

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian pustaka

Kajian pustaka dalam bab ini akan membahas tentang beberapa penelitian serupa yang telah dilakukan sebelumnya. Penelitian-penelitian tersebut menjadi dasar dan acuan yang digunakan penulis dalam melakukan penelitian yang akan dilakukan saat ini dengan judul “Peramalan Jumlah Pemakaian Air dengan Metode *Extreme Learning Machine* (ELM) dan *Ant Colony Optimization* (ACO)”. Untuk penelitian terkait yang menjadi dasar dalam penelitian ini dijelaskan didalam Tabel 2.1.

Penelitian selanjutnya adalah penelitian yang dilakukan oleh Khotimah dkk, dalam penelitian yang membandingkan metode ELM dengan metode Backpropagation. Dalam penelitian ini menunjukkan bahwa nilai keakuratan kesalahan backpropagation dengan nilai MSE = 1,99% dan MAPE = 0,4342% dimana nilainya lebih besar dibandingkan dengan ELM yang menghasilkan nilai MSE = 0,96 dan MAPE = 0,31 (Khotimah, et al., 2010).

Penelitian selanjutnya adalah penelitian yang dilakukan oleh Virginia dkk, penelitian untuk peramalan data *time series* (Studi Kasus: Saham Bank BRI) melakukan peramalan penutupan harga saham harian Bank BRI. Penelitian tersebut menghasilkan bahwa hasil dari peramalan bergantung pada arsitektur jaringan yang di bangun seperti banyak *layer* dan *neuron* yang digunakan, percobaan ini melakukan 15 kali percobaan dengan jumlah *hidden layer* yang berbeda-beda dan didapatkan nilai RMSE terkecil dengan nilai 21.58 yang menggunakan 7 *hidden layer* (Sari, 2017).

Penelitian berikutnya dilakukan oleh Pangaribuan (2016) yang menggunakan metode *Extreme Learning Machine* untuk diagnosa penyakit diabetes meilitus. Dengan menggunakan data set UCI yang menggunakan 8 buah variabel *input*, menghasilkan bahwa jumlah *hidden neuron* berpengaruh pada hasil prediksi. Dengan membandingkan prediksi dengan ELM yang memiliki nilai MSE sebesar 0.393 dibandingkan prediksi dengan *backpropagation* dengan nilai MSE 0.842, sehingga didapatkan bahwa ELM melakukan prediksi 2.14 kali lebih baik dibandingkan dengan *backpropagation* (Pangaribuan, 2016).

Penelitian berikutnya yang dilakukan oleh Kadri dkk, penelitian ini bertujuan mendeteksi kesalahan dalam maufaktur. Penelitian ini menggunakan algoritme Ant Colony Optimization (ACO) untuk deteksi pemantauan kondisi klin semen puta. Algoritme ACO menemukan fitur subset dan *Extreme Learning Machine* (ELM) yang memungkinkan akurasi yang lebih baik. Untuk mengevaluasi hasil penelitian tersebut menggunakan dataset dan menghasilkan efektivitas dan efisiensi yang lebih tinggi dibandingkan dengan metode lainnya (Kadri & Mouss, 2017).

