

## BAB II

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 2.1 Tinjauan Pustaka

*Distributed Generation*(DG) atau pembangkit tersebar memiliki banyak fungsi tergantung dari fungsi tujuan seperti dapat memperbaiki profil tegangan, mengurangi rugi – rugi daya, meningkatkan keandalan, dan kualitas daya. Pada penelitian[1] menggunakan metode *Flower Pollination Algorithm* (FPA) dalam penentuan besarnya kapasitas dan lokasi yang optimal suatu pembangkit tersebar dalam sistem distribusi. Metode yang diujikan adalah pada sistem distribusi standar IEEE 33-bus dan 69-bus. Dari hasil penelitian didapatkan nilai rugi aktif berhasil diturunkan sebesar 82.87 kW dan meningkatkan profil tegangan menjadi 0.96 p.u[1].

Seiring dengan pertumbuhan penduduk, dibutuhkan pula energi listrik yang bisa memenuhi kebutuhan yang tinggi. Untuk itu diperlukan suatu keandalan pada sistem distribusi. Energi terbarukan menjadi salah satu tujuan untuk mengoptimalkan, pada penelitian[2] menggunakan *wind turbine* sebagai energi terbarukan. Investasi dalam implementasi *Distributed Generation*(DG) berbasis angin (*wind turbine*) umumnya dianggap terlalu mahal untuk diimplementasikan. Sehingga digunakan algoritma *Flower Pollination Algorithm*(FPA) untuk mencapai solusi penempatan DG yang optimal. Penelitian ini berhasil meningkatkan keandalan ditandai dengan reduksi indeks energi yang tidak tersuplai[2].

Penelitian yang dilakukan oleh(Reddy et al, 2017) berfokus pada pengaruh dari *Distributed Generation*(DG) pada sistem distribusi. Integrasi DG merupakan perubahan dari sistem distribusi tradisional menjadi sistem dengan banyak sumber. Pengaruh akibat penambahan DG seperti peningkatan level hubung singkat, pengurangan rugi daya, peningkatan keandalan, dan perbaikan profil tegangan. Keuntungan tersebut bergantung dari posisi dan lokasi DG yang ideal. Ideal posisi ini bergantung dari metode indeks vektor dari *Ant Lion Optimizer Algorithm*. ALO dimodelkan berdasarkan pada keunikan tingkah laku berburu pada semut singa.

Algoritma ini diujikan pada sistem distribusi standar IEEE 15-bus, sistem 33-bus, sistem 69-bus, dan sistem 85-bus. Dari hasil analisis didapatkan hasil yang optimal untuk injeksi DG tipe III dengan operasi faktor daya 0.9[3].

Penelitian yang dilakukan oleh(arabi-nowdes et al, 2018) membahas metode optimasi dengan tujuan untuk mengoptimalkan masalah ukuran dan penempatan pembangkit tersebar. Optimasi dilakukan dengan metode terbaru yakni *Ant Lion Optimizer* bergantung perbedaan fungsi tujuan. Fungsi tujuannya penelitian ini adalah mengurangi biaya pembelian, peningkatan keandalan, mengurangi biaya penggunaan, mengurangi rugi-rugi DG, dan mengurangi deviasi tegangan bus. Masalah ini diselesaikan sebagai optimasi multi tujuan dalam penambahan optimasi tujuan tunggal. Hasil yang ditunjukkan bahwa ALO memiliki performa terbaik dalam penerapan masalah penempatan dan ukuran DG yang optimal dengan perbandingan pada *Particle Swarm Optimization*(PSO) dan *Genetic Algorithm*(GA)[4].

*Distributed Generation*(DG) adalah sebuah pembangkit listrik kecil yang terhubung ke konsumen dalam sistem distribusi untuk meningkatkan profil tegangan, regulator tegangan, stabilitas, mengurangi rugi daya, dan keuntungan ekonomi. Keuntungan diatas dapat diperoleh dengan penempatan DG yang optimal. Sebuah algoritma yang terinspirasi alam disebut *Dragonfly Algorithm* digunakan untuk menentukan ukuran yang optimal dari unit DG. Ini dikembangkan berdasarkan perilaku aneh dari *Dragonflies* di alam. Paper[5] berfokus pada *Dragonflies* bagaimana mereka mencari makanan atau menjauh dari musuh. Algoritma ini diterapkan pada Sistem Distribusi standar IEEE 15-bus, 33-bus, dan 69-bus. Hasilnya menunjukkan bahwa algoritma ini dapat dibandingkan dengan algoritma evolusi yang lain. Ketika dibandingkan dengan algoritma lain, algoritma *Dragonflies* memberikan hasil terbaik, hasil terbaik berdasarkan pada DG dengan tipe III pada faktor daya 0.9

Pada penelitian yang dilakukan oleh(Bouktir et al, 2019) optimasi banyak tujuan diterapkan untuk mengoptimalkan masalah aliran daya. Secara efektif hasil tujuan ini, algoritma multi objektif (MOALO) ditujukan untuk menemukan optimal pareto untuk multi tujuan *Optimal Power Flow*(OPF). Tujuan pada paper ini adalah

memberikan solusi terbaik untuk masalah daya aktif dan daya reaktif *Optimal Power Flow*(OPF) dengan mengoptimasi 4 konflik fungsi tujuan secara bersamaan. Seperti untuk biaya generator, emisi polusi lingkungan, rugi daya aktif, dan deviasi tegangan. Performa dari algoritma MOALO diterapkan pada sistem IEEE 30-bus, 57-bus, 118-bus, dan 300-bus, serta pada Algerian DZ114 sistem bus. Hasilnya menunjukkan bahwa algoritma ini meningkatkan banyak hal ketika diaplikasikan ke sistem yang besar. Paling efektif dari metode tujuan telah dikonfirmasi dengan membandingkan hasil berdasarkan algoritma lain yang diberikan berdasarkan literatur yang sama[6].

Pertumbuhan yang tinggi pada eksploitasi sumber daya energi pada sistem distribusi menyebabkan pelanggaran tegangan dan beban lebih pada *grid* distribusi yang menyebabkan tidak terkontrol dan tidak dapat terprogram akibat injeksi generator tersebar. Dengan masalah beban lebih dalam sistem distribusi, operator sistem distribusi telah mengevaluasi pembangkit maksimal yang dapat ditambahkan pada sistem distribusi *grid*. Pada penelitian[7] memiliki perbedaan metodologi evaluasi untuk penambahan kapasitas yang dilihat ulang, model untuk menentukan kapasitas penambahan bergantung pada ketidakpastian parameter-parameter *grid* dan banyaknya generator yang dihubungkan[7].

Penelitian yang dilakukan(Mirjalili, 2015) bertujuan untuk inspirasi algoritma yang disebut dengan *Ant Lion Optimizer*(ALO). Algoritma ini terinspirasi dari mekanisme berburu dari semut singa di alam. Ada 5 langkah kerjanya seperti semut berjalan secara acak, membangun jebakan, memasukan semut dalam jebakan, menangkap mangsa, dan membangun ulang jebakan. Tujuan algoritma ini yakni untuk, pertama diset 19 fungsi matematika, kedua adalah 3 klasifikasi masalah keteknikan, terakhir bentuk dari baling-baling kapal sebagai fungsi kekangan dalam desainnya. Hasil dari algoritma ini sangat kompetitif, menghindari lokal optima, eksplorasi, konvergen. Algoritma ini juga menyelesaikan dalam kekangan kasus ruang pencarian yang beragam[8].

Penelitian yang telah dilakukan oleh(Syahputra, et al, 2015) bertujuan untuk meningkatkan jaringan Distribusi tipe Radial dengan integrasi *Distributed Generation*(DG) menggunakan metode *Particle Swarm Optimization*(PSO)

*Algorithm*. Performa tertinggi dalam jaringan distribusi adalah sebuah jaringan yang memiliki rugi daya rendah, profil tegangan yang baik, dan beban penyulang yang seimbang. Usaha untuk meningkatkan performa pada jaringan distribusi adalah dengan membuat optimasi konfigurasi ulang pada jaringan distribusi. Optimasi menjadi sebuah isu penting dengan penambahan DG. Metodologi yang telah diuji coba dalam 2 model dari sistem distribusi radial standar IEEE. Hasilnya menunjukkan bahwa konfigurasi ulang yang optimal dari jaringan distribusi dapat mengurangi rugi daya dan peningkatan profil tegangan di jaringan secara signifikan[9].

Penelitian yang dilakukan(Syahputra et al, 2017) bertujuan meningkatkan teknik optimasi untuk konfigurasi di jaringan distribusi, yang mana dapat dianggap sebagai modifikasi dari pengembangan *Evolutionary Algorithm* (EA) saat ini. Multi tujuan *Evolutionary Algorithm* diterapkan dalam masalah optimasi biaya total secara serentak meminimalkan rugi daya dan memaksimalkan magnitude tegangan. Jaringan Distribusi Standar IEEE model 69-bus digunakan dalam studi kasus ini, dan hasil tes menunjukkan bahwa algoritma ini bisa diterapkan untuk optimal solusi yang tidak didominasi. Hal itu memperbolehkan peralatan untuk mendapatkan sistem terbaik dari biaya termurah[10].

