

BAB II

DASAR TEORI

2.1 Tinjauan Pustaka

Pada penelitian sebelumnya Penyelesaian analisis aliran daya menggunakan metode Gauss-Seidel dengan bantuan program MATLAB memberikan hasil yang cepat dan akurat (I Putu Suka Asra, 2004), tetapi metode Gauss-Seidel hanya cocok untuk sistem tenaga listrik yang memiliki sedikit bus.

Selanjutnya, Emmy Hosea, dkk (2005) melakukan penelitian untuk membandingkan analisis aliran daya menggunakan metode Algoritma Genetika dengan metode Newton-Raphson. Hasil penelitian diperoleh bahwa, metode iterasi Newton-Raphson maupun metode Algoritma Genetika dapat digunakan untuk menentukan nilai parameter Bus dalam perhitungan aliran daya. Tetapi dilihat dari proses komputasi, metode Newton-Raphson dapat menyelesaikan perhitungan dengan waktu komputasi yang lebih cepat dibandingkan dengan waktu komputasi pada metode Algoritma Genetika untuk mencapai kriteria berhenti yang sama.

Kemudian pada penelitian Metode Injeksi Arus (*Current Injections Method*) merupakan metode baru hasil pengembangan dari metode Newton-Raphson yang digunakan untuk menganalisa aliran daya. Metode Injeksi Arus memiliki struktur matriks Jacobian yang diperbaharui pada setiap iterasi sehingga lebih cepat dalam melakukan analisis aliran daya. Manuaba, dkk (2009), melakukan penelitian mengenai analisis aliran daya dengan metode injeksi arus pada sistem distribusi 20 kV. Hasil penelitian menunjukkan bahwa, metode Injeksi Arus memerlukan

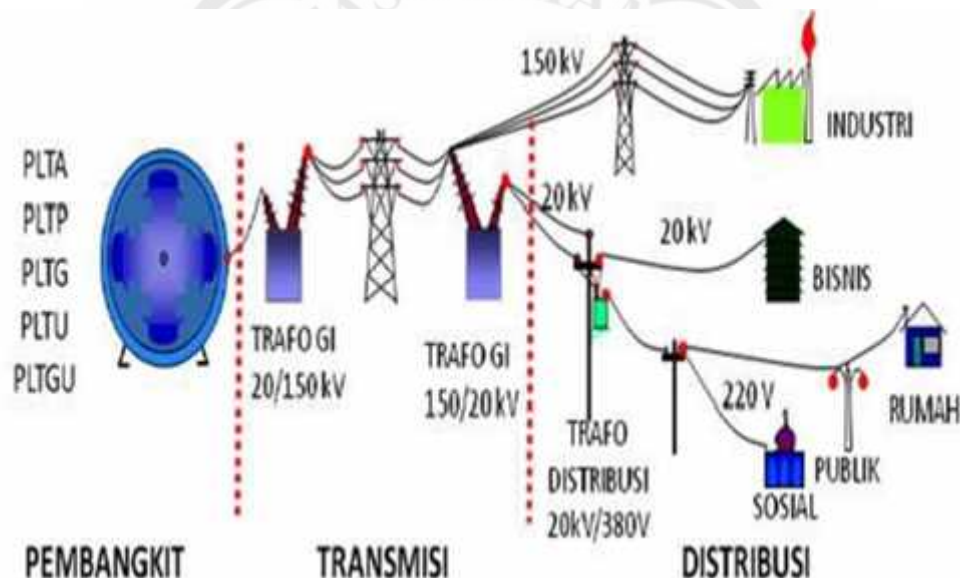
waktu komputasi yang lebih sedikit dibandingkan dengan metode Newton-Rhapson. Waktu yang dibutuhkan dalam proses komputasi menggunakan metode Injeksi Arus yang lebih singkat dibandingkan dengan metode Newton-Rhapson dapat terjadi karena metode Injeksi Arus memiliki struktur matriks Jacobian yang identik dengan matriks admitansi bus.

Penelitian lain yang berhubungan dengan aliran daya Abdul Jaleel dan Shabna, 2013 dengan judul analisis aliran beban dan evaluasi keandalan sistem tenaga 220 kV di Kerala. Studi aliran-aliran penting untuk merencanakan perluasan sistem tenaga kerja di masa depan dan juga dalam menentukan operasi terbaik dari sistem yang ada dan Newton Raphson salah satu metode efisien yang digunakan. Informasi utama yang diperoleh dari analisis aliran beban adalah besarnya sudut dan fase tegangan pada masing-masing bus, dan daya nyata dan reaktif mengalir di setiap garis. Diagram garis tunggal dimodelkan dengan menggunakan software Electrical Transient Analyzer Program (ETAP). Berbagai indeks reliabilitas juga diperoleh yang dapat digunakan sebagai ukuran keandalan sistem. Penempatan kapasitor optimal juga dilakukan dengan menggunakan ETAP. Aliran beban juga dilakukan dengan menggunakan software MATLAB.

Aroop Bardalai, Sanjib Hazarika dan Satyajit Bhuyan (2014) dalam penelitiannya mengenai analisis aliran daya pada sistem bus IEEE 57 menggunakan MATLAB. Tujuannya untuk menentukan besarnya voltase dan sudut yang sesuai untuk semua bus jaringan.

2.2 Sistem Tenaga Listrik

Sistem tenaga listrik merupakan suatu kesatuan yang terintegrasi mulai dari unit pembangkit listrik, unit transmisi listrik, sampai unit distribusi listrik dalam upaya menyalurkan listrik dari produsen kepada konsumen dengan dilengkapi sistem proteksi pada kesatuan tersebut. Menurut PT. PLN (Persero) (2010b : 2), ada tiga bagian penting dalam proses penyaluran tenaga listrik, yaitu Pembangkitan, Penyaluran (Transmisi), dan Distribusi. Secara umum skema sistem tenaga listrik diperlihatkan pada gambar 2.1



Gambar 2.1 Sistem Tenaga Listrik

<http://kuliahdaring.dikti.go.id>

2.3 Sistem Distribusi

Sistem distribusi merupakan keseluruhan komponen dari sistem tenaga listrik yang menghubungkan secara langsung antara sumber daya yang besar (seperti gardu transmisi) dengan konsumen tenaga listrik. Secara umum yang termasuk ke dalam sistem distribusi antara lain :

1. Gardu Induk (GI)

Pada bagian ini jika sistem pendistribusian tenaga listrik dilakukan secara langsung, maka bagian pertama dari sistem distribusi tenaga listrik adalah Pusat Pembangkit Tenaga Listrik dan umumnya terletak di pinggiran kota. Untuk menyalurkan tenaga listrik ke pusat - pusat beban (konsumen) dilakukan dengan jaringan distribusi primer dan jaringan distribusi sekunder. Jika sistem pendistribusian tenaga listrik dilakukan secara tak langsung, maka bagian pertama dari sistem pendistribusian tenaga listrik adalah Gardu Induk yang berfungsi menurunkan tegangan dari jaringan transmisi dan menyalurkan tenaga listrik melalui jaringan distribusi primer.

2. Jaringan Distribusi Primer

Jaringan distribusi primer merupakan awal penyaluran tenaga listrik dari Gardu Induk (GI) ke konsumen untuk sistem pendistribusian langsung. Sedangkan untuk sistem pendistribusian tak langsung merupakan tahap berikutnya dari jaringan transmisi dalam upaya menyalurkan tenaga listrik ke konsumen. Standar tegangan menengah di Indonesia adalah 20 kV (Abdul Kadir 2006 : 149). Untuk wilayah kota, tegangan diatas 20kV tidak diperkenankan, mengingat pada tegangan 20 kV akan terjadi gejala-gejala korona yang dapat mengganggu frekuensi radio, televisi, telekomunikasi dan telepon.