Tabel 2.1 Kajian pustaka

No.	Penulis	Objek	Metode	Hasil
1	(Khotimah, et al., 2010)	Objek: sistem peramalan <i>Input:</i> Data parameter, data training, data testing, bobot	Metode perbandingan ELM dan Backpropagation Tahap: 1. Proses autokorelasi 2. Proses normalisasi 3. Pelatihan data 4. Menentukan bobot terbaik 5. Hitung evaluasi 6. Pengujian data	Hasil: hasil peramalan Akurasi: MSE = 1,100 %
2	(Sari, 2017)	Objek: Penutupan harga saham harian Bank BRI Data: harga penutupan saham 1 Januari 2016 – 24 Januari 2016	Metode <i>Extreme Learning Machine</i> Tahap: 1. Normalisasi data 2. Inisialisasi bobot dan bias 3. <i>Input</i> parameter 4. Proses feedforward 5. Cek stoping condition	Hasil: hasil penelitian ini membuktikan bahwa nilai evaluasi bergantung pada arsitektur jaringan yang dibuat. Hasil terbaik dari 15 percobaan menghasilkan RMSE sebesar 21.580
3	(Pangaribuan, 2016)	Objek: diagnosa diabetes melitus Data: <i>data set</i> diabetes UCI <i>Input:</i> 1. Umur 2. Index masa tubuh 3. Tekanan darah rendah 4. Glukosa darah 5. Jumlah kehamilan 6. Fungsi asal-usul diabetes 7. Serum insulin 8. Ketebalan lipatan kulit	Metode Extreme Learning Machine (ELM) Tahap: 1. Normalisasi data 2. Inisialisasi bobot dan bias 3. <i>Training</i> 4. <i>Testing</i>	Hasil: 9. ELM lebih cepat dibanding <i>backpropagation</i> 10. Akurasi ELM lebih bagus dibanding <i>backpropagation</i> Kecepatan proses: 0.004 MSE : 0.393
5	(Kadri & Mouss, 2017)	Objek: deteksi kesalahan pada mesin pemutar semen <i>Input:</i> Data kriteriai	Metode ELM dan ANT CO Tahap: 1. Inisialisasi <i>pheromone</i> 2. Hitung FH(a) 3. Evaluasi solusi yang ditemukan oleh semut	Hasil: hasil deteksi kesalahan alat Akurasi: 77%

Tabel 2.1 (lanjutan)

No.	Penulis	Objek	Metode	Hasil
			<ol style="list-style-type: none"> 4. <i>Upgrade phenomone</i> dari net dengan solusi optimal 5. Uji stoping kondisi 	
5	(Kadri & Mouss, 2017)	<p>Objek: deteksi kesalahan pada mesin pemutar semen</p> <p><i>Input:</i> Data kriteriai</p>	<p>Metode ELM dan ANT CO</p> <p>Tahap:</p> <ol style="list-style-type: none"> 1. Inisialisasi <i>pheromone</i> 2. Hitung FH(a) 3. Evaluasi solusi yang ditemukan oleh semut 4. <i>Upgrade phenomone</i> dari net dengan solusi optimal 5. Uji stoping kondisi 	<p>Hasil: hasil deteksi kesalahan alat</p> <p>Akurasi: 77%</p>
6	(Penulis, 2018)	<p>Objek: Pemakaian jumlah air sebagai pembangkit listrik tenaga uap</p> <p><i>Input:</i> jumlah pemakaian air</p>	<p>Metode ELM-ACO</p> <p>Tahap:</p> <ol style="list-style-type: none"> 1. Inisialisasi parameter ACO 2. Menentukan node awal semut 3. Menentukan node selanjutnya 4. <i>Update pheromone local</i> 5. Memperbarui nilai node awal 6. Konversi nilai parameter 7. <i>Trainig</i> ELM 8. <i>Testing</i> ELM 9. <i>Update</i> feromon global 10. <i>Testing</i> ELM 11. Evaluasi hasil peramalan 	<p>Hasil: hasil peramalan pemakaian air</p> <p>Akurasi: Mape sebesar 0.170 %</p>

2.2 Air

Air merupakan sumber energi yang murah dan mudah didapat, pada air terdapat energi potensial dan energi kinetik. Kebutuhan air adalah jumlah air yang di butuhkan untuk memenuhi kebutuhan makhluk hidup diantaranya keperluan rumah tangga, pengelolaan kota, industri dan yang lainnya. Energi yang dimiliki air dapat dimanfaatkan dan di gunakan dalam wujud energi mekanis dan energi listrik (Jatmiko, et al., 2012). Kebutuhan air diperlukan untuk bermacam-macam tujuan

diantaranya kebutuhan domestik dan kebutuhan non domestik (Suryadmaja, et al., 2015).

