MODUL PRAKTIKUM METODE NUMERIS

Laboratorium Komputasi Proses
Jurusan Teknik Kimia

Universitas Sultan Ageng Tirtayasa

MODUL

Pengenalan Matlab dan Pengantar Pemograman

Persamaan Linier

Persamaan Tak Linier

Regresi Linier dan Non Linier

Integrasi

Persamaan Diferensial Biasa

Persamaan Diferensial Parsial

FT UNTIRTA
Jl. Jend. Sudirman Km. 3
Cilegon - Banten 42435

Daftar Isi

Hal

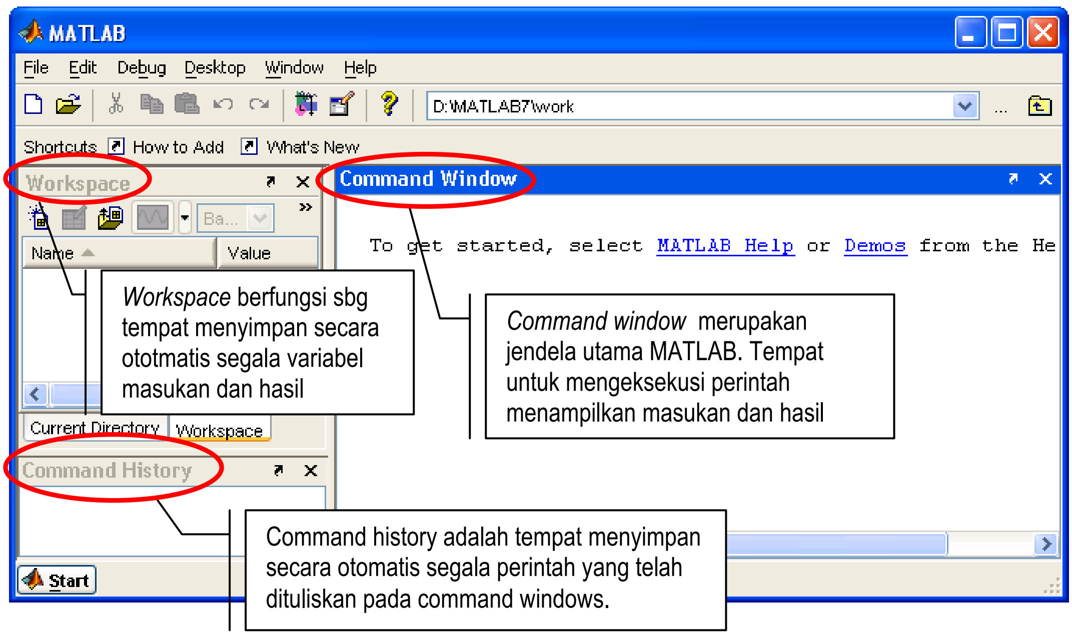
Bab 1
Pengenalan Matlab & Pengantar Pemrograman