**Tabel 2.1 Review Hasil Penelitian Sebelumnya**

No.	Penulis	Judul (Tahun)	Hasil
1.	M. R. Ali	Optimasi Penempatan dan Kapasitas DG dengan metode <i>Flower Pollination Algorithm</i> (2015)	Penempatan DG layak untuk diinvestasikan dengan nilai BCR yang baik dan meningkatkan profil tegangan menjadi 0.96 P.U, menurunkan rugi – rugi daya aktif menjadi 82.87 kW

**Tabel 2.1 (Lanjutan)**

2.	S. Shaddiq	Peningkatan keandalan sistem distribusi dengan penempatan <i>optimal wind-based</i> DG menggunakan FPA (2017)	Peningkatan keandalan ditandai dengan terjadinya reduksi indeks energi yang tak tersuplai dengan penempatan optimal <i>wind-based</i> DG
3.	Dinakara prasasd Reddy P, Veera Reddy VC, Gowri Manohar T	<i>Antlion Optimization algorithm for optimal sizing of renewable energy resources for loss reduction in distribution systems</i> (2017)	Dari hasil analisis hasil terbaik telah didapatkan dari tipe DG III dengan operasi pada faktor daya 0.9
4.	Mohammad jafar hadidian-moghaddam, saber arabi-nowdes, Mehdi bigdeli, Davood Azizian	<i>Multi-objective optimal sizing and sitting of distributed generation using Antlion optimization technique</i> (2018)	Diperoleh hasil bahwa ALO memiliki performa terbaik dalam menampilkan solusi masalah ukuran dan penempatan DG yang optimal dibandingkan PSO dan GA

**Tabel 2.1 (Lanjutan)**

5.	M. C. V. Suresh and E. J. Belwin	<i>Optimal DG Placement for benefit maximization in distribution networks by using Dragonfly algorithm</i> (2018)	Hasil terbaik menunjukkan bahwa DG dengan tipe III saat operasi faktor daya sebesar 0.9
6.	O. Herbadji, L. Slimani, and T. Bouktir	<i>Optimal Power Flow with four Conflicting Objective Functions using Antlion Algorithm: A Case Study of the Algerian Electrical Network</i> (2019)	Terbukti bahwa uji coba keberagaman dari algoritma dapat diterapkan pada sistem yang besar. Peningkatan efektifitas ditujukan berdasarkan konfirmasi perbandingan hasil yang didapatkan dengan hasil algoritma lain sesuai literatur dengan sistem tes yang sama.
7.	S. Mina Mirbagheri, Davide Falabretti, Valentin Ilea, Marco Merlo	<i>Hosting Capacity analysis: A review and a new evaluation method in case of parameters uncertainly and multi-generator</i> (2018)	Mengevaluasi perbedaan yang efektif antara ketidakpastian parameter-parameter dan multi generator. Pada akhirnya metode komputasi terbaru yang efektif untuk mengevaluasi penambahan kapasitas dalam masalah ketidakpastian parameter <i>grid</i> dan multi generator terhubung yang dituju.

**Tabel 2.1 (Lanjutan)**

8.	Seyedali Mirjalili	<i>The Ant Lion Optimizer</i> (2015)	Meningkatkan fungsi eksplorasi, menghindari optimal lokal, eksplotasi, dan konvergen. Algoritma ini juga dapat menyelesaikan kekangan masalah dalam ruang pencarian yang bebas.
9.	R. Syahputra, I. Robandi, and M. Ashari	<i>Performance Improvement of Radial Distribution Network with Distributed Generation Integration Using Extended Particle Swarm Optimization Algorithm</i> (2015)	Hasil yang didapatkan menunjukkan bahwa konfigurasi ulang yang optimal pada jaringan distribusi tipe radial dengan metode <i>Particle Swarm Optimization</i> , dapat mengurangi rugi daya dan perbaikan profil tegangan secara signifikan.
10.	R. Syahputra, R. O. Wiyagi, S. Suripto, and I. Soesanti	<i>Optimization of Distribution Network Configuration Using Evolutionary Algorithm Approach</i> (2017)	Algoritma ini dapat menentukan solusi optimal yang tidak didominasi. Memperbolehkan peralatan untuk mendapatkan sistem terbaik dengan biaya terendah.

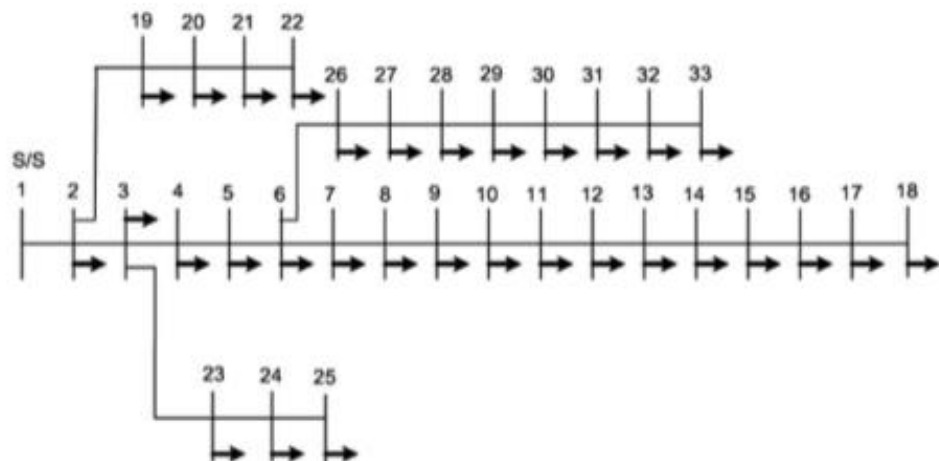
## 2.2 Landasan Teori

### 2.2.1 Konfigurasi Jaringan Sistem Distribusi

Distribusi Sistem Tenaga Listrik adalah bagian akhir dari pengiriman daya listrik. Sistem distribusi membawa listrik dari sistem transmisi ke setiap individu pengguna listrik. Saluran sistem distribusi primer membawa daya tegangan menengah ke trafo distribusi yang terletak di dekat lokasi pelanggan. Trafo distribusi menurunkan dari level tegangan menengah ke level tegangan peralatan rumah atau yang sejenisnya untuk memberikan asupan ke beberapa pelanggan melalui saluran distribusi sekunder pada level tegangan ini[1].

#### 1. Jaringan Distribusi tipe Radial

Penyulang terpisah menyebar dari satu Gardu Induk dan memberikan suplai ke distributor dengan hanya satu akhiran disebut sistem radial. Sistem Distribusi Radial adalah tipe dari distribusi daya, dimana daya dikirim melalui cabang utama ke sub-cabang lalu dipisahkan lagi dari sub-cabang seperti yang ditunjukkan pada **Gambar 2.1**, dimana daya dikirim dari titik akar dan lalu memisah pada *line* 1(L1). Struktur radial mengakibatkan tidak ada perputaran daya pada jaringan dan setiap bus terhubung 8 kepada sumber melalui tepat satu jalur. Konfigurasi jaringan ini yang murah, tetapi kurang handal dalam penyaluran tenaga listrik. Sistem distribusi radial banyak digunakan pada area yang perkembangannya tersebar. [1]



**Gambar 2.1 Sistem Distribusi tipe Radial Standar IEEE**

Sumber : IEEE tes 33-bus



Kriteria di bawah ini di asumsikan untuk penomoran titik dan saluran:

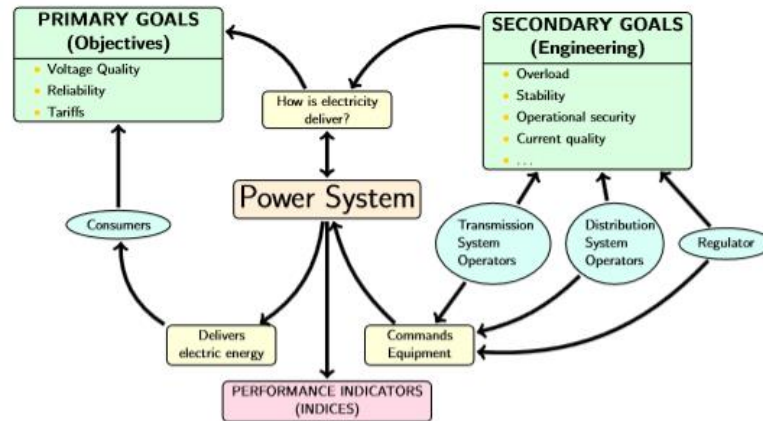
- a. Setiap titik diberi nomor secara berurutan dari yang terkecil sampai terbesar dilanjutkan dari setiap lapisan ke lapisan lain, dengan cara ini jalur apapun dari titik akar ke suatu titik terminal bertemu dengan titik yang diberi nomer dari urutan terkecil sampai terbesar.
- b. Setiap cabang dimulai dari bus pengirim dari sisi titik akar dan diidentifikasi dengan nomor dari bus penerima (Bompard, 2000).

Kelebihan utama dari jaringan distribusi radial adalah pada konstruksinya yang sederhana, biaya awal yang murah, berguna saat pembangkitan pada level tegangan rendah. Jaringan radial direkomendasikan jika gardu induk terletak di tengah-tengah beban[1].

### 2.2.2 *Distribution Generation Hosting Capacity (DG HC)*

*Distribution Generation Hosting Capacity* atau penambahan kapasitas pada pembangkit tersebar merupakan suatu teknik optimasi di dalam sistem distribusi tenaga listrik untuk melakukan penambahan kapasitas pembangkit tersebar dan terhubung ke sistem distribusi yang sudah ada. Banyak permasalahan di sistem distribusi radial seperti jatuh tegangan, rugi-rugi daya, dan biaya yang tidak murah. Hal ini yang mendasari untuk mencapai tujuan utama tersebut.

Teknik optimisasi *hosting capacity* ini tujuannya adalah untuk mengurangi tingkat beban lebih, meningkatkan stabilitas atau keandalan, pengamanan sistem distribusi, dan juga kualitas arus. Jadi bagaimana ketika terdapat penambahan *host* berupa sistem pembangkit terbarukan dan dihubungkan ke sistem distribusi yang sudah ada tidak mengalami kendala atau mengganggu sistem distribusi yang telah ada. Hal ini dapat dilihat seperti **Gambar 2.2** adalah tujuan dari *hosting capacity*.



**Gambar 2.2** Sistem *Hosting Capacity*[11]

Dalam menentukan *hosting capacity* terdapat beberapa tipe-tipe DG yang digunakan sebagai suplai daya. Hal ini akan mempengaruhi hasil dari beberapa tipe DG. Tipe-tipe DG dapat dikarakterisasi[3] sebagai berikut :

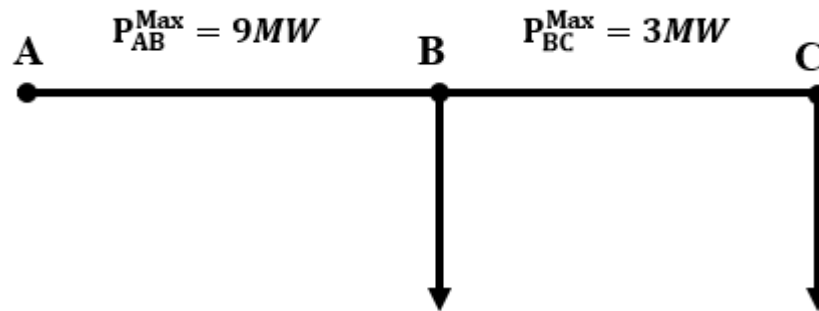
- a. Tipe 1 : Injeksi daya nyata (P). Ini beroperasi pada pf 1. PV dan sel bahan bakar, turbin mikro.
- b. Tipe II : Injeksi daya reaktif (Q). Seperti : Kapasitor, Sinkron, dan kompensator kVAR, dll.
- c. Tipe III : Injeksi daya nyata (P) dan reaktif (Q). misal : mesin sinkron (kogenerasi, turbin gas, dll.)
- d. Tipe IV : Menggunakan daya reaktif (Q) tetapi injeksi daya nyata(P). misal : generator induksi di pembangkit tenaga angin.