Dalam pembangkitan listrik tenaga uap di PT. PJB memanfaatkan air laut, air laut tersebut dimurnikan atau dilakukan desalinasi untuk menjadi air tawar yang disebut dengan *make-up water* atau air yang sudah dimurnikan. Setelah air laut dimurnikan maka akan ditampung ke dalam tangki penampung air yang akan digunakan sebagai bahan bakar sebagai tenaga pembangkit listrik tenaga uap.

Pembangkit Listrik Tenaga Uap (PLTU) adalah pembangkit listrik yang bekerja dengan mengendalikan energi kinetik dari uap air untuk menghasilkan tenaga listrik. Bentuk dari PLTU terdiri dari generator yang dihubungkan dengan turbin, untuk memutar turbin diperlukan suatu energi kinetik dari uap panas atau kering. Konsumsi energi pada peralatan dalam PLTU berasal dari putaran turbin karena uap panas. Untuk menghasilkan uap, perlu adanya proses pemanasan air dengan melakukan pembakaran. Untuk menghasilkan uap panas itu, diperlukan air yang diproses dengan cara *boiling*. Air yang digunakan adalah air yang memiliki standar tertentu. Penggunaan air laut dalam proses industri tidak dapat dilakukan dikarenakan berbahaya bagi mesin-mesin dalam industri. Sehingga diperlukan suatu proses pemurnian air.

Siklus air dalam PLTU akan terus bersirkulasi tanpa adanya pengurangan massa air sehingga pada siklus tersebut tidak perlu penambahan air dari luar siklus. Tetapi pada proses PLTU sering terjadi berkurangnya massa air yang disebabkan oleh kebocoran-kebocoran dalam sistem, *tempering*, dan pembuangan gas yang masih mengandung air. Sehingga walaupun air digunakan dalam siklus namun karena faktor-faktor tersebut siklus masih membutuhkan penambahan air (Johneri, 2014).

2.3 Peramalan

Peramalan adalah suatu perkiraan atau proses meramalkan sesuatu yang belum terjadi dan akan terjadi pada masa mendatang baik berupa kualitas, kuantitas, waktu atau lokasi yang dibutuhkan untuk memenuhi permintaan (Laksana, 2012). Sedangkan prediksi merupakan adalah perkiraan berdasarkan subjek ataupun metode ilmiah, seperti prediksi cuaca, prediksi bencana, prediksi keuangan dan yang lainnya (Ririanti, 2014).

2.3.1 Metode peramalan

Peramalan kuantitatif adalah peramalan yang melakukan proses komputasi matematis pada data historis secara objektif. Penggunaan metode peramalan kuantitatif dapat dilakukan pada saat data historis dan pola aspek masa depan tersedia (Hyndman, 2009). Metode kuantitatif adalah jenis metode peramalan berdasarkan pada model yang bersifat objektif dan sangat dipengaruhi oleh perhitungan matematika. Model asosiatif atau sebab akibat dan *time series* dapat dijadikan sebagai dasar teknik peramalan kuantitatif (Montgomery, et al., 2015).

2.3.2 Evaluasi hasil peramalan

Evaluasi hasil peramalan dapat diukur dengan tujuan mengetahui tingkat keakuratan dan seberapa baik hasil dari peramalan dengan data aktual yang ada. Nilai evaluasi merupakan nilai kesalahan yang merupakan selisih nilai perbandingan yang disebut sebagai nilai *error rate*. Semakin kecil nilai *error rate* yang dihasilkan maka hasil peramalan akan semakin bagus. Ada 4 evaluasi yang dapat digunakan, salah satunya yaitu MAPE (Crone, et al., 2007).

2.3.2.1 Mean Absolute Percentage (MAPE)

Mean Absolute Percentage (MAPE) adalah ukuran kesalahan relatif. MAPE menyatakan persentase kesalahan hasil peramalan terhadap permintaan aktual. Untuk menghitung MAPE yaitu dengan menjumlahkan seluruh rata-rata nilai dari data aktual dan data hasil peramalan, MAPE sering di gunakan dalam proses peramalan ekonomi (Bratu, 2012). Rumus MAPE ditunjukkan dengan Persamaan 2.1 berikut. Untuk menilai kemampuan menunjukkan model berdasarkan nilai MAPE di tunjukkan pada Tabel 2.2.

