

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Proteksi Tenaga Listrik

2.1.1 Pengertian Sistem Proteksi

Sistem proteksi adalah susunan perangkat proteksi secara lengkap yang terdiri dari perangkat utama dan perangkat-perangkat lain yang dibutuhkan untuk melakukan fungsi proteksi untuk mencegah atau membatasi kerusakan pada peralatan terhadap gangguan demi keandalan penyaluran tenaga listrik.

2.1.2 Persyaratan Sistem Proteksi

Adapun syarat syarat system proteksi adalah sebagai berikut

1. Sensitivity (Kepekaan)

Relay harus dapat merasakan gangguan baik di daerahnya maupun di daerah cadangan jauhnya. Sebagai pengaman pada peralatan, relay harus bisa mendeteksi lebih dini adanya gangguan sehingga dapat meminimalisir kerusakan pada peralatan

Relay yang kurang peka menyebabkan banyak terjadinya gangguan yang meluas. Namun relay yang terlalu peka, kan menyebabkan kejadian seringnya trip pemutus tenaga akibat gangguan yang kecil

2. Reliability (Keandalan)

Yaitu tingkat kemampuan bekerja. Dimana relay sangat diandalkan kinerjanya, dimana tidak boleh ada kejadian gagal trip. Relay juga tidak boleh salah kerja. Salah kerja yang dimaksud adalah bekerja tidak sebagaimana mestinya.

3. Selectivity (Selektifitas)

Sistem proteksi relay harus bisa memisahkan daerah kerjanya atau bekerja di dalam daerah pengamannya saja. Untuk itu relay dibuat bermacam macam jenis dan karakteristik. Dengan pemilihan jenis dan setting yang tepat, maka selektifitas yang baik bias didapatkan

4. Speed (Kecepatan)

Untuk meminimalisir kerusakan peralatan, dibutuhkan waktu kerja relay yang sangat cepat. Daerah yang terganggu harus segera diputus secepat mungkin sampai Kawasan yang terganggu benar benar terputus dari Kawasan lainnya yang tidak mengalami gangguan.

2.2 Relay Arus Lebih (OCR)

2.2.1 Pengertian Relay Arus Lebih

Relay arus lebih adalah relay yang bekerja terhadap arus lebih, ia akan bekerja bila arus yang mengalir melebihi nilai setting, baik yang disebabkan adanya gangguan hubung singkat atau kelebihan beban untuk kemudian memberikan perintah *trip* ke PMT sesuai karakter waktunya. Relay arus lebih saat ini memiliki 2

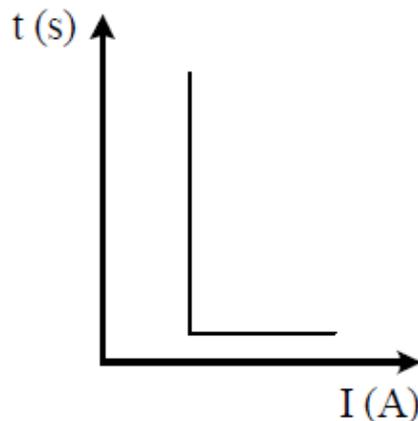
kemampuan yaitu sebagai Relay arus lebih (Over Current Relay) dan Relay gangguan tanah (Ground Fault Relay). Relay arus lebih dapat berkoordinasi dengan Relay lain atau dengan GFR dengan memberikan tunda waktu sesuai setting karakternya

2.2.2 Jenis Relay berdasarkan Karakteristik Waktu

1. Instantaneous OCR (Relay Arus Lebih Waktu Kerja Seketika)

Relay ini akan bekerja dengan seketika tanpa adanya delay waktu jika arus yang mengalir melebihi nilai settingnya.

