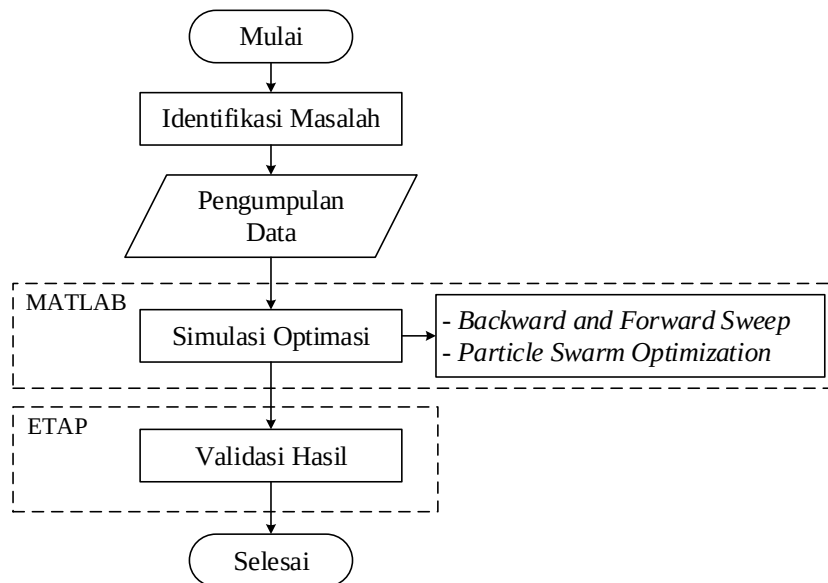


BAB III

METODE PENELITIAN

3.1 Flowchart Penelitian



Gambar 3. 1 Diagram alir penelitian

Dalam menjalankan penelitian tugas akhir ini, beberapa tahapan penelitian dilakukan seperti ditunjukkan pada Gambar 3.1.

1) Identifikasi Masalah

Identifikasi masalah dilakukan untuk mengetahui permasalahan yang timbul dan bersumber dari PT PLN (Persero) UP3 Tasikmalaya sebagai penyedia layanan listrik di daerah Tasikmalaya. Identifikasi masalah juga dilakukan dari literasi jurnal yang membahas topik yang sama mengenai permasalahan pada jaringan distribusi. Sehingga dalam penelitian ini akan dilakukan simulasi upaya teknik optimasi untuk mengurangi rugi-rugi daya terhadap kondisi lapangan berdasarkan hasil literasi.

2) Pengumpulan Data

Pengumpulan data dilakukan di Tasikmalaya sebagai tempat penelitian dengan menggunakan dua variabel data, yaitu data sekunder dan data primer.

a. Data Sekunder

Data sekunder dilakukan dengan cara interview dengan pihak PT PLN (Persero) UP3 Tasikmalaya untuk mendapatkan data jaringan penyulang TMSR, berupa:

- data jumlah transformator;
- data tipe transformator;
- data pembebanan transformator;
- data kapasitas transformator terpasang;
- data *Single Line Diagram* (SLD) penyulang TMSR.

b. Data Primer

Data primer bertujuan untuk data pendukung dan sebagai validitas data sekunder. Data primer didapatkan dengan cara identifikasi lapangan di daerah Kawasan penyulang TMSR mengenai:

- jumlah transformator;
- tipe transformator;
- kapasitas transformator;
- penempatan transformator;
- penghantar saluran;
- jarak saluran.

Pengumpulan data primer ini menggunakan bantuan *Global Positioning System* (GPS) untuk menandai lokasi bus beban (transformator), sehingga

nilai jarak saluran dapat diketahui. Selain itu, digunakan *smartphone* untuk mendokumentasikan transformator yang terdapat di lapangan.

3) Simulasi Optimasi

Data yang terkumpul akan dilakukan perhitungan aliran daya pada jaringan terkait menggunakan metode *Backward and Forward Sweep* (BFS) untuk mengetahui nilai drop tegangan dan rugi-rugi daya sebagai perbandingan dengan nilai hasil optimasi. Penyelesaian perhitungan aliran daya ini menggunakan *software* MATLAB R2023b yang mampu mengkalkulasi hasil dengan penggunaan fungsi dan perintah dalam bentuk algoritma coding.