Sifat pelayanan sistem distribusi sangat luas dan kompleks, karena konsumen yang harus dilayani mempunyai lokasi dan karakteristik

yang berbeda. Sistem distribusi harus dapat melayani konsumen yang terkonsentrasi di kota, pinggiran kota dan konsumen di daerah terpencil. Sedangkan dari karakteristiknya, terdapat konsumen perumahan dan konsumen dunia industri. Sistem konstruksi saluran distribusi terdiri dari saluran udara dan saluran bawah tanah. Pemilihan konstruksi tersebut didasarkan pada pertimbangan yaitu alasan teknis berupa persyaratan teknis, alasan ekonomis, alasan estetika dan alasan pelayanan yaitu kontinuitas pelayanan sesuai jenis konsumen. Pada jaringan distribusi primer terdapat 4 jenis sistem konfigurasi jaringan yaitu :

- a. Sistem *Radial*
- b. Sistem Hantaran Penghubung (*Tie Line*)
- c. Sistem *Loop*
- d. Sistem Spindel

3. Gardu Distribusi (Transformator Distribusi)

Gardu distribusi (Trafo distribusi) berfungsi merubah tegangan listrik dari jaringan distribusi primer menjadi tegangan terpakai yang digunakan untuk konsumen dan disebut sebagai jaringan distribusi sekunder.

Kapasitas transformator yang digunakan pada transformator distribusi ini tergantung pada jumlah beban yang akan dilayani dan luas daerah pelayanan beban. Gardu distribusi (trafo distribusi) dapat berupa transformator satu fasa dan juga berupa transformator tiga fasa.

4. Jaringan Distribusi Sekunder

Jaringan distribusi sekunder atau jaringan distribusi tegangan rendah merupakan jaringan tenaga listrik yang langsung berhubungan dengan konsumen. Oleh karena itu besarnya tegangan untuk jaringan distribusi sekunder ini adalah 220 V untuk satu fasa dan 380 untuk 3 fasa.

2.4 Studi Aliran Daya

Daya listrik akan selalu menuju ke beban, sehingga disebut aliran daya atau aliran beban. Studi aliran daya (*load flow*) digunakan untuk menentukan tegangan, arus, daya aktif atau daya reaktif di berbagai macam titik/bus pada jaringan listrik dalam kondisi operasi normal (Stevenson, 1990). Selain dipergunakan untuk perencanaan pengembangan sistem listrik pada masa mendatang, juga dapat digunakan untuk mengevaluasi kondisi sistem kelistrikan yang sudah ada.

Analisis aliran daya merupakan penentuan atau perhitungan tegangan, arus, daya aktif, daya reaktif, faktor daya yang terdapat pada setiap simpul atau bus suatu sistem tenaga listrik. Perhitungan tersebut dilakukan pada kondisi normal, baik yang sedang berjalan saat ini maupun yang diharapkan akan berkembang di masa mendatang. Dengan analisis aliran daya listrik dapat diketahui efek-efek interkoneksi dengan sistem tenaga lain, beban yang baru, sistem pembangkit yang baru, dan saluran yang baru, (Prabowo, 2007 dalam Dhimas, 2014 :16) .

Tujuan studi aliran daya untuk mengetahui besar vektor tegangan pada tiap bus dan besar aliran daya pada tiap cabang suatu jaringan untuk suatu kondisi beban tertentu dalam kondisi normal. Hasil perhitungan dapat digunakan untuk menelaah berbagai persoalan yang berhubungan dengan jaringan tersebut, yaitu meliputi hal-hal yang berhubungan dengan operasi jaringan yaitu: (Saadat, 1999).

- a. Pengaturan tegangan (*voltage regulation*), perbaikan faktor daya (*power factor*) jaringan, kapasitas kawat penghantar, termasuk rugi-rugi daya.
- b. Perluasan atau pengembangan jaringan, yaitu menentukan lokasi yang tepat untuk penambahan bus beban baru dan unit pembangkitan atau gardu induk baru.
- c. Perencanaan jaringan, yaitu kondisi jaringan yang diinginkan pada masa mendatang untuk melayani pertumbuhan beban karena kenaikan terhadap kebutuhan tenaga listrik.

2.4.1 Analisis Aliran Daya

Menurut Saadat (1999), dalam analisis aliran daya terdapat empat buah besaran pada masing-masing bus jaringan yang ditinjau dan memegang peranan yaitu:

- a. Daya aktif P (*active power*) dengan satuan megawatt (MW).
- b. Daya reaktif Q (*reactive power*) dengan satuan (MVAR)
- c. Besar tegangan $|V|$ (*magnitude*) dengan satuan kilovolt (kV)
- d. Sudut fase tegangan (*angle*) dengan satuan radian

Dua di antara empat besaran yang terdapat pada tiap bus tersebut sudah diketahui, sedangkan dua besaran lainnya merupakan yang akan dihitung melalui proses iterasi. (Kundur, 1993)

Untuk melakukan perhitungan aliran daya, diperlukan data-data untuk menganalisisnya. Referensi data yang diperlukan antara lain (Prabowo, 2007 dalam Dhimas, 2014 : 16) :

1. Data Saluran

Data yang diperoleh dari diagram segaris (single line diagram).

2. Data Bus

Data bus yang diperlukan adalah besaran daya, tegangan, daya aktif dan daya reaktif.

3. Data Spesifikasi

Data yang didapat dari rating-rating setiap komponen, type komponen, merk komponen, frekuensi, dan data asli dari setiap komponen.

Pada umumnya, perhitungan aliran daya diasumsikan sistem dalam keadaan seimbang. Data dan informasi yang didapatkan berguna dalam merencanakan perluasan sistem tenaga listrik dan dalam menentukan operasi terbaik untuk sistem jaringan kelistrikan.

2.4.2 Klasifikasi Sistem Aliran Daya

Menurut Sigit (2015 : 40) dalam penelitiannya, klasifikasi sistem aliran daya dibagi dalam beberapa bagian, antara lain :

1. *Representasi* Transformator

Transformator berfungsi untuk menurunkan tegangan primer 20 kV (dari PLN) menjadi tegangan sekunder 380/220 V. Transformator merupakan komponen yang sangat penting dalam sistem tenaga listrik. Komponen ini dapat memindahkan dan mengubah energi listrik dari rangkaian listrik ke rangkaian listrik lainnya melalui suatu gandingan magnet yang berdasarkan prinsip induksi-elektromagnet.

Penggunaan transformator dalam sistem tenaga listrik memungkinkan pemilihan tegangan yang sesuai dan ekonomis untuk tiap-tiap keperluan misalnya kebutuhan akan tegangan tinggi dalam pengiriman daya listrik jarak jauh. Dalam bidang tenaga listrik pemakaian transformator dikelompokkan menjadi :

- a. Transformator daya
- b. Transformator distribusi
- c. Transformator pengukuran (transformator arus dan transformator tegangan)

Transformator direpresentasikan sebagai resistansi R dan reaktansi bocor X , karena R dan X akan mempunyai nilai persatuan yang sama baik pada sisi tegangan rendah maupun sisi tegangan tinggi pada transformator.

2. *Representasi* Generator

Generator listrik adalah sebuah alat yang memproduksi energi listrik dari sumber energi mekanik, biasanya dengan menggunakan

induksi elektromagnetik. Proses ini dikenal sebagai pembangkit listrik. Walau generator dan motor punya banyak kesamaan, tapi motor adalah alat yang mengubah energi listrik menjadi energi mekanik.

Generator mendorong muatan listrik untuk bergerak melalui sebuah sirkuit listrik eksternal, tapi generator tidak menciptakan listrik yang sudah ada di dalam kabel lilitannya. Hal ini bisa dianalogikan dengan sebuah pompa air, yang menciptakan aliran air tapi tidak menciptakan air di dalamnya. Sumber energi mekanik bisa berupa resiprokot maupun turbin mesin uap, air yang jatuh melalui sebuah turbin maupun kincir air, mesin pembakaran dalam, turbin angin, engkol tangan, energi surya atau matahari, udara yang dimampatkan, atau apa pun sumber energi mekanik yang lain.

3. *Representasi Bus* (Busbaar)

Busbar adalah suatu penghantar dengan impedansi rendah dimana beberapa *sirkuit* listrik dapat dihubungkan secara terpisah dengan setiap keluaran tertuju ke dasar kerangka dengan tiga busbar fasa dan satu fasa netral.