$$MAPE = \frac{\sum_{i=1}^n \left(\frac{|y_t - y'_t|}{y_t} \right) \times 100\%}{n} \quad (2.1)$$

Keterangan:

- y_t = nilai data aktual
- y'_t = nilai hasil prediksi
- n = jumlah data

Tabel 2.2 Kemampuan menunjukkan model

MAPE	Signifikan
< 10%	Kemampuan peramalan sangat baik
10% - 20%	Kemampuan peramalan baik
20% - 50%	Kemampuan peramalan layak
>50%	Kemampuan peramalan buruk

Sumber: (Chang, et al., 2007)

2.4 Jaringan Syaraf Tiruan (JST)

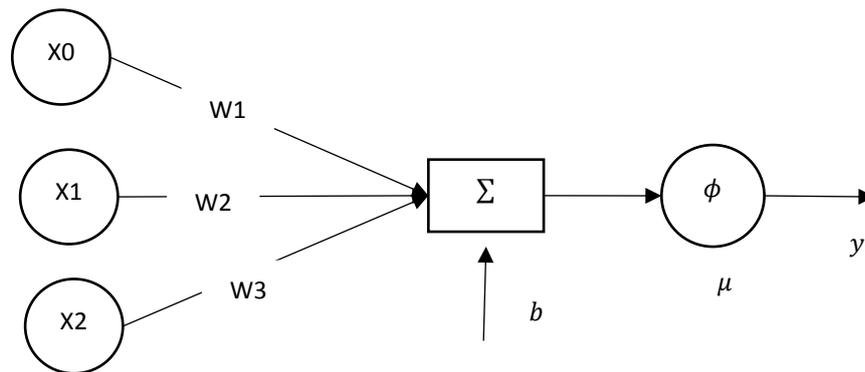
Jaringan Syaraf Tiruan (JST) adalah suatu kelompok dari jaringan yang saling berhubungan yang menyerupai kinerja sistem saraf biologis, seperti halnya susunan dalam proses syaraf manusia. JST merupakan salah satu tiruan dari otak manusia yang dapat melakukan proses pembelajaran seperti otak manusia (Lesnussa, et al., 2015). Kunci dari struktur sistem pengolahan informasi yang terdiri dari sejumlah besar elemen-elemen pemrosesan yang saling terhubung dan berkaitan dan bekerja sama untuk menyelesaikan suatu permasalahan. Cara kerja JST seperti halnya cara kerja jaringan pada tubuh manusia. JST diadaptasi oleh suatu proses pembelajaran atau lebih sering disebut *learning* untuk mengenali pola dari permasalahan (Bakhrun, 2013).

2.4.1 Model JST

Seberapa baik suatu model JST ditentukan oleh beberapa hal berikut:

1. Arsitektur jaringan, merupakan sebuah arsitektur yang menentukan pola yang terbentuk antar neuron.
2. Metode pembelajaran, metode metode untuk proses pembelajaran dan pengenalan pola dari suatu masalah.
3. Fungsi aktivasi.

Proses matematis pada JST dijelaskan pada Gambar 2.3.



Gambar 2.1 Struktur neuron Jaringan Syaraf Tiruan

2.5 Extreme Learning Machine (ELM)

Extreme Learning Machine (ELM) adalah suatu metode pembelajaran *feedforward* sederhana dengan satu buah *hidden layer* atau lebih sering disebut *Single Hidden Layer Feedforward Neural Network* (SLFNs). Metode ELM pertama kali diperkenalkan oleh Huang pada tahun 2004. Metode ini dikembangkan untuk memperbaiki kekurangan-kekurangan dari metode yang telah ada sebelumnya (Sun, et al., 2008).

JST *feedforward* memiliki *speed* rendah karena beberapa alasan sebagai berikut:

1. Dalam melakukan proses *training* menggunakan *slow gradient based learning algorithm*.
2. Proses *training* digunakan untuk menentukan seluruh parameter pada jaringan yang ditentukan secara iteratif.

