

IMPLEMENTASI JARINGAN SYARAF TIRUAN DALAM MEMPREDIKSI FREKUENSI RESONANSI ATENA MIKROSTRIP

Khairi Budayawan¹⁾, Yuhandri²⁾, Gunadi Widi Nurcahyo³⁾

¹⁾Prodi Pendidikan Teknik Elektronika, Universitas Negeri Padang

²⁾Fakultas Ilmu Komputer, Universitas Putra Indonesia YPTK

e-mail : [1\)khairi@ft.unp.ac.id](mailto:khairi@ft.unp.ac.id), [2\)yuhandri_yunus@gmail.com](mailto:yuhandri_yunus@gmail.com), [3\)gunadiwidi@yahoo.co.id](mailto:gunadiwidi@yahoo.co.id)

ABSTRACT

The resonant frequency of an antenna is determined by the dimensional parameters and permittivity of the antenna substrate. Generally, to get the resonant frequency, a complex mathematical formula is needed to solve. For this reason, an intelligent method is offered to determine the resonant frequency more easily. In this study, an artificial neural network method with Backpropagation algorithm is used to overcome the problem. The data used were consisting of 80 training data and 15 testing data. The results have shown that the artificial neural network learning method with the backpropagation algorithm was successfully utilized to calculate the resonant frequency of microstrip antennas, where the precision of the resonant frequency obtained of 93.33% at an error of $\leq 1\%$, and 100% at an error of $\leq 2\%$.

Keywords: Artificial Neural Network, Backpropagation, Microstrip Antenna, Resonant Frequency

INTISARI

Frekuensi resonansi antena sangat ditentukan oleh parameter dimensi dan permitivitas substrat antena. Umumnya untuk mendapatkan frekuensi resonansi tersebut dibutuhkan rumus matematis yang rumit. Untuk itu dibutuhkan suatu metode agar dapat menentukan frekuensi resonansi yang lebih mudah. Penelitian ini digunakan metode JST dengan algoritma Backpropagation untuk mengatasi masalah tersebut. Data yang digunakan dalam penelitian sebanyak 80 data training dan 15 data uji. Hasil penelitian memperlihatkan bahwa metode pembelajaran Jaringan Syaraf Tiruan menggunakan algoritma Backpropagation berhasil dengan baik digunakan untuk menghitung frekuensi resonansi antena mikrostrip, dimana presisi dari frekuensi resonansi yang diperoleh adalah sebesar 93.33% pada error $\leq 1\%$, dan 100% pada error $\leq 2\%$.

Kata kunci : jaringan syaraf tiruan, backpropagation, antena mikrostrip, frekuensi resonansi

PENDAHULUAN

Dalam sistem komunikasi wireless, antena merupakan salah satu komponen yang sangat penting. Rancangan antena yang baik dapat membuat sistem bekerja dengan mudah dan dapat meningkatkan performa sistem secara keseluruhan. Selama 50 tahun terakhir, teknologi antena telah menjadi mitra revolusi komunikasi yang tak tergantikan. Banyak kemajuan yang telah terjadi selama periode ini dan umum digunakan saat ini, dan merupakan efek dari semakin banyaknya tuntutan akan performa sistem yang lebih besar. Permintaan akan aplikasi *wireless* yang terus bertambah mengakibatkan permintaan akan desain

antena yang terintegrasi dan memiliki *gain* yang besar semakin meningkat [1].

Dari berbagai macam type antena, salah satu yang populer digunakan adalah antena mikrostrip. Antena mikrostrip memiliki fitur menarik yaitu profil rendah, ringan, *narrow bandwidth*, kompatibel terhadap permukaan *planar* dan *non-planar*, mudah dipabrikasi dan integrasi dengan rangkaian *microwave* [2]. Antena mikrostrip telah populer digunakan dalam komunikasi bergerak [3-4] yang sedang trend saat ini.

Setiap antena dibuat sesuai dengan kebutuhan frekuensi yang diminta. Penghitungan frekuensi resonansi antena sulit dilakukan secara

matematis, karena banyak parameter-parameter yang mempengaruhi dalam perhitungan, di antaranya jenis substrat, ketebalan substrat, dimensi antena, dan bahkan kondisi lingkungan juga mempengaruhi. Tabel 1 merupakan perbandingan hasil perhitungan antena [5] dengan hasil simulasi menggunakan aplikasi CST Microwave Studio memperlihatkan perbedaan yang cukup signifikan yaitu rata-rata persentase *error* mencapai 3.33%. Hasil tersebut menunjukkan bahwa perhitungan frekuensi antena dengan menggunakan persamaan matematis tersebut tidak dapat memprediksi frekuensi resonansi antena secara akurat.