1.1 Perangkat Lunak MATLAB

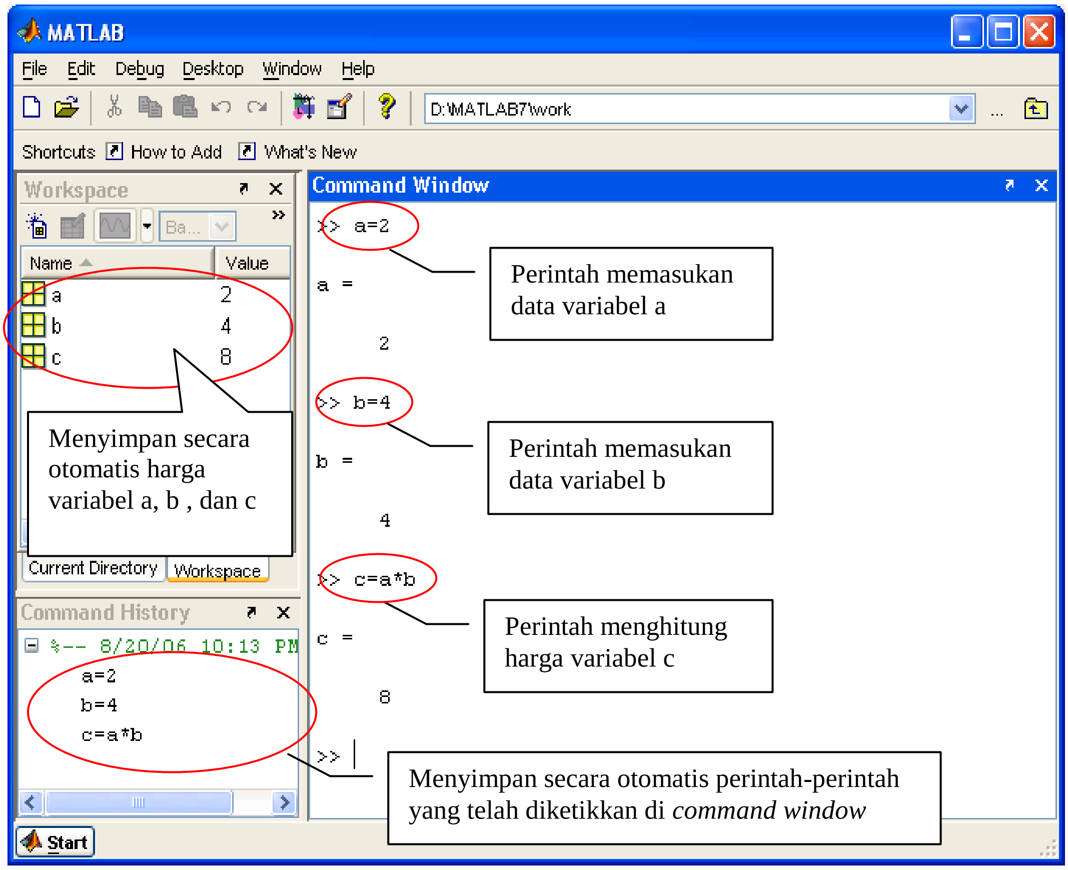
MATLAB merupakan perangkat lunak produk dari The MathWorks,Inc yang memadukan kemampuan perhitungan, pencitraan, dan permograman dalam satu paket. MATLAB merupakan bahasa komputasi teknik yang lebih mudah dan lebih canggih dalam penggunaannya dibandingkan dengan bahasa teknik pendahulunya seperti FORTRAN, BASIC, PASCAL. Sebetulnya MATLAB tidaklah berbeda dengan kalkulator *scientific* yang sehari-hari kita (orang teknik) kenal.

Secara garis besar lingkungan kerja MATLAB terdiri atas beberapa unsur, yaitu:

1. *Command window* (layar kendali)
2. *Workspace* (rak data)
3. *Command history* (layar pengingat)
4. M-file (editor)→ akan dibahas pada bagian khusus.



Gambar 1.1 Lingkungan kerja MATLAB 7.0



Gambar 1.2 Sistem kerja MATLAB

1.2 Matrik, Vektor dan MATLAB

MATLAB adalah singkatan dari *matrix laboratory*. Oleh karena itu pemahaman terhadap konsep matrik harus memadai agar dapat memanfaatkan MATLAB sebagai bahasa komputasi dengan maksimal. Vektor merupakan matrik yang hanya terdiri atas satu kolom atau satu baris saja.