Pada algoritme ELM parameter-parameter yang digunakan yaitu *input*, bobot, dan *hidden bias* ditentukan secara acak atau *random* dari nilai tertentu. Sehingga ELM memiliki proses pembelajaran atau *learning speed* yang cepat. Dengan menggunakan *random* pada *range* tertentu, sehingga dapat memberikan hasil prediksi yang lebih stabil (Ramadhanty, 2017).

Perhitungan dengan algoritme ELM terbagi menjadi 2 tahap, yaitu proses *training* dan proses *testing*.

2.5.1 Proses *Training*

Langkah awal metode ELM untuk melakukan suatu peramalan yaitu melakukan proses *training*. Tujuan dari proses ini adalah menghasilkan bobot keluaran dengan nilai *error* yang rendah. Tahapan proses *training* metode ELM adalah:

1. Normalisasi data

Normalisasi data digunakan untuk mengubah data menjadi angka bernilai antara 0 sampai 1 dengan *min-max normalization* (Saranya & Manikandani, 2013). Untuk menghitung normalisasi data dapat menggunakan Persamaan 2.2.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2.2)$$

Keterangan:

- x' = data hasil normalisasi
- x = data yang akan dinormalisasi
- x_{max} = data terbesar dataset
- x_{min} = data terkecil dataset

2. Inisialisasi *input weight* serta bias dengan nilai *random* antara 0 sampai dengan 1 (Liang, et al., 2006).
3. Menghitung seluruh keluaran di *hidden layer* dengan menggunakan fungsi aktivasi.

Fungsi aktivasi sigmoid biner memiliki nilai antara 0 sampai 1, karena vektor *input* dan target *output* yang dipresentasikan sesuai dengan fungsi aktivasi sigmoid maka nilainya harus diantara 0 sampai 1. Fungsi aktivasi sigmoid dapat dirumuskan dengan Persamaan 2.3.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.3)$$

Keterangan:

- $f(x)$ = fungsi aktivasi sigmoid biner
- e^{-x} = eksponen

Perhitungan output hidden layer ditunjukkan pada Persamaan 2.4.

$$H_{init\ ij} = \left(\sum_{k=1}^n w_{jk} \times x_{ik} \right) + b_j \quad (2.4)$$

Keterangan:

- H_{init} = matriks keluaran *hidden layer*
- i = [1,2,...,N], N merupakan jumlah data
- j = [1,2,...,M], M merupakan jumlah *hidden neuron*
- n = jumlah *input neuron*

w = bobot awal
 x = *input*
 b = bias

4. Menghitung *output weight* dari *hidden layer* ke *output layer*. Dalam melakukan perhitungan *output weight*, langkah awal yaitu menghitung matriks *Moore Penrose Pseudo Invers* dari hasil output *hidden layer* dengan menggunakan fungsi aktivasi sigmoid. Kemudian baru dihitung *output* bobot menggunakan Persamaan 2.5 dan menghitung hasil prediksi dengan Persamaan 2.6.

$$\beta = H \times T \quad (2.5)$$

$$y = H \times \beta \quad (2.6)$$

Keterangan:

β = matriks bobot keluaran dari *hidden layer*
 H = matriks keluaran *hidden layer*
 H^+ = matriks *Moore Penrose Pseudo Invers* dari matriks H
 T = matriks target
 y = hasil peramalan

2.5.2 Proses *Testing*

Setelah proses pelatihan selesai, maka dilakukan suatu proses *testing*. Proses *testing* digunakan untuk melakukan uji coba hasil *training* untuk mendapatkan nilai akurasi peramalan (Siwi, 2016). Beberapa tahapan yang dilakukan ketika melakukan proses *testing* antara lain (Cholissodin, et al., 2017):

1. Mendapat nilai w_{mn} , b , dan β dari proses *training* yang telah dilakukan.
2. Menghitung matriks keluaran pada *hidden layer* dengan Persamaan 2.7.

$$H = \frac{1}{1 + \exp(-(x_{test}w^T + b))} \quad (2.7)$$

Keterangan:

