihak Manajemen (Pengelola) untuk membuat penjadwalan pemeliharaan AC secara berkala atau secara tidak berkala berdasarkan kriteria kerusakan dan data pemeliharaan di perumahan. Sehingga pelaksanaan program fasilitas pelayanan di perumahan lebih baik dan dapat memberikan kenyamanan kepada karyawannya yang tinggal di Perumahan.

2. TINJAUAN TEORI

2.1 Jaringan Syaraf Tiruan

Jaringan saraf tiruan (JST) atau *Artificial Neural Network* (ANN) disebut juga *Simulated Neural Network* (SNN), atau biasanya hanya disebut *Neural Network* (NN), adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan saraf manusia. JST merupakan sistem adaptif yang dapat merubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut. Secara sederhana, JST adalah sebuah alat pemodelan data statistik *non-linier*. JST dapat digunakan untuk memodelkan hubungan yang kompleks antara *input* dan *output* untuk menemukan pola-pola pada data.

Komponen Jaringan Syaraf ada beberapa tipe jaringan syaraf, namun demikian, hampir semuanya memiliki komponen-komponen yang sama. Seperti halnya otak manusia, jaringan saraf juga terdiri dari beberapa *neuron* dan ada hubungan antara *neuron-neuron* tersebut. *Neuron-neuron* tersebut akan mentransformasikan informasi yang di terima melalui sambungan keluarnya menuju ke *neuron-neuron* yang lain. Pada jaringan saraf, hubungan ini dikenal dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut.



Gambar 1. Komponen Neuron

2.2 Arsitektur jaringan Syaraf Tiruan

JST yang telah dan sedang dikembangkan merupakan pemodelan matematika dari jaringan saraf, berdasarkan asumsi :

Pemrosesan informasi terjadi pada banyak elemen pemroses sederhana yang disebut *neuron*.

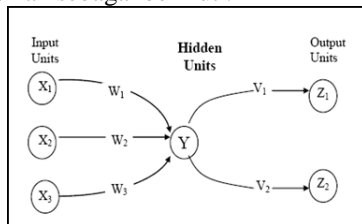
- a) Sinyal dilewatkan antar *neuron* yang membentuk jaringan *neuron*.
- b) Setiap elemen pada jaringan *neuron* memiliki 1 (satu) pembobot.

- c) Sinyal yang dikirimkan ke lapisan *neuron* berikutnya adalah info dikalikan dengan pembobot yang bersesuaian.
- d) Setiap *neuron* mengerjakan fungsi aktivasi untuk mendapatkan nilai *output*nya.

Karakteristik dari JST secara umum adalah :

1. Arsitektur (pola koneksi antar *neuron*)
2. Pelatihan/ pembelajaran
3. Fungsi aktivasi

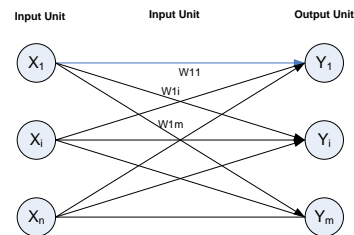
Suatu jaringan *neuron* sederhana digambarkan sebagai berikut :



Gambar 2 Neural Network yang sederhana

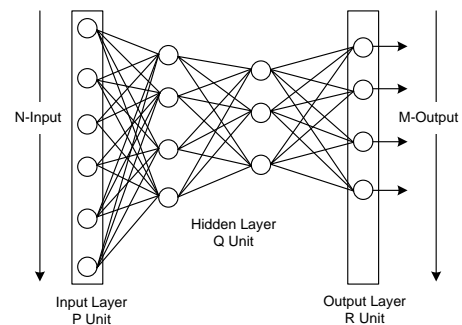
Pada Gambar 2 merupakan suatu model *neural network* yang sangat sederhana yang terdiri dari unit-unit *input*, *output* dan satu unit *hidden* yang terletak pada suatu lapisan tersembunyi (*hidden layer*). Susunan dari *neuron-neuron* di dalam layer dan bentuk hubungan di dalam dan diantara layer disebut *net architecture* (arsitektur jaringan). *Neural net* pada umumnya diklasifikasikan berdasarkan pada jumlah *layer* (*input unit* tidak dihitung sebagai *layer*, karena tidak mengerjakan fungsi aktivasi).