Beberapa hal dalam analisis *hosting capacity* adalah sebagai berikut :

#### A. *Hosting Capacity & Power*

*Hosting Capacity* yang ditambahkan pada distribusi ini adalah dengan penambahan DG daya. Agar suplai daya yang dikirim sebanding dengan daya disisi penerima, sehingga rugi daya akibat saluran distribusi bisa diminimalkan.

**Gambar 2.3** menunjukkan sistem distribusi listrik tanpa adanya penambahan DG ketika tanpa beban penuh:



Gambar 2.3 Sistem Distribusi tanpa DG [11]

$$P_{LB}^{max} = 4MW \quad P_{LC}^{max} = 2MW$$

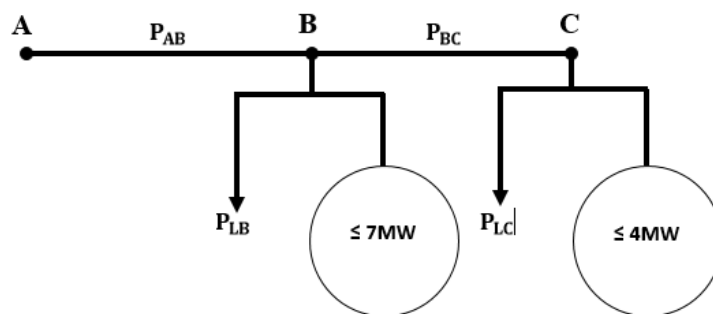
$$P_{LB}^{min} = 1MW \quad P_{LC}^{min} = 1MW$$

Hal ini dapat dirumuskan seperti pada persamaan 2.1 dan persamaan 2.2:

$$P_{delivered}^{max} < P_{LB}^{max} + P_{LC}^{max} \rightarrow \text{Konsumen} \quad (2.1)$$

$$P_{delivered}^{max} < P_{AB}^{max} + P_{BC}^{max} \rightarrow \text{Daya yang dikirim} \quad (2.2)$$

Pada **Gambar 2.4** adalah sistem distribusi pada saat kondisi adanya penambahan DG dengan teknik *hosting capacity*.



Gambar 2.4 Sistem Distribusi dengan penambahan DG[11]

Dari **Gambar 2.4** diketahui bahwa nilai dari masing – masing daya dapat dilihat seperti perhitungan (2.3) sampai (2.6):

$$P_{AB}^{HC1} = (4 + 2) + (1 + 1) = 8MW \quad (2.3)$$

$$P_{BC}^{HC1} = (2) + (1) = 3MW \quad (2.4)$$

$$P_{AB}^{HC2} = (9) + (1 + 1) = 11MW \quad (2.5)$$

$$P_{BC}^{HC2} = (3) + (1) = 4MW \quad (2.6)$$

Dimana, untuk pemodelan **Gambar 2.4** Dapat dirumuskan sebagai persamaan (2.7) dan persamaan (2.8):

$$P_{generated}^{max} < P_{consumed}^{max} + P_{consumed}^{min} \rightarrow 1^{st} \text{ Hosting capacity (HC1)} \quad (2.7)$$

$$P_{generated}^{max} < P_{feeder}^{max} + P_{consumed}^{min} \rightarrow 2^{st} \text{ Hosting capacity (HC2)} \quad (2.8)$$

### B. Hosting Capacity & Voltage

*Hosting capacity & Voltage* digunakan untuk mengatur tegangan apabila terjadi *overvoltage* atau tegangan lebih dan *undervoltage* atau tegangan jatuh. Masalah ini sering terjadi dalam sistem distribusi, biasanya terjadi jatuh tegangan yang diakibatkan adanya penggunaan listrik saat beban puncak. Sedangkan tegangan lebih terjadi saat konsumsi listrik rendah dengan suplai daya yang maksimal. Untuk itu dapat dirumuskan dengan persamaan (2.13):

$$\Delta V = |U - V| = |Z_L I| \quad (2.9)$$

$$\Delta V = |R(1 + \alpha j) \frac{P}{V} (1 + kj)| \quad (2.10)$$

$$\Delta V = \frac{PR}{V} |1 - \alpha k + (\alpha + k)j| \quad (2.11)$$

$$\frac{\Delta V}{V} = \frac{PR}{V^2} \sqrt{(1 - \alpha k)^2 + (\alpha + k)^2} \quad (2.12)$$

$$P = \frac{\delta V V^2}{R} \frac{1}{\sqrt{(1 - \alpha k)^2 + (\alpha + k)^2}} \quad (2.13)$$

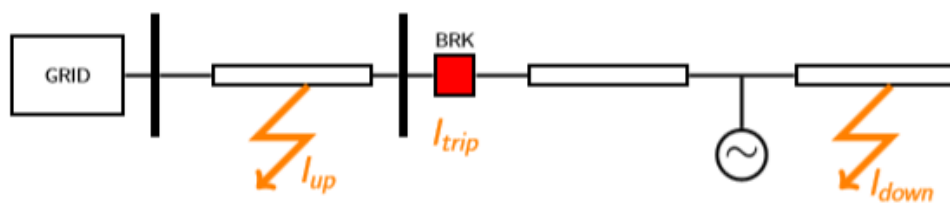
### C. Hosting Capacity & Protection

Sistem proteksi akibat adanya *hosting capacity* terlihat seperti **Gambar 2.5**. Hal ini dapat dirumuskan berdasarkan persamaan (2.14) dan persamaan (2.15) berikut, pada saat kondisi tanpa generator didapatkan persamaan (2.14):

$$I_{down}^{min} > I_{trip} \quad I_{up} = 0 \quad (2.14)$$

Pemodelan sistem distribusi pada saat terdapat DG dapat dirumuskan dengan persamaan matematis. Dimana pada Gambar 2.5 menampilkan arus trip yang dihitung dan didapatkan dengan persamaan (2.15):

$$I_{down}^{min} > I_{trip} > I_{up}^{max} \quad (2.15)$$



**Gambar 2.5** Proteksi *Hosting Capacity Distributed Generation*[11]

Kondisi-kondisi yang menyebabkan sistem proteksi bekerja pada *hosting capacity* :

- Kondisi tanpa tentu terjadi ketika suatu DG mengalami kenaikan sistem seperti tegangan dan daya.
- Panjang *feeder* suatu saluran distribusi sangat panjang, sehingga DG mengalami penurunan atau DG tidak dapat menyuplai secara efektif
- Pembangkit yang terletak jauh dari Gardu Induk maupun pusat-pusat beban sehingga DG ini memiliki kenaikan penggunaan.
- Arus proteksi bukan menjadi solusi ketika terjadi penetrasi DG yang tinggi.

Dengan menggunakan teknik optimasi metaheuristik ini diharapkan peletakkan suatu DG dan kapasitasnya dapat memberikan dampak *hosting capacity* yang baik untuk mengurangi permasalahan tersebut.

### 2.2.3 *Distributed Generation Photovoltaic* (DG Pv)

Panel surya atau fotovoltaik adalah alat yang mengubah energi cahaya menjadi energi listrik menggunakan [efek fotoelektrik](#). Dibuat pertama kali pada

tahun 1880 oleh Charles Fritts. Pembangkit listrik tenaga surya tipe fotovoltaik adalah pembangkit listrik yang menggunakan perbedaan tegangan akibat efek fotoelektrik untuk menghasilkan [listrik](#). Solar panel terdiri dari 3 lapisan, lapisan panel P di bagian atas, lapisan pembatas di tengah, dan lapisan panel N di bagian bawah. Efek fotoelektrik adalah di mana [sinar matahari](#) menyebabkan [elektron](#) di lapisan panel P terlepas, sehingga hal ini menyebabkan [proton](#) mengalir ke lapisan panel N di bagian bawah dan perpindahan arus proton ini adalah arus listrik[23].

Menurut[12] bahwa *Photovoltaic Distributed Generation* (Pv-DG) dapat diklasifikasikan sebagai skala mikro, kecil, menengah dan utilitas :

- a. Skala mikro *Photovoltaic Distributed Generation* (Pv-DG) terdiri dari unit fase tunggal ukuran Volt Ampere(VA) yang sangat terdistribusi (misal, 200 W).
- b. *Photovoltaic Distributed Generation* (Pv-DG) skala kecil terdiri dari beberapa unit *single phase* ukuran kW (misal, 5 hingga 10 kW) seperti yang dipasang di atap rumah tangga tempat tinggal.
- c. *Photovoltaic Distributed Generation* (Pv-DG) skala menengah terdiri dari unit tiga fase ukuran kW yang lebih besar (misal, 100 hingga 500 kW) seperti yang dipasang di atap bangunan komersial kecil.
- d. Skala Pv-Utilitas terdiri dari unit-unit tiga ukuran MW yang memiliki kapasitas nominal yang dapat dikelola oleh pengumpulan dan gardu distribusi, misalnya, antara 500 kW hingga 15 MW.

#### 2.2.3.1 Pengaruh *Photovoltaic Distributed Generation* (Pv-DG) terhadap Sistem Distribusi

Menurut[13] beberapa dampak paling umum dampak penetrasi rendah maupun tinggi yang didorong oleh integrasi *Photovoltaic Distributed Generation* (Pv-DG) pada sistem distribusi adalah:

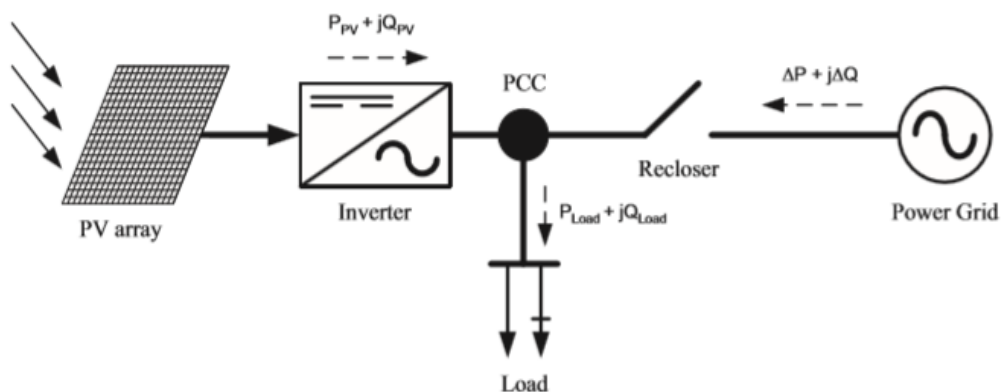
- a. Efek fluktuasi tegangan
- b. Kenaikan tegangan
- c. Ketidakseimbangan Tegangan
- d. Harmonisa

### 2.2.3.2 Deteksi dan Operasi *Islanding*

Pengertian *islanding* dapat didefinisikan sebagai situasi dimana *Renewable Energy Source* (RES) yang diterapkan dalam jadwal sistem perbaikan tegangan dan frekuensi diantara batas yang diijinkan setelah sistem dihubungkan dalam *grid*. **Gambar 2.6** menunjukkan tipe sistem konfigurasi dari deteksi *islanding* yang mana *Point of Common Coupling* (PCC) beraksi sebagai antarmuka antara *grid* dengan sistem distribusi lokal[13].