Tabel 1 Perbandingan Hasil Perhitungan dan Hasil Simulasi Antena

No	θ	h	L	W	X_0	f_r Hitung	f_r CST	error %
1	2	1	105.57	106.07	19.28	1.000	0.990	1.00
2	2	1	52.51	53.03	9.59	2.000	1.962	1.90
3	2	1	34.80	35.36	6.36	3.000	2.922	2.60
4	2	1	25.94	26.52	4.74	4.000	3.882	2.95
5	2	1	20.62	21.21	3.77	5.000	4.818	3.64
6	3	2	86.09	86.60	14.05	1.000	0.984	1.60
7	3	2	42.65	43.30	6.96	2.000	1.944	2.80
8	3	2	28.11	28.87	4.58	3.000	2.898	3.40
9	3	2	20.81	21.65	3.39	4.000	3.846	3.85
10	3	2	16.41	17.32	2.68	5.000	4.788	4.24
11	5	3	66.75	67.08	9.48	1.000	0.978	2.20
12	5	3	32.87	33.54	4.67	2.000	1.920	4.00
13	5	3	21.47	22.36	3.05	3.000	2.856	4.80
14	5	3	15.72	16.77	2.23	4.000	3.792	5.20
15	5	3	12.26	13.42	1.74	5.000	4.710	5.80
Rata-rata persentase <i>error</i> %								3.33

Banyak metode yang dapat digunakan dalam menghitung frekuensi resonansi antena mikrostrip, seperti metode *Transmission Line* (TLM), *Cavity Model*, dan *Method of Moment* (MoM). Namun, untuk memperoleh hasil perhitungan yang mendekati dengan metode ini sangat sulit dilakukan dan memakan waktu. Beberapa metode atau teknik yang berbeda seperti Logika Fuzzy, Algoritma Genetik, dan Jaringan Syaraf Tiruan (JST) telah digunakan untuk memperkirakan frekuensi resonansi dan bandwidth antena mikrostrip [6-7]. Berdasarkan hal tersebut maka masalah yang ada dapat dirumuskan yaitu bagaimana menggunakan metode Jaringan Syaraf Tiruan dengan algoritma *backpropagation* dalam memprediksi frekuensi resonansi antena mikrostrip.

Tujuan dari penelitian adalah menggunakan model Jaringan Syaraf Tiruan dengan algoritma *backpropagation* dalam memprediksi frekuensi resonansi Antena Mikrostrip. Proses pembelajaran untuk Jaringan Syaraf Tiruan menggunakan sebanyak 80 dan 15 data sebagai data pelatihan dan

pengujian. Penelitian ini diharapkan dapat memudahkan dalam memprediksi frekuensi resonansi antena mikrostrip berdasarkan parameter dimensi antena dan permitivitas substrat.

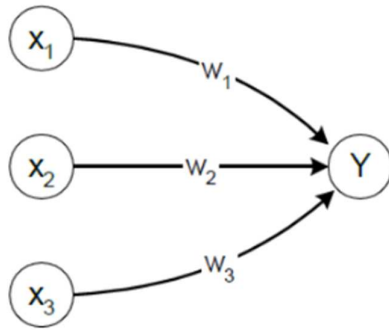
PENDEKATAN PEMECAHAN MASALAH

JST adalah suatu model yang terinspirasi dari kerja jaringan syaraf biologis khususnya otak yang digunakan untuk memprediksi atau pendekatan fungsi yang bergantung pada sejumlah besar input dan pada umumnya tidak diketahui [8]. Sistem syaraf memiliki jutaan sel-sel syaraf di mana antara satu sama lain saling terhubung dan tiap-tiap syaraf tersebut melakukan tugas tertentu. Karena interkoneksi antara sel-sel itu rumit, hasil dari multi tugas sederhana ini adalah pelaksanaan tugas yang lebih kompleks dan lebih besar [9]. Mengadopsi cara kerja jaringan syaraf biologi, JST terdiri atas sejumlah besar *neuron* interkoneksi, di mana dari hal tersebut JST dapat meniru bagaimana sistem syaraf dalam memproses informasi pada manusia.

JST merupakan penggeneralisasi model matematika yang berdasarkan jaringan syaraf secara biologi [10] dengan anggapan bahwa:

- 1) Proses informasi berada pada *neuron-neuron*
- 2) Sinyal dikirimkan antar *neuron-neuron* dengan menggunakan penghubung.
- 3) Penghubung antara *neuron* ini mempunyai nilai bobot yang dapat berfungsi untuk memperkuat atau memperlemah sinyal
- 4) Untuk menghasilkan *output*, pada setiap *neuron* memakai fungsi aktivasi yang diberikan pada setiap jumlah input yang diperoleh. Nilai *output* dan batas ambang nilai kemudian dibandingkan.