Jumlah dari *layer* pada *net* dapat didefinisikan menjadi jumlah *layer* dari pembobotan interkoneksi (*links*) diantara *neuron-neuron*, karena pembobot pada suatu *net* berisi info yang amat penting. Sebuah *net* yang memiliki satu *layer* dari hubungan pembobot-pembobotnya, diantara unit-unit X_1 , X_2 , dan X_3 serta W_1 , W_2 , dan W_3 merupakan pembobot antar koneksi dari X_1 , X_2 , X_3 terhadap *neuron* Y . Jaringan *input* y_{in} pada *neuron* Y merupakan penjumlahan dari perkalian *neuron input* dengan pembobot yang bersesuaian (Gambar 3), sedangkan sebuah *net* dengan satu atau lebih *hidden layer* diantara unit-unit *input* dengan unit-unit *output* dikenal dengan *multilayer neural network* (Gambar 4). Untuk beberapa kasus, mungkin *multi layer* lebih menguntungkan, tetapi pada umumnya dengan satu *layer* saja sudah memadai untuk menyelesaikan berbagai masalah.



Gambar 3. Single Layer Neural Network

Input dengan unit-unit *output* disebut *single-layer neural network* (Gambar 3). Sedangkan, sebuah *net* dengan satu atau lebih *hidden layer* diantara unit-unit *input* dengan unit-unit *output* dikenal dengan *multi layer neural network* (Gambar 4). Untuk beberapa kasus, mungkin *multi layer* lebih menguntungkan, tetapi pada umumnya dengan satu *layer* saja sudah memadai untuk menyelesaikan berbagai masalah.



Gambar 4. Multi Layer Neural Network

2.3 Jaringan Saraf Tiruan Propagasi Balik

Metode Neural Network atau Jaringan Saraf Tiruan (JST) yang akan digunakan pada tugas akhir ini menggunakan algoritma *backpropagation*. Aturan belajar algoritma ini adalah menggunakan *error* atau ketidaksesuaian *output* dengan *target* untuk koreksi bobotnya. bobot di koreksi sampai *error* dapat diterima (memenuhi toleransi yang kita berikan) atau sampai dengan jumlah *epoch* tertentu.

Prosedur pengajaran atau pembentukan bobot-bobot yang digunakan adalah sebagaimana yang digunakan dalam pengajaran jaringan yang bersifat *supervised learning* (pengajaran yang menggunakan *target*). Sehingga aturan ini memerlukan pasangan *output* untuk tiap *input* yang akan diajarkan.

Dengan keadaan bobot awal *random*, tiap *input* dilewatkan ke bobot tersebut dan di hasilkan *output* untuk saat itu. *Output* tersebut di bandingkan dengan *target* yang diinginkan. Besar perbedaan yang terjadi digunakan sebagai faktor

pengubah pembobot yang menghubungkan *input* dengan *output* tersebut (*Update Weight*). Sehingga, dengan bobot yang baru akan mengarahkan *output* ke target yang seharusnya. Proses perubahan bobot berdasarkan *error* ini dilakukan terus sampai *output* yang di hasilkan sesuai dengan yang di targetkan, atau mempunyai *error* yang dapat diterima.

Logika umumnya adalah sebuah metode untuk meminimalisasi *error* rata-rata total (*gradient descent*), yang dihitung oleh net (jaringan *neuron*). Tujuannya, yaitu melatih sistem jaringan untuk mencapai suatu keseimbangan dalam merespon secara benar model *input* yang telah dilatihkan dan kemampuan untuk memberikan respon yang masuk akal bagi *input* yang mirip tetapi tidak identik dengan *input* pada saat pembelajaran. Pada proses pengajaran, diperlukan semua pola data *input* yang akan diajarkan dan target yang telah di tentukan sebelumnya. Setiap pola yang dimasukkan akan diolah dan diproses melalui bobot yang ada, dan hasilnya dibandingkan dengan data target yang diinginkan, kemudian dihitung *error*-nya (ketidaksamaan hasil saat itu dengan hasil yang diinginkan). Dimana, *error* tersebut diumpanbalikkan (*backpropagation*) ke bobot yang menghubungkan *layer* tersebut sebagai sinyal koreksi bobot, agar dengan bobot yang baru *error*nya berkurang sampai dengan harga yang diterima.

Prinsip algoritma *backpropagation* memiliki 3 fase, yaitu:

1. Fase feedforward pada pola input pembelajaran.
2. Fase kalkulasi dan *backpropagation error* yang didapat.
3. Fase penyesuaian bobot.

Arsitektur yang digunakan adalah jaringan *perceptron* lapis banyak (*multi layer perceptron*), hal ini merupakan generalisasi dari arsitektur jaringan *perceptron* lapis tunggal. Secara umum, algoritma jaringan ini memerlukan waktu pembelajaran yang memang lambat, namun setelah pembelajaran selesai, aplikasinya akan memberikan *output* yang sangat cepat dikarenakan faktor pembobot yang lebih baik.