H = matriks keluaran *hidden layer*
 x_{test} = matriks data normalisasi
 w^T = matriks *transpose* bobot
 i_{test} = jumlah data uji
 b = bias

3. Menghitung hasil keluaran *hidden layer* dengan menggunakan Persamaan 2.4.
4. Denormalisasi hasil peramalan dengan menggunakan Persamaan 2.8.

$$x = x'(max - min) + min \quad (2.8)$$

Keterangan:

x'	= nilai data asli
x	= nilai data normalisasi
min	= nilai minimum data set
max	= nilai maksimum data set

5. Evaluasi kesalahan menggunakan MAPE dengan Persamaan 2.1.

2.6 Ant Colony Optimization (ACO)

Ant Colony Optimization (ACO) pada awalnya diperkenalkan oleh Dorigo dan Stuzle pada tahun 1990 awal, metode ACO didasarkan pada perilaku dari koloni semut. ACO adalah strategi optimasi yang bersifat *stochastic*. ACO adalah populasi semut yang bergerak secara independen. Komunikasi antar semut dalam suatu koloni menggunakan jejak yang ditinggalkan berupa *pheromone*. *Pheromone* adalah zat yang seiring waktu berjalan akan menguap (Ariyasingha & Fernando, 2015).

2.6.1 Karakteristik ACO

Metode ACO memiliki beberapa karakteristik antara lain (Putra & Fatichah, n.d.):

1. Memanfaatkan interaksi semut, yang mana semut hanya melakukan tugas untuk menemukan solusi.
2. Informasi dari iterasi sebelumnya berupa feromon yang digunakan untuk menentukan hasil selanjutnya.
3. Menambahkan feromon adalah mekanisme ulas balik pada suatu node dan *evaporaton* feromon untuk mengurangi jumlah *pheromone*.

2.6.2 Algoritme ACO

Tahap-tahap metode ACO atau metode semut adalah sebagai berikut (Dorigo, et al., 2006):

1. Inisialisasi awal parameter ACO

Pada inisialisasi ini perlu menentukan posisi awal, jumlah semut dan inisialisasi awal nilai dari parameter.

2. Konstruksi solusi semut

Seluruh semut ditempatkan pada setiap titik secara *random*, kemudian secara bergantian dan berurutan berjalan menelusuri titik yang saling terhubung dan membentuk suatu solusi.

3. Menghitung nilai *pheromone* tiap semut

Semut terbaik dapat ditentukan dengan melihat semut yang memiliki feromon paling banyak.

4. Memeriksa jumlah iterasi

Jika iterasi sudah mencapai maksimum iterasi yang telah ditentukan, maka iterasi akan berhenti, dan *feature subset* yang dihasilkan oleh semut terbaik akan digunakan sebagai solusi.

5. *Update pheromone*

Jika semua semut menghasilkan *feature subset* dan *pheromone* masing-masing maka membentuk *vertex* yang telah dilewati semut. Perubahan *pheromone* tersebut disebut dengan feromon lokal. Dan perubahan *pheromone* pada *feature subset* terbaik disebut dengan feromon global.

6. *Generate* populasi baru

Setiap semut di jalur kunjungan dihapus, dan melakukan proses pembentukan populasi semut baru dan kembali ke langkah 3.

7. Menentukan kondisi berhenti dari kualifikasi yang ditemukan.

2.7 Langkah-Langkah Metode ELM dengan optimasi ACO

Langkah-langkah dalam perhitungan metode ELM dengan optimasi ACO adalah (Hong, et al., 2011) :

1. Inisialisasi parameter ACO

Inisialisasi yang dibutuhkan untuk melakukan proses perhitungan dalam ACO diantaranya:

- Jumlah semut (m)
- Ordinat $[0, 9]$
- Iterasi maksimum (NC_{max})
- Tetapan penguapan *pheromone* (ρ) dengan nilai $0 < \rho < 1$
- Tetapan pengendalian intensitas visibilitas (β)
- Tetapan pengendalian *pheromone* (α)
- *Pheromone* awal (τ_0)
- Tetapan siklus semut (q_0) dengan nilai $[0,1]$

Setelah inisialisasi parameter ACO, langkah selanjutnya yaitu menentukan *range* untuk parameter ELM

2. Menentukan node semut

Semut akan mencari jalur secara acak (Lixing, et al., 2010). Langkah-langkah dalam menentukan node semut adalah:

Langkah 1. Menentukan secara acak posisi dari node awal (r) dengan nilai $[0, 9]$

Langkah 2. Menentukan secara acak nilai (q) antara 0 sampai 1 (Hong, et al., 2011).

