

# Pemodelan Arus *Arcing* Tegangan Rendah pada Kabel Fleksibel (Serabut) menggunakan *Elman Neural Network*

Liga Primabaraka, Dimas Anton Asfani, dan I Made Yulistya Negara

Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

*e-mail:* [liga.primabaraka12@mhs.ee.its.ac.id](mailto:liga.primabaraka12@mhs.ee.its.ac.id), [anton@ee.its.ac.id](mailto:anton@ee.its.ac.id), [yulistya@ee.its.ac.id](mailto:yulistya@ee.its.ac.id)

**Abstrak**—Penggunaan listrik dapat menyebabkan bahaya jika tidak diperlakukan dengan baik. Salah satu bahaya yang dapat terjadi adalah kebakaran akibat dari adanya *arc flash* yang berasal dari peristiwa hubung singkat. Maka dari itu, diperlukan sebuah analisa bahaya listrik pada tegangan rendah sehingga keamanan pada sisi pelanggan dapat tetap terjaga. Arus *Arc flash* pada tegangan rendah memiliki karakteristik berupa bentuk gelombang yang nilainya tinggi, namun durasi yang relative singkat. Hal ini berakibat pada tidak bereaksinya alat pengaman dalam mengatasi gangguan ini. Dalam tugas akhir ini, penulis akan membuat sebuah pemodelan terhadap arus *arcing* pada tegangan rendah. Tujuan dari melakukan pemodelan adalah untuk memudahkan analisa terhadap fenomena *arc flash* pada tegangan rendah sehingga nantinya ditemukan solusi untuk mengatasi bahaya yang dapat diakibatkan oleh fenomena *arc flash* itu sendiri. Pemodelan ini menggunakan metode *Elman Neural Network* yang berfungsi untuk membentuk karakteristik dari bentuk arus *arc flash* dengan menentukan nilai hambatan pada tempat terjadinya *arcing*. Pemodelan dilakukan berdasarkan kasus *arc flash* pada kabel fleksibel dengan jumlah serabut yang berbeda-beda. Keluaran dari tugas akhir ini adalah sebuah arus *arcing* pemodelan yang bentuknya mirip dengan arus *arcing* hasil percobaan. Hasil yang didapatkan adalah arus *arcing* pemodelan memiliki bentuk yang mirip dengan arus *arcing* percobaan.

**Kata Kunci**—*arc flash*, *elman neural network*, kabel fleksibel, pemodelan, tegangan rendah

## I. PENDAHULUAN

DI era modern ini, listrik menjadi kebutuhan yang sangat vital. Dalam beberapa tahun terakhir rasio elektrifikasi mengalami peningkatan. Dengan banyaknya pelanggan listrik skala rumah tangga, keamanan instalasi listrik tegangan rendah menjadi hal yang sangat penting untuk diperhatikan. Salah satu bahaya yang dapat terjadi yang diakibatkan oleh listrik adalah kebakaran. Arus hubung singkat dapat menimbulkan *arc flash* (busur api) yang dapat memacu terjadinya kebakaran. *Arcing* pada tegangan rendah memiliki nilai yang cukup tinggi, namun memiliki durasi yang cenderung singkat sehingga tidak dapat dideteksi oleh perangkat pengaman seperti fuse maupun circuit breaker. Analisa mengenai *arcing* sangat perlu