2.4 Pembelajaran dan Pelatihan

Pada jaringan saraf tiruan belajar adalah proses pembentukan konfigurasi harga-harga bobot dari jaringan. Pembentukan ini memiliki tujuan akhir agar *input* yang diberikan padanya akan direspon melalui bobot-bobot tersebut sehingga menghasilkan *output* yang sesuai dengan *target* atau mendekati, untuk *input* yang bersangkutan. Terdapat banyak cara di dalam proses pembelajaran dari JST, secara umum dapat dikategorikan dalam 2 jenis

proses, yaitu :

a. Supervised training

Pada tipe belajar dengan pengawasan, tiap pola *input* memiliki pola *target*. Sehingga pada pembelajaran tipe ini, masing-masing *input* memiliki pasangan *output* yang bersesuaian.

b. Unsupervised training

Algoritma ini tidak membutuhkan vektor *target* untuk keluarannya, sehingga tidak ada perbandingan untuk menentukan respon yang ideal. Langkah-langkah Algoritma a. Algoritma Pelatihan

Pelatihan suatu jaringan dengan algoritma *backpropagation* meliputi 2 tahap : perambatan maju dan perambatan mundur.

b. Prosedur Pelatihan

Langkah 0 : Inisialisasi bobot. (sebaiknya diatur pada nilai acak yang kecil),

Langkah 1 : Jika kondisi tidak tercapai, lakukan langkah 2-9,

Langkah 2 : Untuk setiap pasangan pelatihan, lakukan langkah 3-8,

Perambatan Maju :

Langkah 3 : Tiap unit masukan ($x_i, i = 1, \dots, n$) menerima sinyal x_i dan menghantarkan sinyal ini ke semua unit lapisan di atasnya (unit tersembunyi),

Langkah 4 : Setiap unit tersembunyi ($x_i, i = 1, \dots, p$) jumlahkan bobot sinyal masukannya,

$$y_in_k = w_{ok} + \sum_{j=1}^n z_j w_{jk} \quad \text{wok}$$

= bias pada unit keluaran k dan aplikasikan fungsi aktivasinya untuk menghitung sinyal keluarannya, $y_k = f(y_in_k)$.

Perambatan Mundur :

Langkah 6 : Tiap unit keluaran ($y_k, k = 1, \dots, m$) menerima pola target yang saling berhubungan pada masukan pola pelatihan, hitung kesalahan informasinya,

$$\delta_k = (t_k - y_k) f'(y_in_k) \quad \text{hitung}$$

koreksi bobotnya (digunakan untuk memperbaharui wjk nantinya),

$$\Delta w_{jk} = \alpha \delta_k z_j \quad \text{hitung koreksi}$$

biasnya (digunakan untuk memperbaharui wok nantinya), dan kirimkan δ_k ke unit-unit pada lapisan dibawahnya, Langkah 7 : Setiap unit lapisan tersembunyi ($z_j, j = 1, \dots, p$) jumlahkan hasil perubahan masukannya (dari unit-unit lapisan diatasnya),

$$\Delta_in_j = \sum_{k=1}^m \delta_k w_{jk} \quad \text{kalikan dengan}$$

turunan fungsi aktivasinya untuk menghitung informasi kesalahannya,

$$\delta_j = \delta_{in_j} f'(z_{in_j})$$

hitung

koreksi bobotnya (digunakan untuk memperbaharui voj nanti), Langkah 8 : Tiap unit keluaran ($y_k, k = 1, \dots, m$) *update* bias dan bobotnya ($j = 0, \dots, p$) :

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

Tiap

unit lapisan tersembunyi ($z_j, j = 1, \dots, p$) *update* bias dan bobotnya ($i = 0, \dots, n$) :

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

Langkah 9 : Test kondisi berhenti.

PROSEDURE PENGUJIAN : Setelah pelatihan, jaringan saraf *backpropagation* diaplikasikan dengan hanyamenggunakan tahap perambatan maju dari algoritma pelatihan. Prosedur aplikasinya adalah sebagai berikut :

Langkah 0 : Inisialisasi bobot (dari algoritma pelatihan).

Langkah 1 : Untuk tiap vektor masukan, lakukan langkah 2-4.

Langkah 2 : for $i = 1, \dots, n$: atur aktivasi unit masukan x_i .

Langkah 3 : for $j = 1, \dots, p$:

$$z_{in_j} = v_{oj} + \sum_{i=1}^n x_i v_{ij}$$

$$z_j = f(z_{in_j})$$

Langkah 4

: for $k = 1, \dots, m$:

$$y_{in_k} = w_{ok} + \sum_{j=1}^p z_j w_{jk}$$

$$y_k = f(y_{in_k})$$

Langkah 5

: Jika $y_k \geq 0,5$ maka $y_k = 1$, else $y_k = 0$.