Selanjutnya, simulasi optimasi dilakukan setelah mendapatkan hasil dari studi aliran daya, yang mana nilai rugi-rugi daya akan dioptimalkan dengan cara rekonfigurasi jaringan menggunakan metode perhitungan *Particle Swarm Optimization* (PSO) dalam *software* MATLAB R2023b.

4) Validasi Hasil

Validasi hasil dilakukan untuk memastikan bahwa tingkat error sangat kecil atau bahkan tidak ada error pada simulasi *software* MATLAB R2023b. Validasi dilakukan sebanyak dua kali yaitu sebelum dilakukan optimasi dan setelah dilakukan optimasi. Validasi ini dilakukan menggunakan *software* ETAP 19.0.1 yang mampu memodelkan jaringan distribusi dan melakukan percobaan serta analisis aliran daya.

Validasi hasil dilakukan dengan menggunakan persamaan standar deviasi dalam perhitungan data tunggal sebagai berikut:

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}} \quad (3.1)$$

Dimana :

- S = Standar deviasi
 x_i = Data ke-i (Hasil perhitungan ke-i)
 $\bar{x}$ = Rata-rata data
 n = Banyak data

3.2 Fungsi Objektif

Fungsi objektif yang digunakan dalam penelitian ini merupakan fungsi tujuan untuk mencari rugi-rugi daya aktif terkecil pada saluran dengan persamaan matematis sebagai berikut (Musdir et al., 2022).

$$f(x) = \min \left(\sum_{(i,j) \in B(x)}^N P_{L(i,j)} \right) \quad (3.2)$$

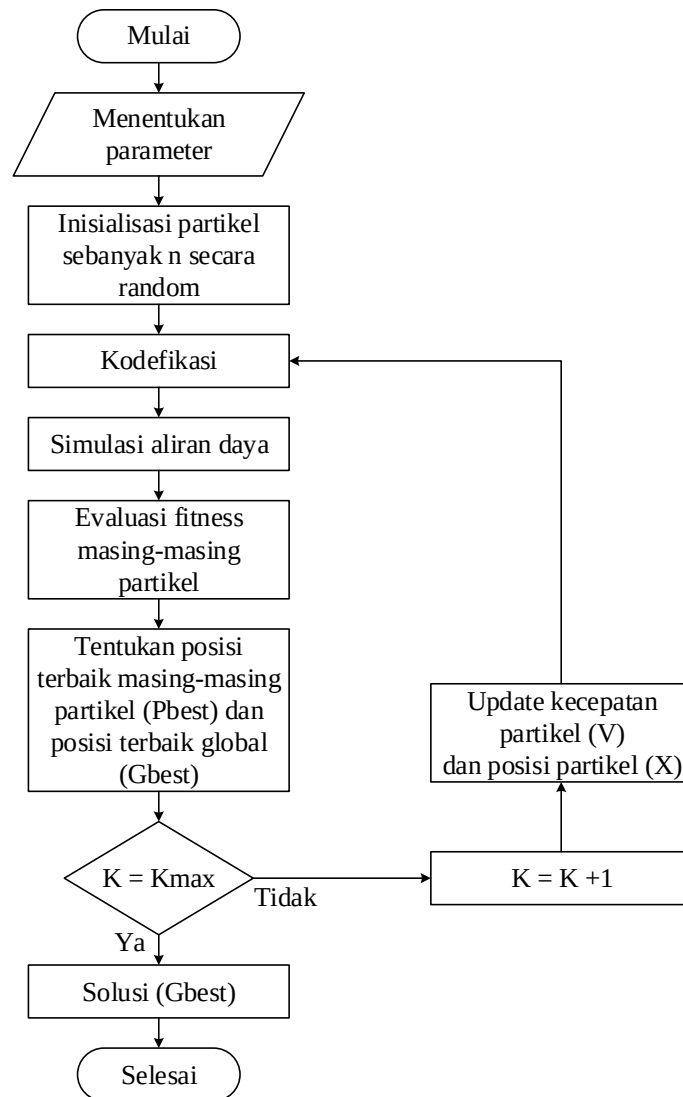
$$f(x) = \min \left(\sum_{(i,j) \in B(x)}^N I_{(i,j)}^2 R_{(i,j)} \right) \quad (3.3)$$

Keterangan:

- $f(x)$ = Fungsi rugi daya saluran pada jaringan (kW)
 $B(x)$ = Saluran pada konfigurasi jaringan x
 $P_{L(i,j)}$ = Rugi-rugi daya aktif pada saluran ke-(i,j) (kW)
 $I_{(i,j)}$ = Arus pada saluran ke-(i,j) (A)
 $R_{(i,j)}$ = Resistansi saluran ke-(i,j) (Ohm)
 N = Jumlah Saluran

3.3 Flowchart Optimasi *Particle Swarm Optimization* (PSO)

Dalam menjalankan optimasi, metode PSO memiliki tahapan-tahapan sebagai berikut.