*Islanding* sering tidak diijinkan dan pertimbangan utama dalam pekerjaan akhir untuk mengamankan sistem sebelum semua sistem jatuh. Hal ini terjadi karena :

- Pengaman *Hazard* : bagian dari sistem untuk meningkatkan PCC selama terdapat energi pada mode *islanding*.
- Kerusakan sistem : ketidaksinkron fase tegangan antara sistem PV dan sistem *grid* selama penutupan dapat merusak peralatan yang terhubung ke sistem



**Gambar 2.6 Sistem Konfigurasi *Photovoltaic* [13]**

Menurut[13] bahwa solusi teknik yang dapat diberikan dari permasalahan penambahan *Photovoltaic Distributed Generation* (Pv-DG) adalah secara konvensional yakni dengan *On Load Tap Changer* (OLTC), *Switched Capacitor* (SC) dan *Step Voltage Regulator* (SVR) digunakan

untuk mengontrol level tegangan. Penambahan *channel* komunikasi juga dibangun untuk tujuan koordinasi.

Dalam mengurangi ketidakseimbangan tegangan, sebuah sistem dengan topologi Inverter dan algoritma kontrol berdasarkan pada DVR dan DSTATCOM pada PCC mereka yang telah dituju. Tujuan ini sistem *pole-shift* berdasarkan pada sebuah kontrol umpan balik yang telah diterapkan. Sistem dapat mengontrol tegangan pada keluaran dari DVR dan DSTATCOM dalam jalur dimana keseimbangan tegangan dapat dihubungi dalam jaringan[13].

Untuk menyelesaikan permasalahan harmonik dalam Generator Distribusi, filter yang secara umum digunakan. Secara konvensional, tipe induktor yang paling banyak digunakan. Induktor kapasitor adalah induktor filter dengan pengurangan yang sering digunakan dalam menangani harmonik jaringan[13].

### 2.2.3.3 Persyaratan Teknik Tambahan untuk Pembangkit Listrik Tenaga Surya

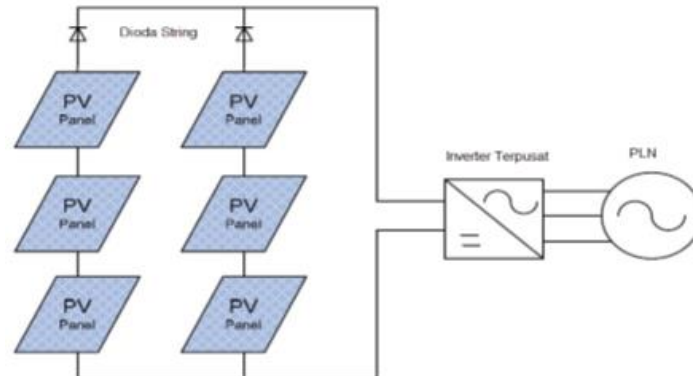
Penyambungan dan pengoperasian paralel Pembangkit Listrik Tenaga Surya Fotovoltaik harus memenuhi persyaratan fungsi proteksi dan fungsi pencegahan interferensi sistem. Beberapa komponen utama yang harus ada adalah penyambungan DG ke sistem distribusi seperti : Inverter, Trafo *step-up*, dan perangkat kontrol serta proteksi[14]. Berikut beberapa metode konfigurasi pembangkit *Photovoltaic*(PV) sebagai pembangkit tersebar ke sistem distribusi :

#### 1. Konfigurasi Inverter Terpusat

Topologi penyambungan pembangkit tipe ini paling murah, karena hanya memiliki satu *inverter* DC ke AC terpusat untuk *string* panel fotovoltaik yang dihubungkan secara seri dan paralel untuk menghasilkan tegangan dan arus DC tertentu. Kelemahan lain adalah permasalahan keandalan sistem, dengan adanya gangguan atau kerusakan pada inverter maka akan menyebabkan seluruh pembangkit



berhenti beroperasi[14], seperti dimodelkan pada **Gambar 2.7** berikut.

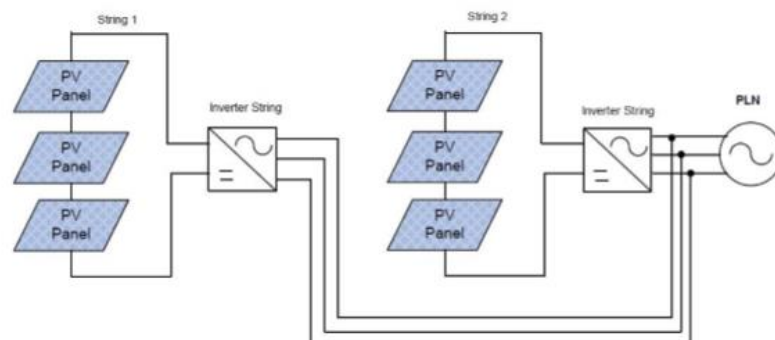


**Gambar 2.7 Konfigurasi Fotovoltaik Inverter Terpusat**

Sumber : (SPLN 2014)

## 2. Konfigurasi *Inverter Individual String*

*Inverter String Array* yang mana rangkaian seri tunggal panel *Fotovoltaic* dihubungkan secara seri dan *string* terhubung ke inverter tunggal dan ada satu inverter untuk setiap *string*. Keuntungan dari konfigurasi ini adalah bahwa string inverter memiliki kemampuan pelacakan titik daya maksimum atau *Maximum Power Point Tracking* (MPPT) secara terpisah dari setiap *string Fotovoltaic* untuk mengurangi kerugian dari ketidakcocokan dan bayangan parsial sehingga dapat meningkatkan produksi energi. Kelemahan konfigurasi ini adalah diperlukan jumlah Inverter yang lebih banyak. (Sumantri, 2014), hal ini seperti dimodelkan pada **Gambar 2.8**:

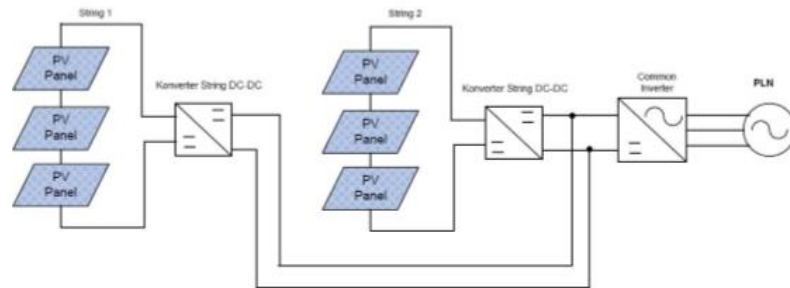


**Gambar 2.8 Konfigurasi Individual String Inverter**

Sumber : (SPLN 2014)

### 3. Konfigurasi *Multi - String Inverter*

Konfigurasi ini memungkinkan untuk mengoptimalkan efisiensi pengoperasian setiap *string* secara terpisah, dan integrasi berbagai orientasi surya untuk memaksimalkan produksi energi. (Sumantri, 2014), hal ini seperti konfigurasi pada **Gambar 2.9** berikut:

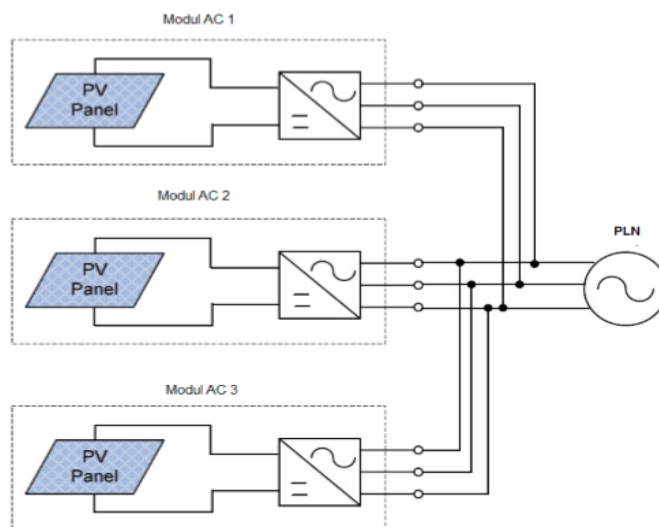


**Gambar 2.9 Konfigurasi Multi-string Inverter**

Sumber : (SPLN 2014)

### 4. Konfigurasi Inverter Modul AC

Pada **Gambar 2.10** menjelaskan bahwa pada inverter tipe ini tiap-tiap panel masuk dalam setiap modul, kemudian diubah menjadi tegangan AC di masing – masing modul.



**Gambar 2.10 Konfigurasi AC Modul Inverter**

Sumber : (SPLN 2014)

#### 2.2.3.4 Persyaratan teknik tambahan untuk Pembangkit Listrik Tenaga Surya (PLTS)

Karena sifat intermiten Pembangkit Listrik Tenaga Surya dan potensi dampak yang merugikan pada profil tegangan dari penyulang, terutama pada kondisi beban rendah dan kondisi awan yang cepat tertutup, maka Pembangkit Listrik Tenaga Surya Fotovoltaik dengan kapasitas terpasang 2 MW atau lebih tinggi memiliki kemampuan untuk [14]:

- a. Membatasi *ramp inverter* (naik atau turun) pada laju 10% per menit dari kapasitas inverter, berlaku untuk *start up* dan *shut down*, operasi normal, dan perintah pembatasan, kecuali selama terjadi penurunan radiasi surya.
- b. Mengatur waktu *restart* untuk inverter ganda pada 15 detik atau lebih untuk setiap blok 1 MW.