3. METODOLOGI PENELITIAN

Metode penelitian yang dilakukan untuk penelitian ini adalah :

- Identifikasi Masalah
- Studi Literatur
- Pengumpulan Data
- Pengolahan Data
- Pengujian (Evaluasi)

Dalam tahap ini dilakukan pengujian dengan aplikasi *software Matlab* untuk menganalisa tingkat kesalahan yang dihasilkan dalam pengujian ini.

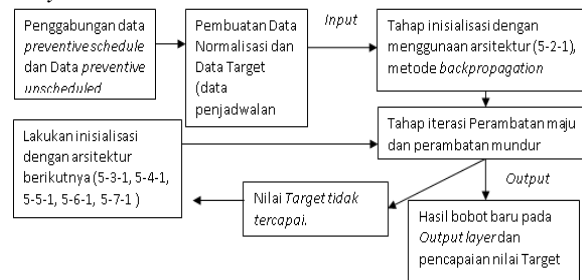
Mekanisme pengujian dilakukan dalam beberapa tahap, yaitu :

- Tahap penentuan Kriteria Konsentrasi
- Tahap penentuan lapisan *input*, lapisan tersembunyi dan lapisan *output*
- Tahap pelatihan data (*training*) dan pengujian data (*testing*)
- Tahap dokumentasi

4. ANALISA dan PENGUJIAN

4.1 Analisa Data

Dalam analisa data ini ada beberapa tipe perumahan yang diambil data pemeliharaannya. Data ini sebagai data yang sudah dilakukan pemeliharaan AC, yang merupakan sebagai data masukan (*input*). Data yang pertama berdasarkan jadwal pemeliharaan secara berkala (*preventive schedule*) dan data yang kedua berdasarkan bukan jadwal pemeliharaan secara tidak berkala. Kemudian data tersebut dinormalisasikan dengan memberi target akhir untuk nilai 1 adalah Penjadwalan *preventive schedule* dan nilai 0 untuk penjadwalan *preventive unscheduled*. Dengan mempergunakan metode *backpropagation* maka akan diperoleh target akhirnya 1 atau 0 dengan menggunakan *input layer*, *hidden layer* dan *output layer*.



Gambar 5. Diagram Data Penjadwalan

Data kriteria permasalahan yang ditemui di lapangan dalam penjadwalan AC secara tidak berkala yaitu :

- Not cold* (NC), penanganannya permasalahannya yaitu : ganti bearing dan kapasitor *indoor/outdoor*, ganti *module outdoor*, tambah gas Freon kencangkan *neple* dan *check* ampere, perbaikan pada sensor *remote indoor*, perbaikan kabel *power* dan bersihkan es dari *indoor* unit, ganti fan *motor indoor* dan kapasitor *fan indoor*, ganti kapastor sistem A,B dan C.
- Noise* (N), penanganan permasalahannya, yaitu : perbaikan dan *flashing drainase indoor*, ganti *bearing module indoor*, ganti *kapasitor indoor / outdoor*, *check* kebocoran, perbaikan kedudukan pada *lifer indoor* unit.
- Not work* (NW), penanganan permasalahannya, yaitu : penyemenan lubang *indoor*, ganti sensor *remote* perbaikan *module indoor / outdoor*, flaring ulang tubing di *condensing*, ganti bearing, ganti *kapasitor indoor/outdoor*.
- Waterleak* (WL), penanganan permasalahannya, yaitu : perbaikan *drainase indoor/outdoor*, cuci *filter indoor, flashing drain indoor*, ganti bearing dan *kapasitor indoor*.

5. Icing (I), penanganan permasalahannya, yaitu :
buang es di *evaporator relay kontektor*, check kebocoran, perbaikan *nepel outdoor*, perbaikan *modul indoor unit*.

Tabel.1 Data penjadwalan AC secara Berkala (preventive scedule)dalam setahun

N o	Description	Perumahan				
		Leuser	Merapi	Seulawah	Sibayak	Talang
1	Jumlah rumah	18	58	51	47	27
2	Jumlah Service rutin setahun	58	133	131	125	17
3	Laporan Service pada kerusakan setahun	62	167	161	127	17

Tabel 2. Data Gabungan Pemeliharaan AC secara berkala (preventive schedule) dan bukan secara berkala (preventive unschedule)

N o	Nama	Perumahan				
		Leuser	Merapi	Seulawah	Sibayak	Talang
1	Januari 2012	6	15	21	14	0
2	Pebruari 2012	17	29	22	12	3
3	Maret 2012	7	15	28	27	3
4	April 2012	9	17	19	22	3
5	Mei 2012	12	25	34	18	5
6	Juni 2012	18	30	22	26	0
7	Juli 2012	10	31	36	33	17
8	Agustus 2012	3	19	19	9	0
9	Septemb er 2012	13	29	28	28	2
10	Oktober 2012	10	31	32	31	1
11	Nopemb er 2012	8	27	16	11	0
12	Desembe r 2012	7	32	15	21	0

