

## BAB II

### LANDASAN TEORI

#### 2.1 Algoritma

##### 2.1.1 Pengertian Algoritma

Istilah algoritma pertama kali diperkenalkan oleh seorang ahli matematika dan astronomi Persia yaitu Abu Ja'far Muhammad Ibnu Musa Al Khawarizmi (diperkirakan wafat pada tahun 850). Beliau merupakan pencetus pertama algoritma karena di dalam buku "Aljabar wal muqabala" Abu Ja'far menjelaskan langkah-langkah dalam menyelesaikan berbagai persoalan aritmatika (aljabar).

Pengertian algoritma (dalam Suarga, 2006) :

1. Suatu urutan dari barisan langkah-langkah atau instruksi guna menyelesaikan suatu masalah.
2. Teknik penyusunan langkah-langkah penyelesaian masalah dalam bentuk kalimat dengan jumlah kata terbatas tetapi tersusun secara logis dan sistematis.
3. Suatu prosedur yang jelas untuk menyelesaikan suatu persoalan dengan menggunakan langkah-langkah tertentu dan terbatas jumlahnya.
4. Susunan langkah yang pasti, yang bila diikuti maka akan mentransformasi data *input* menjadi *output* yang berupa informasi.

### 2.1.2 Ciri Algoritma

Menurut Donald E. Knuth (dalam Suarga, 2006), ada beberapa ciri algoritma, yaitu:

1. Algoritma mempunyai awal dan akhir, suatu algoritma harus berhenti setelah mengerjakan serangkaian tugas. Dengan kata lain, suatu algoritma memiliki langkah yang terbatas.
2. Setiap langkah harus didefinisikan dengan tepat sehingga tidak memiliki arti ganda, tidak membingungkan (*not ambiguous*).
3. Memiliki masukan (*input*) atau kondisi awal.
4. Memiliki keluaran (*output*) atau kondisi akhir.
5. Algoritma harus efektif, bila diikuti benar-benar maka akan menyelesaikan persoalan.

### 2.1.3 Sifat Algoritma

Sifat utama suatu algoritma adalah sebagai berikut:

1. *Definiteness*: Langkah-langkah yang dituliskan dalam algoritma terdefinisi dengan jelas sehingga mudah dilaksanakan oleh pengguna algoritma.
2. *Finiteness*: Suatu algoritma harus memberi kondisi akhir atau *output* setelah sejumlah langkah yang terbatas jumlahnya dilakukan terhadap setiap kondisi awal atau *input* yang diberikan.
3. *Effectiveness*: Setiap langkah dalam algoritma bisa dilaksanakan dalam suatu selang waktu tertentu sehingga pada akhirnya didapatkan solusi sesuai dengan yang diharapkan.

4. *Generality*: Langkah-langkah algoritma berlaku untuk setiap himpunan *input* yang sesuai dengan persoalan yang diberikan, tidak hanya untuk himpunan tertentu.

#### 2.1.4 Komponen Algoritma

Dalam merancang sebuah algoritma ada 3 (tiga) komponen yang harus ada yaitu:

1. Komponen masukan (*input*)

Komponen *input* merupakan kondisi awal suatu algoritma sebelum dilaksanakan yang berupa nilai-nilai peubah yang diambil dari himpunan khusus biasanya terdiri dari pemilihan variabel, jenis variabel, tipe variabel, konstanta dan parameter (dalam fungsi).

2. Komponen keluaran (*output*)

Komponen ini merupakan hasil *output* atau kondisi akhir yang diperoleh dari nilai *input* yang telah di proses melalui algoritma. Karakteristik keluaran yang baik adalah benar (menjawab) permasalahan dan tampilan yang ramah (*friendly*).