Terdapat 3 hal yang sangat menentukan pada JST yaitu: 1) pola hubungan antara *neuron* yang disebut arsitektur jaringan; 2) metode untuk penentuan bobot penghubung, biasa disebut dengan metode *training* (pelatihan) / *learning* (pembelajaran) / algoritma; dan 3) fungsi aktivasi. Contoh digambarkan pada Gambar 1 *neuron Y* dengan input x_1, x_2 dan x_3 .



Gambar 1. Pembobotan pada JST

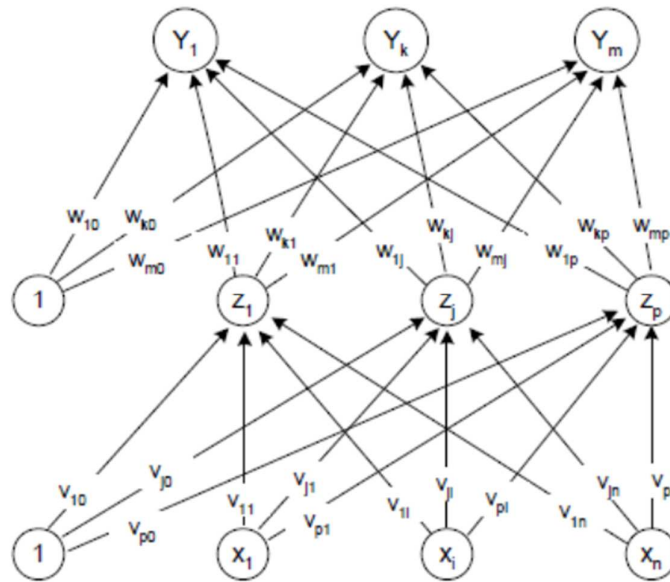
Backpropagation

Salah satu algoritma biasa dipakai dalam membantu menyelesaikan masalah yang kompleks adalah *Backpropagation*. Ini dapat terjadi karena jaringan yang memakai algoritma tersebut dilatih menggunakan metode pembelajaran terpandu/terbimbing. Pada jaringan akan diberikan

sepasang pola yang meliputi pola *input* dan pola yang diharapkan. Saat pola diberikan, dilakukan perubahan pada bobot dengan tujuan untuk memperkecil beda pola *output* dengan pola yang diharapkan. Latihan diberikan secara berulang hingga didapatkan pola *output* yang diharapkan.

Backpropagation

Backpropagation mempunyai beberapa unit yang berada pada satu atau lebih *hidden layer*. Arsitektur *Backpropagation* dengan sejumlah n *input* dan *hidden layer* yang terdiri atas sejumlah p unit, serta sejumlah m unit *output* dijelaskan pada Gambar 2. V_{ji} adalah nilai bobot garis dari unit *input* X_i ke unit *hidden layer* Z_j (V_{j0} adalah nilai bobot garis yang berfungsi menghubungkan bias pada unit *input* ke unit *hidden layer* Z_j). W_{kj} adalah nilai bobot dari unit *hidden layer* Z_j menuju unit *output* Y_k (W_{k0} adalah nilai bobot bias *hidden layer* ke unit *output* Z_k).



Gambar 2. Arsitektur *Backpropagation*

Algoritma Backpropagation

Pelatihan pada *Backpropagation* memiliki tiga fase yaitu: 1) fase propagasi maju; 2) fase propagasi mundur; dan 3) fase perubahan bobot. Ketiga fase tersebut dilakukan secara berulang-ulang hingga kondisi penghentian terpenuhi. Kondisi penghentian biasa dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika apabila jumlah iterasi yang dilakukan sudah

melampaui jumlah maksimum iterasi yang ditetapkan, atau jika kesalahan yang terjadi sudah lebih rendah dari batas toleransi yang dibolehkan. Tahapan lebih detail untuk proses pelatihan dan proses pengujian diuraikan dalam algoritma.