Penulisan matrik di MATLAB

Tanda pisah antar elemen matrik

Tanda koma (,) atau spasi digunakan untuk memisahkan elemen-elemen satu baris. Tanda titik koma(;) digunakan untuk memisahkan elemen-elemen satu kolom.

```
>> a=[1,2,3]
a =
     1     2     3

>> b=[1;2;3]
b =
     1
     2
     3

>> A=[1 2 3;4 5 6;7 8 9]
A =
     1     2     3
     4     5     6
     7     8     9
```

Matrik transposisi

```
>> A'
ans =

     1     4     7
     2     5     8
     3     6     9
```

Menentukan ukuran matrik

```
>> size(A)
ans =
     3     3
```

Menentukan determinan matrik

```
>> det(A)
ans =
     0
```

Menentukan invers matrik

```
>> inv(A)
Warning: Matrix is close to singular or badly scaled.
         Results may be inaccurate. RCOND = 1.541976e-018.
ans =
   1.0e+016 *
```

```
   -0.4504    0.9007   -0.4504
    0.9007   -1.8014    0.9007
   -0.4504    0.9007   -0.4504
```

Perhitungan invers matrik A menggunakan MATLAB ternyata memunculkan peringatan yang menyatakan bahwa matrik A adalah singular (tak wajar). Hal ini bisa diketahui lebih awal dengan melihat harga determinan A. Apabila determinan A berharga nol dapat dipastikan matrik A adalah matrik singular.

Vektor baris adalah matrik yang terdiri atas satu baris saja.

```
>> B=[2:6]
B =
     2     3     4     5     6
```

Penulisan seperti di atas akan menghasilkan vektor baris dengan selisih 1

```
>> C=[2:2:6]
C =
     2     4     6
```

Penulisan seperti di atas akan menghasilkan vektor baris dengan selisih 2

Vektor kolom adalah matrik yang terdiri atas satu kolom saja

```
>> V=[2:0.5:4]'
V =
    2.0000
    2.5000
    3.0000
    3.5000
    4.0000
```

Penulisan seperti di atas akan menghasilkan vektor kolom dengan selisih 0.5

Menentukan ukuran vektor

```
>> length(V)
ans =
     5
```

Aljabar Matrik

Operasi aljabar matrik maupun skalar menggunakan simbol yang tidak jauh berbeda. Berikut ini hirarki operasi aljabar dalam MATLAB. Pertama ^ kedua * ketiga / atau \ dan terakhir + dan -. Keterangan:

^	Pangkat
*	Perkalian
/	Pembagian matrik kanan (mis: B/A = B*inv(A))
\	Pembagian matrik kiri (mis: A\B = inv(A)*B)
+	Penambahan
-	Pengurangan

Penjumlahan dan pengurangan

Hanya dapat dilakukan jika matrik-matrik yang akan dijumlahkan dan dikurangkan memiliki orde sama.

$$\begin{bmatrix} 2 & 3 & 1 & 6 \\ 1 & 4 & 5 & 2 \end{bmatrix} + \begin{bmatrix} 2 & 3 & 1 & 6 \\ 1 & 4 & 5 & 2 \end{bmatrix} = \begin{bmatrix} 4 & 6 & 2 & 12 \\ 2 & 8 & 10 & 4 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 3 & 1 & 6 \\ 1 & 4 & 5 & 2 \end{bmatrix} - \begin{bmatrix} 2 & 3 & 1 & 6 \\ 1 & 4 & 5 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

```
>> A =[2 3 1 6;1 4 5 2]
A =

     2     3     1     6
     1     4     5     2

>> A+A
ans =
     4     6     2    12
     2     8    10     4
>> A-A
ans =
     0     0     0     0
     0     0     0     0
```

Perkalian matrik

AB Syarat → jumlah kolom A = jumlah kolom baris B

$AB \neq BA$

Misal

$$A = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$AB = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = 1 + 4 + 9 = 14$$

(1 x 3) (3 x 1)=(1 x 1)

$$BA = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{bmatrix}$$

Operasi perkalian matrik dalam MATLAB dilakukan dengan simbol *

```
>> A=[1,2,3]

A =
     1     2     3


>> B=[1;2;3]
B =
     1
     2
     3
>> A*B
ans =
  14
>> B*A
ans =
     1     2     3
     2     4     6
     3     6     9
```