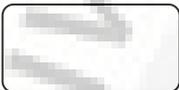
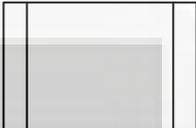
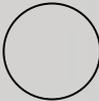
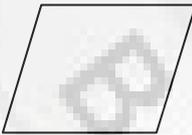
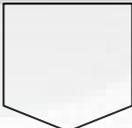
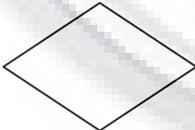
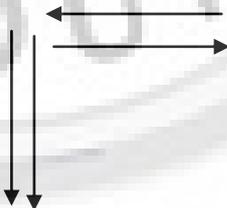
3. Komponen proses (*processing*)

Komponen ini merupakan bagian utama dan terpenting dalam merancang sebuah algoritma. Dalam bagian ini terdapat logika masalah, logika algoritma (sintaksis dan semantik), rumusan, metode (rekursi, perbandingan, penggabungan, pengurangan, dan lain-lain).

## 2.2 Flowchart

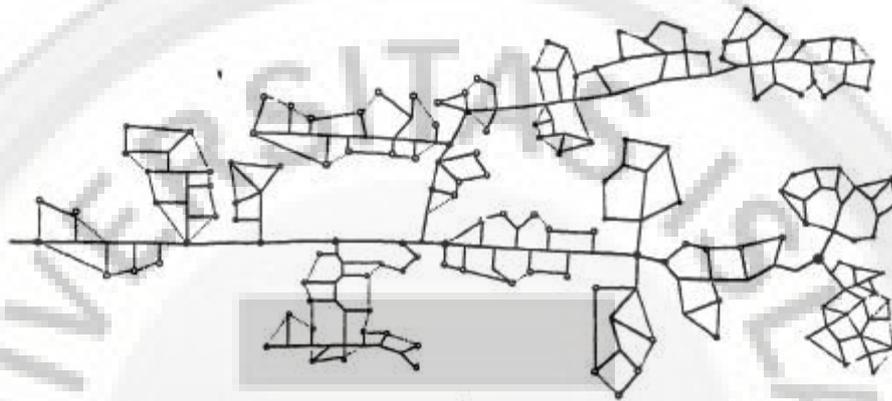
*Flowchart* adalah untaian simbol gambar (*chart*) yang menunjukkan aliran (*flow*) dari proses terhadap data. *Flowcharting* merupakan suatu teknik untuk menyusun rencana program yang telah diperkenalkan dan telah dipergunakan oleh kalangan pemrograman komputer sebelum algoritma menjadi populer.

Tabel 2.1 Daftar Simbol untuk *Flowchart*

Gambar	Keterangan	Gambar	Keterangan
	<i>Terminator</i> : Mulai atau selesai		<i>Predefined-Process</i> : Lambang fungsi atau sub-program
	<i>Process</i> : Menyatakan proses terhadap data		<i>Connector</i> : Penghubung pada halaman yang sama
	<i>Input/ Output</i> : Menerima <i>input</i> atau menampilkan <i>output</i>		<i>Page Connector</i> : Penghubung pada halaman yang berbeda
	<i>Selection</i> : Memilih aliran berdasarkan syarat		<i>Flow Direction</i> : Garis alir
	<i>Predefined-Data</i> : definisi awal dari variabel atau data		

### 2.3 Aliran fluida

Jaringan pipa minyak mentah pada kenyataannya dapat membentuk rangkaian seri dan paralel atau banyak cabang yang disebut sistem jaringan pipa kompleks.



Gambar 2.1 Contoh Jaringan Kompleks Pipa

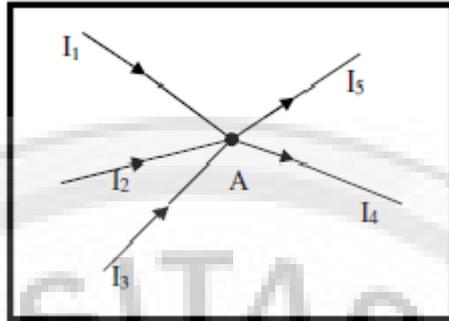
#### Hukum Kekekalan Massa

Hukum kekekalan massa atau dikenal juga sebagai hukum Lomonosov-Lavoisier adalah suatu hukum yang menyatakan massa dari suatu sistem tertutup akan konstan meskipun terjadi berbagai perubahan fisika dan kimia.