Busbar pada dasarnya merupakan ril penghubung dua atau lebih rangkaian listrik. Karena itu busbar dapat disebut ril penghubung rangkaian. Semua generator atau sumber listrik dalam pusat tenaga listrik disalurkan melalui bus atau ke ril pusat listrik (Dhimas , 2014 : 11). Dalam sistem tenaga listrik terdapat jenis-jenis bus yaitu :

- a) Bus berayun (*swing bus*, yang sering juga disebut *floating bus*, *slack bus* atau bus referensi, dipilih di antara bus generator atau penyedia daya yang mempunyai kapasitas tertinggi di antara yang terpasang dalam jaringan yang ditinjau. Bus ini mempunyai besar tegangan dan nilai sudut fasa tertentu. Bus berayun ini harus mampu membangkitkan daya aktif dan daya reaktif yang dibutuhkan untuk melayani bus beban dan mengimbangi rugi daya pada saluran.
- b) Bus kontrol tegangan (*voltage controlled*) atau bus generator, yaitu bus yang mempunyai nilai tegangan dan daya reaktif tertentu. Tegangan pada bus ini dapat dikendalikan dengan mengatur daya reaktif yang disuplai atau diserap bus. Daya reaktif ini dispesifikasi dalam jangkauan batas minimum dan maksimum tertentu. Daya aktif dapat diatur untuk menjaga tegangan tertentu kecuali bila batas daya reaktif yang dispesifikasikan terlampaui. Jika batas ini terlampaui, maka daya reaktif ditetapkan pada batas tersebut dan tegangan akan diberikan pada nilai yang diperlukan untuk menyelesaikan persamaan aliran daya.
- c) Bus beban (*load bus*), yaitu bus yang mempunyai nilai daya aktif dan daya reaktif tertentu yang diperoleh berdasarkan pengukuran pada saat tertentu. Nilai tegangan bus beban harus dicari melalui proses iterasi sampai tercapai nilai tertentu yang konvergen dengan toleransi ketelitian yang diinginkan.

4. Representasi Kapasitor Bank

Kapasitor bank berfungsi sebagai salah satu alternatif dalam peralatan listrik terhadap koreksi faktor daya (*power factor*). Kapasitor akan memberikan daya reaktif lebih tinggi jika bekerja pada tegangan yang lebih tinggi. Namun walau bekerja pada tegangan yang lebih tinggi dari pada tegangan jaringan standar, kapasitor akan bekerja secara efektif dan tahan lama.

5. Representasi Beban

Jenis beban terbagi menjadi dua jenis yaitu :

a. *Static Load* (Beban Statis)

Beban statis dalam pemakaiannya selalu stabil dan tidak membutuhkan daya yang besar saat awalan atau mulai mengoperasikannya. contoh dari beban statis adalah beban penerangan atau lampu.

b. *Dinamic Load* (Beban Dinamis)

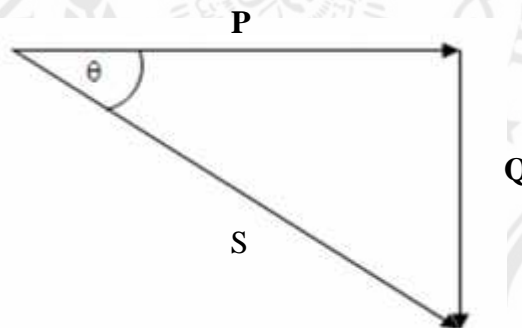
Dinamic load adalah beban yang membutuhkan daya yang besar dalam pengoperasiannya. Biasanya beban ini merupakan beban motor (induksi, sinkron, atau serempak). Motor induksi merupakan motor arus bolak balik (AC) yang paling banyak digunakan.

6. Representasi Sistem Proteksi

Representasi sistem proteksi adalah sebagai pengaman baik untuk mengamankan peralatan listrik maupun pengguna peralatan listrik (manusia). Sistem pengaman diharapkan dapat mencegah masalah-masalah yang disebabkan akibat hubung singkat (*short circuit*) dan kelebihan beban (*over load*) dalam aliran daya listrik.

2.5 Konsep Dasar Aliran Daya

Dalam persamaan maupun perhitungan daya, hal pokok yang harus dipahami adalah dengan memahami konsep segitiga daya. Berikut ini akan dijelaskan tentang segitiga daya menggunakan gambar disertai penjelasan dan perhitungannya (Stevenson, 1990 dalam Dhimas, 2014 : 17).



Gambar 2.2 Segitiga daya

Keterangan :

- P : Daya Aktif
- Q : Daya Reaktif
- S : Daya Semu
- Cos : Faktor Daya

Dalam sistem tenaga listrik dikenal tiga jenis daya, yaitu daya aktif atau *real power* (P), daya reaktif atau *reactive power* (Q), dan daya nyata atau *apparent power* (S).

Menurut Sigit (2015 : 19), daya aktif (P) adalah daya listrik yang dibangkitkan di sisi keluaran generator, kemudian dapat dimanfaatkan oleh konsumen, dapat dikonversi ke bentuk energi lainnya seperti energi gerak pada motor, bisa juga menjadi energi panas pada heater ataupun dapat diubah ke bentuk energi listrik lainnya. Perlu diingat bahwa daya ini memiliki satuan watt (W), kilowatt (kW) atau tenaga kuda (HP).

Sedangkan daya reaktif (Q) adalah suatu besaran yang digunakan untuk menggambarkan adanya fluktuasi daya pada saluran transmisi dan distribusi akibat dibangkitkannya medan / daya magnetik atau beban yang bersifat induktif (seperti : motor listrik, trafo, dan las listrik). Walaupun namanya adalah daya, namun daya reaktif ini tidak nyata dan tidak bisa dimanfaatkan. Daya ini memiliki satuan volt-ampere-reaktif (VAR) atau kilovar ($kVAR$).

Daya semu (S) merupakan jumlah daya total yang terdiri dari daya aktif (P) dan daya reaktif (Q).

Konsep dasar aliran daya listrik sangat penting untuk membantu perhitungan dalam analisis aliran daya listrik. Dalam suatu analisis sistem tenaga listrik khususnya pada analisis aliran daya selalu mengacu pada konsep-konsep dasar aliran daya sebagai berikut :

1. Daya Listrik Pada Rangkaian Satu Fasa

Daya yang diserap oleh suatu beban pada setiap saat sama dengan jatuh tegangan (*voltage drop*) pada beban tersebut dalam volt dikalikan dengan arus yang mengalir lewat beban dalam ampere, jika terminal-terminal beban digambarkan sebagai a dan n serta jika tegangan dan arus dinyatakan dengan (Stevenson, 1990 : 14),

$$V_{an} = V_{\max} \cos t \text{ dan } I_{an} = I_{\max} \cos (t - \theta) \quad (2.1)$$

θ = positif (+), untuk arus lagging

θ = negatif (-), untuk arus leading

Maka daya sesaat (S) :

$$S = V_{\max} I_{\max} \cos t \cos (t - \theta) \quad (2.2)$$

$$S = V \times I \quad (2.3)$$

$$= |V||I| \cos \theta (1 + \cos 2 t) + |V||I| \sin \theta \sin 2 t \quad (2.4)$$

Keterangan :

$|V|$ dan $|I|$ adalah harga efektif dari tegangan dan arus

$|V||I| \cos \theta (1 + \sin t)$ selalu positif, dengan harga rata-rata :

$$P = |V||I| \cos \theta$$

$|V||I| \sin \theta \sin 2 t$ mempunyai harga positif dan negatif dengan harga rata-rata nol

$$Q = |V||I| \sin \theta$$

Keterangan :

P : disebut daya nyata atau aktif (Watt)

Cos : disebut faktor kerja

Q : disebut daya reaktif

“positif” untuk beban induktif

“negatif” untuk beban kapasitif

2. Daya Listrik Pada Rangkaian Tiga Fasa

Daya yang diberikan oleh generator tiga fasa atau yang diserap oleh beban tiga fasa adalah jumlah daya dari tiap-tiap fasa. Pada sistem tiga fasa seimbang berlaku rumus-rumus (Sigit, 2015 : 15) :

$$P = 3V_p I_p \cos \theta_p \quad (2.5)$$

$$Q = 3V_p I_p \sin \theta_p \quad (2.6)$$

Keterangan :

θ_p : sudut antara arus fasa (*lagging*) dan tegangan fasa Hubungan yang terjadi pada rangkaian sistem tiga fasa umumnya ada dua hubungan yaitu hubungan bintang (Y) dan segitiga (Δ).