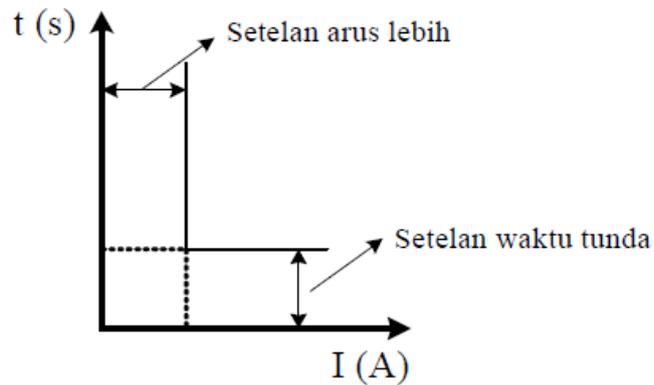
Karakteristiknya sebagai berikut



Gambar 2.1 Kurva waktu Instan

2. Definite Time OCR (Relay Arus Lebih Waktu Kerja Tertentu)

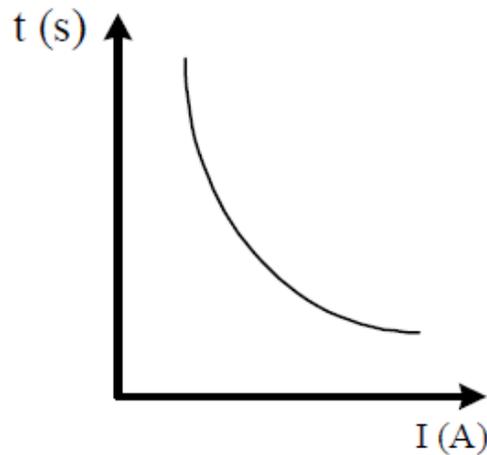
Relay ini bekerja dengan waktu tunda yang telah ditentukan. Jenis ini memungkinkan setting menjadi bervariasi untuk mengatasi besar arus gangguan yang berbeda dengan menggunakan waktu operasi berbeda



Gambar 2.2 Kurva waktu Definite

3. Invers Time Relay (Relay Arus Lebih Kerja Terbalik)[6]

Cara kerja relay ini pada dasarnya adalah semakin besar arus gangguan maka semakin cepat waktu kerja dari relay tersebut. Keuntungan dari relay ini adalah untuk arus yang sangat tinggi, waktu untuk membuka (trip) menjadi sangat pendek didapatkan tanpa resiko terhadap selektivitas.

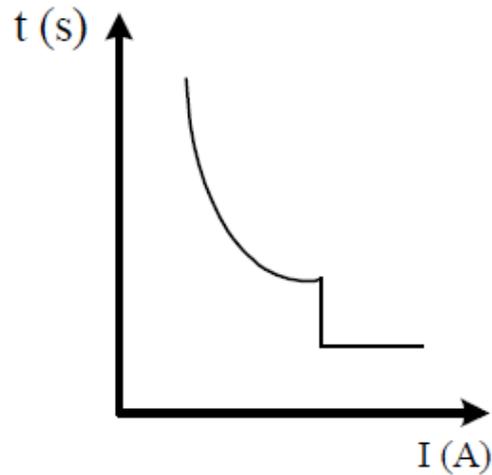


Gambar 2.3 Kurva waktu Inverse

4. *Invers Definite Minimum Time* OCR (Relay Arus Lebih IDMT)

Semakin besar arus gangguan yang terjadi maka akan semakin cepat relay bekerja. Tetapi pada saat tertentu yaitu pada saat mencapai waktu

yang telah ditentukan maka kerja relay tidak lagi ditentukan oleh arus gangguan tetapi oleh waktu. Keuntungan menggunakan relay jenis ini adalah sebagai pengaman banyak saluran.



Gambar 2.4 Kurva waktu IDMT

Tabel 2.1 IDMT SI standar IEC 60255

Karakteristik Rele	Standar IEC 60255
Standard Inverse (SI)	$T = TMS \times \frac{0.14}{I_r^{0.02}-1}$
Very Inverse (VI)	$T = TMS \times \frac{13.5}{I_r-1}$
Extremely Inverse (EI)	$T = TMS \times \frac{80}{I_r^2-1}$
Long time standart earth fault	$T = TMS \times \frac{120}{I_r-1}$

2.2.3 Prinsip Kerja OCR

Prinsip kerja Relay OCR adalah mendeteksi adanya arus yang melebihi setting di dalam belitan relay, baik akibat gangguan hubung singkat maupun overload. Kemudian memberikan perintah TRIP ke Circuit Breaker. Pada kondisi normal arus mengalir ke

dalam relay namun arus masih di bawah setting, maka arus tidak bekerja

2.2.4 Prinsip Kerja GFR

Prinsip kerja GFR mendeteksi adanya hubung singkat ke tanah. Dimana pada saat beban seimbang, arus di fasa RST nilainya sama sehingga di titik netral tidak timbul arus. Bila terjadi gangguan arus hubung singkat ke tanah sehingga menyebabkan beban tidak seimbang, maka akan timbul arus urutan nol pada kawat Netra yang menyebabkan relay GFR bekerja

2.2.5 Setting OCR dan GFR

1. Setting OCR

Yang diperlukan dalam menentukan setting OCR adalah terlebih dahulu menghitung arus nominal trafo tenaga.