#### Hukum Kirchoff I

Hukum Kirchoff I dapat digunakan untuk menganalisa suatu jaringan kompleks. Hukum Kirchoff I menyatakan bahwa jumlah arus yang menuju suatu titik cabang jaringan listrik adalah jumlah arus yang meninggalkan titik cabang tersebut. Dengan kata lain :

$\sum I_{\text{menuju titik cabang}} = \sum I_{\text{meninggalkan titik cabang}}$



Gambar 2.2 Arus-arus pada Titik Cabang

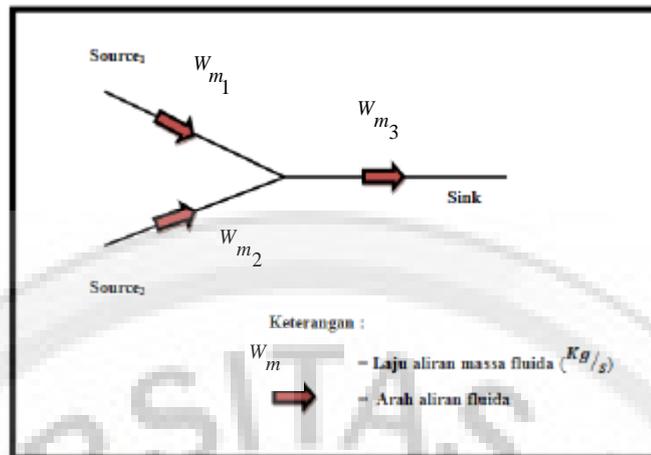
Pada Gambar 2.2 arus  $I_1$ ,  $I_2$ , dan  $I_3$  menuju titik cabang A. Sedangkan arus  $I_4$  dan  $I_5$  meninggalkan titik cabang A. Maka pada titik cabang A berlaku persamaan :

$$I_1 + I_2 + I_3 = I_4 + I_5$$

Berdasarkan Hukum Kekekalan Massa, jaringan pipa tertutup yang terdiri dari beberapa *source* menuju sebuah *sink* yang di dalamnya mengalir tiga buah fasa yaitu minyak, pasir, dan gas tidak mengalami perubahan massa ketika mengalir di sepanjang pipa alir. Menurut Mucharam dan Adewumi (1990), Hukum Kirchoff I dapat diterapkan pada jaringan tertutup pipa minyak. Selisih laju aliran massa fluida yang masuk ( $W_{m_{\text{masuk}}}$ ) dengan laju aliran massa fluida yang keluar ( $W_{m_{\text{keluar}}}$ ) dari setiap cabang harus nol atau dengan kata

lain :

$$\sum W_{m_{\text{masuk}}} = \sum W_{m_{\text{keluar}}} \quad (2.1)$$



**Gambar 2.3** Laju Aliran Massa Fluida pada Titik Cabang

Pada Gambar 2.3, laju aliran massa fluida dari  $Source_1$  dan  $Source_2$  masing-masing adalah  $W_{m_1}$  dan  $W_{m_2}$  menuju  $sink$ , sedangkan besarnya laju aliran massa pada  $sink$  adalah  $W_{m_3}$ . Berdasarkan persamaan (2.1) akan didapat :

$$\sum W_{m_{source}} = \sum W_{m_{sink}}$$

atau dengan kata lain :

$$W_{m_1} + W_{m_2} = W_{m_3}$$

Inisiasi karakteristik fluida dapat diimplementasikan untuk memprediksi properti fluida yang digunakan untuk mengatur persamaan pada sistem jaringan. Menggunakan pengembangan prosedur iterasi, besar tekanan pada setiap *node*, kecepatan fluida dan fraksi volumetrik fluida pada setiap *leg* dapat diprediksi.

Tekanan pada segmen atau *leg j* dapat dihitung melalui persamaan berikut :

$$P = \left[ P_j \right] = \left[ \frac{PN_{j_{\text{masuk}}} + PN_{j_{\text{keluar}}}}{2} \right] \quad (2.2)$$

dengan  $P_j$  adalah tekanan pada leg  $j$ .  $PN_{j_{\text{masuk}}}$  adalah tekanan pada *node* saat fluida masuk leg  $j$  sedangkan  $PN_{j_{\text{keluar}}}$  adalah tekanan pada *node* saat fluida keluar leg  $j$ ,  $j = 1, 2, 3, \dots, m$ .