Hubungan bintang (Y) :

$$V_p = \frac{V_1}{\sqrt{3}}; I_p = I_1$$

Hubungan segitiga (Δ) :

$$V_p = V_1; I_p = \frac{I_1}{\sqrt{3}}$$

Kemudian dimasukkan ke persamaan di atas menjadi :

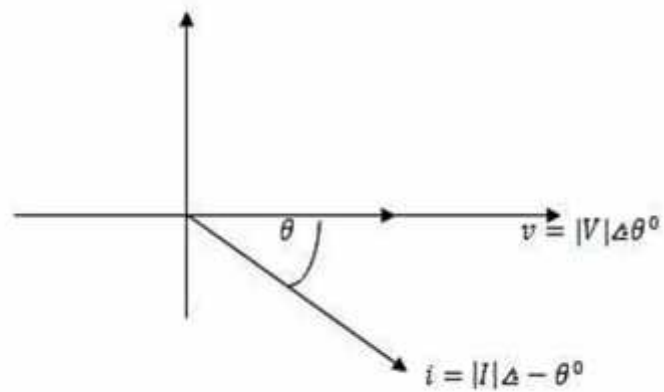
$$P = \sqrt{3} V_1 I_1 \cos \theta_p$$

$$Q = \sqrt{3} V_1 I_1 \sin \theta_p$$

Sehingga :

$$S = \sqrt{P^2 + Q^2} = \sqrt{3} V_1 I_1$$

3. Bentuk Kompleks dari Besaran Tegangan dan Arus Listrik

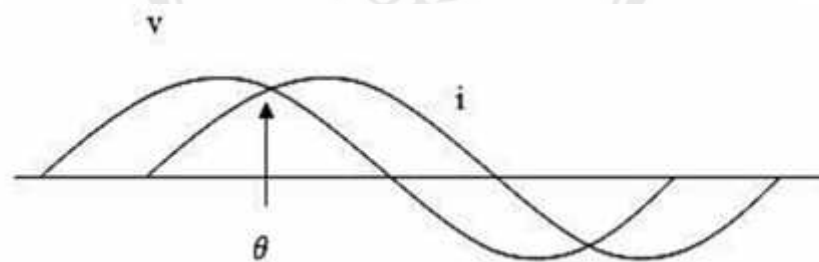


Gambar 2.3 Bentuk kompleks dari besaran tegangan dan arus listrik

V dan I mempunyai bentuk gelombang yang sama (*sinus*) dengan *frekuensi* yang sama pula. Tetapi yang membedakan hanya *magnitude* (harge efektif) dan satu fasanya. Dalam bentuk kompleks besaran dari arus dan tegangan adalah :

$$v = |V|\angle\theta^0 \text{ (digunakan sebagai referensi)}$$

$$i = |I|\angle-\theta^0 \text{ (lagging)}$$



Gambar 2.4 Diagram fasor antara tegangan dan arus

4. Faktor Daya

Daya rata-rata bukan lagi fungsi rms (*root mean square*) dari arus dan tegangan saja, tetapi ada unsur perbedaan sudut fasa arus dan tegangan dari persamaan sephasa dan $\varphi = 0^0$, menurut Zuhail, 1998 dalam Dhimas (2014 : 31), maka persamaan daya menjadi :

$$P = V.I \cos\varphi \quad (2.7)$$

Untuk :

$$\varphi = 60^0 ; \text{ maka } P = V.I \cos (60^0) = 0,3 V.I$$

$$\varphi = 90^0 ; \text{ maka } P = V.I \cos (90^0) = 0$$

Arus yang mengalir pada sebuah tahanan akan menimbulkan tegangan pada tahanan tersebut, yaitu sebesar :

$$P = V_r . I_m \cos \quad (2.8)$$

Keterangan :

P : daya aktif (Watt)

V_r : tegangan (Volt)

I_m : arus maksimal (Ampere)

cos : faktor daya

Karena tidak ada sudut fasa antara arus dengan tegangan pada tahanan, maka sudut $= 0^0$. Sehingga :

$$P = V.I \quad (2.9)$$

Tegangan dikalikan dengan arus disebut daya semu.

Daya rata-rata dibagi daya semu disebut dengan faktor daya.

Untuk arus dan tegangan sinusoid, faktor daya dapat dihitung :

$$\text{Faktor daya} = \frac{P}{V.I} = \frac{V.I.\cos \varphi}{V.I} = \cos \varphi \quad (2.10)$$

adalah sudut faktor daya, sudut ini menentukan kondisi terdahulu atau tertinggal tegangan terhadap arus. Bila sebuah beban diberikan tegangan, impedansi dari beban tersebut maka dapat menentukan besar arus dan sudut fasa yang mengalir pada beban tersebut. Faktor daya merupakan petunjuk yang menyatakan sifat suatu beban.

Perbandingan antara daya aktif (P) dan daya nyata (S) inilah yang dikenal dengan istilah faktor daya atau *power factor* (PF). Apabila dilihat pada segitiga daya, perbandingan daya aktif (P) dan daya nyata (S) merupakan nilai $\cos \varphi$. Oleh karena hal ini, istilah faktor daya juga sering dikenal dengan sebutan nilai $\cos \varphi$.

Beban yang sering digunakan pada konsumen pada instansi pendidikan kebanyakan bersifat induktif. Peningkatan beban yang bersifat induktif ini pada sistem tenaga listrik dapat menurunkan nilai faktor daya dalam proses pengiriman daya. Penurunan faktor daya ini dapat menimbulkan berbagai kerugian, antara lain (Sigit, 2015 : 18) :

- a. Memperbesar kebutuhan kVA
- b. Penurunan efisiensi penyaluran daya
- c. Memperbesar rugi-rugi panas kawat dan peralatan
- d. Mutu listrik menjadi rendah karena adanya drop tegangan

5. *Losses* (Rugi Daya)

Konduktor pada bagian penyaluran energi listrik mempunyai resistansi terhadap arus listrik, jadi ketika sistem beroperasi pada bagian penyaluran ini akan terjadi rugi daya yang berubah menjadi energi panas. Rugi daya pada gardu induk relatif kecil, sehingga rugi daya dalam sistem tenaga listrik dapat dianggap terdiri dari rugi daya pada jaringan transmisi dan jaringan distribusi. Jika energi listrik disalurkan melalui jaringan arus bolak-balik tiga fasa, maka rugi daya pada jaringan tersebut adalah :

$$P_1 = 3.I^2.R \text{ (watt)} \quad (2.11)$$

Keterangan :

I : arus konduktor (Ampere)

R : resistansi konduktor ()

2.5.1 Metode Analisis Aliran Daya

Diagram satu garis sistem tenaga listrik dapat dinyatakan dengan diagram admitansi yang nilainya berbanding terbalik dengan impedansi. Mengacu pada diagram admitansi yang telah merepresentasikan kondisi jaringan, kemudian dibentuk persamaan yang menggambarkan diagram tersebut menggunakan hukum kirchoff arus. Persamaan dirumuskan dalam bentuk persamaan matrik untuk menggambarkan kondisi sistem yang kompleks. Tegangan dan arus dihubungkan melalui persamaan:

$$[I] = [Y] [V] \quad (2.12)$$

Keterangan :

[I] : Vektor arus yang mengalir pada jaringan.

[V] : Vektor tegangan yang mengalir pada jaringan.

[Y] : Matriks admitansi jaringan

Persamaan (2.12) adalah sebuah persamaan aljabar linier dengan koefisien kompleks jika [I] atau [V] telah diketahui.

Secara umum matriks admitansi dibentuk sesuai persamaan (2.12) adalah sebagai berikut:

$$y = \begin{bmatrix} y_{11} & y_{12} & y_{13} & \dots & y_{1n} \\ y_{21} & y_{22} & y_{23} & \dots & y_{2n} \\ \vdots & & \ddots & & \vdots \\ y_{n1} & y_{n2} & y_{n3} & \dots & y_{nn} \end{bmatrix} \quad (2.13)$$

Keterangan :

y_{ii} = jumlah seluruh admitansi yang terhubung pada bus i ($i=1, 2, 3, \dots, n$)

y_{jk} = negative dari admitansi yang menghubungkan antara bus j dan bus k

karena $y_{jk} = y_{kj}$ maka akan terbentuk matriks admitansi yang simetris.