Tabel 3. Data Normalisasi

N o	Nama	Perumahan				
		Leuser	Merapi	Seulawah	Sibayak	Talang
1	Januari 2012	0,26	0,1	0,33	0,27	0,1
2	Pebruari 2012	0,85	0,76	0,37	0,2	0,24
3	Maret 2012	0,31	0,1	0,6	0,71	0,24
4	April 2012	0,42	0,19	0,25	0,53	0,24

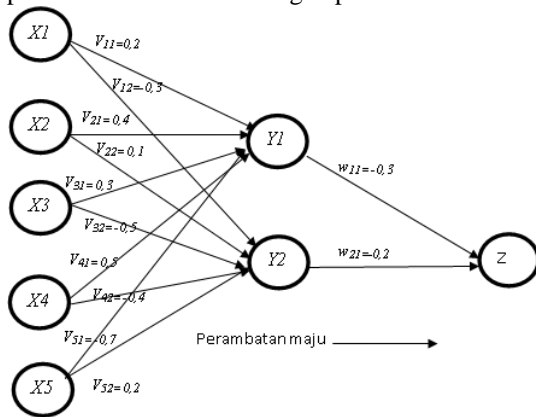
5	Mei 2012	0,58	0,57	0,82	0,41	0,34
6	Juni 2012	0,9	0,81	0,37	0,67	0,1
7	Juli 2012	0,47	0,85	0,9	0,9	0,9
8	Agustus 2012	0,1	0,29	0,25	0,1	0,1
9	Septemb er 2012	0,63	0,76	0,6	0,7	0,19
10	Oktober 2012	0,47	0,85	0,75	0,83	0,15
11	Nopemb er 2012	0,37	0,66	0,14	0,17	0,1
12	Desembe r 2012	0,31	0,9	0,1	0,5	0,1

Data target adalah 1 menunjukkan konsentrasi pemeliharaan secara berkala(*Preventive schedule*) sedangkan 0 menunjukkan konsentrasi pemeliharaan bukan secara berkala (*Preventive unscheduled*)

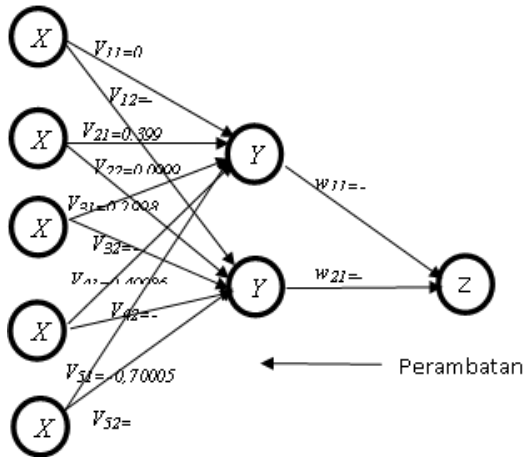
Tabel 4. Data Target

N o	Bulan	Perumahan					
		Leuser	Merapi	Seulawah	Sibayak	Talang	Target
1	Januari 2012	0,26	0,1	0,33	0,27	0,1	1
2	Pebruari 2012	0,85	0,76	0,37	0,2	0,24	0
3	Maret 2012	0,31	0,1	0,6	0,71	0,24	0
4	April 2012	0,42	0,19	0,25	0,53	0,24	1
5	Mei 2012	0,58	0,57	0,82	0,41	0,34	0
6	Juni 2012	0,9	0,81	0,37	0,67	0,1	0
7	Juli 2012	0,47	0,85	0,9	0,9	0,9	1
8	Agustus 2012	0,1	0,29	0,25	0,1	0,1	0
9	Septemb er 2012	0,63	0,76	0,6	0,7	0,19	0
10	Oktober 2012	0,47	0,85	0,75	0,83	0,15	1
11	Nopemb er 2012	0,37	0,66	0,14	0,17	0,1	0
12	Desembe r 2012	0,31	0,9	0,1	0,5	0,1	0

Korelasi hasil jaringan syaraf tiruan terhadap penentuan konsentrasi dengan pola 5-2-1.