#### 2.2.4 Metode Penyelesaian Aliran Daya Sistem Distribusi

Studi aliran daya atau *Load Flow* dilakukan untuk menentukan kondisi operasi *steady state* dari sistem tenaga listrik. *Load flow* sangat penting digunakan untuk perencanaan sistem tenaga listrik, operasi sistem, optimasi, dan kontrol. Banyak faktor yang mempengaruhi sistem distribusi radial, salah satunya adalah naik turunnya beban. Sejak tahun 1960 sudah banyak algoritma aliran daya yang berbasis komputasi digital, seperti metode *fast decoupled*, *newton raphson*, dan *gaus seidel*, serta metode *forward/backward sweep*. Menurut [15] bahwa metode *forward/backward sweep* memiliki komputasi efektif disetiap iterasi, serta solusi dan perhitungan numeriknya tepat [16].

##### 1. Backward/Forward Sweep(BFS)

Pada penyelesaian aliran daya di sistem distribusi matriks admittans tidak digunakan, hal ini berbeda dengan metode aliran daya pada sistem transmisi. Pada kasus ini, karena rendahnya keterhubungan antara saluran satu dengan lainnya, persamaan *Kirchhoff's Voltage Law* (KVL) dan *Kirchhoff's Current Law* (KCL) secara langsung dibentuk pada bagian saluran [15].

## 2. Backward Sweep

Prosedur penyelesaian aliran daya dimulai dengan *backward sweep*. Pada iterasi pertama, tegangan dari semua titik diasumsikan sama dengan tegangan pada sumber utama. Jika ada beberapa sumber pada jaringan, arus injeksi kompensasi pada sumber tersebut bernilai nol pada iterasi pertama. Di sisi lain, tegangan pada masing-masing titik dan arus injeksi kompensasi dihitung pada iterasi sebelumnya. Ketika tegangan pada masing-masing titik dan arus injeksi kompensasi sudah diketahui, arus beban dapat dicari dengan persamaan (2.16)[15]:

$$I_{ldi} = \left[ \frac{P_i + jQ_i}{V_i} \right] \quad (2.16)$$

Dengan,

$I_{ldi}$  : Arus beban pada titik  $i$  (A)

$P_i$  : Kebutuhan daya aktif pada titik  $i$  (W)

$Q_i$  : Kebutuhan daya reaktif pada titik  $i$  (Var)

$V_i$  : Tegangan pada titik  $i$  (Volt)

## 3. Forward Sweep

Pada analisa *forward sweep*, dimulai dari titik sumber utama dimana nilai tegangannya diketahui, impedansi dan arus yang mengalir pada masing-masing saluran sudah diketahui, semua tegangan titik diperbarui mengabaikan sumber lainnya jika ada. Hal ini dirumuskan pada persamaan (2.17):

$$V_i = V_{vi} - (Z_i \times I_{Li}) \quad (2.17)$$

Dimana,

$i = 1, \dots, N$

$V_i$  : Tegangan pada titik  $i$  (V)

$V_{vi}$  : Tegangan pada atas dari titik  $i$  (V)

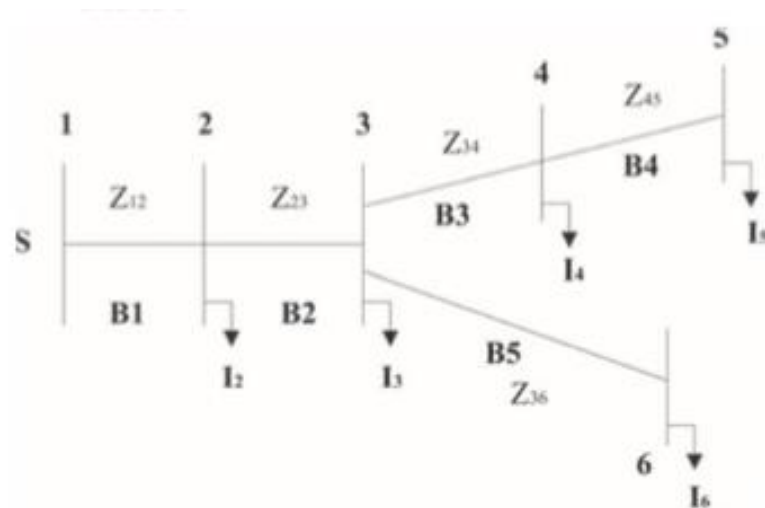
$Z_i$  : Impedansi saluran  $i$

$I_{Li}$  : Arus yang mengalir pada saluran  $i$  (A)

*Backward sweep* dimulai dari ujung terjauh dari jaringan arus beban pada titik beban. Oleh karena itu, arus yang mengalir pada saluran dihitung menurut asumsi atau kalkulasi dari tegangan pada iterasi sebelumnya. Setelah menghitung arus yang mengalir pada saluran, pada *backward sweep* dimulai dari titik sumber, tegangan dari masing-masing titik bus diperbarui. Setelah *forward sweep*, kompensasi arus injeksi dihitung. Lalu kriteria konvergensi yang disesuaikan dengan titik tegangan beban atau arus saluran, dan masukan daya ke jaringan[17].

### 2.2.5 Pemodelan Struktur Jaringan

Berdasarkan contoh sistem distribusi radial di **Gambar 2.11**, dapat dibuat formasi dengan menggunakan matriks BBIC,BCBV, dan DLF.



**Gambar 2.11** Contoh Sistem Distribusi Radial 6 bus[16]

#### A. Matriks Bus Injection to Branch Current (BBIC)

Pada sistem distribusi, bus ke- $i$  beban kompleks dapat diekspresikan dengan persamaan (2.18) dan nilai ekuivalen arus injeksi yang bersangkutan dapat diekspresikan dengan persamaan (2.19).

$$S_i = (P_i + jQ_i) \quad i = 1, 2, \dots, N \quad (2.18)$$

$$I_i = (S_i/V_i) \quad (2.19)$$

Vektor injeksi arus dari contoh sistem diatas dapat dibuat pada **Tabel 2.2**:

**Tabel 2.2** keterangan vektor arus injeksi

Bus No.	2	3	4	5	6
Arus Injeksi	$I_2$	$I_3$	$I_4$	$I_5$	$I_6$

Untuk sistem pada **Gambar 2.11**, diaplikasikan *Kirchhoff's Current Law* (KCL), arus cabang dapat diekspresikan dengan nilai arus injeksi ekuivalen yang dibuat menjadi matrik seperti persamaan (2.25):

$$B_1 = I_2 + I_3 + I_4 + I_5 + I_6 \quad (2.20)$$

$$B_2 = I_3 + I_4 + I_5 + I_6 \quad (2.21)$$

$$B_3 = I_4 + I_5 + I_6 \quad (2.22)$$

$$B_4 = I_5 + I_6 \quad (2.23)$$

$$B_5 = I_6 \quad (2.24)$$

$$\begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} I_2 \\ I_3 \\ I_4 \\ I_5 \\ I_6 \end{bmatrix} \quad (2.25)$$

Persamaan arus cabang di atas dapat disusun dengan bentuk umum persamaan 2.27:

$$[B] = [BBIC][I] \quad (2.27)$$

#### B. Matrik *Branch Current to Bus Voltage* (BCBV)

Hubungan antara arus cabang dan tegangan bus dapat diekspresikan seperti persamaan 2.28:

$$V_2 = V_1 - B_1 Z_{12} \quad (2.28)$$

$$V_3 = V_2 - B_2 Z_{23} \quad (2.29)$$

$$V_4 = V_3 - B_3 Z_{34} \quad (2.30)$$

$$V_5 = V_4 - B_4 Z_{45} \quad (2.31)$$

$$V_6 = V_5 - B_5 Z_{56} \quad (2.32)$$

Dengan mensubstitusikan persamaan (2.28) dan persamaan (2.29) ke persamaan (2.30), maka diperoleh tegangan pada bus 4 seperti pada persamaan (2.33):

$$V_4 = V_1 - B_1 Z_{12} - B_2 Z_{23} - B_3 Z_{34} \quad (2.33)$$

Sama seperti nilai tegangan pada bus lain dapat ditulis seperti persamaan (2.36):

$$V_3 = V_1 - B_1 Z_{12} - B_2 Z_{23} \quad (2.34)$$

$$V_5 = V_1 - B_1 Z_{12} - B_2 Z_{23} - B_3 Z_{34} - B_4 Z_{45} \quad (2.35)$$

$$V_6 = V_1 - B_1 Z_{12} - B_2 Z_{23} - B_3 Z_{34} - B_4 Z_{45} - B_5 Z_{36} \quad (2.36)$$

Persamaan (2.27), persamaan (2.32), persamaan (2.33), persamaan (2.34), persamaan (2.35) dapat disusun seperti pada persamaan (2.37):

$$\begin{bmatrix} V_1 \\ V_1 \\ V_1 \\ V_1 \\ V_1 \end{bmatrix} - \begin{bmatrix} V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \end{bmatrix} = \begin{bmatrix} Z_{12} & 0 & 0 & 0 & 0 \\ Z_{12} & Z_{23} & 0 & 0 & 0 \\ Z_{12} & Z_{23} & Z_{34} & 0 & 0 \\ Z_{12} & Z_{23} & Z_{34} & Z_{45} & 0 \\ Z_{12} & Z_{23} & 0 & 0 & Z_{36} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ B_5 \end{bmatrix} \quad (2.37)$$

$$[\Delta V] = [BCBV][B] \quad (2.38)$$

Kemudian, substitusikan persamaan (2.27) ke persamaan (2.38) dan hasil persamaannya dapat diekspresikan ke persamaan (2.39) dan persamaan (2.40):

$$[\Delta V] = [BCBV][BIBC][I] \quad (2.39)$$

$$[\Delta V] = [DLF][I] \quad (2.40)$$

Dengan DLF adalah matriks *Distribution Load Flow* (DLF) yang ditunjukkan dengan matriks (2.41)[16]:

$$\begin{bmatrix} Z_{12} & Z_{12} & Z_{12} & Z_{12} & Z_{12} \\ Z_{12} & Z_{12} + Z_{23} & Z_{12} + Z_{23} & Z_{12} + Z_{23} & Z_{12} + Z_{23} \\ Z_{12} & Z_{12} + Z_{23} & Z_{12} + Z_{23} + Z_{34} & Z_{12} + Z_{23} + Z_{34} & Z_{12} + Z_{23} \\ Z_{12} & Z_{12} + Z_{23} & Z_{12} + Z_{23} + Z_{34} & Z_{12} + Z_{23} + Z_{34} + Z_{45} & Z_{12} + Z_{23} \\ Z_{12} & Z_{12} + Z_{23} & Z_{12} + Z_{23} & Z_{12} + Z_{23} & Z_{12} + Z_{23} + Z_{36} \end{bmatrix} \quad (2.41)$$