Perhitungan nilai arus bus (I bus) menggunakan persamaan:

$$I_i = \frac{(P_i + jQ_i)^*}{V_i^*} \quad (2.14)$$

Tanda * menunjukkan konjugat dari sebuah bilangan kompleks.

$$\left[\frac{P - jQ}{V^*} \right] = [Y] \cdot [V] \quad (2.15)$$

Persamaan (2.15) dapat dirinci untuk persamaan tiap bus menjadi:

$$P_i + jQ_i = V_i \left[\sum_{k=1}^n Y_{ik} V_k \right]^* \quad (2.16)$$

Atau

$$P_i + jQ_i = |V_i|^2 Y_{ii}^* + \sum_{\substack{k=1 \\ k \neq i}}^n Y_{ik}^* V_i V_k^* \quad (2.17)$$

Persamaan (2.15), (2.16), dan (2.17) bersifat nonlinier, maka tidak dapat dipecahkan secara langsung, sehingga perhitungan aliran beban dilakukan dengan cara iterasi. Metode komputasi yang sering dipakai untuk menyelesaikan perhitungan aliran beban antara lain metode Newton-Raphson, Fast Decoupled, dan Gauss-Seidel.

Metode Newton-Raphson secara matematis lebih baik dibandingkan dengan metode Gauss-Seidel. Untuk sistem yang besar, metode Newton-Raphson jauh lebih efisien dan lebih praktis. Banyaknya iterasi yang diperlukan dengan metode Newton-Raphson tidak tergantung pada banyaknya rel. Waktu yang diperlukan untuk metode Gauss-Seidel (admitansi rel) meningkat hampir sebanding dengan banyaknya rel. Waktu yang dibutuhkan untuk menghitung matriks jacobian cukup lama, namun untuk menghitung tiap iterasinya lebih singkat.

2.5.2 Perhitungan Aliran Daya Menggunakan Metode Newton-Raphson

Pada perhitungan dengan menggunakan metode Newton Raphson, iterasi yang diperlukan untuk memperoleh pemecahan ditentukan berdasarkan ukuran sistem. Persamaan aliran daya merupakan persamaan *aljabar non-linear*,

sehingga tidak mempunyai solusi *eksak*. Persamaan tersebut dapat diselesaikan dengan metode iterasi dari beberapa metode *numerik*. Harga *konvergensi* pada proses iterasi ditentukan oleh besarnya indeks presisi antara 0.01 hingga 0.00001 atau sesuai dengan yang dikehendaki. Jumlah iterasi menentukan besarnya presisi yang dikehendaki, semakin presisi semakin banyak jumlah iterasi yang harus dilakukan.

Metode Newton-Raphson pada dasarnya merupakan metode Gauss-Siedel yang diperluas dan disempurnakan. Metode Newton-Raphson adalah uraian dari deret Taylor untuk satu fungsi dengan dua variabel atau lebih untuk memecahkan persoalan aliran daya yaitu dengan mencari daya aktif, daya reaktif, tegangan dan faktor daya.

Metode Newton-Raphson mempunyai perhitungan lebih baik dari pada metoda Gauss-Siedel bila untuk sistem tenaga yang besar karena lebih efisien dan praktis. Jumlah iterasi (perhitungan) yang dibutuhkan untuk memperoleh pemecahan ditentukan berdasarkan ukuran sistem.

Waktu komputasi pada perhitungan dengan metoda Newton-Raphson jauh lebih cepat dibandingkan pada penggunaan metode Algoritma-Genetika (Sulasno, 1993 dalam Dhimas, 2015 : 24). Sehingga banyak dipakai menyelesaikan analisis aliran daya.

Prosedur penyelesaian analisa aliran daya menggunakan metode Newton Raphson yaitu:

1. Penyelesaian analisa aliran daya dimulai dengan menyusun matriks admitansi dari data impedansi yang sudah diketahui. Matriks admitansi ini dibangun dalam bentuk polar untuk memudahkan perhitungan selanjutnya.
2. Identifikasi nilai-nilai yang diketahui pada tiap bus. Pada *swing bus* nilai magnitude dan sudut tegangan telah diketahui atau ditentukan. Nilai tegangan ini yang menjadi patokan dan bersifat tetap. Pada bus PV dan bus beban nilai magnitude tegangan yang belum diketahui ditentukan nilai awlanya sebesar 1,00 pu ($|V|=1,00$), sedangkan sudut tegangan yang belum diketahui ditentukan sebesar 0° ($\delta=0$).
3. Menentukan daya aktif dan daya reaktif *schedule* yaitu daya yang telah diketahui pada bus beban dan bus PV yang nantinya akan menjadi patokan untuk menghitung P dan Q .
4. Proses analisa dilanjutkan dengan menghitung daya ditiap bus dengan persamaan (2.17) menggunakan nilai tegangan perkiraan awal.

Persamaan (2.17) dapat diuraikan menjadi

$$P_i = \sum_{j=1}^n |V_i||V_j||Y_{ij}|\cos(\theta_{ij}-\delta_i+\delta_j) \quad (2.18)$$

$$Q_i = \sum_{j=1}^n |V_i||V_j||Y_{ij}|\sin(\theta_{ij}-\delta_i+\delta_j) \quad (2.19)$$

5. Setelah didapatkan nilai daya aktif dan daya reaktif yang baru dari hasil perhitungan persamaan (2.18) dan (2.19), dihitung nilai P dan Q dengan menggunakan persamaan

$$P^{(k)} = P_{sch} - P^{(k)} \quad (2.20)$$

$$Q^{(k)} = Q_{sch} - Q^{(k)} \quad (2.21)$$

6. Saat $P^{(k)}$ dan $Q^{(k)}$ tidak melebihi dari batas toleransi yang ditentukan, maka nilai tegangan yang telah ditentukan diawal sudah mendekati nilai yang sesungguhnya dan dapat digunakan untuk mencari nilai dari parameter-parameter yang belum diketahui lainnya. Namun apabila $P^{(k)}$ dan $Q^{(k)}$ masih diatas batas toleransi, maka perhitungan akan dilanjutkan dengan membentuk matriks *Jacobian*.
7. Matriks *Jacobian* merupakan ciri khas dari analisa perhitungan aliran daya menggunakan metode Newton Raphson. Matriks *Jacobian* merupakan matriks yang berisi nilai *parsial deverbatim* dari daya aktif dan reaktif terhadap nilai magnitude dan sudut tegangan pada tiap bus.

Perubahan kecil nilai tegangan pada persamaan 2.17 dapat dikarenakan perubahan kecil daya pada bus. Sebuah persamaan linier dapat difungsikan untuk penyelesaian perubahan tersebut Pada sistem yang terdiri n bus, persamaan untuk penyelesaian aliran daya sebanyak (n-1), yang dalam matriks dapat dinyatakan dalam bentuk (Stevenson, 1990 : 187) :

$$\begin{bmatrix} \Delta P_1 \\ \Delta Q_{n-1} \\ \dots \\ \Delta \Phi_1 \\ \Delta \Phi_{n-1} \end{bmatrix} = \begin{bmatrix} \frac{\partial P_1}{\partial \Phi_1} & \frac{\partial P_1}{\partial \Phi_{n-1}} & \frac{\partial P_1}{\partial V_1} & \frac{\partial P_1}{\partial V_{n-1}} \\ \frac{\partial P_{n-1}}{\partial \Phi_1} & \frac{\partial P_{n-1}}{\partial \Phi_{n-1}} & \frac{\partial P_{n-1}}{\partial V_1} & \frac{\partial P_{n-1}}{\partial V_{n-1}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial Q_1}{\partial \Phi_1} & \frac{\partial Q_1}{\partial \Phi_{n-1}} & \frac{\partial Q_1}{\partial V_1} & \frac{\partial Q_1}{\partial V_{n-1}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial Q_{n-1}}{\partial \Phi_1} & \frac{\partial Q_{n-1}}{\partial \Phi_{n-1}} & \frac{\partial Q_{n-1}}{\partial V_1} & \frac{\partial Q_{n-1}}{\partial V_{n-1}} \\ \frac{\partial \Phi_1}{\partial \Phi_1} & \frac{\partial \Phi_1}{\partial \Phi_{n-1}} & \frac{\partial \Phi_1}{\partial V_1} & \frac{\partial \Phi_1}{\partial V_{n-1}} \end{bmatrix} \begin{bmatrix} \Delta \Phi_1 \\ \Delta \Phi_{n-1} \\ \dots \\ \Delta V_1 \\ \Delta V_{n-1} \end{bmatrix} \quad (2.22)$$

Persamaan di atas dapat disederhanakan menjadi :

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} \Delta \Phi \\ \Delta V \end{bmatrix} \quad (2.23)$$

Atau

$$\begin{bmatrix} \Delta \Phi \\ \Delta V \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \quad (2.24)$$

Keterangan :

P dan Q : Selisih daya aktif dan daya reaktif antara nilai yang diketahui dan nilai yang dihitung.