dilakukan agar dapat menghindari bahaya yang mungkin dapat diakibatkan oleh fenomena tersebut. Bahaya yang mungkin dapat ditimbulkan adalah *electric shock*, *arc flash burn*, *arc flash blast*, *intense light*, *sound wave*, *projectiles* [1]. Dalam rangka memudahkan analisa dari *arcing* tegangan rendah maka diperlukan sebuah pemodelan dari arus *arcing*. Tujuan dari melakukan pemodelan adalah dari faktor kemudahan agar ketika hendak melakukan analisa *arcing* tidak perlu melakukan pengambilan data karena bentuk gelombangnya sudah dimodelkan. *Artificial Neural Network* (ANN) atau Jaringan Saraf Tiruan (JST) telah banyak diterapkan di berbagai permasalahan, seperti lalu lintas, komunikasi, dan rekayasa transportasi. Oleh karena itu, disini kami hendak memanfaatkan JST dalam melakukan pemodelan arus *arcing*. Maka dari itu, penulis hendak melakukan penelitian yang berjudul “Pemodelan Arus *Arcing* Tegangan Rendah pada Kabel Fleksibel (Serabut) menggunakan *Elman Neural Network*” dimana pemodelan ini akan menggunakan JST yang membentuk karakteristik dari gelombang *arc flash* dan memanfaatkan parameter parameter lainnya, seperti energi, jumlah serabut, dan waktu.

## II. DASAR TEORI

### A. Pemodelan Arus *Arcing* Tegangan Rendah

*Arcing* adalah sebuah peluahan elektrik yang mengalir diantara dua elektroda melalui gas atau uap air [2]. *Arc Flash* dapat terjadi akibat adanya hubung singkat. Hubung singkat dapat terjadi karena kontak langsung kedua konduktor yang berbeda potensial memiliki isolasi yang tidak standar, sehingga bagian dalam kabel saling bersentuhan.

*Arc* memiliki tiga wilayah umum: katoda, anoda, dan kolom positif. Drop tegangan didalam katoda ada di sekitar 20 V atau kurang [3]. Drop tegangan dalam wilayah anoda biasanya jauh lebih kecil. Kolom positif, atau plasma, memiliki gradient tegangan yang cukup seragam [4].

Beberapa karakteristik umum arus busur api listrik pada beban resistif yang telah diidentifikasi antara lain di setiap setengah *cycle*, arus memiliki dua nilai yaitu tidak bernilai (seolah-olah bernilai nol) dan bernilai, begitu seterusnya setiap setengah *cycle*. Laju kenaikan dan maksimum (*max*) arus biasanya selalu lebih besar dibanding arus beban normal.



**B. Pengambilan Data**

Setelah semua sudah terangkai dengan baik seperti layout pada gambar 3.1, proses pengambilan data pun dimulai. Pengambilan data dilakukan sebanyak 10 kali untuk masing-masing serabut dengan jumlah serabut 1,3,6,12, dan 24.

**C. Pengolahan Data**

Setelah dilakukan akuisisi data, maka proses selanjutnya adalah pengolahan dari data tersebut. beberapa parameter kemudian ditentukan untuk dapat merepresentasikan karakteristik fenomena dari busur api pada tegangan rendah. Adapun parameter yang pertama adalah energi. Energi adalah jumlah daya dalam satuan waktu. Dalam percobaan ini, *energy* digunakan untuk menentukan R arc. Energi yang diambil adalah energi di tempat terjadi arc.

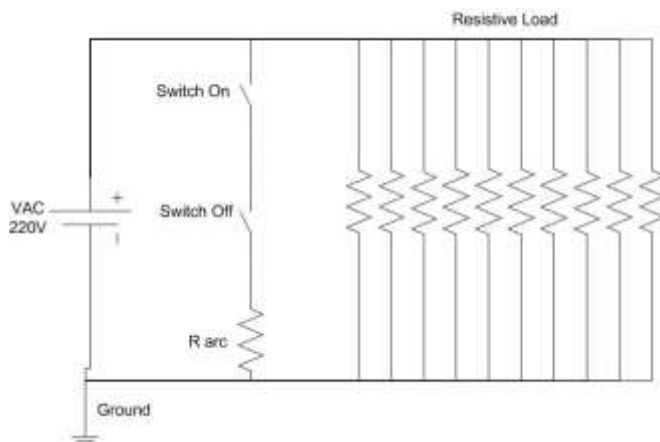
Parameter yang kedua adalah hambatan *arcing*. Nilai hambatan yang dimaksud disini adalah resistansi pada tempat terjadinya busur api listrik, kita namakan dengan Rarc. Nilai dari R arc didapatkan dari perhitungan hambatan pada kabel paralel dikurangi hambatan kabel hitam dan kabel biru. R arc digunakan sebagai target pelatihan *neural network*

**D. Pembuatan Model**

Pemodelan dilakukan menggunakan software matlab. Rangkaian ini terdiri dari power supply, hambatan dalam, R kabel hitam dan biru, R arc, *switch on*, *switch off*, dan R beban.