Besarnya tekanan suatu aliran fluida tiga fasa pada jaringan kompleks pipa minyak mentah dapat mempengaruhi besarnya properti fasa minyak dan fasa gas serta properti fluida tiga fasa.

### 2.3.1 Perhitungan Properti Fasa Minyak dan Fasa Gas

Properti fasa minyak dan fasa gas pada aliran fluida tiga fasa meliputi massa jenis minyak, massa jenis gas, viskositas gas, serta viskositas minyak.

#### 1. Massa Jenis Minyak

Massa jenis minyak ( $\rho_L$ ) dengan satuan  $lb/ft^3$  dapat dihitung dengan menggunakan rumus Standing Correlation (Ahmed, 1989) yang ditunjukkan oleh persamaan berikut :

$$\rho_L = \frac{62.4 * SGG + 0.0136 * RS * SGG}{0.972 + 0.000147 \left[ RS \left( \frac{SGG}{SGL} \right)^{0.5} + 1.25 * T \right]^{1.175}}$$

dengan :

$$RS = SGG \left( \left[ \frac{P + 14.7}{18.2} + 1.4 \right] * 10^{(0.0125 * API - 0.00091 * T)} \right)^{1.2048}$$

$$API = \frac{141.5}{SGL} - 131.5$$

Keterangan :

$\rho_L$  = Massa Jenis Minyak ( $lb/ft^3$ )

$P$  = Tekanan rata-rata fluida ( $Psi$ )

$T$  = Suhu rata-rata fluida ( $Rankine$ )

$SGG$  = Konstanta gravitasi spesifik untuk gas

$SGL$  = Konstanta gravitasi spesifik untuk minyak

## 2. Massa Jenis Gas

Massa jenis gas ( $\rho_G$ ) dengan satuan  $lb/ft^3$  dapat dihitung dengan menggunakan rumus Standing Correlation (Ahmed, 1989) yang ditunjukkan oleh persamaan berikut :

$$\rho_G = \frac{Mwa * (P + 14.7)}{Z * R * (T + 459.67)}$$

dengan :

$$Mwa = 29 * SGG$$

$$R = 10.732$$

$$Z = A + (1 - A) * \exp(-B) + C * P_{pr}^D$$

$$A = 1.39 \left( T_{pr} - 0.92 \right)^{0.5} - 0.36 * T_{pr} - 0.101$$

$$B = P_{pr} \left( 0.62 - 0.23 * T_{pr} \right) + P_{pr}^2 \left( \frac{0.065}{T_{pr} - 0.86} - 0.037 \right) + \frac{0.32 * P_{pr}^6}{10 \left( 9 * T_{pr} - 1 \right)}$$

$$C = 0.132 - 0.32 * \log(T_{pr})$$

$$D = 10 \left( 0.3106 - 0.49 * T_{pr} + 0.1824 * T_{pr}^2 \right)$$

$$T_{pr} = \frac{T}{T_{pc}}$$

$$T_{pc} = 168 + 325 * SGG - 12.5 * SGG^2$$

$$P_{pr} = \frac{P * 0.000145}{P_{pc}}$$

$$P_{pc} = 677 + 15 * SGG - 37.5 * SGG^2$$

### 3. Viskositas Gas

Viskositas gas ( $\mu_G$ ) dengan satuan  $cp$  adalah kekentalan dari gas.

Berdasarkan korelasi Lee-Gonzales-Eakin (Ahmed, 1989) viskositas gas dapat dituliskan sebagai berikut :

$$\mu_G = 10^{-4} * K * \exp \left( X * \left( \frac{\rho_g}{62.4} \right)^Y \right)$$

dengan :

$$K = \frac{(9.4 + 0.02 * Mwa) * T^{1.5}}{209 + 19 * Mwa + T}$$

$$X = 3.5 + \frac{986}{T} + 0.01 * Mwa$$

$$Y = 2.4 - 0.2 * X$$

#### 4. Viskositas Minyak

Viskositas minyak ( $\mu_L$ ) dengan satuan *cp* adalah kekentalan dari minyak.