V dan : Selisih tegangan bus dan sudut fasa.

J1, J2, J3, dan J4 disebut sub matriks Jacobian dari matriks Jacobian J.

2.6 Electric Transient and Analysis Program (ETAP)

ETAP adalah suatu *software* analisis yang *comprehensive* untuk mendesain dan mensimulasikan suatu sistem rangkaian tenaga. Analisis yang ditawarkan oleh ETAP yang digunakan oleh penulis adalah drop tegangan, dan losses jaringan.

ETAP juga bisa memberikan *warning* terhadap bus – bus yang *under voltage* dan *over voltage* sehingga pengguna bisa mengetahui bus mana yang tidak beroperasi optimal. Untuk menganalisa suatu rangkaian diperlukan data rangkaian yang lengkap dan akurat sehingga hasil perhitungan ETAP bisa di pertanggung jawabkan.

ETAP mengintegrasikan data–data rangkaian tenaga listrik seperti kapasitas pembangkit, panjang jaringan, resistansi jaringan per km, kapasitas busbar, rating trafo, impedansi urutan nol, positif, dan negatif suatu peralatan listrik seperti trafo, generator dan penghantar.

ETAP memungkinkan anda untuk bekerja secara langsung dengan diagram satu garis grafis dan sistem kabel bawah tanah *raceway*. Program ini telah dirancang berdasarkan tiga konsep kunci:

1. *Virtual reality operasi*

Program operasi menyerupai sistem operasi listrik nyata sedekat mungkin. Sebagai contoh, ketika membuka atau menutup sebuah pemutus sirkit, tempat elemen dari layanan, atau mengubah status operasi dari motor, unsur *de-energized* dan sub-sistem yang ditunjukkan pada diagram satu garis berwarna abu-abu. ETAP menggabungkan konsep-konsep baru untuk menentukan perangkat pelindung koordinasi langsung dari diagram satu garis.

2. Integrasi total data

ETAP menggabungkan listrik, atribut logis, mekanik, dan fisik dari elemen sistem dalam *database* yang sama. Misalnya, kabel tidak hanya berisi data yang mewakili sifat listrik dan dimensi fisik, tapi juga informasi yang menunjukkan *raceways* melalui yang disalurkan. Dengan demikian, data untuk kabel tunggal dapat digunakan untuk analisis aliran daya atau sirkuit pendek (yang membutuhkan listrik dan parameter koneksi) serta kabel *ampacity derating* perhitungan (yang memerlukan rute fisik data). Integrasi ini menyediakan konsistensi data di seluruh sistem dan menghilangkan *multiple entry* data untuk unsur yang sama.

3. Kesederhanaan di data entri

ETAP melacak data rinci untuk setiap alat listrik. Editor data dapat mempercepat proses entri data dengan meminta data minimum untuk studi tertentu. Untuk mencapai hal ini, kita telah terstruktur editor properti dengan cara yang paling logis untuk memasukkan data untuk berbagai jenis analisis atau desain. ETAP diagram satu garis mendukung sejumlah fitur untuk membantu dalam membangun jaringan dari berbagai kompleksitas. Misalnya, setiap elemen secara individu dapat memiliki berbagai orientasi, ukuran, dan simbol-simbol *display* (IEC atau ANSI). Diagram satu garis juga memungkinkan untuk menempatkan beberapa alat pelindung antara sirkuit cabang dan bus.

ETAP menyediakan berbagai pilihan untuk menampilkan atau melihat sistem listrik. Pandangan ini disebut presentasi. Lokasi, ukuran, orientasi,

dan simbol setiap unsur dapat berbeda di masing-masing presentasi. Selain itu, alat pelindung dan *relay* dapat ditampilkan atau disembunyikan untuk presentasi tertentu. Misalnya, satu presentasi dapat menggunakan tampilan *relay* di mana semua perangkat pelindung ditampilkan. Presentasi lain mungkin menunjukkan diagram satu garis dengan beberapa pemutus sirkuit ditampilkan dan sisanya tersembunyi (tata letak paling cocok untuk hasil aliran daya).

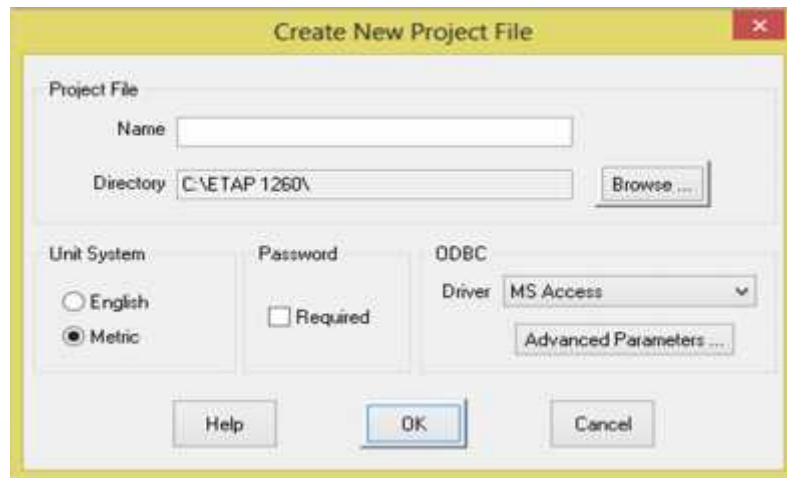
Beberapa hal yang perlu diperhatikan dalam bekerja dengan ETAP adalah:

1. ***One Line Diagram***, menunjukkan hubungan antar komponen/peralatan listrik sehingga membentuk suatu sistem kelistrikan.
2. ***Library***, informasi mengenai semua peralatan yang akan dipakai dalam sistem kelistrikan. Data elektris maupun mekanis dari peralatan yang detail dapat mempermudah dan memperbaiki hasil simulasi/analisa.
3. **Standar yang dipakai**, biasanya mengacu pada standar IEC atau ANSII, frekuensi sistem dan metode-metode yang dipakai.
4. ***Study Case***, berisikan parameter-parameter yang berhubungan dengan metode studi yang akan dilakukan dan format hasil analisa.

2.6.1 Membuat proyek baru

Berikut ini merupakan langkah-langkah untuk membuat proyek baru:

1. Klik tombol *New* atau klik menu *File* lalu akan muncul kotak dialog sebagai berikut:



Gambar 2.5 *create new project file*

2. Lalu ketik nama *file project*, misalnya: Pelatihan. Lalu klik OK.
3. Akan muncul kotak dialog *User Information* yang berisi data pengguna *software*. Isi nama dan deskripsikan proyek anda. Lalu klik OK.
4. Anda telah membuat file proyek baru dan siap untuk menggambar *one line diagram*.

2.6.2 Menggambar *single line diagram*

Menggambar *single line diagram* dilakukan dengan cara memilih simbol peralatan listrik pada menu bar disebelah kanan layar. Klik pada simbol, kemudian arahkan kursor pada media gambar. Untuk menempatkan peralatan pada media gambar, klik kursor pada media gambar.

Untuk mempercepat proses penyusunan *single line diagram*, semua komponen dapat diletakkan secara langsung pada media gambar. Untuk mengetahui kontinuitas antar komponen dapat di cek dengan *Continuity Check* pada menu bar utama.

Pemakaian Continuty Check dapat diketahui hasilnya dengan melihat warna *komponen/branch*. Warna hitam berarti telah terhubung, warna abu-abu berarti belum terhubung.

2.7 MATLAB (*Matrix Laboratory*)

Matlab singkatan dari *MATrix LABoratory*, merupakan bahasa pemrograman yang dikembangkan oleh *The Mathwork. Inc*. Program Matlab merupakan program interaktif yang digunakan untuk melakukan perhitungan numerik keteknikan, komputasi simbolik, visualisasi, grafis, analisis data matematis, statistika, simulasi dan pemodelan dengan dasar matriks dan bidang ilmu pengetahuan dan teknik rekayasa.