Adapun perincian dari masing-masing komponen pemodelan adalah sebagai berikut:

**1. Blok Rangkaian Utama**



Gambar 3. Blok Rangkaian Utama

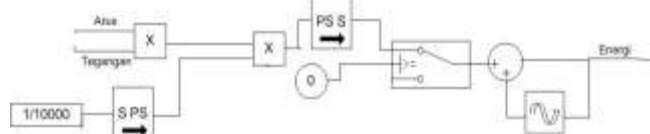
Adapun bagian-bagian dari blok ini dimulai dari Blok *Power Supply*. Blok *power supply* merupakan representasi dari sumber listrik PLN. Pada blok ini terdapat hambatan dalam yang nilainya diatur sedemikian rupa sehingga memiliki *output* pemodelan yang sesuai dengan *output* percobaan. Pada komponen sumber AC terdapat kolom bertuliskan peak amplitude. Karena sumber tegangan 220 V, maka peak amplitude menjadi  $200 \text{ VRMS} \times \sqrt{2} = 311.13 \text{ V}$ . Beban dalam eksperimen ini adalah lampu bohlam sebanyak 10 buah dengan masing-masing memiliki daya 100 Watt. Didalam pemodelan ini, beban direpresentasikan dengan sebuah resistor. Adapun nilai

dari masing-masing beban adalah sebagai berikut 484  $\Omega$ . Berikutnya adalah Kabel hitam dan kabel biru yang saling terhubung seri dipasang secara paralel terhadap sumber. Kedua kabel ini adalah kabel fleksibel dengan panjang 17m. Nilai resistansi dari kedua kabel ini berdasarkan table yang sudah dijelaskan diatas

Terdapat dua sensor utama, yaitu sensor arus dan sensor tegangan. Sensor tegangan digunakan untuk mengukur tegangan yang diletakkan dekat dengan sumber. Adapun sensor arus digunakan untuk mengukur arus sumber dan arus arc. Tegangan sumber dan arus arc kemudian dihubungkan dengan time scope. Sedangkan arus sumber digunakan untuk menghitung energi sumber.

Kejadian hubung singkat yang mengakibatkan terjadinya busur api direpresentasikan dengan dua buah *switch* dan sebuah resistor variabel. Yang pertama adalah Rangkaian *Switch On* yang berfungsi untuk mengatur kapan hubung singkat mulai terjadi. Kemudian ada Rangkaian *Switch Off* yang berfungsi untuk mengatur kapan hubung singkat berakhir. Kemudian ada Rangkaian R arc yang mendapatkan *input* berasal dari blok *neural network*.

Kemudian ada rangkaian penghitung *energy*. Rangkaian ini berfungsi untuk menghitung *energy* total yang dimulai ketika terjadi hubung singkat. Blok ini mendapatkan masukan dari keluraran sensor arus arc dan sensor tegangan. Keluaran dari rangkaian ini dimasukkan ke *output & display* dan JST.