Algoritma Proses Pelatihan

Backpropagation:

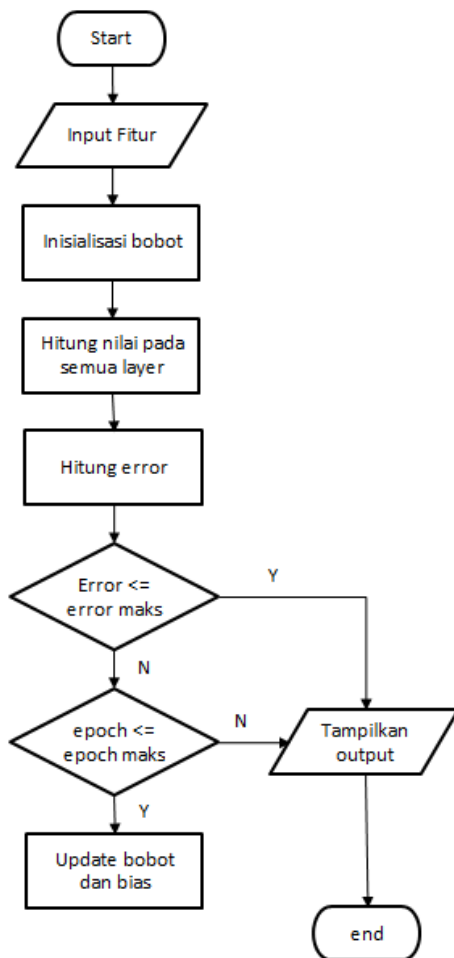
1. Input fitur
2. Inisialisasi bobot
3. Hitung nilai pada semua *layer*
4. Hitung *error*
5. Update bobot dan bias
6. Tampilkan *output*

Algoritma Proses Pengujian

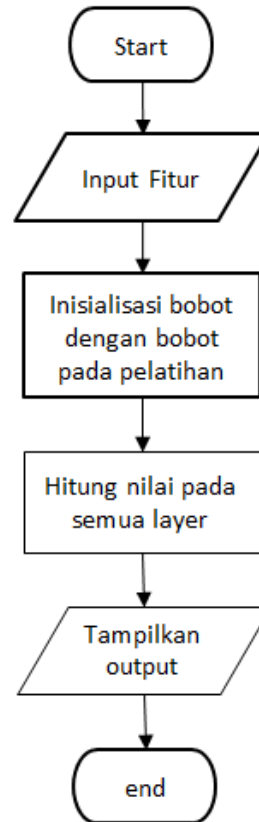
Backpropagation:

1. Input fitur
2. Inisialisasi bobot sesuai hasil pelatihan
3. Hitung nilai pada semua *layer*
4. Tampilkan *output*

Berdasarkan penjelasan tentang algoritma *Backpropagation* dan kemudian diterjemahkan ke dalam bentuk *flowchart* yang kemudian implementasi dan pengujian menggunakan bahasa pemrograman Matlab. *Flowchart* algoritma pelatihan dan pengujian seperti terlihat pada Gambar 3 dan Gambar 4.



Gambar 3. Flowchart Pelatihan



Gambar 4. Flowchart Pengujian

Algoritma pelatihan jaringan untuk yang memiliki satu *hidden layer* menggunakan fungsi aktivasi *sigmoid biner* dapat diuraikan atas [11]:

- a) Tahap 0 : Dilakukan inisialisasi semua nilai bobot berdasarkan bilangan acak kecil
- b) Tahap 1 : Bila belum didapatkan kondisi penghentian, laksanakan tahap 2 hingga 9
- c) Tahap 2 : bagi setiap pasangan data pelatihan, laksanakan tahap 3 hingga 8

Fase I : Propagasi maju

- d) Tahap 3 : setiap unit *input* menerima sinyal kemudian meneruskan ke unit *hidden* yang ada di atasnya
- e) Tahap 4 : Hitung nilai semua *output* pada unit *hidden* z_j ($j = 1, 2, \dots, p$)

$$z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \quad (1)$$

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}} \quad (2)$$

- f) Tahap 5 : Hitung nilai semua keluaran jaringan pada unit y_k ($k = 1, 2, \dots, m$)

$$y_net_k = w_{k0} + \sum_{j=1}^n z_j w_{kj} \quad (3)$$

$$y_k = f(y_net_k) = \frac{1}{1 + e^{-y_net_k}} \quad (4)$$

Fase II : Propagasi mundur

- g) Tahap 6 : Lakukan perhitungan faktor δ unit output berdasarkan atas kesalahan pada setiap unit output y_k ($k = 1, 2, \dots, m$)

$$\delta_k = (t_k - y_k) f'(y_net_k) = (t_k - y_k) y_k (1 - y_k) \quad (5)$$

δ merupakan unit kesalahan dan digunakan untuk perubahan nilai bobot layer pada tingkat dibawahnya (tahap 7).