Matlab menyediakan sejumlah fasilitas, diantaranya:

1. Manipulasi mudah untuk membentuk matriks,
2. Sejumlah rutin yang mudah diakses dan dimodifikasi,
3. Fasilitas canggih untuk mendapatkan gambar berdimensi dua atau tiga,
4. Kemudahan untuk menuliskan program yang singkat, sederhana, dan dapat dikembangkan sesuai kebutuhan.

Karakteristik Matlab disebutkan dalam buku *Analisis dan Design Sistem Kontrol dengan Matlab* oleh Thomas Wahyu Dwi Hartanto dan Y. Wahyu Agung Prasetyo (2009) sebagai berikut:

1. Bahasa pemrogramannya didasarkan pada matriks (baris dan kolom),
2. Program Matlab lebih lambat karena bahasanya langsung diartikan,

3. *Automatic memory management*, kita tidak harus mendeklarasikan *arrays* terlebih dahulu,
4. Tersusun rapi,
5. Memiliki waktu pengembangan program yang lebih cepat dibandingkan bahasa C dan *Fortran*,
6. Dapat diubah ke bahasa C lewat Matlab Compiler untuk efisiensi yang lebih baik, dan
7. Tersedia banyak *toolbox* untuk aplikasi-aplikasi khusus.

Matlab merupakan sistem dengan elemen dasar basis data *array* yang dimensinya tidak perlu dinyatakan secara khusus. Hal ini memungkinkan untuk memecahkan banyak masalah perhitungan teknik, khususnya melibatkan matriks dan vektor.

Sistem pada Matlab terdiri atas 5 sistem pokok yaitu:

1. Lingkungan kerja meliputi *command window*, *command history*, *editor and debugger*, dan fasilitas *help*, *workspace*, dan *search path*,
2. *Matlab Mathematical Function Library* atau fungsi matematis dalam Matlab mulai dari fungsi-fungsi sederhana aritmatika, logika, trigonometri, bilangan kompleks, dan matriks,
3. *Matlab Language* atau bahasa pemrograman Matlab,
4. Garfik pada Matlab baik dua dimensi maupun tiga dimensi, dan
5. Aplikasi Program *Interface* pada Matlab yang memungkinkan membangun program menggunakan bahasa C dan *Fortran* di Matlab.

Program Matlab memberikan banyak kemudahan dengan adanya perintah dan fungsi-fungsi bawaan dalam listing program Matlab.

Perintah dan fungsi dalam *listing* program Matlab antara lain :

disp	memperlihatkan matriks atau teks
end	mendefinisikan akhir dari sebuah struktur kontrol, tanda batas blok statemen for, while, switch, dan if
fprintf	mencetak informasi terformat
sprintf	menulis data terformat menjadi <i>string</i>
grid	memasukkan grid (garis-garis grid) dalam sebuah gambar
pause	sementara menghentikan sebuah program, untuk melanjutkannya ketik sembarang tombol
plot	membuat sebuah gambar xy linear
text	keterangan teks
title	menambahkan sebuah judul pada sebuah gambar
xlabel	menambahkan label sumbu-x pada sebuah gambar
ylabel	menambahkan label sumbu-y pada sebuah gambar
while	membuat sebuah struktur putaran, mengulang blok statemen dengan jumlah perulangan tidak terbatas
for	loop (pengulangan) sebanyak beberapa kali
sum	menyatakan jumlah elemen

untuk mengetahui perintah-perintah yang ada dalam program Matlab dapat dilihat dengan menggunakan perintah berikut:

```
>> help <instruksi/perintah>.
```

Perintah *help* digunakan untuk menggunakan bantuan yang tersedia pada Matlab. Pada menu *help* juga tersedia demo dan keterangan lengkap tentang Matlab. Dengan mengetahui fungsi masing-masing perintah, maka dengan mudah dapat dibuat deretan program untuk menjalankan perhutingan dengan menggunakan Matlab.

Untuk menyimpan hasil-hasil dari operasi Matlab ke dalam media yang lebih permanen menggunakan perintah:

```
>> save nama_file.mat
```

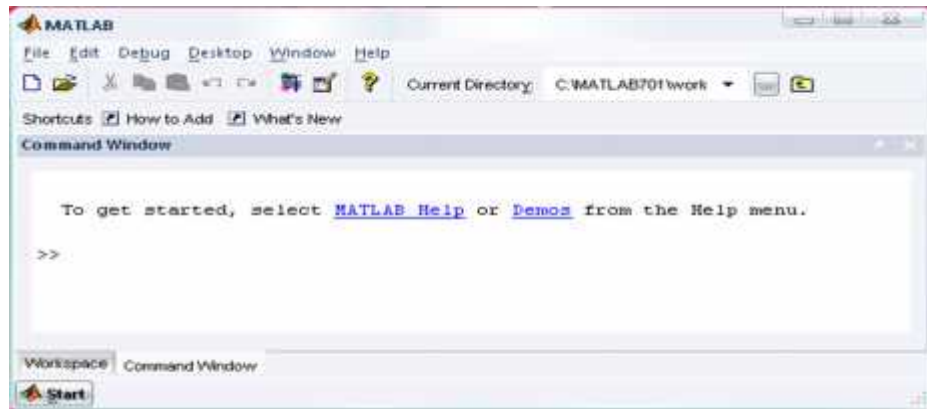
Pada perintah diatas semua atau sebagian variable yang saat itu aktif dalam memory disimpan kedalam sebuah file. Jika hanya menggunakan perintah *save*, maka semua variable kerja yang aktif akan disimpan dalam file dengan ekstensi *matlab.mat*. Nilai-nilai yang tersimpan dalam file tersebut dapat dipanggil kembali dengan perintah *load* seperti berikut:

```
>>load nama_file
```

2.7.1 Lingkungan Kerja Matlab

Matlab menyediakan lingkungan kerja yang mendukung dalam membangun aplikasi. Lingkungan terpadu ini terdiri atas beberapa *form* atau *window* yang memiliki kegunaan masing-masing. Fasilitas-fasilitas pada lingkungan kerja ini terus bertambah sesuai dengan perkembangan dari versi Matlab sendiri. *Form* atau *window* induk yang tersedia di Matlab 7.0.1. terdiri dari *command window*, *command history*, *current directory*, *workspace* dan *M-file*.

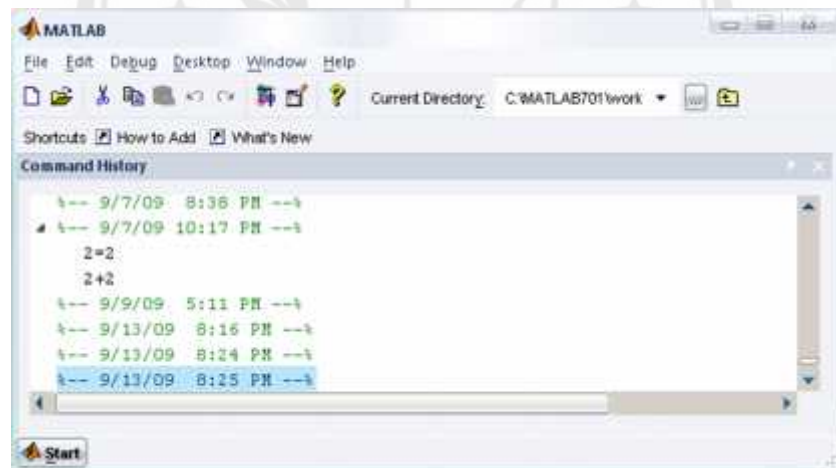
1. *Command Window*



Gambar 2.6 Tampilan *Command Window*

Command window berfungsi sebagai window untuk menjalankan fungsi dan program yang sudah dimasukan. Selain itu *command window* digunakan untuk memasukan variabel-variabel yang dibutuhkan.