Langkah 3. Menentukan node yang akan dikunjungi selanjutnya (u) dengan peluang perpindahan semut dari node awal (r) dengan menggunakan Persamaan 2.8 dan untuk menghitung nilai variabel acak dengan Persamaan 2.9 berikut (Lixing, et al., 2010).

$$P(r, u) = \begin{cases} \arg \max_{u \in J(r)} \{[\tau(r, u)]^2 \times [\tau(r, u)]^\beta\}, & \text{if } q \leq q_0 \\ S & \end{cases} \quad (2.8)$$

$$P(r, s) = \begin{cases} \frac{[\tau(r, u)]^\alpha \times [\tau(r, u)]^\beta}{\sum_{u \in J(r)} [\tau(r, u)]^\alpha \times [\tau(r, u)]^\beta}, & \text{if } s \in J(r) \\ 0, & \text{lainnya} \end{cases} \quad (2.9)$$

Keterangan:

q = nilai acak [0, 1]

q_0 = nilai tetapan siklus semut

τ = nilai intensitas feromon

r = node awal

u = node selanjutnya

α = tetapan pengendali intensitas feromon

β = tetapan pengendali visibilitas

$J(r)$ = kumpulan node pada matriks yang tersedia

s = nilai acak antara 0 sampai 9

Langkah 4. Update nilai feromon lokal $\tau(r, u)$ dengan dengan Persamaan 2.10 berikut (Lixing, et al., 2010).

$$\tau(r, u) = (1 - \rho) \times (\tau(r, u)) + \rho \tau_0 \quad (2.10)$$

Keterangan:

r = node awal

u = node selanjutnya

ρ = tetapan penguapan jejak feromon lokal

τ = nilai intensitas feromon

τ_0 = nilai intensitas feromon awal

Langkah 5. Setelah mendapat node yang baru, node selanjutnya (u) menjadi node awal (r) berikutnya. Ulangi Langkah 2 sampai semua node dilewati oleh semut.

3. Konversi nilai parameter ELM

$$NormW = \left[\left(\frac{A - A_{min}}{A_{max} - A_{min}} \right) \times (W_{max} - W_{min}) \right] + W_{min} \quad (2.11)$$

$$Normb = \left[\left(\frac{A - Amin}{Amax - Amin} \right) \times (bmax - bmin) \right] + bmin \quad (2.12)$$

Keterangan:

A = hasil optimasi ACO

$Amin$ = nilai minimal optimasi ACO

$Amax$ = nilai maksimal optimasi ACO

$Wmax$ = batas atas parameter *input weight*

$Wmin$ = batas bawah parameter *input weight*

$bmax$ = batas atas parameter bias

$bmin$ = batas bawah parameter bias

4. Melakukan proses ELM menggunakan parameter yang didapat dari ACO.
5. Menghitung *update* feromon global pada jalur menggunakan Persamaan 2.13.

$$\tau(r, s) = (1 - \delta) \times (\tau(r, s)) + \delta \Delta\tau(r, s) \quad (2.13)$$

$\Delta\tau(r, s)$ digunakan untuk meningkatkan *phenomone* pada jalur dengan Persamaan 2.14.

$$\tau(r, s) = \begin{cases} \frac{1}{L}, & \text{jika } (r, s) \in \text{globalbestroute} \\ 0, & \text{lainnya} \end{cases} \quad (2.14)$$

Keterangan:

r = node awal

u = node selanjutnya

τ = nilai intensitas feromon

$\Delta\tau(r, u)$ = matriks perubahan intensitas feromon global

L = MAPE *global best tour*