Hitung suku perubahan nilai bobot w_{kj} (nanti digunakan untuk merubah nilai bobot w_{kj}) dengan laju percepatan α .

$$\Delta w_{kj} = \alpha \delta_k z_j \quad k = 1, 2, \dots, m ; j = 0, 1, \dots, p \quad (6)$$

- h) Tahap 7 : Hitung faktor δ unit hidden berdasarkan kesalahan pada setiap unit hidden

$$z_j (j = 1, 2, \dots, p) \quad \delta_net_j = \sum_{k=1}^m \delta_k w_{kj} \quad (7)$$

Faktor δ unit *hidden*:

$$\delta_j = \delta_net_j f'(z_net_j) = \delta_net_j z_j (1 - z_j) \quad (8)$$

Hitung suku perubahan nilai bobot v_{ji} (nanti dipakai dalam merubah nilai bobot v_{ji})

$$\Delta v_{ji} = \alpha \delta_j x_i \quad j = 1, 2, \dots, p ; i = 0, 1, \dots, n \quad (9)$$

Fase III : Perubahan Bobot

- i) Tahap 8 : Hitung semua nilai perubahan bobot
Perubahan nilai bobot garis yang menuju ke unit *output*:

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj} \quad (k = 1, 2, \dots, m ; j = 0, 1, \dots, p) \quad (10)$$

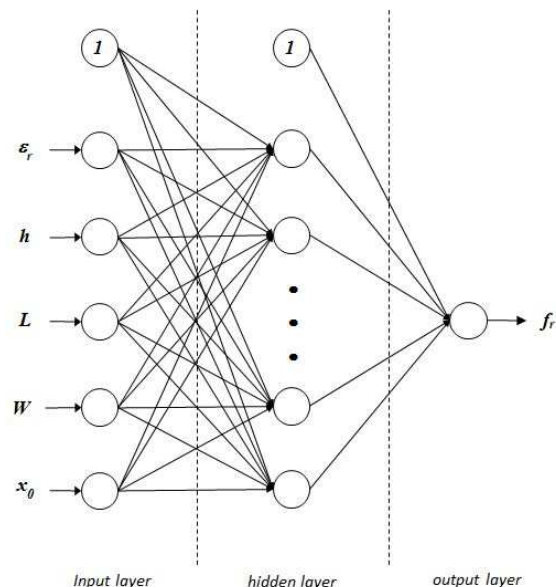
Perubahan nilai bobot garis yang menuju ke unit *hidden* :

$$v_{ji}(\text{baru}) = v_{ji}(\text{lama}) + \Delta v_{ji} \quad (j = 1, 2, \dots, p ; i = 0, 1, \dots, n) \quad (11)$$

Setelah pelatihan selesai dilaksanakan, jaringan dapat digunakan untuk melakukan pengenalan pola. Pada tahap ini hanya propagasi maju (tahap 4 serta 5) yang digunakan dalam menentukan *output* jaringan. Apabila *sigmoid biner* tidak digunakan sebagai fungsi aktivasi, maka tahap 4 dan 5 harus disesuaikan. Demikian pula untuk turunannya di tahap 6 dan 7

Proses Pelatihan

Arsitektur jaringan merupakan gambaran umum hubungan antar *layer* yang digunakan dalam proses pelatihan. Arsitektur Jaringan Syaraf Tiruan dengan algoritma *backpropagation* dalam prediksi frekuensi resonansi antenna mikrostrip memiliki 5 buah masukan (ϵ_r, h, L, W, x_0) dengan 1 bias, 1 buah *hidden layer* dengan 1 bias, serta 1 buah target pada *output layer* (f_r), seperti terlihat pada Gambar 5.



Gambar 5. Arsitektur Jaringan Syaraf Tiruan yang diusulkan

Berdasarkan arsitektur tersebut dapat ditentukan variabel pada input layer dan output

layer, dimana ϵ , h , L , W , x_0 masing-masing diwakili oleh $X1$, $X2$, $X3$, $X4$, dan $X5$ untuk *input layer*, dan f_r diwakili dengan Y pada *output layer*.

Sebelum memulai proses pelatihan, data yang digunakan harus dipastikan apakah sudah sesuai dengan fungsi aktivasi yang digunakan. Pada penelitian ini, fungsi aktivasi yang digunakan adalah fungsi aktivasi *sigmoid biner*. Karena fungsi aktivasi *sigmoid biner* memiliki range $[0, 1]$, maka data ditransformasikan terlebih dahulu ke interval $[0, 1]$ dengan menggunakan persamaan:

$$x' = \frac{0.8(x-a)}{b-a} + 0.1 \tag{12}$$

di mana a merupakan nilai data minimum, dan b adalah nilai data maksimum.

Metode

Penelitian dimulai dari mengidentifikasi masalah, menentukan tujuan, studi literatur, kemudian dilanjutkan dengan pengumpulan data yang akan digunakan untuk pelatihan pada Jaringan Syaraf Tiruan, merancang arsitektur Jaringan Syaraf Tiruan, implementasi algoritma *backpropagation* dan dilanjutkan pengujian hasil. Urutan tahapan kerangka kerja penelitian digambarkan pada Gambar 5.