Gambar 4. Rangkaian Penghitung Energi

Pada gambar diatas, PS Product sebanyak dua buah. PS Product yang pertama berfungsi untuk mengalikan nilai Arus yang berasal dari *Current sensor* dengan Tegangan yang berasal dari *Voltage Sensor*. PS Product yang kedua terdapat dua *input*. *Input* yang pertama adalah hasil pengali dari PS Product yang sebelumnya, yakni  $V \times I$ . *Input* yang kedua adalah representasi dari  $\Delta t$  sebesar 1/10000. *Output* dari PS Product yang kedua adalah energi. Sum adalah komponen yang berfungsi untuk menjumlahkan dua *input*. Salah satu *input* adalah feedback dari *output* pada time step sebelumnya. Sedangkan delay berfungsi untuk memberikan jarak waktu untuk penjumlahan feedback sesuai dengan nilai  $\Delta t$ , yakni sebesar 0.0001 s. *Switch* memiliki dua jenis *input*. Yakni control port dan *input* port. Pada pemodelan ini *input* port diisi oleh clock. *Switch* memiliki nilai *threshold*. Nilai *threshold* menjadi batas *switch* untuk on ke sisi *input*. Ketika nilai control port sudah mencapai nilai *threshold*, *switch* akan on ke sisi *input* port. Dengan demikian, rangkaian ini berfungsi untuk mengatur kapan nilai *energy* dapat mulai dihitung. Dibawah ini adalah gambar *switch* dan contoh pemilihan nilai *threshold* yang dalam penelitian ini berupa waktu.

**1. Blok Jaringan Saraf Tiruan**

Pada penelitian kali ini digunakan sebuah JST yaitu sebagai penentu nilai dari R arc. *Input* dari JST ini adalah jumlah

serabut dan *energy*. Adapun metode yang digunakan adalah elman neural. Blok JST ini didapatkan menggunakan library matlab yang kemudian dilatih sesuai dengan parameter-parameter yang telah ditentukan. Neural *network* yang digunakan adalah neural *network* dari masing-masing serabut. Adapun *switch* control menentukan neural *network* mana yang akan dieksekusi.

Seperti yang sudah diterangkan sebelumnya, *input* dari JST kali ini adalah jumlah serabut dan jumlah energi. Adapun *output*nya adalah Rarc. Dalam rangkaian diatas, nilai S adalah representasi dari jumlah serabut, nilai E adalah representasi dari jumlah *energy* dan nilai R adalah representasi dari nilai Rarc

## 2. Komponen Pendukung

Komponen pendukung disini memiliki peran untuk melengkapi rangkaian-rangkaian utama yang ada diatas. Dimulai dengan komponen *solver configuration*. Komponen ini adalah komponen fisikal yang harus ada dalam setiap rangkaian fisikal pada Simulink. *Solver configuration* berfungsi untuk menentukan banyaknya frekuensi sampling pada pemodelan. Karena dalam percobaan menggunakan frekuensi sampling 10.000/s. maka, pada sample time diinputkan nilai 0.0001. Kemudian ada komponen *From & GoTo*. Komponen ini adalah komponen Simulink. *From* berfungsi untuk menerima sinyal dari *GoTo* yang memiliki label yang sama. *GoTo* mendapatkan masukan dari rangkaian yang ada dibelakangnya, adapun *from* memberikan masukan pada rangkaian yang ada didepannya. Simulink to *Physical Signal Converter* (S-PS) adalah sebuah komponen yang berfungsi untuk mengubah nilai simulink menjadi nilai fisikal. Sedangkan *Physical to Simulink Converter* (PS-S) adalah sebuah komponen yang berfungsi untuk mengubah nilai fisikal menjadi nilai Simulink. Kedua komponen ini digunakan dalam rangkaian pemodelan yang dapat berupa komponen fisikal maupun komponen Simulink sehingga dapat saling terhubung satu sama lain. Komponen *display* berfungsi untuk menampilkan nilai dari sebuah masukan dalam timesetep tertentu. Komponen *scope* berfungsi untuk menampilkan sebuah masukan dalam bentuk gambar. Dalam penelitian ini, *scope* digunakan untuk menampilkan bentuk gelombang dari tegangan sumber dan arus *arcing* yang sebelumnya terbaca oleh *voltage sensor* dan *current sensor*

## IV. HASIL DAN PEMBAHASAN

### A. Pelatihan Jaringan Saraf Tiruan (JST)

Dalam melakukan pelatihan pada jaringan saraf tiruan dengan metode Elman Neural *Network*, telah disebutkan sebelumnya bahwa pembuatan perangkat JST menggunakan toolbox pada matlab.