Berdasarkan korelasi Glasso (Ahmed,1989) viskositas minyak dapat dituliskan sebagai berikut :

$$\mu_L = 3.141 * 10^{10} * (T - 460)^{-3.444} * (\log(API))^a$$

dengan :

$$a = 10.313 * \log(T - 460) - 36.447$$

#### 2.3.2 Perhitungan Properti Fluida Tiga Fasa

Perhitungan properti fluida tiga fasa meliputi perhitungan laju volumetrik total, laju volumetrik fasa fluida, kecepatan superfisial aliran fluida, fraksi volumetrik, fraksi aliran gas dan minyak, koefisien drag, serta kecepatan minimum fluida.

## 1. Laju Volumetrik Total

Massa jenis atau *density* adalah ukuran konsentrasi massa zat cair dan dinyatakan dalam bentuk massa ( $m$ ) per satuan volume ( $V$ ). Dalam satuan massa jenis dapat dinyatakan dengan  $kg/m^3$ .

$$\rho = \frac{m}{V}$$

Jika dihitung dalam per satuan waktu, maka persamaan di atas dapat menjadi

$$\rho = \frac{W_m}{Q_T}$$

dengan :

$$W_m = \text{laju aliran massa total (Kg/s)}$$

$$Q_T = \text{laju volumetrik total (m}^3/\text{s)}$$

Persamaan di atas dapat ditulis dalam bentuk

$$Q_T = \frac{W_m}{\rho_M}$$

dengan  $\rho_M$  adalah massa jenis campuran.

## 2. Laju Volumetrik Fasa Fluida

Laju volumetrik fluida suatu fasa A ( $Q_A$ ), (dalam Ahmed, 1989), dapat dinyatakan dalam persamaan sebagai berikut :

$$Q_A = H_A * Q_T$$

dengan  $H_A$  adalah fraksi volumetrik fluida suatu fasa A.

### 3. Kecepatan Superfisial Aliran Fluida

Kecepatan superfisial aliran fluida suatu fasa A ( $US_A$ ) atau kecepatan semu dari suatu fasa A adalah rasio laju volumetrik fluida suatu fasa A ( $Q_A$ ) terhadap luas penampang pipa alir ( $A$ ) (Menon, 2005). Secara matematis ditulis sebagai berikut :

$$US_L = \frac{Q_L}{A}$$

$$US_G = \frac{Q_G}{A}$$

$$US_S = \frac{Q_S}{A}$$

### 4. Fraksi Volumetrik

Karena pada segmen pipa besarnya tekanan dapat berubah yang menyebabkan terjadinya perubahan fasa maka diperlukan perhitungan untuk memprediksi perubahan fraksi volumetriknya. Menurut Danielson (2007) (dalam Bello, 2008) fraksi volumetrik fluida gas ( $H_G$ ), fraksi volumetrik fluida pasir ( $H_S$ ), dan fraksi volumetrik fluida minyak ( $H_L$ ) diperoleh dari persamaan sebagai berikut :

$$H_G = \frac{US_G}{1.2 * U_{LG} * U_L} \quad (2.3)$$

$$H_S = \frac{-(US_L + US_S - U_C) + \sqrt{(US_L + US_S - U_C)^2 + 4 * U_C * US_S}}{2 * U_C} \quad (2.4)$$

$$H_L = 1 - (H_G + H_S) \quad (2.5)$$

dengan :

$$\rho_{LG} = \rho_G * H_G + \rho_L * H_L$$

$$U_{LG} = \frac{4}{\pi * D^2} * \left( \frac{\rho_L * Q_L + \rho_G * Q_G}{\rho_{LG}} \right)$$

$$U_L = \frac{US_L}{H_L}$$

$$K = 0.23$$

$$S = \frac{\rho_S}{\rho_L}$$

$$V = \frac{\mu_L}{\rho_L}$$

$$U_C = K * V^{-1/9} * dp^{1/9} * (g * D * (S - 1))^{5/9}$$

##### 5. Fraksi Aliran Gas dan Minyak

Fraksi aliran gas ( $\lambda_G$ ) dan fraksi aliran minyak ( $\lambda_L$ ) (Ahmed, 1989)

dapat dinyatakan sebagai persamaan berikut :