### C. Arus Injeksi Kompensasi

Setelah *forward sweep* dilakukan, arus injeksi kompensasi diperbarui. Arus injeksi kompensasi ada akibat adanya sumber lain selain sumber utama. Jika tidak ada sumber lain atau hanya ada satu sumber utama, arus injeksi kompensasi bernilai nol pada setiap titik. Sebaliknya, untuk jumlah sumber yang banyak, nilai arus injeksi kompensasi untuk masing-masing titik yang tidak terhubung pada sumber bernilai nol dan pada titik yang terhubung sumber bernilai sama dengan persamaan (2.42)[1].

$$I_{Ci\ new} = I_{Ci\ old} + (V_i - V_{si}) / \left(\sum_2\right) Z_{si} \quad (2.42)$$

Dimana,

$I_{Ci\ new}$  : Arus injeksi kompensasi pada titik i pada iterasi sekarang

$I_{Ci\ old}$  : Arus injeksi kompensasi pada titik i pada iterasi sebelumnya

$V_i$  : Tegangan pada titik i

$V_{si}$  : Tegangan sumber pada titik i

$Z_{si}$  : Impedansi antara sumber utama dan titik i

$S$  : Jumlah titik sumber, termasuk sumber utama

### 2.2.6 Metode Metaheuristik

Dalam dekade terakhir, kajian dan studi mengenai bidang metaheuristik untuk menyelesaikan permasalahan optimasi tumbuh secara signifikan, baik dari segi teknik metaheuristik maupun dari segi aplikasinya. Metode ini menunjukkan dalam penyelesaian masalah *combinatorial hard combinatorial* dengan skala yang cukup besar dengan waktu komputasi yang kompetitif. Pendekatan metaheuristik diinspirasi dari berbagai fenomena alam seperti : fenomena evolusi, perilaku hewan



dalam mencari makan atau proses pendinginan baja (*metallic annealing*) dan yang lain[20].

Metode metaheuristik ini telah banyak berkembang dari beberapa penemuan sebelumnya, tujuannya adalah untuk menemukan suatu metode yang paling optimal. Beberapa contoh metode metaheuristik adalah sebagai berikut : *Genetic Algorithm*(GA), *Ant Colony Optimization*(ACO), *Particle Swarm Optimization*(PSO), *Flower Pollination Algorithm*(FPA), *Antlion Algorithm*(ALO), dan *Multiobjective Ant Lion Optimizer*(MOALO). Pada penelitian ini menggunakan metode *Ant Lion Optimizer*(ALO) sebagai komputasi dari permasalahannya.

Metode Optimasi *Ant Lion Optimizer*(ALO) *algorithm* merupakan metode optimasi terbaru yang dikeluarkan oleh Mirjalili Seyedali tahun tahun 2015. Menurut[4] didalam papernya bahwa metode ALO memiliki optimasi yang lebih baik dibandingkan dengan metode PSO dan GA. Dalam paper tersebut dibuktikan beberapa kategori seperti nilai fungsi tujuan dan grafik *Index Fuzzy*.

#### 2.2.6.1 *Ant Lion Optimizer Algorithm*

*Ant Lion Optimizer*(ALO) adalah sebuah cerita yang terinspirasi dari alam dengan algoritma yang dibuat oleh Seyedali Mirjalili tahun 2015. Algoritma ALO meniru dari mekanisme berburu dari *ant lion* (semut singa) di alam. Ada 5 babak utama untuk berburu membunuh mangsanya seperti kemampuan membangun jebakan dari jalan semut agar semut terperangkap, terjadi penangkapan, dan membangun kembali jebakan untuk diimplementasikan[18].

*Ant Lion*(semut singa) masih menjadi kelas dari kupu-kupu. Perputaran hidup dari semut singa masuk ke dalam 2 fase utama : larva dan dewasa. Total masa dewasa di alam dapat mencapai 3 tahun, dimana kebanyakan terjadi di larva (hanya 3-5 minggu selama masa dewasa). Semut singa menjalani metamorfosis dalam sebuah kepompong untuk menjadi dewasa. Mereka kebanyakan berburu didalam larva dan periode masa dewasa hanya untuk bereproduksi. Sebuah semut singa menggali lubang dalam bentuk kerucut di pasir dengan bergerak di sepanjang jalan melingkar dan membuat pasir dengan rahangnya yang besar. Setelah menggali jebakan, larva bersembunyi di bawah

bagian kerucut dan menunggu untuk serangga (utamakan semut) untuk terjebak didalam lubang. Ujung kerucup cukup tajam sehingga serangga mudah jatuh ke dasar perangkap. Begitu semut singa menyadari bahwa mangsa ada didalam perangkap, ia mencoba menangkapnya[18].

Secara umum ada 5 langkah utama dalam berburu seperti *random walk of agents, building traps, entrapment of ant in trap, catching prey and rebuilding traps. The roulette wheel and random walks of ants in ALO optimizer* dapat mengeliminasi optima lokal.

a. Semut – semut berjalan secara acak

Semut – semut yang berjalan secara acak dapat dirumuskan seperti persamaan (2.43):

$$X(t) = [0, \text{cumsum}(2r(t_1) - 1), \text{cumsum}(2r(t_2) - 1), \dots, \text{cumsum}(2r(t_n) - 1)] \dots (2.43)$$

Dimana :

cumsum : perhitungan kumulatif sum,

n : nomer maksimum dari jumlah iterasi

t : langkah-langkah dari perjalanan acak

$$r(t) = \begin{cases} 1 & \text{if } rand > 0.5 \\ 0 & \text{if } rand < 0.5 \end{cases} \quad (2.44)$$

Dimana :

rand : nomer generator diantara [0,1]

Dengan mengikuti persamaan (2.44), *random walk* dapat dibuat untuk memasukkan ke ruang pencarian seperti persamaan (2.45) dibawah:

$$X_i^t = \frac{(X_i^t - a_i)x(d_i^t - c_i^t)}{(b_i - a_i)} + c_i^t \quad (2.45)$$

Dimana :

a<sub>i</sub> : nilai minimal dari *random walk* dari variabel ith

$b_i$  : nilai maksimum dari *random walk* dari variabel *ith*

$c_i^t$  : nilai minimum dari variabel *ith* dalam *tth* iterasi

$d_i^t$  : nilai maksimum dari variabel *ith* dalam *tth* iterasi

Lokasi dari semut-semut(*ants*) bergantung pada kecocokan fungsi matriks yang diberikan seperti persamaan (2.46) dan persamaan (2.47):

$$M_{ant} = \begin{pmatrix} A_{1,1} & A_{1,2} & \dots & \dots & A_{1,d} \\ A_{2,1} & A_{2,2} & \dots & \dots & A_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{n,1} & A_{n,2} & \dots & \dots & A_{n,d} \end{pmatrix} \quad (2.46)$$

$$M_{ant} = \begin{bmatrix} f([A_{11} & A_{12} & \dots & \dots & A_{1d}]) \\ f([A_{21} & A_{22} & \dots & \dots & A_{2d}]) \\ \vdots \\ f([A_{n1} & A_{n2} & \dots & \dots & A_{nd}]) \end{bmatrix} \quad (2.47)$$

Dimana  $M_{OA}$  adalah matrik yang menampilkan kecocokan nilai dari lokasi semut-semut(*ants*) matrik  $M_{ANT}$ . Jika semut-semut (*ants*) dan semut singa(*ant lion*) bersembunyi didalam ruang pencarian, kemudian lokasi yang sesuai dan kecocokan matrik diberikan seperti persamaan (2.48) dan persamaan (2.49):

$$M_{antlion} = \begin{pmatrix} A_{1,1} & A_{1,2} & \dots & \dots & A_{1,d} \\ A_{2,1} & A_{2,2} & \dots & \dots & A_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{n,1} & A_{n,2} & \dots & \dots & A_{n,d} \end{pmatrix} \quad (2.48)$$

$$M_{OAL} = \begin{bmatrix} f([A_{11} & A_{12} & \dots & \dots & A_{1d}]) \\ f([A_{21} & A_{22} & \dots & \dots & A_{2d}]) \\ \vdots \\ f([A_{n1} & A_{n2} & \dots & \dots & A_{nd}]) \end{bmatrix} \quad (2.49)$$

Dimana  $M_{OAL}$  adalah matrik yang menampilkan kecocokan nilai dari lokasi semut singa(*ant lion*).