2. *Command History*

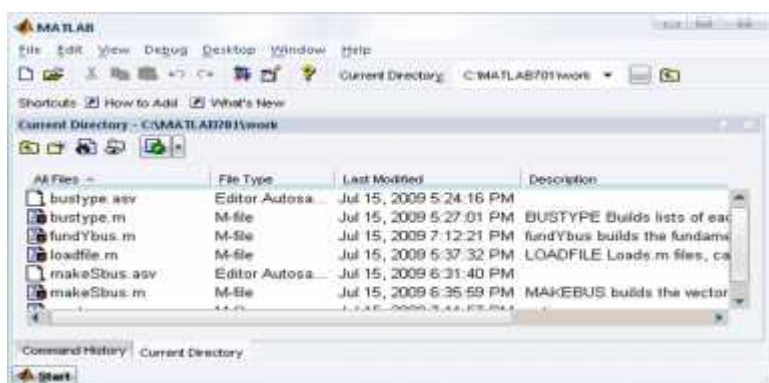


Gambar 2.7 Tampilan *Command History*

Semua *statement* atau perintah yang telah di-*run* pada *command window* akan secara otomatis tersimpan pada *command history*. Untuk

menghapus perintah yang ada di *command history* pilih menu *edit* kemudian pilih *clear command history*.

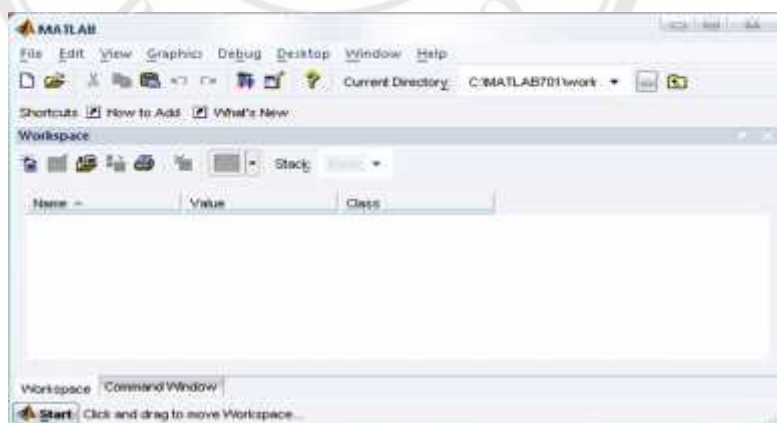
3. *Current Directory*



Gambar 2.8 Tampilan *Current Directory*

Current directory digunakan untuk mencari program yang telah digunakan untuk menjadi acuan.

4. *Workspace*



Gambar 2.9 Tampilan *Workspace*

Workspace menampilkan nama-nama variabel yang sudah digunakan. Nama variabel pada suatu program harus unik. Ada

kemungkinan programmer lupa nama dari variabel yang telah ditulis sebelumnya. Dengan *workspace* maka adanya kesamaan nama variabel dapat teratasi karena semua nama variabel tercantum didalamnya.

5. M-files

File-file yang mengandung berekstensi *.m* disebut m-files. Ada dua jenis m-files, yaitu *script file* dan *function file*. *Script file* tidak mengandung *argument input* atau mengembalikan *argument output*. *Function file* dapat mengandung *argument input* atau mengembalikan *argument output*.

Untuk membuka m-file, klik menu **file** kemudian pilih **new** dan klik **m-file**. Setelah itu akan muncul jendela Matlab *editor/debugger*. jendela Matlab *editor/debugger* digunakan untuk menuliskan dan mengedit program.



Gambar 2.10. Tampilan jendela Matlab editor.

Dalam pembuatan program menggunakan Matlab, terdapat 2 cara yaitu:

1. Menggunakan *Command Window*

Cara ini biasa digunakan untuk membangun program-program sederhana. Jika digunakan untuk membangun program yang panjang dan rumit, akan menemui kesulitan saat mengevaluasi program secara keseluruhan karena biasanya perintah dilakukan tiap baris.

2. Menggunakan *M-file*

Program dapat dibangun pada jendela *M-file* dengan menuliskan *listing* program. Program ditulis sesuai urutan eksekusinya. Untuk menjalankan program bisa dengan cara *setting path* dan memanggil file program dari *Command Window* atau dengan cara *listing* tersebut *dicopy* ke *Command Window* terlebih dahulu kemudian tekan *enter*.

2.7.2 Bilangan dan Operasi Matematis Dasar dalam Matlab

Matlab dirancang untuk melakukan untuk memudahkan perhitungan-perhitungan Terdapat 3 jenis bilangan dalam Matlab yaitu bilangan bulat (*Integer*), bilangan nyata (*Real*), bilangan kompleks. Untuk melengkapi penjelasan bilangan tersebut, Matlab mempunyai 3 variabel *nounnumbers* yaitu:

1. $-\text{Inf}$ (*Negative infinity*),
2. Inf (*Infinity*),
3. Nan (*Not a number*) misalnya $0/0$.

Program Matlab dilengkapi fasilitas untuk melaksanakan operasi aritmatika sederhana, relasi dan operasi logika seperti berikut:

\wedge	Perpangkatan
$*$	Perkalian
/ atau \	Pembagian
$+$	Penjumlahan
$-$	Pengurangan
$<$	Kurang dari
$<=$	Kurang dari sama dengan
$>$	Lebih dari
$>=$	Lebih dari sama dengan
$==$	Sama dengan
$\sim=$	Tak sama dengan
$ $	Operator logika or
$\&$	Operator logika and
\sim	Operator logika not

selain karakter-karakter yang telah disebutkan diatas, masih ada karakter-karakter lain yang mempunyai fungsi khusus seperti:

- () membentuk tulisan bawah garis
- [] membentuk matriks
- , memisahkan tulisan bawah garis atau elemen matriks
- ; memisahkan perintah atau baris matriks
- % menyatakan komentar

- . perkalian skalar
- : membentuk matriks
- ' tranpose matriks

Didalam Matlab juga terdapat fungsi-fungsi matematis dan trigonometri khusus, antara lain:

isreal (x) : untuk mengetahui apakah bilangan x merupakan bilangan real

sin : sinus

sinh : sinus *hyperbolic*

asin : inverse sinus

asinh : inverse sinus *hyperbolic*

cos : cosinus

tan : tangen

sec : secan

csc : cosecan

cot : cotangen

exp : eksponensial

abs : absolute value

angle : sudut dari bilangan kompleks polar

conj : conjugate dari suatu bilangan kompleks

inv : invers suatu matrik

2.7.3 Kontrol Arus Aliran Program

Perhitungan komputasi numerik yang akan dilakukan mengandung kontrol arus aliran, dimana terdapat perhitungan berulang sampai didapatkan angka yang tidak melebihi dari tolrransi yang ditentukan. Matlab menyediakan perintah khusus untuk mengontrol arus perintah, antara lain:

1. Loop *For*

Sintaks dari loop *for* ditunjukkan seperti dibawah ini:

```
for k = array  
commands  
end
```

perintah-perintah antara pernyataan *for* dan *end* dilaksanakan untuk semua nilai yang tersimpan dalam *array*. Loop *for* hanya digunakan jika metode-metode lain tidak dapat digunakan.

2. Loop *While*

Sintaks dari loop *while* ditunjukkan seperti dibawah ini:

```
while expression  
statement  
end
```

Loop ini digunakan jika programmer tidak tahu jumlah pengulangan yang akan dilakukan dengan jelas.

3. Loop *If-else-end*

Sintaks yang paling sederhana untuk loop *if-else-end* adalah sebagai berikut:

if expression

command

end

Konstruksi ini digunakan jika hanya ada satu alternatif. Jika ada dua alternatif maka memerlukan konstruksi sebagai berikut:

if expression

command (evaluated if expression is true)

else

command (evaluated if expression is false)

end

Untuk loop dengan banyak alternatif, konstruksi yang digunakan adalah sebagai berikut:

if expression 1

commands (evaluated if expression 1 is true)

elseif expression 2

commands (evaluated if expression 2 is true)

elseif...

.

.

.

else

commands (executed if all previous expressions evaluated to false)

end

4. Loop *Switch case*

Sintaks dari konstruksi *switch case* adalah sebagai berikut:

switch expression (scalar or string)

case value 1 (executes if expression evaluates to value1)

commands

case value 2 (executes if expression evaluates to value2)

commands

•
•

otherwise

statements

end

Switch membandingkan ungkapan input dengan masing-masing nilai *case*. Sekali kesesuaian didapat, *switch* akan melaksanakan perintah yang berkaitan.