#### 1. Parameter Pelatihan

Seperti yang sudah disebutkan diatas, metode pelatihan TRAINLM adalah yang paling presisi, namun, hal ini perlu dibuktikan lebih lanjut. Maka dari itu, akan diuji berbagai macam metode pelatihan yang digunakan untuk memodelkan gelombang serabut 1. Parameter-parameter lainnya seperti

jumlah hidden layer, jumlah neuron, dan lain-lain dibuat sama. Hasilnya adalah TRAINLM

#### 2. Jumlah *Hidden Layer*

Sesuai dengan pernyataan sebelumnya, bahwa jumlah hidden layer dengan *input* lebih dari satu sebaiknya tidak lebih dari 2. Namun disini akan diuji berbagai jumlah hidden layer dengan parameter-parameter lainnya dibuat sama pula. Hasilnya adalah 2 hidden layer yang terbaik

#### 3. Jumlah Neuron

Pengujian yang terakhir adalah dengan menentukan jumlah neuron. Jika mengikuti script neural *network* pada matlab, *default* dari neuron adalah 10. Hasilnya adalah 20 neuron yang terbaik

#### 4. Pembuatan Neural *Network*

Setelah berbagai percobaan diatas, diketahui parameter-parameter terbaik dibanding dengan parameter lainnya. Dapat disimpulkan jenis-jenis parameter yang dipilih adalah:

Metode	Metode Train	Jumlah Hidden Layer	Jumlah Neuron
Elman Backprop	TRAINLM	2	20

#### 5. Penentuan *Input* & Target Pelatihan

Setelah arsitektur NN dibuat, kemudian yang dilakukan adalah menentukan *input* dan target pelatihan NN tersebut. dalam penelitian kali ini, *input* dari neural *network* ada dua, yakni serabut dan jumlah *energy*. Sedangkan targetnya adalah *Rarcing*

#### 6. Memulai Pelatihan & Monitoring

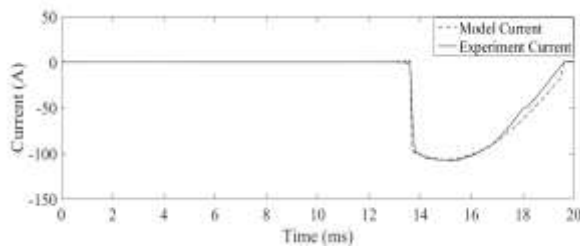
Setelah *input* dan target pelatihan ditentukan, berikutnya adalah menentukan parameter-parameter pelatihan di dalam neural *network* yang telah dibuat. Contohnya adalah *epochs* (jumlah iterasi), target, dan lain-lain. Setelah parameter ditentukan, train dapat dimulai

#### 7. Hasil Pelatihan Neural *Network*

Pelatihan neural *network* dalam percobaan ini menggunakan parameter yang terbaik hasil dari percobaan diatas. Metode pelatihan menggunakan trainlm, jumlah hidden layer 2 dan jumlah neuron 20. Dalam penelitian ini, serabut yang dilatih adalah serabut 1,3,6,12 dan 24.

#### B. Hasil Pemodelan dengan Data Training

Simulasi model dilakukan sebanyak lima kali, sebanyak serabut yang akan dimodelkan, yaitu serabut 1,3,6,12,24. Data simulasi model tersebut kemudian dibandingkan dengan data percobaan yang digunakan untuk melakukan training JST menggunakan metode *Mean Absolute Percentage Error*, sehingga didapatkan tingkat akurasi dari masing-masing pemodelan. Arus *arcing* selesai menggunakan pengaturan *switch off* secara manual.