Gambar 3. 2 Diagram alir metode optimasi *Particle Swarm Optimization* (PSO)

Berdasarkan Gambar 3,2 di atas, langkah-langkah algoritma dalam menentukan suatu keputusan menggunakan metode PSO diantaranya sebagai berikut:

1) Menentukan Parameter

Dalam hal ini, data merupakan sebuah parameter yang dijadikan sebagai batasan sebelum memulai optimasi. Parameter dalam metode *Particle Swarm Optimization* (PSO) diantaranya:

- a. Fungsi objektif atau fitness
 - b. Jumlah partikel (n_{partikel})
 - c. Posisi partikel (posisi_partikel)
 - d. Kecepatan partikel pada iterasi ke- i (v)
 - e. *Learning rate* untuk kemampuan individu ($c1$) dan pengaruh sosial kelompok ($c2$)
 - f. Bilangan acak antara 0 s.d. 1 dalam distribusi normal ($r1, r2$)
 - g. Posisi terbaik untuk setiap partikel dalam kelompok (P_{best})
 - h. Posisi terbaik dalam kelompok (G_{best}).
- 2) Melakukan inisialisasi partikel
- Inisialisasi partikel merupakan pemanggilan partikel yang mewakili posisi saluran baru (NS dan NR) yang dilakukan secara acak (*random*) dengan tujuan untuk memberikan informasi awal dalam rekonfigurasi saluran baru.
- 3) Kodefikasi
- 4) Melakukan simulasi aliran daya
- Silumasi aliran daya dilakukan untuk mengetahui nilai rugi-rugi daya saluran terhadap konfigurasi jaringan yang dijalankan.
- 5) Melakukan evaluasi terhadap nilai fitness pada masing-masing nilai partikel
- Evaluasi dilakukan untuk mengetahui perbandingan nilai fitness hasil perhitungan terhadap konfigurasi yang dijalankan berdasarkan kondisi posisi partikel.
- 6) Menentukan P_{best} dan G_{best}

$Pbest$ merupakan partikel terbaik dan $Gbest$ merupakan global terbaik yang merupakan pengalaman atau hasil dari setiap partikel dalam menghasilkan solusi terbaik berupa saluran baru (NS dan NR) ataupun kombinasi beberapa saluran baru.

7) *Update* kecepatan partikel dan posisi partikel

Secara matematis, proses pencarian posisi partikel dipengaruhi oleh kecepatan partikel v , yang dirumuskan sebagai berikut:

$$v_p^{k+1} = w \cdot v_p^k + c1r1(Pbest_p^k - X_p^k) + c2r2(Gbest^k - X_p^k) \quad (3.4)$$

Dimana:

w	= inersia
v_p^{k+1}	= kecepatan partikel p pada iterasi selanjutnya ($k=1$)
v_p^k	= kecepatan partikel p pada iterasi k
$c1, c2$	= <i>cognitive</i> dan <i>social learning</i>
$r1, r2$	= nilai acak antara 0 sampai 1
$Pbest_p^k$	= posisi terbaik masing-masing partikel p pada iterasi k
$Gbest^k$	= posisi terbaik dari seluruh partikel pada iterasi k
X_p^k	= posisi partikel p pada iterasi k

sedangkan persamaan *update* posisi partikel dirumuskan sebagai berikut:

$$X_p^{k+1} = X_p^k + v_p^{k+1} \quad (3.5)$$

- 8) Lakukan langkah 2 sampai 4, kemudian cek iterasi. Apabila sudah memenuhi nilai optimal, iterasi dapat dihentikan dan didapatkan nilai $Gbest$.

3.4 Model Jaringan

Model jaringan yang digunakan dalam penelitian ini yaitu penyulang TMSR (Tamansari) berupa penyulang tipe radial yang mencakup 40 bus, yang terdiri dari 1 bus GI sebagai bus utama, 1 bus *recloser* sebagai proteksi penyulang, dan 38 bus beban yang menampung beban-beban konsumen di lapangan seperti yang terlampir pada Lampiran.