Gambar 5. *Flowchart* Pengujian

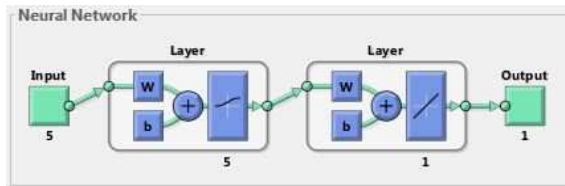
Berdasarkan kerangka kerja penelitian, maka dapat diuraikan sebagai berikut:

1. **Identifikasi Masalah**
 Pada tahap inidilakukan proses penentuan masalah dan mendefenisikan batasan masalah yang akan diteliti, sehigga tujuan penelitian dapat tercapai dengan baik.
2. **Studi Literatur**
 Pada tahap ini pengumpulan referensi dilakukan dan diambil dari penelitian terdahulu yang terkait dengan penelitian yang menggunakan Jaringan Syarat Tiruan untuk memprediksi suatu masalah.
3. **Pengumpulan Data**
 Data dikumpulkan dari hasil simulasi menggunakan program aplikasi CST Microwave Studio, merupakan program aplikasi yang digunakan untuk simulasi elektromagnetik 3D untuk komponen frekuensi tinggi. Data yang diperoleh digunakan sebagai data untuk pelatihan Jaringan Syaraf Tiruan.
4. **Merancang Arsitektur Jaringan Syaraf Tiruan**
 Rancangan arsitektur yang tepat akan menentukan tingkat keberhasilan target yang akan dicapai, karena tidak semua permasalahan dapat dipecahkan dengan menggunakan arsitektur yang sama. Pada penelitian ini, karena permasalahan yang akan dipecahkan cukup sulit, maka arsitektur yang dipilih adalah jaringan dengan banyak lapisan (*multilayer net*), dengan 5 buah input masing-masing untuk parameter permitivitas ϵ , ketebalan substrat h , panjang patch L , lebar patch W , dan posisi feeding point x , dan 1 buah output untuk parameter Frekuensi Resonansi f_r .
5. **Implementasi Metode Back propagation**
 Pada tahap ini Jaringan Syarat Tiruan dilatih/training dengan menggunakan data yang telah dikumpulkan sehingga siap untuk digunakan dalam memprediksi output yang diinginkan sesuai dengan input yang diberikan.

HASIL DAN PEMBAHASAN

Parameter-parameter yang diperlukan dalam membangun Jaringan Syaraf Tiruan ini meliputi *learning rate*, jumlah *neuron* pada bagian *hidden layer*, dan *goal*. Parameter *epoch* dan nilai *learning rate* ditentukan dengan melihat nilai *means square error* (MSE) saat pelatihan, dimana semakin kecil nilai MSE maka semakin bagus kinerja Jaringan Syaraf Tiruan.

Nilai awal parameter arsitektur jaringan ditetapkan memiliki 5 input, 1 *hidden layer* dengan 5 *neuron* dan 1 *output* seperti terlihat pada Gambar 6. Fungsi aktivasi yang dipilih adalah fungsi aktivasi *logsig* pada lapisan *input*, fungsi aktivasi *purelin* untuk lapisan *output*, serta *traingdx* sebagai fungsi pelatihan jaringan. Inisialisasi nilai bobot dan bias dilakukan secara acak dengan range -1 hingga +1.

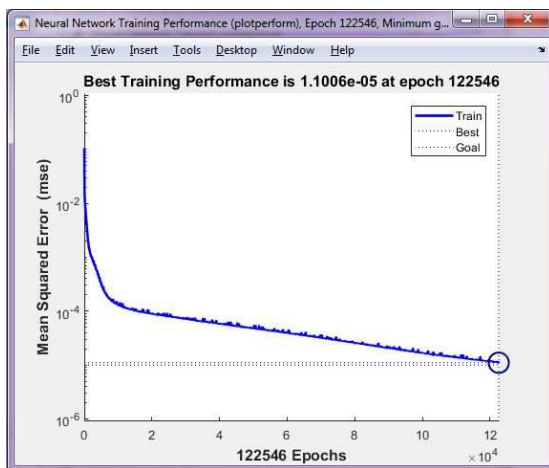


Gambar 6. Arsitektur Jaringan Syaraf Tiruan algoritma Backpropagation.