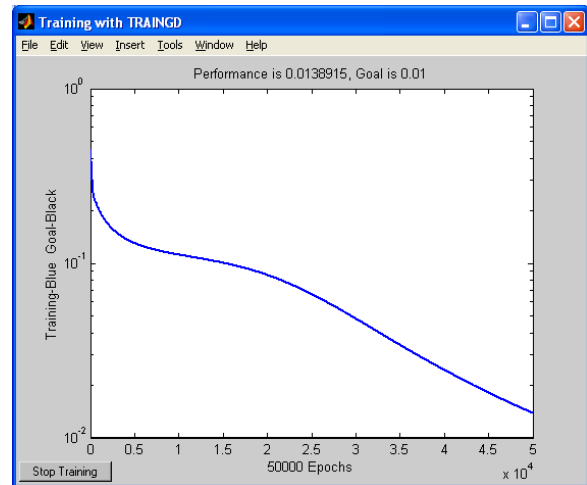
Gambar 6. Perambatan Maju



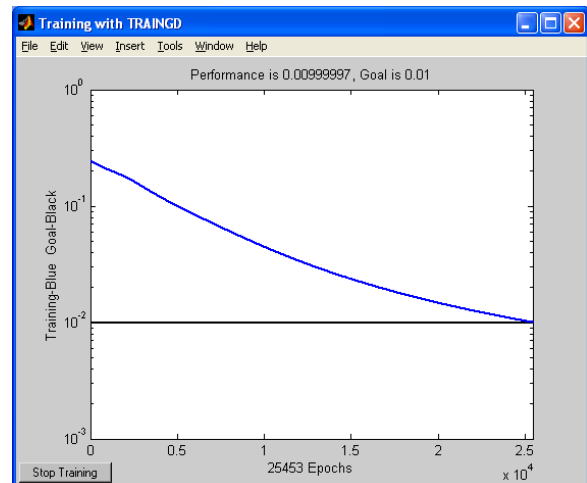
Gambar 7. Perambatan Mundur

4.2. Pengujian

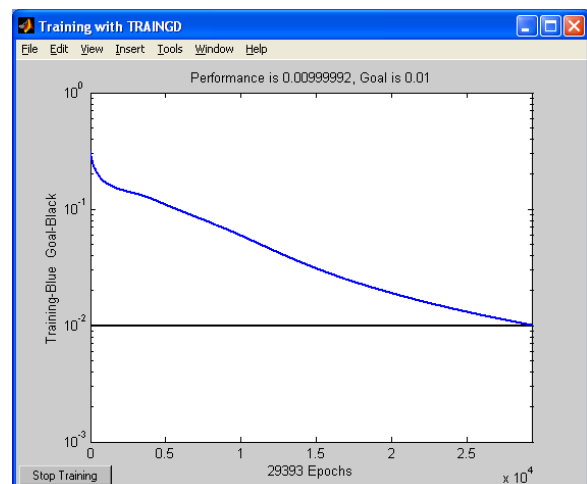
Proses pengujian dilakukan dengan menggunakan *software* aplikasi *Matlab 6.1*. Model Jaringan Syaraf Tiruan yang digunakan adalah 5-3-1, model 5-4-1, model 5-5-1, model 5-6-1 dan model 5-7-1. Untuk hasil perbandingan pengujian yang lebih rinci dapat dilihat pada tabel 8. Dari kelima model yang digunakan penulis dalam proses pengujian sistem Jaringan Syaraf Tiruan dengan algoritma backpropagation dengan menggunakan *Matlab 6.1* dengan sampel data sebanyak 12 data dalam waktu 12 bulan (setahun), maka penulis menyimpulkan bahwa model terbaik adalah model 5-7-1 dengan proses perulangan (epoch) pada saat pelatihan dengan nilai epoch = 35160 dan pencapaian MSE pada saat pengujian dengan MSE = 0.00999954. Adapun data perbandingan dari masing-masing model dapat dilihat pada tabel 5.



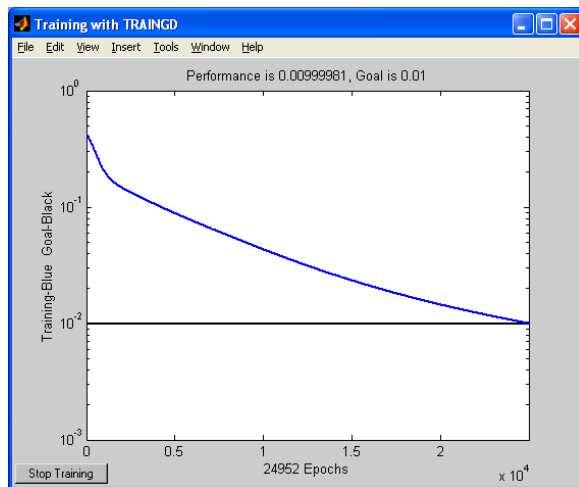
Gambar 8. Pelatihan Pola 5-3-1 Tidak Mencapai Goal (Target)