Rumus menghitung arus nominal adalah

$$I_{nom} = \frac{VA}{V \cdot \sqrt{3}} \dots\dots\dots (2.1)$$

Arus setting untuk relay OCR baik di sisi primer maupun sekunder adalah

$$I_{set} (prim) = 1,05 \times I_{nom} \text{ trafo} \dots\dots\dots (2.2)$$

Dimana nilai tersebut adalah nilai di sisi primer. Maka untuk menentukan setting untuk diinput ke dalam relay adalah

$$I_{set} (sek) = I_{set} (prim) : \text{Rasio CT} \dots\dots\dots (2.3)$$

Untuk formulasi kurva waktu adalah sebagai berikut

2. Setting GFR

Yang diperlukan dalam menentukan setting GFR adalah terlebih dahulu menghitung arus nominal trafo tenaga. Arus setting untuk relay GFR baik di sisi primer maupun sekunder adalah

$$I_{set} (\text{Prim}) = 10\% \times I_{nom} \text{ trafo} \dots\dots\dots (2.4)$$

Dimana nilai tersebut adalah nilai di sisi primer. Maka untuk menentukan setting untuk diinput ke dalam relay adalah

$$I_{set} (\text{sek}) = I_{set} (\text{prim}) : \text{Rasio CT} \dots\dots\dots (2.5)$$

2.2.6 Perhitungan Arus Gangguan Hubung Singkat

1. Arus gangguan hubung singkat tiga fasa

Perhitungan arus gangguan hubung singkat tiga fasa dengan 3, kedalam persamaan:

$$I_{hs3ph} = \frac{V_{L-N}}{Z_1} \dots\dots\dots (2.6)$$

Dengan cara yang sama maka untuk perhitungan arus gangguan hubung singkat tiga fasa pada titik lokasi gangguan penyulang 20 kV Transformator Daya yang lain

2. Arus gangguan hubung singkat 2 fasa

Perhitungan arus gangguan hubung singkat dua fasa kedalam persamaan:

$$I_{hs2ph} = \frac{V_{L-L}}{Z_1 + Z_2} \dots\dots\dots (2.7)$$

Dengan cara yang sama maka untuk perhitungan arus gangguan hubung singkat dua fasa pada titik lokasi gangguan

penyulang 20 kV Transformator Daya yang

3. Arus gangguan hubung singkat 1 fasa ke tanah

Perhitungan arus gangguan hubung singkat dua fasa kedalam persamaan:

$$I_{hs1ph} = \frac{3V_{L-N}}{Z_1 + Z_2 + Z_3} \dots\dots\dots (2.8)$$

Dengan cara yang sama maka untuk perhitungan arus gangguan hubung singkat satu fasa ke tanah pada titik lokasi gangguan penyulang 20 kV Transformator Daya yang lain

2.3 Gangguan di Gardu Induk

Dalam sistem tenaga listrik tentunya tidak terlepas dari gangguan yang menyebabkan sistem tenaga listrik di Gardu induk tersebut mengalami masalah. Permasalahan yang dimaksud adalah menyebabkan terjadinya gangguan keseimbangan system 3 fasa atau menyebabkan kerusakan pada suatu peralatan. Jenis gangguan yang menyebabkan relay Overcurrent bekerja terdiri dari 2 jenis. Yaitu

2.3.1 Gangguan Beban Lebih (Overload)

Gangguan jenis ini terjadi akibat dari kesalahan pengaturan sistem dimana bukan terjadi akibat kesalahan fungsi peralatan tegangan tinggi. Namun gangguan ini bisa membahayakan jika dibiarkan terus menerus karena kapasitas arus yang mengalir melebihi setting kapasitas peralatan

2.3.2 Gangguan Hubung Singkat

Sama seperti gangguan jenis sebelumnya, gangguan ini diakibatkan

adanya kelainan power system fault. Tegangan lebih bisa dipengaruhi oleh akibat petir yaitu loncatan listrik diantara awan dengan bumi. Dimana secara tiba tiba ada tegangan yang sangat besar yang mengalir pada peralatan di Gardu Induk

2.4 *Particle Swarm Optimization (PSO)*

Particle Swarm Optimization merupakan metode komputasi algoritma yang memiliki kesamaan sifat dengan algoritma genetika atau biasa disebut dengan alternatif genetika algoritma yang didasarkan pada *swarm intelligence behaviorally inspired*

Dalam konteks optimasi multivariabel, kawanan diibaratkan memiliki ukuran tertentu dengan setiap partikel berada di suatu lokasi yang acak dalam ruang multidimensi. Setiap partikel diasumsikan memiliki dua parameter, yaitu posisi dan kecepatan yang bergerak dalam ruang tertentu dan mengingat posisi terbaik yang pernah dilalui atau ditemukan terhadap sumber makanan yang akan dituju atau nilai fungsi objektif. Setiap partikel menyampaikan informasi atau posisi terbaiknya kepada partikel yang lain sehingga partikel lain dapat mengikuti posisi dan kecepatan masing-masing berdasar informasi partikel yang menemukan sumber terbaik. Berikut merupakan tahap dari PSO.