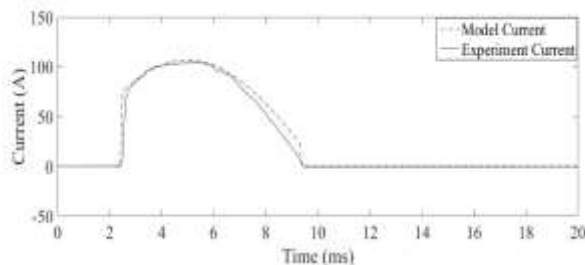
Gambar 5. Perbandingan Arus Pemodelan dan Percobaan (Data Training) Serabut 1

Tabel 3.  
Error Pemodelan

Jumlah Serabut	Jenis Pemodelan	MSE
Serabut 1	Tegangan	58.72276883
	Arus	16.65445803
Serabut 3	Hambatan	49.93474312
	Tegangan	100.0965751
Serabut 6	Arus	17.77324287
	Hambatan	200.3116234
Serabut 12	Tegangan	201.8148888
	Arus	143.9601619
Serabut 24	Hambatan	31.36391
	Tegangan	318.5309863
Serabut 12	Arus	46.71694074
	Hambatan	4.440189086
Serabut 24	Tegangan	9627,118
	Arus	1244,925
	Hambatan	0,0093

C. Hasil Pemodelan dengan Data Non-Training

Hasil pemodelan dengan data non-training maksudnya adalah menguji program menggunakan data yang berbeda dengan data yang digunakan untuk melakukan training JST. Arus *arcing* selesai menggunakan pengaturan *switch off* secara manual.



Gambar 6. Perbandingan Arus Pemodelan dan Percobaan (Data Non-Training) Serabut 1

V. KESIMPULAN

Dari Pemodelan arus *arcing* diatas, didapatkan nilai MSE dan untuk masing-masing serabut adalah:

- Serabut 1: MSE = 16.34543
- Serabut 3: MSE = 16.24647
- Serabut 6: MSE = 44.12924
- Serabut 12: MSE = 21.32577
- Serabut 24: MSE = 1244.925

Durasi *arcing* cenderung meningkat seiring dengan bertambahnya jumlah serabut, namun dalam beberapa kasus

terjadi sebaliknya. Adapun nilai arus puncak dari *arcing* cenderung sama.

Arc flash tegangan rendah tidak membahayakan bagi pengguna, namun jika arc flash terjadi dekat dengan bahan yang mudah terbakar dapat memicu terjadinya kebakaran.

DAFTAR PUSTAKA

- [1] J. Maida, P.E., “*Arc Flash Overview and Qualifications*”, Maida Engineering Inc, Fort Washington, 2009
- [2] T. Gammon and J. Matthews, “*The Historical Evolution of Arcing-Fault Models for Low-Voltage Systems*”, Industrial & Commercial Power Systems Technical Conference, 1999 IEEE, pp. 1-6
- [3] J. J. Lowke, “Simple Theory of Free-Burning Arcs,” *Journal of physics D: Applied Physics*, vol. 12, 1979, pp. 1873-1886
- [4] T. H. Lee, “Plasma physics and the interruption of an electric circuit,” *Proc. of the IEEEX*, vol. 57, Mar. 1969, pp. 307-323
- [5] Albert M. Smoak, P.E., Adam J. Keeth, “*An Investigation of Low Voltage Arc Flash Exposure*”, IEEE 2013, pp. 184
- [6] Albert M. Smoak, P.E., Adam J. Keeth, “*An Investigation of Low Voltage Arc Flash Exposure*”, IEEE 2013, pp. 184
- [7] C.T. Lin, and C.S.G. Lee, *Neural fuzzy systems: A neuro-fuzzy synergism to intelligent systems*. Upper Saddle River, NJ: Prentice-Hall Inc., 1996
- [8] L. Fausset, *Fundamentals of neural networks: architectures, algorithms, and applications*, New Jersey: Prentice-Hall, 1994
- [9] O. R. Schurig, “Fault voltage drop and impedance at short-circuit currents in low-voltage circuits,” *AIEE Trans.*, vol. 60, 1941, pp. 479-487
- [10] N. Sanjaya, “*Pemodelan Busur Api (Arc Flash) Listrik Tegangan berbasis Jaringan Syaraf Tiruan (Artificial Neural Network)*”, Universitas Dian Nuswantoro, Semarang, 2015