Setelah dilakukan beberapa kali pelatihan pada jaringan untuk mencari parameter-parameter pelatihan yang paling cocok [12], maka didapatkan hasil konfigurasi parameter untuk sistem Jaringan Syaraf Tiruan seperti terlihat pada Tabel 2.

Tabel 2 Konfigurasi Parameter Pelatihan Jaringan

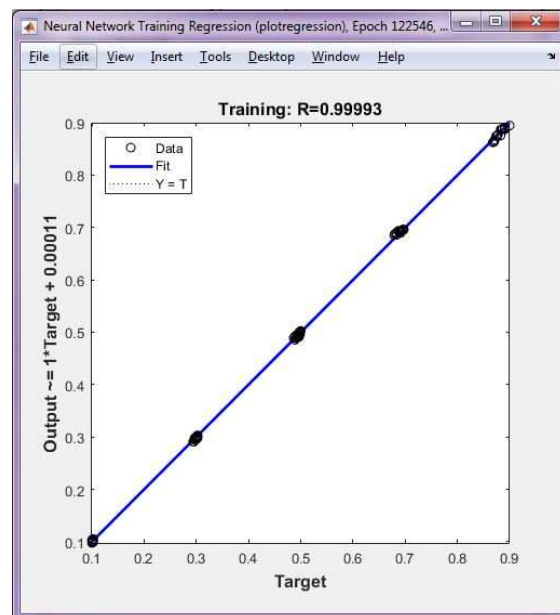
No	Nama Parameter	Jumlah
1	Arsitektur Jaringan	5-5-1
2	Neuron Input	5
3	Neuron Pada Hidden Layer	5
4	Neuron Output	1
5	Learning Rate	0.1
6	Goal	0.00001
8	Momentum	0.95



Gambar 7. Grafik Perubahan Nilai MSE terhadap Epoch

Pada Gambar 7 terlihat bahwa hingga *epoch* 1000 terjadi penurunan yang sangat tajam dari nilai MSE 10^{-1} hingga 10^{-3} . Penurunan gradient yang signifikan pada proses menuju target menunjukkan bahwa terjadi perbaikan kesalahan, di mana *output* yang memiliki *error* besar telah diperkecil dengan melakukan penyesuaian pada masing-masing bobot saat proses *Backpropagation*. Penurunan saat menuju *goal* kemudian cenderung landai yang berarti kesalahan yang terjadi sudah semakin kecil.

Pada pelatihan ini, performa terbaik jaringan diperoleh pada nilai MSE $1.1006e-05$ di *epoch* 122546. Berdasarkan hal itu maka dapat dijelaskan bahwa masing-masing bobot jaringan dapat melakukan perbaikan kesalahan dan mengenali *input* lebih cepat dari *epoch* yang ditetapkan.



Gambar 8. Grafik Regresi Pelatihan

Akurasi dari pelatihan yang telah dilakukan dapat dilihat dari analisis regresi untuk melihat respon jaringan dan target yang diinginkan. Pada Gambar 8 terlihat hasil regresi dari output (*Y*) dan target (*X*) menunjukkan gradient garis hasil regresi linear hampir sama pada satu titik, titik perpotongan dengan sumbu *Y* pada titik 0 dan koefisien korelasi (*R*) mendekati 1 (0.99993). Dari ketiga parameter tersebut dapat disimpulkan bahwa output jaringan hampir sama dengan target, berarti 80 data dapat dikenali dengan baik saat pelatihan. Hasil evaluasi analisis regresi menyatakan bahwa jaringan siap digunakan untuk menganalisa dan mengidentifikasi data uji yang diberikan dengan menggunakan parameter-parameter yang diperoleh saat pelatihan.

Tabel 3. Hasil Pengujian Jaringan Syaraf Tiruan Algoritma *Backpropagation* Prediksi Frekuensi Resonansi Antena Mikrostrip

No	a	h	L	W	X_0	f_c GHz (CST)	f_c GHz (JST)	error %	error $\leq 1\%$	error $\leq 2\%$
1	4.4	1.53	35.53	35.75	5.22	1.944	1.927	0.87	√	√
2	4.4	1.53	13.85	14.30	2.04	4.740	4.726	0.30	√	√
3	4.4	1	23.69	23.84	3.48	2.910	2.896	0.49	√	√
4	4.4	2	23.33	23.84	3.43	2.874	2.871	0.11	√	√
5	3	2.5	42.41	43.30	6.92	1.938	1.944	0.30	√	√
6	3	2.5	20.51	21.65	3.34	3.840	3.853	0.35	√	√
7	2	2	29.05	30.30	5.30	3.390	3.399	0.25	√	√
8	4	2	20.82	21.43	3.14	3.342	3.354	0.36	√	√
9	5	2	18.66	19.17	2.65	3.336	3.370	1.01	√	√
10	2.5	3	61.94	63.25	10.63	1.458	1.462	0.25	√	√
11	2.5	3	19.36	21.08	3.32	4.344	4.359	0.34	√	√
12	3.5	2	19.31	20.04	3.02	3.828	3.843	0.40	√	√
13	4.6	1.53	19.64	19.98	2.85	3.348	3.370	0.66	√	√
14	2	2	41.24	42.43	7.53	2.424	2.447	0.97	√	√
15	2.5	3	30.06	31.62	5.15	2.898	2.903	0.15	√	√
Rata-rata persentase error %								0.45		
Rata-rata Presisi %									93.33	100