2.4.1 Proses Inisiasi

Inisiasi sekumpulan *particle* secara random dimana setiap *particle* merepresentasikan solusi yang mungkin untuk masalah optimasi. Pada tahap ini biasanya beberapa parameter *particle* memiliki batasan-batasan pada masing-

masing kasus yang akan dioptimasi. Hal ini dilakukan untuk mencegah nilai random inisialisasi yang terlalu jauh.

Parameter yang sangat penting diinisialisasi pada algoritma PSO adalah posisi dari setiap *particle* (X_i), kecepatan dari setiap *particle* (V_i), dan posisi terbaik dari setiap *particle* (P_i). Ketiga parameter ini dapat berupa lebih dari satu parameter yang akan dioptimasi atau biasa disebut dengan *multi objective*. Kemudian parameter lainnya yang harus diinisialisasi adalah nilai-nilai konstanta, seperti c_1 , c_2 , c_3 , r_1 , dan r_2 . Keempat konstanta tersebut dapat diperbaharui setiap iterasi nantinya atau hanya ditentukan sebuah angka (0 sampai 1).

2.4.2 Proses Iterasi

Pada proses ini parameter-parameter penting yang sudah diinisiasi pada tahap sebelumnya masuk ke dalam tahap pembaharuan nilai-nilai parameter untuk mendapatkan nilai yang terbaik. Adapun rumus untuk menghitung nilai-nilai parameter yang akan diperbaharui pada tahap iterasi adalah sebagai berikut

$$v_{k+1} = w_k v_k + c_1 r_1 (P_{best} - x_k) + c_2 r_2 (x_k - P_{notbest}) + c_3 r_3 (G_{best} - x_k) \dots \dots \dots (2.9)$$

Dimana :

w_k = inertia weight

c_1 = faktor learning untuk *particle*

c_2 = faktor learning untuk *swarm*

r_1 = random nilai 1 (0 sampai 1)

r_2 = random nilai 2 (0 sampai 1)

r_3 = random nilai 3 (0 sampai 1)

v_k = kecepatan setiap *particle*

x_k = posisi setiap *particle*

v_{k+1} = kecepatan baru setiap *particle*

P_{best} = posisi terbaik dari *particle*

$P_{notbest}$ = posisi bukan terbaik dari *particle*

G_{best} = posisi terbaik dari populasi

Nilai dari w_k disesuaikan dengan permasalahan optimasi yang diinginkan. Hal ini dikarenakan nilai dari formula untuk mencari nilai w_k adalah beragam. Pemilihan nilai w_k yang tepat akan mempercepat proses optimasi. Dalam algoritma PSO terdapat *update* pembatas kecepatan. Hal ini merupakan salah satu pembaharuan dari algoritma PSO. Hal ini dilakukan untuk mempercepat proses konvergensi [16]. Formula untuk membatasi kecepatan pada algoritma PSO adalah sebagai berikut :

$$v(i,j)=R \times (X_{max}-X_{min}) \dots\dots\dots (2.10)$$

$$R=R_{initial}+(R_{final}-R_{initial}) \times (iteriter_max) \dots\dots\dots (2.11)$$

Dimana :

$$R_{initial} > R_{final}$$

Setiap nilai dari posisi setiap *particle* (X_i) dimana X_i ditentukan nilai kenaikan dan penurunannya sesuai dengan permasalahan optimasinya lalu dibandingkan nilai terbaik yang dicapai oleh *local best* (P_i). Jika nilai dari fluktuasi $X_i < P_{best}$ maka P_{best} diganti dengan nilai fluktuasi dari X_i sedangkan jika nilai fluktuasi $X_i > P_{best}$ maka $P_{notbest}$ diganti dengan nilai fluktuasi dari X_i . Selain itu, setiap nilai fluktuasi dari *particle* akan dibandingkan dengan nilai terbaik yang dicapai dalam populasi G_{best} .

Jika *update* nilai posisi setiap *particle* telah mencapai kondisi dimana nilai iterasi maksimum atau perulangan telah mencapai nilai optimum maka perulangan berhenti. Kemudian nilai optimumnya akan diambil sebagai hasil dari optimasi PSO. Jika belum mendapatkan hasil yang optimal, iterasi akan diulangi dengan masuk pada batasan antara posisi setiap *particle*, posisi terbaik *particle*, dan posisi terbaik populasi.