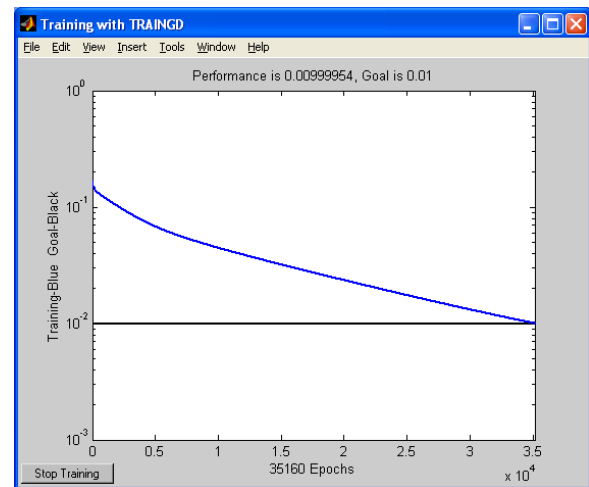
Gambar 9. Pelatihan Pola 5-4-1 Mencapai Goal (Target)



Gambar 10. Pelatihan Pola 5-5-1 Mencapai Goal (Target)



Gambar 11. Pelatihan Pola 5-6-1 Mencapai Goal (target)



Gambar 12. Pelatihan Pola 5-7-1 Mencapai Goal (Dengan menggunakan Matlab)

Tabel 5. Perbandingan Epoch dan MSE dari kelima pola yang di uji

	5-3-1	5-4-1	5-5-1	5-6-1	5-7-1
Epoch Pelatihan	50000	24523	29393	24952	35160
MSE Pengujian	0.00138 915	0.00999 997	0.00999 992	0.00999 981	0.00999 954
Akurasi (100-MSE)	99,9986 1085	99,9900 0003	99,9900 0008	99,9900 0019	99,9900 0046

Tabel 6 Hasil dan Error Data Input (8 buah Data) Pelatihan Data dengan Model 5 – 7– 1 (Pengujian Dengan Matlab)

No	Bulan	Perumahan						JST Pola 5-7-1	
		Leuser	Merapi	Seulawah	Sibayak	Talang	Target	Actual	Error
1	Januari 2012	0,26	0,1	0,33	0,27	0,1	1	0,5875	0,4125
2	Pebruari 2012	0,85	0,76	0,37	0,2	0,24	0	0,2750	- 0,2750
3	Maret 2012	0,31	0,1	0,6	0,71	0,24	0	0,7160	- 0,7160
4	April 2012	0,42	0,19	0,25	0,53	0,24	1	0,6244	0,3756
5	Mei 2012	0,58	0,57	0,82	0,41	0,34	0	0,1391	- 0,1391
6	Juni 2012	0,9	0,81	0,37	0,67	0,1	0	0,4991	- 0,4991
7	Juli 2012	0,47	0,85	0,9	0,9	0,9	1	0,9173	0,0827
8	Agustus 2012	0,1	0,29	0,25	0,1	0,1	0	0,3943	- 0,3943

Tabel 7. Hasil dan Error Data Input (4 Buah Data)
Pengujian dengan Model 5-7-1 (Pengujian dengan Matlab)

No	Bulan	Perumahan						JST Pola 5-7-1	
		Leuser	Merapi	Seulawah	Sibayak	Talang	Target	Act	Error
9	September 2012	0,63	0,76	0,6	0,7	0,19	0	0,0210	- 0,0210
10	Oktober 2012	0,47	0,85	0,75	0,83	0,15	1	0,0093	0,9907
11	Nopember 2012	0,37	0,66	0,14	0,17	0,1	0	0,0294	- 0,0294
12	Desember 2012	0,31	0,9	0,1	0,5	0,1	0	0,0514	- 0,0514

Tabel 8. Hasil Pengujian Sistem

No	Pengujian dengan Matlab Model 5-3-1			Pengujian dengan Matlab Model 5-4-1			Pengujian dengan Matlab Model 5-5-1			Pengujian dengan Matlab Model 5-6-1			Pengujian dengan Matlab Model 5-7-1		
	Target	Output	Error	Target	Output	Error	Target	Output	Error	Target	Output	Error	Target	Output	Error
1	0	0,0092	- 0,0092	0	0,0683	-0,0683	0	0,0006	- 0,0006	0	0,0243	-0,0243	0	0,0210	- 0,0210
2	1	0,0061	0,0039	1	0,0498	0,9502	1	0,0001	0,9999	1	0,0904	0,9096	1	0,0093	0,9907
3	0	0,1728	- 0,1728	0	0,0073	-0,0073	0	0,0121	- 0,0121	0	0,0212	-0,0212	0	0,0294	- 0,0294
4	0	0,6590	- 0,6590	0	0,0012	0,0012	0	0,0025	- 0,0025	0	0,0607	-0,0607	0	0,0514	- 0,0514

Dari tabel 8 yang merupakan pengujian dengan Matlab dengan menggunakan model 5-3-1, 5-4-1, 5-5-1, 5-6-1 dan 5-7-1 menghasilkan target yang sesuai dengan yang diharapkan yaitu, $\text{Target} = \text{Output} + \text{Error}$.