Setelah melalui tahap proses pelatihan, dilakukan proses pengujian dengan menggunakan 15 data uji. Berdasarkan hasil pengujian yang telah dilakukan seperti terlihat pada Tabel 3 didapat rata-rata persentase *error* adalah 0.45%. Rata-rata presisi hasil testing adalah 93.33% pada nilai *error* $\leq 1\%$ dan 100% pada nilai *error* $\leq 2\%$. Hasil pengujian tersebut menunjukkan bahwa pembelajaran Jaringan Syaraf Tiruan *backpropagation* dengan target *goal* sebesar 0.00001 dapat digunakan dengan baik untuk memprediksi frekuensi resonansi antena mikrostrip dengan presisi 93.33% pada nilai *error* $\leq 1\%$, dengan rata-rata *error* sebesar 0.45%. Hasil nilai presisi dapat ditingkatkan dengan cara mengatur nilai target *goal* mendekati nilai 0.

KESIMPULAN

Berdasarkan penelitian dapat dibuat kesimpulan bahwa prediksi frekuensi resonansi antena mikrostrip berdasarkan data *input* parameter antena dapat dilakukan dengan menggunakan model Jaringan Syaraf Tiruan dengan algoritma *backpropagation* dengan tingkat presisi 93.33% dicapai pada nilai *error* $\leq 1\%$, dengan rata-rata *error* 0.45%.

DAFTAR PUSTAKA

[1] Musa A, Isa A. Prediction of Resonance Frequency of Aperture Coupled Microstrip Antennas by Artificial Neural Network. *International Journal of Engineering Sciences & Research Technology*. 5 252-260. 2016.

[2] Shivendra R, Syed S.U., Tanveer S.K. Design of Microstrip Antenna Using Artificial Neural Network. *International Journal of Engineering Research and Applications*. 3 pp. 461-464. 2013.

[3] Akdagli A, Toktas A, Kayabasi A, Develi I. An Application of Artificial Neural Network To Compute The Resonant Frequency of E-Shaped Compact Microstrip Antennas. *Journal of Electrical Engineering*. 64 pp. 317-322. 2013.

[4] Sakshi L, Singh V.K. A Novel Approach for Analysis of Bandwidth of Microstrip Patch Antenna Using Neural Network. *International Journal of Advanced Research in Computer Science and Software Engineering*. 4 pp.637-642. 2014.

[5] Milligan T.A. Modern Antenna Design. John Wiley & Sons, Inc. pp. 285-355. 2015.

[6] Kapusuz K.Y., Can S, Dagdeviren B. Artificial Neural Network base Bandwidth Estimation of a CPW-Fed Patch Antenna. *International Journal of Computer Application*. 69 pp. 37-40. 2013.

[7] Pritam S.R., Shamik C. Optimization and improved Bandwidth of Fork shape Microstrip Antenna via Artificial Neural Network. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, pp. 11 7-12. 2016.

[8] Malvika S, Singhal P.K. Comparative Analysis of Different Artificial Neural Networks for Circular Microstrip Antenna Design. *International Research Journal of Engineering and Technology (IRJET)*. 4. Pp. 2291-2297. 2017.

[9] Rojas R. *Neural Networks: a Systematic Introduction*. Springer-Verlag, Berlin. 1996.

[10] Siang JJ. *Jaringan Syaraf Tiruan dan Pemrogramannya menggunakan Matlab*. Andi Offset. 2005.

[11] Vivek S.K., Geetam S.T. Design and Analysis of Microstrip Patch Antennas Using Artificial Neural Network. dlm Sudipta Chattopadhyay. "Trends in Research on Microstrip Antennas." IntechOpen pp. 55-75. 2017.

[12] Yuhandri, Sarifuddin M., Eri P.W., Karmilasari. Pattern Recognition and Classification Using Backpropagation Neural Network Algorithm for Songket Motifs Image Retrieval. *International Journal on Advanced Science Engineering Information Technology*. 7. Pp. 2343-2349. 2017.