$$\lambda_G = \frac{Q_T * H_G}{US_L + US_G}$$

$$\lambda_L = \frac{Q_T * H_G}{US_L + US_G}$$

## 6. Koefisien Drag

Koefisien drag ( $CD$ ) adalah koefisien yang mempengaruhi gaya drag pada aliran fluida. Koefisien drag ( $CD$ ) (Bello, 2010) dapat dinyatakan dalam persamaan sebagai berikut :

$$U_{\text{slip}} = \frac{US_L}{1 - H_S} - \frac{US_S}{H_S}$$

$$Re_P = \frac{\rho_L * U_{\text{slip}} * dp}{\mu_L}$$

$$CD = \frac{48.5}{\left(1 + 4.5 * \Psi^{0.35}\right)^{0.8} * Re_P^{0.64}} + \left(\frac{Re_P}{Re_P + 100 + 100 * \Psi}\right)^{0.32} \left(\frac{1}{\Psi^{18} + 1.05 * \Psi^{0.8}}\right)$$

## 7. Kecepatan Minimum Fluida

Menurut Bello (2008) kecepatan minimum suatu fluida dapat dituliskan dalam persamaan berikut :

$$U_{\min} = 1.08(1 + 3.64H_S)^{1.09} (1 - H_S)^{0.55n} \left(\frac{\mu_L}{\rho_L}\right)^{-0.09} dp^{0.18} \left(2g \frac{\rho_S - \rho_L}{\rho_L}\right)^{0.54} D^{0.46} \quad (2.6)$$

### 2.3.3 Model Aliran Tiga Fasa (Gas, Minyak, dan Pasir) Bello dan Modifikasinya

Model tekanan aliran tiga fasa (gas, minyak, dan pasir) yang dibuat oleh Bello melibatkan beberapa asumsi yaitu :

- Sudut elevasi dari pipa adalah 0.
- Fasa cair dan padat taktermampatkan (*incompressible*) sehingga densitas fluida tidak berubah sepanjang pipa alir.
- Fluida dalam keadaan tunak (*Steady state*) dan seragam.
- Temperatur dalam jaringan konstan (*isothermal*).
- Penurunan tekanan akibat komponen pipa diabaikan.
- Fluida yang mengalir adalah fluida Newton (*Newtonian Fluid*).

Sehingga membentuk model matematika dengan total penurunan tekanan aliran tiga fasa (gas, minyak, dan pasir) di segmen pipa horizontal (Bello, 2010) adalah sebagai berikut :

$$\left(\frac{\Delta P}{\Delta x}\right)_T = \left(\frac{\Delta P}{\Delta x}\right)_{LG} + \left(\frac{\Delta P}{\Delta x}\right)_{LS} - \left(\frac{\Delta P}{\Delta x}\right)_L \quad (2.7)$$

Keterangan :  $P$  = tekanan fluida (*Pascal*)

$x$  = panjang pipa ( $m$ )

$T$  = Total

$LG$  = Minyak-Gas ;  $LS$  = Minyak-Pasir ;  $L$  = Minyak

Pada jaringan pipa sederhana, model pada persamaan (2.7) dapat digunakan untuk memprediksi tekanan pada *sink* atau segmen pipa penampungan minyak dengan ketentuan bahwa tekanan pada *source* telah diketahui. Namun

untuk jaringan pipa kompleks, model tersebut tidak dapat digunakan karena perhitungan penurunan tekanan satu segmen tidak dapat dilakukan tanpa memperhitungkan pengaruh dari segmen lain. Apalagi pada jaringan yang mengandung *loop* tidak diketahui secara pasti arah alirannya sehingga perhitungan harus dilakukan secara simultan untuk memprediksi aliran. Jadi model Bello pada persamaan (2.7) dimodifikasi oleh Permanasari, dkk (2014) sehingga dapat diterapkan pada jaringan kompleks. Menurut Permanasari, dkk (2014), perubahan tekanan pada *leg j* dapat dituliskan sebagai persamaan berikut :

$$\Delta P_j^{(n+1)} = \frac{1}{\eta_j} W_{mj}^{(n+1)} \quad (2.8)$$

Sehingga diperoleh laju aliran massa total pada *leg j* adalah sebagai berikut :

$$W_{mj}^{(n+1)} = \eta_j \Delta P_j^{(n+1)} \quad (2.9)$$

dengan :

$$\eta_j = \frac{1}{\Delta x \left[ \delta_{1j} W_{mj(n)} + \frac{\delta_{2j}}{W_{mj(n)}} \right]} \quad (2.10)$$

$n = 0, 1, 2, \dots, N$  merupakan iterasi dan  $N$  adalah bilangan bulat positif yang menunjukkan iterasi pada saat komputasi berhenti.