Untuk pencapaian target 1 yaitu *preventive schedule* dapat diketahui pada data pengujian dengan menjumlahkan nilai *output* dengan nilai *error*. Sedangkan pada pencapaian target 0 yaitu *preventive unscheduled* dapat diketahui pada data pengujian dengan menjumlahkan nilai *output* dengan nilai *error* yang menghasilkan nilai 0. Untuk pencapaian target 1 pada *preventive schedule* menggunakan pola pelaksanaan dalam 3 bulan dapat terpenuhi berdasarkan dari data pengujian yang dihasilkan pada metode *back propagation*.

- Pada pengujian model 5-4-1 dimana Target (1) = 0,0498 (*output*) + 0,9502 (*error*).
- Pada pengujian model 5-5-1 dimana Target (1) = 0,0001 (*output*) + 0,9999 (*error*).
- Pada pengujian model 5-6-1 dimana Target(1) = 0,0904 (*output*) + 0,9096 (*error*).
- Pada pengujian model 5-7-1 dimana Target(1) = 0,0093 (*output*) + 0,9907 (*error*).

Perancangan arsitektur sistem jumlah *hidden layer* ditentukan dengan cara trial dan error, dalam arti hasil pembelajaran yang tercepat dan terbaik itulah yang menentukan jumlah *hidden layer* tersebut. Mengenai jumlah banyaknya *hidden layer* yang dibutuhkan tidak ada ketentuan khusus karena tidak ada teori yang dengan pasti dapat dipakai.

5. KESIMPULAN dan SARAN

Kesimpulan yang dapat diambil setelah melakukan penelitian dan pengujian pada sistem dalam penentuan target penjadwalan pemeliharaan *preventive schedule* dengan menggunakan *algoritma backpropagation* adalah :

1. Jaringan syaraf tiruan dengan *algoritma backpropagation* mampu menganalisa dan menguji konsentrasi yang sesuai dengan pola yang dilatih berdasarkan nilai *input* yang menjadi kriterianya.
2. Proses pelaksanaan jaringan syaraf tiruan menggunakan data sejumlah 12 buah data, Dimana data tersebut dibagi menjadi 2 bagian, bagian pertama sejumlah 8 buah data digunakan untuk data pelatihan (*training*) dan bagian kedua sejumlah 4 buah data digunakan untuk data pengujian (*testing*).
3. Tingkat keakuratan dari penentuan konsentrasi tersebut dibuktikan dengan dengan tingkat konsentrasi pada MSE yang terendah. Dimana Akurasi = $(100 - \text{MSE}) = 99,99000046$ mendekati 100 % pada pola 5-7-1.
4. Semakin banyak data yang dilatih maka *output* akan lebih akurat.

Dari hasil penelitian dan implementasi jaringan syaraf tiruan dengan menggunakan *algoritma Backpropagation* untuk penentuan konsentrasi jadwal *preventive schedule* maka disarankan hal-hal yang berikut ini :

1. Untuk lebih memudahkan penggunaan maka perlu dibuat suatu *interface* seperti *Graphical User Interface* (GUI).
2. Dalam penentuan konsentrasi jadwal *preventive schedule* yang dilakukan tidak pada perumahan (*housing*) saja namun bisa dilakukan pada perkantoran (*Office*), bangunan untuk umum (*public building*) dan juga bangunan sekolah.

DAFTAR PUSTAKA

- Anita Desiani dan Muhammad Arhami (2006)
“Konsep Kecerdasan Buatan”. CV. Andi Offset Yogyakarta
- Suyanto, ST, Msc (2007), “Artificial Intelligence”, Informatika Bandung
- Drs. Jong Jek Siang, Msc (2009), “Jaringan Syaraf Tiruan dan Pemogramannya menggunakan Matlab”, CV. Andi Offset Yogyakarta
- Kiki, Sri Kusumadewi, “Analisis Jaringan Syaraf Tiruan dengan Metode Backpropagation Untuk

Mendeteksi Gangguan Psikologi”, Universitas Islam Indonesia

M.F. Andrijasa dan Mistianingsih, “Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Jumlah Pengangguran di Kalimantan Timur Dengan Menggunakan Algoritma Pembelajaran Backpropagation.

Enireddy.Vamsidhar, K.V.S.R.P. Varma, P.Sankara Rao, Ravikanth satapati, “Prediction of Rainfall Using Backpropagation Neural Network Model”, Gitam University Visakhapatnam.

M.S.S Sai, P.Thrimurthy, Dr.S. Purushothaman, “Implementation Of Backpropagation Algorithm for Renal datamining”, India.