#### b. Membangun Perangkat

Untuk mendapatkan probabilitas yang lebih tinggi untuk menangkap semut digunakan roda *roulette(roulette wheel)*. Mekanisme ini mengidentifikasi singa semut yang paling cocok.

c. Menjebak dalam lubang semut singa

Persamaan matematika untuk menjebak semut-semut diberikan sesuai persamaan (2.50):

$$\begin{aligned}c_i^t &= Antlions_t^t + c^t \\d_i^t &= Antlions_t^t + d^t\end{aligned}\tag{2.50}$$

Dimana :

- $c_i^t$  : nilai minimum dari semua variabel pada  $i$ th iterasi
- $d_i^t$  : nilai maksimum dari semua variabel pada  $i$ th iterasi
- $c^t$  : nilai minimum dari semua variabel untuk  $i$ th semut
- $d^t$  : nilai maksimum dari semua variabel untuk  $i$ th semut

d. Semut – semut tergelincir ke arah semut singa

Untuk menggeser semut – semut ke arah semut singa, semut singa menembakkan pasir ke luar. Secara matematik dapat dimodelkan seperti pada persamaan (2.51):

$$\begin{aligned}C^t &= \frac{c^t}{I} \\D^t &= \frac{d^t}{I}\end{aligned}\tag{2.51}$$

Dimana :

- $I$  :  $10^{\omega \frac{t}{T}}$
- $t$  : *Current iteration*
- $T$  : maksimal dari nomer iterasi

e. Menangkap mangsa dan membangun kembali lubang

Tahap terakhir dari perilaku berburu dari semut singa adalah menangkap seekor semut yang mencapai batas dasar lubang dan kemudian kembali untuk memperbarui posisi dari posisi paling akhir dengan mengikuti persamaan (2.52):

$$Antlion_j^t = Ant_i^t \quad \text{if} \quad f(Ant_i^t) > f(Antlion_j^t) \quad (2.52)$$

f. Elitisme (*elitism*)

Yang terpenting dari evolusi algoritma adalah menemukan solusi terbaik utama. Ini dapat dimodelkan dengan persamaan (2.53):

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \quad (2.53)$$

Dimana :

$R_A^t, R_E^t$  : jalan-jalan secara acak di sekitar semut singa dengan roda roulette

g. Penerapan dari ALO

Menurut[19] secara detail kerja dari algoritma ini adalah sebagai berikut:

1. Baca data saluran dan beban dari sistem dan menyelesaikan aliran penyulang saluran untuk sistem yang menggunakan cara aliran beban.
2. Temukan lokasi DG terbaik menggunakan metode indeks vektor.
3. Inisialisasi dari populasi atau solusi dan  $it_{max} = 100$ , nomer dari lokasi DG d=1 untuk  $dg_{min} = 40, dg_{max} = 16800$ .
4. Hasilkan populasi dari ukuran DG secara acak menggunakan persamaan (2.54):

$$populasi = (dg_{max} - dg_{min}) \times rand() + dg_{min} \quad (2.54)$$

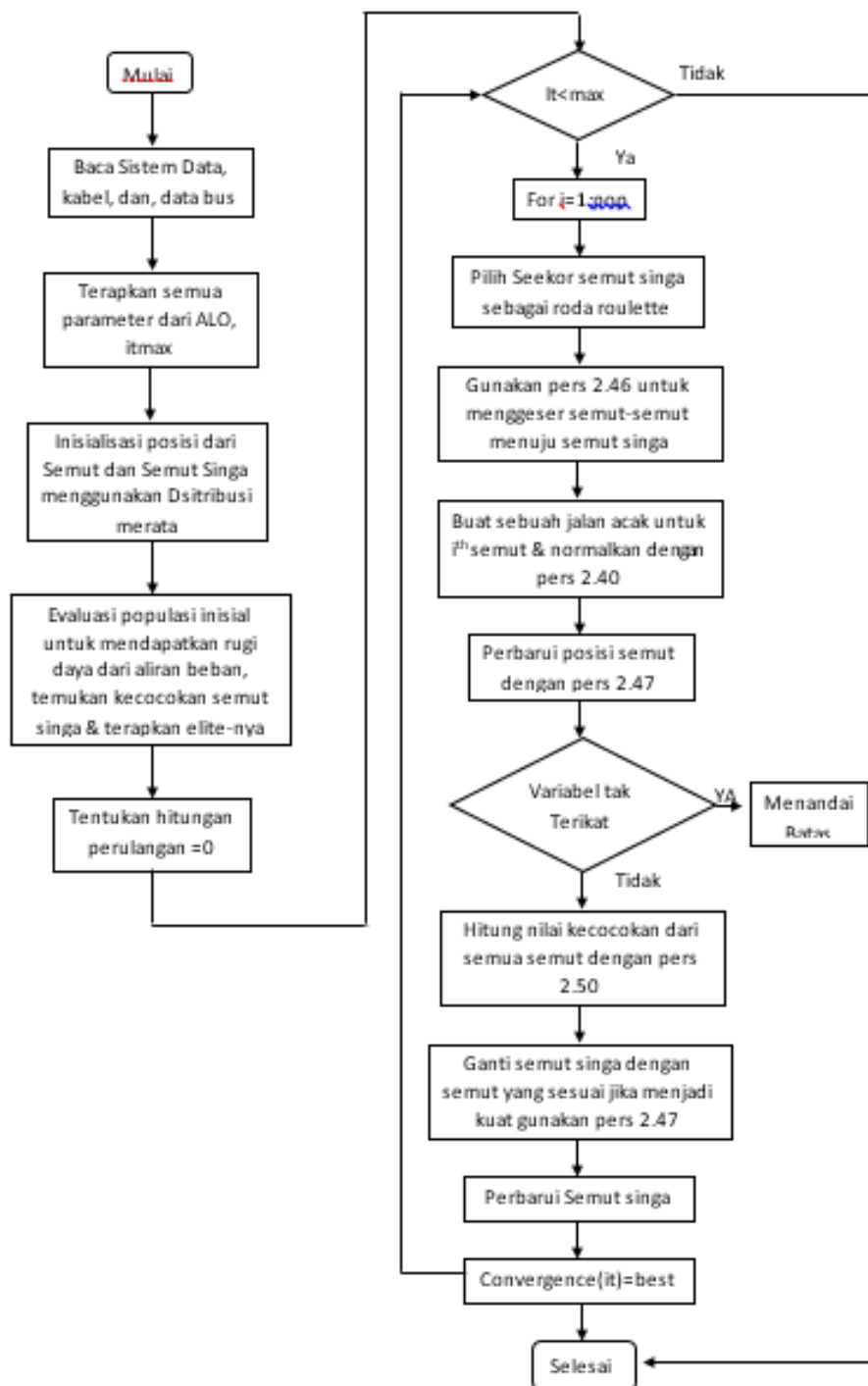
Dimana  $dg_{min}$  dan  $dg_{max}$  adalah batas minimum dan maksimum dari ukuran DG.

5. Temukan rugi daya untuk populasi yang dihasilkan.
6. Nilai DG dengan rugi-rugi rendah adalah solusi arus terbaik.

7. Menggunakan persamaan (2.43) dan (2.46) perbarui posisi dari semut singa.
8. Untuk memperbarui populasi menentukan rugi-rugi dengan kemampuan aliran daya.
9. Jika diperoleh rugi-rugi maka kurangi dan ganti solusi arus terbaik dengan it atau else kembali ke langkah 7.
10. Cetak hasil iterasi maksimum yang telah ditentukan.

#### 2.2.7 *Flowchart* cara kerja ALO

Untuk memahami secara mendetail cara kerja dari program ALO dibuatlah bagan alirnya. Pada **Gambar 2.12** adalah cara kerja *Ant Lion Optimizer*(ALO) *algorithm* sebagai penempatan pembangkit tersebar. Dimana ALO ini layak digunakan sebagai metode metaheuristik dalam menentukan lokasi dan kapasitas DG dari gejolak permasalahan di Sistem Distribusi tipe Radial.



Gambar 2.12 Cara kerja ALO[3]

## 2.2.8 Fungsi Kekangan

### 1. Kekangan keseimbangan daya

Keseimbangan daya sebagai fungsi pembatas dapat dirumuskan dengan persamaan (2.55):

$$P + \sum_{k=1}^N P_{DG} = P_d + P_{loss} \quad (2.55)$$

## 2. kekangan kapasitas DG

Hasil optimasi juga harus memenuhi ketersediaan batas kapasitas DG sesuai dengan kapasitas standart yang digunakan. Namun dalam paper[3] disebutkan batasan kapasitas DG sebagai berikut :

$$40 < P_{DG} < 168000$$

Dimana nilai batasan kapasitas *photovoltaic Distributed Generation* tersebut pada saat keadaan daya dihasilkan maksimal. Data DG tersebut dihasilkan berdasarkan pemantauan *real time* dari *Queensland University Photovoltaic*[22].

### 2.2.9 Penentuan Kapasitas DG

Tujuan adanya penerapan DG baru adalah salah satunya untuk meminimalkan rugi daya. Cara yang digunakan untuk meminimalkan rugi daya aktif pada saluran dapat dirumuskan seperti persamaan (2.56)[1].

$$f(x) = \min \sum_{i=1}^N P_{Li} \quad (2.56)$$

Pendekatan pertama dalam penentuan DG adalah dengan pendekatan analitis. Total rugi daya aktif pada sistem tenaga listrik direpresentasikan dengan persamaan (2.57)[1], atau lebih populer disebut “*exact loss formula*” (Elgerd, 1973) :

$$P_L = \sum_{i=1}^N \sum_{j=1}^N [\alpha_{ij}(P_i P_j + Q_i Q_j) + \beta_{ij}(Q_i P_j - P_i Q_j)] \quad (2.57)$$

dengan,

$$\alpha_{ij} = \frac{r_{ij}}{v_i v_j} \cos(\delta_i - \delta_j)$$

$$\beta_{ij} = \frac{r_{ij}}{v_i v_j} \sin(\delta_i - \delta_j)$$

$V_i < \delta_i$  : Tegangan kompleks pada bus ke- $i$



- $r_{ij} + j_{ij} = Z_{ij}$  : Elemen ke- $ij$  dari matrik impedans [ $Z_{bus}$ ]  
 $P_i$  dan  $P_j$  : Injeksi daya aktif masing-masing pada bus ke- $i$  dan ke- $j$   
 $Q_i$  dan  $Q_j$  : Injeksi daya reaktif masing – masing pada bus ke- $i$  dan ke- $j$   
 $N$  : jumlah bus

Anggapan awal bahwa  $\alpha = (sign) \tan(\cos^{-1}(PF_{DG}))$ , maka keluaran daya reaktif DG dapat diekspresikan dengan persamaan (2.58)[1]:

$$Q_{DGi} = \alpha P_{DGi} \quad (2.58)$$

Dengan,

Sign = +1 : DG menginjeksi daya reaktif

Sign = -1 : DG mengambil daya reaktif

$PF_{DG}$  adalah faktor daya dari DG

Daya aktif (P) dan daya reaktif (Q) yang diinjeksikan pada bus  $i$ , dimana DG ditempatkan, diekspresikan dengan persamaan (2.57) dan (2.58)[1] :

$$P_i = P_{DGi} - P_{Di} \quad (2.59)$$

$$Q_i = Q_{DGi} - Q_{Di} = \alpha P_{DGi} - Q_{Di} \quad (2.60)$$

Dari persamaan (2.56), (2.57), rugi daya aktif dapat dituliskan seperti pada persamaan 2.61[1]:

$$P_L = \sum_{i=1}^N \sum_{j=1}^N \{ \alpha_{ij} [P_{DGi} - P_{Di}] P_j + (\alpha P_{DGi} - Q_{Di}) Q_j \} + \beta_{ij} [(\alpha P_{DGi} - Q_{Di}) P_j - (P_{DGi} - P_{Di}) Q_j] \} \quad (2.61)$$