Keterangan :

$$\eta_j, \delta_{1j}, \delta_{2j} = \text{faktor koefisien pada leg } j$$

$$j = 1, 2, 3, \dots, m$$

dengan :

$$\delta_1 = \frac{\alpha + \beta + \gamma_1}{\rho_M^2}$$

$$\delta_2 = \frac{\gamma_2}{\rho_M^2}$$

$$\alpha = \frac{f_L \rho_L H_L^2}{2DA^2}$$

$$\beta = \frac{f_{TP} \rho_{TP}}{2DA^2} \left( \frac{H_G}{E} + \frac{H_L}{(1-E)} \right)^2$$

$$\gamma_1 = \frac{1}{A^2} \left( \frac{H_L}{\rho_L} - \frac{H_S}{\rho_S} \right)^2 \left( \frac{3}{4} CD \frac{\rho_S H_S}{dp} (1-H_S) - 2.06 \right) + \frac{2\rho_{LS}}{DA^2} (f_L + f_{SS}) + \frac{dp^2 \rho_S H_S H_L}{D^2 A V_p}$$

$$\gamma_2 = \frac{\pi \rho_{LS}}{4 V_p} dp^2 (u')^2$$

Keterangan :

$\rho_S$  = massa jenis pasir ( $Kg/m^3$ )

$\rho_{TP}$  = massa jenis untuk turbulensi aliran gas dan minyak(  
 $Kg/m^3$ )

$\rho_{LS}$  = massa jenis minyak dan pasir ( $Kg/m^3$ )

$f_L$  = faktor gesekan minyak

$f_{TP}$  = faktor gesekan untuk turbulensi aliran gas-minyak

$f_{SS}$  = faktor gesekan antar partikel pasir

$D$  = diameter dalam pipa (  $m$  )

$E$  = fraksi void, area pada penampang lintang relatif yang terisi oleh gas

$dp$  = diameter partikel pasir (  $m$  )

$V_p$  = volume campuran (  $m^3$  )

$u'$  = kecepatan aliran turbulen (  $m/s$  )

Model matematika hasil modifikasi pada persamaan (2.8) secara teoritis sudah cukup baik. Namun karena tidak tersedianya data lapangan, model tersebut belum divalidasi dengan keadaan di lapangan.

Perubahan tekanan dengan satuan *Pascal* pada *leg j* adalah selisih antaran tekanan *node* yang masuk dengan tekanan *node* yang keluar dari *leg j*. Sehingga tekanan pada *node* yang masuk *leg j* adalah jumlah antara perubahan tekanan pada *leg j* dengan tekanan pada *node* yang keluar *leg j* atau dapat dituliskan menjadi persamaan berikut :

$$PN_{j \text{ masuk}} = \Delta P_j + PN_{j \text{ keluar}} \quad (2.11)$$

## 2.4 Pemrograman Matlab

Matlab (*Matrix Laboratory*) adalah program untuk analisis dan komputasi numerik, merupakan suatu bahasa pemrograman matematika numerik, merupakan

suatu bahasa pemrograman matematika lanjutan yang dibentuk dengan dasar pemikiran menggunakan sifat dan bentuk matriks.

Matlab telah berkembang menjadi sebuah *environment* pemrograman yang canggih dan berisi fungsi-fungsi *built-it* untuk melakukan tugas pengolahan sinyal, aljabar linier, dan kalkulasi matematis lainnya. Matlab juga berisi “toolbox” yang berisi fungsi-fungsi tambahan untuk aplikasi khusus. Matlab bersifat *extensible*, dalam arti bahwa seorang pengguna dapat menulis fungsi baru untuk ditambahkan di “library” jika fungsi-fungsi *built-in* yang tersedia tidak dapat melakukan tugas tertentu.

Matlab merupakan bahasa pemrograman tingkat tinggi berbasis matriks yang sering digunakan untuk teknik komputasi numerik, menyelesaikan masalah-masalah yang melibatkan operasi matematika elemen, matriks, optimasi, aproksimasi, dan lain lain. Matlab banyak digunakan pada:

- Matematika dan komputasi;
- Pengembangan dan algoritma;
- Pemrograman modeling, simulasi, dan pembuatan prototipe;
- Analisa data, eksplorasi, dan visualisasi;
- Analisis numerik dan statistik;
- Pengembangan aplikasi teknik.

Matlab tidak mengenal deklarasi, maka setiap variabel akan memiliki tipe yang sama dengan nilai yang diberikan. Semua teks sesudah tanda “%” akan menjadi komentar dan tidak akan dianggap sebagai bagian dari algoritma.

### 2.4.1 Fungsi - fungsi pada Matlab

Matlab mempunyai berbagai fungsi matematika umum yang biasa digunakan dalam matematika. Sebagian besar fungsi tersebut hampir sama dengan menuliskannya secara matematis. Beberapa contoh fungsi matematika dasar pada Matlab yaitu :

1. Fungsi “Pi”

Fungsi “pi” telah disediakan dalam Matlab dengan nilai  $\pi = 3.1416$

2. Fungsi “Sqrt”

Fungsi “sqrt” merupakan fungsi akar kuadrat.

3. Fungsi “Exp”, “Log”, “Log10”

Fungsi “exp” adalah perpangkatan dengan bilangan dasar e. “Log” digunakan untuk menghitung nilai logaritma natural sedangkan “log10” untuk menghitung logaritma berbasis 10.

4. Fungsi Invers Matriks

Fungsi “inv” dalam Matlab digunakan untuk menghitung invers matriks. Fungsi “pinv” juga telah disediakan Matlab untuk menghitung invers pada matriks bujur sangkar yang singular.

5. Fungsi Trigonometri

Fungsi yang disediakan Matlab adalah : “sin”, “cos”, “tan”, “sinh”, “cosh”, “tanh”, dan lain-lain.

6. Fungsi “Abs”

Fungsi “abs” digunakan untuk mencari nilai mutlak dari suatu bilangan real.

## 2.4.2 Instruksi pada Matlab

### 1. Instruksi “Function”

Sebuah fungsi pada *editor script* Matlab dimulai dengan kata kunci “function”.

contoh :

```
function hasil=nama_fungsi;
```

### 2. Instruksi *Input* dan *Output*

Mengisi nilai dari piranti masukan dinamakan operasi pembacaan data. Di dalam Matlab, instruksi pembacaan nilai dilakukan dengan notasi “input”. Untuk pencetakan data berbentuk karakter atau bilangan dilakukan dengan notasi “disp” atau dengan notasi “fprintf”.

contoh :

```
variabel=input(':');
disp('teks ....');
disp(variabel); atau
fprintf('teks....%g\n',variabel);
```

### 3. Instruksi Pemilihan

Dalam algoritma ada kalanya sebuah instruksi dikerjakan jika dalam kondisi tertentu dipenuhi. Kondisi adalah persyaratan yang dapat bernilai benar atau salah. Dalam pemilihan dikenal struktur “if-else”. Berikut ini contoh struktur “if-else” :

```
if (kondisi)
    Aksi=1
else
    Aksi=2
end
```

#### 4. Instruksi Pengulangan

Statement “while” digunakan untuk melakukan proses pengulangan suatu statement atau blok statement terus menerus selama kondisi ungkapan logika pada “while” masih bernilai logika seperti yang dicontohkan oleh bentuk umum di bawah ini :

```
while (kondisi)
.....
end
```

Sedangkan untuk mengulang statement atau blok statement berulang kali yang jumlahnya telah diketahui sebelumnya biasanya digunakan statement “for”.

contoh :

```
for var=awal:perubahan:akhir
.....
end
```