Total rugi-rugi daya aktif pada sistem akan minimum jika turunan parsial dari persamaan (2.61) terhadap injeksi daya aktif dari DG ke bus  $i$  menjadi 0. Setelah disederhanakan dan penyusunan ulang seperti pada persamaan (2.62)[1].

$$\frac{\partial PL}{\partial P_{DGi}} = 2 \sum_{j=1}^N [\alpha_{ij}(P_j + \alpha Q_j) + \beta_{ij}(\alpha P_j - Q_j)] = 0 \quad (2.62)$$

Persamaan (2.63) dapat dituliskan dapat dituliskan sebagai berikut :

$$\alpha_{ii}(P_i + \alpha Q_i) + \beta_{ii}(\alpha P_i - \alpha Q_i) + \beta_{ij}(\alpha P_j - Q_j) = 0 \quad (2.63)$$

$$\alpha \sum_{j=1, j \neq i}^N (\alpha_{ij} Q_j - \beta_{ij} P_j) = 0 \quad (2.64)$$

$$\text{set} \begin{cases} X_i = \sum_{j=1, j \neq i}^N (\alpha_{ij} P_j - \beta_{ij} Q_j) \\ Y_i = \sum_{j=1, j \neq i}^N (\alpha_{ij} Q_j - \beta_{ij} P_j) \end{cases} \quad (2.65)$$

Dari persamaan (2.59), persamaan (2.60), persamaan (2.64), dan persamaan (2.65), dapat dikembangkan seperti persamaan (2.66)[1]:

$$\alpha_{ii}(P_{DGi} - P_{Di} + \alpha^2 P_{DGi} - \alpha Q_{DGi}) + \beta_{ii}(Q_{Di} - \alpha P_{Di}) + X_i + \alpha Y_i = 0 \quad (2.66)$$

Dari persamaan (2.66), nilai kapasitas DG yang optimal di setiap bus  $i$  untuk meminimalkan rugi daya aktif dapat dituliskan dengan persamaan (2.67):

$$P_{DGi} = \frac{\alpha_{ii}(P_{Di} + \alpha Q_{Di}) + \beta_{ii}(\alpha P_{Di} - Q_{Di}) - X_i - \alpha Y_i}{\alpha^2 \alpha_{ii} + \alpha_{ii}} \quad (2.67)$$

Faktor daya dari DG sangat mempengaruhi pada kondisi operasi dan tipe DG. Sehingga dari tipe – tipe DG tersebut memiliki pengaruh yang berbeda dalam mengoptimalkan rugi daya. Dimana nilai tersebut dimasukan pada masing – masing bus  $i$ , menurut[1] tiap-tiap tipe DG dapat dibuat persamaan matematis sebagai berikut :

- a. DG tipe 1 : faktor daya bernilai 1,  $PF_{DG} = 1$ ,  $\alpha = 0$ . Dari persamaan (2.67), kapasitas optimal dari DG dari masing-masing bus  $i$  untuk meminimalkan rugi-rugi daya dengan mereduksi persamaan (2.68) berikut :

$$P_{DGi} = P_{Di} - \frac{1}{\alpha_{ii}} [\beta_{ii} Q_{Di} + \sum_{j=1, j \neq i}^N (\alpha_{ij} P_j - \beta_{ij} Q_j)] \quad (2.68)$$

- b. DG tipe 2 : diasumsikan  $PF_{DG} = 0, \alpha = \infty$ , dari persamaan (2.58) dimasukkan ke persamaan (2.61) kapasitas optimal dari DG dari masing-masing bus  $i$  untuk meminimalkan rugi daya dengan mereduksi persamaan (2.69)

$$Q_{DGi} = Q_{Di} + \frac{1}{\alpha_{ii}} [\beta_{ii} P_{Di} - \sum_{j=1, j \neq i}^N (\alpha_{ij} Q_j - \beta_{ij} P_j)] \quad (2.69)$$

- c. DG tipe 3 : diasumsikan  $0 < P_{DG} < 1, sign = +1$  dan " $\alpha$ " adalah konstanta, kapasitas optimal DG pada masing-masing bus  $i$  untuk meminimalkan rugi-rugi daya secara berurutan dapat menggunakan persamaan (2.67) dan persamaan (2.60).
- d. DG tipe 4 : diasumsikan  $0 < P_{DG} < 1, sign = -1$  dan " $\alpha$ " adalah konstanta, kapasitas DG optimal pada masing-masing bus  $i$  untuk meminimalkan rugi-rugi daya secara berurutan dapat menggunakan persamaan (2.67) dan persamaan (2.60).

### 2.2.10 Fungsi Tujuan

#### 1. Meminimalkan rugi daya

Dalam meminimalkan rugi daya akibat jatuh tegangan, maka dapat dirumuskan seperti persamaan (2.70):

$$P_{loss} = \sum_i^n I_i^2 R^2 \quad (2.70)$$

#### 2. Mengurangi biaya pembelian energi dari jaringan pembangkit

Keuntungan utama dari pembangkit tenaga listrik melalui unit-unit DG dalam sistem distribusi adalah untuk mengurangi daya injeksi dari jaringan pembangkit. Tambahan, dalam riset ini dijadwalkan selama 20 tahun periode, inflasi dan suku bunga seharusnya diterapkan dalam perhitungan harga listrik. Terlebih, laba yang

berasal dari pembukaan unit-unit DG baru ke jaringan distribusi ditentukan dengan persamaan (2.71)[4]:

$$C_{DG}^{Purchase} = \sum_{i=1}^{N_{dg}} P_{DG}^i \times \rho = P_{DG}^i \times \rho^0 \times \left( \frac{1+InfR}{1+IntR} \right)^{i-1} \quad (2.71)$$

Dimana :

$P_{DG}^i$  : Kapasitas daya pembangkit tersebar

$\rho$  : Harga energi listrik

$N_{DG}$  : nomer dari DGs

$i$  : Periode waktu belajar (Tahun)

$\rho^0$  : harga listrik dalam tahun pertama (\$/MWh)

$InfR$  : besarnya inflasi/tahun

$IntR$  : besarnya suku bunga/tahun

### 3. Peningkatan keandalan

Keandalan adalah keuntungan lain dari penerapan DG secara optimal dalam sistem distribusi. Karena gangguan terjadi dalam cabang-cabang sistem distribusi, beberapa pelanggan dapat terkena dampak dalam jangka waktu yang lama dalam lokasi gangguan. Selain itu, beberapa pelanggan di bagian yang tidak terkena gangguan dari sistem distribusi yang terhubung, secara otomatis menjadi terpisah dari bagian yang terjadi gangguan. Dalam kondisi ini, DG dapat digunakan untuk mengembalikan daya ke bagian distribusi cabang gangguan dan meningkatkan keandalan dari sistem distribusi. Jadi sistem pembangkit tersebar secara optimal diterapkan untuk mengurangi biaya dari sistem distribusi dalam kerusakan pelanggan. Dalam studi ini keandalan dimodelkan dalam CENS dari sistem distribusi. *Cost of Energy not supply* (CENS) didefinisikan seperti pada persamaan (2.72)[4].

$$C_{DG}^{ENS} = \sum_{i=1}^{N_{branch}} \times \sum_{j=1}^{N1} \times \lambda_i \times L_i \times \rho_0^{int} \times \left( \frac{1+InfR}{1+IntR} \right)^{i-1} \times t_i \times D_j \quad (2.72)$$

Dimana :

$C_{DG}^{ENS}$  : biaya energi yang tidak tersuplai/tahun

- $N_i$  : nomer dari cabang *grid*  
 $L_i$  : panjang cabang sistem distribusi  
 $t_i$  : waktu yang dipakai untuk pergantian gangguan  
 $\rho_0^{int}$  : biaya energi yang tidak disuplai di tahun pertama  
 $\lambda_i$  : tingkat atau pesat kegagalan cabang , dimana  
 $\lambda_i$  :  $\frac{\text{jumlah gangguan selama selang waktu tertentu}}{\text{jumlah selang waktu pengamatan}}$   
 $D_j$  : Permintaan beban pada titik j

#### 4. Mengurangi biaya penggunaan DG (DGAC)

Biaya penggunaan DG masuk dalam investasi ( $C_{DG}^{Invs}$ ), operasi ( $C_{DG}^{Oper}$ ) dan biaya-biaya perawatan ( $C_{DG}^{Main}$ ). Biaya penggunaan DG dapat didefinisikan dengan persamaan (2.73)[4].

$$DGAC = C_{DG}^{Invs} + C_{DG}^{Oper} + C_{DG}^{Main} \quad (2.73)$$

Dimana,  $C_{DG}^{Invs}$  (\$/MWh) akan di bayar di awal dan itu terkait ke pembelian dan pemasangan masing-masing DG. Dapat dihitung dengan persamaan (2.74):

$$C_{DG}^{Invs} = \sum_{i=1}^{Ndg} C_i^{investment} \quad (2.74)$$

Dimana,

$C_i^{investment}$  : biaya pembelian dan pemasangan dari i-th unit-unit DG.

$C_{DG}^{Main}$  (\$/MWh) adalah biaya tahunan yang menampilkan suku bunga dan suku kenaikan dan dapat dimodelkan seperti persamaan (2.75):

$$C_{DG}^{Main} = \sum_{i=1}^{Ndg} C_i^{maintenance} \quad (2.75)$$

Dimana,  $C_i^{maintenance}$  : biaya perawatan dari i-th DG dan dapat dihitung seperti persamaan (2.76):

$$C_i^{maintenance} = C_0^{maintenance} x \left( \frac{1+InfR}{1+IntR} \right)^{i-1} \quad (2.76)$$

Dimana,

$C_0^{maintenance}$  : biaya perawatan awal (nilai = 0)

#### 5. *Load Line Index*(LLI)

*Load Line Index* (LLI) merupakan indek kemungkinan penambahan beban di masa yang akan datang. Menurut[4] pertumbuhan jumlah beban saluran setiap tahun sekitar 1%. Sehingga pada penelitian ini dengan pengujian sistem Distribusi tipe Radial Standar IEEE bisa mengetahui perkembangan beban bus untuk 20 tahun yang akan datang. *Load Line Index*(LLI) bisa dirumuskan seperti pada persamaan (2.77):

$$L_i = L_o x (1 + L_{growth})^{i-1} \quad (2.77)$$

Dimana:

$L_i$  (*Load Increment*) : Besar kenaikan beban bus

$L_o$  : Beban pada saat tahun pertama

$L_{growth}$  : pertumbuhan beban setiap tahun