Pembagian matrik kanan

$xA = c$

$x = cA^{-1}$

$x = c / A$

Misalkan:

$$x \begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 4 \\ 4 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 20 & 15 & -8 \end{bmatrix}$$

```
>> A=[1 2 3;2 5 4;4 3 1]
A =
     1     2     3
     2     5     4
     4     3     1
>> c=[20 15 -8]
c =
    20    15    -8
>> x=c/A
x =
   -8.6667    3.0952    5.6190
```

Pembagian matrik kiri

$Ax = c$

$x = A^{-1}c$

$x = A \backslash c$

Misalkan:

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 4 \\ 4 & 3 & 1 \end{bmatrix} x = \begin{bmatrix} 20 \\ 15 \\ -8 \end{bmatrix}$$

```
>> A=[1 2 3;2 5 4;4 3 1]
A =
     1     2     3
     2     5     4
     4     3     1
```

```
>> c=[20;15;-8]
c =
    20
    15
    -8

>> x=A\c
x =
   -1.0000
   -4.7143
   10.1429

>> A=[1 2;3 4]
A =
     1     2
     3     4

>> A.*A
ans =
     1     4
     9    16

>> A./A'
ans =

    1.0000    0.6667
    1.5000    1.0000

>> A.\A'
ans =
    1.0000    1.5000
    0.6667    1.0000

>> A.^A
ans =
     1     4
    27   256
```

1.3 Membuat Grafik

Grafik 2 Dimensi

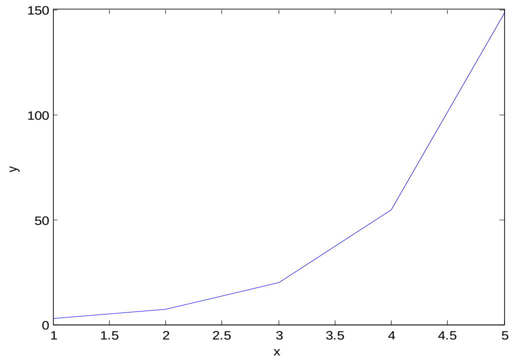
Perintah menggambar grafik 2D

plot(x,y)

Misalkan:

x	1	2	3	4	5
y	2.7	7.4	20.1	54.6	148.4

```
>> x=[1,2,3,4,5]
x =
     1     2     3     4     5
>> y=[2.7,7.4,20.1,54.6,148.4]
y =
    2.7000    7.4000   20.1000   54.6000  148.4000

>> plot(x,y)
>> xlabel('x')
>> ylabel('y')
```



Grafik 3 Dimensi

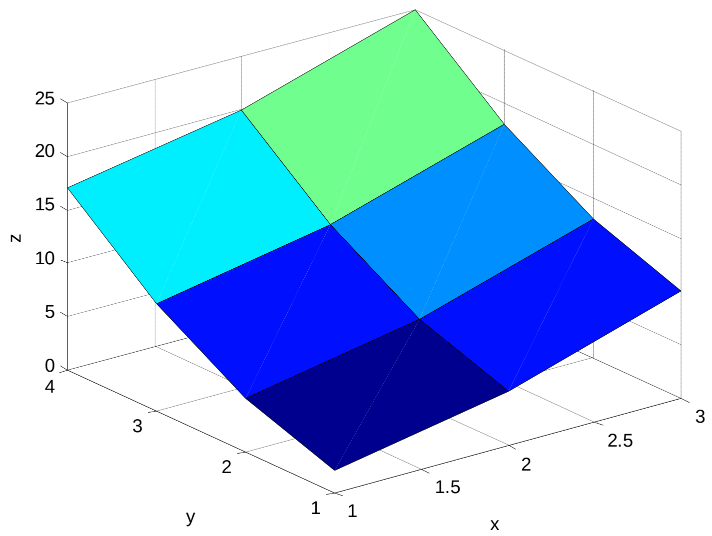
Perintah menggambar grafik 3D

surf(x,y,z)

Misalkan:

x	y	z(x=1)	z(x=2)	z(x=3)
1	1	2	5	10
2	2	5	8	13
3	3	10	13	18
	4	17	20	25

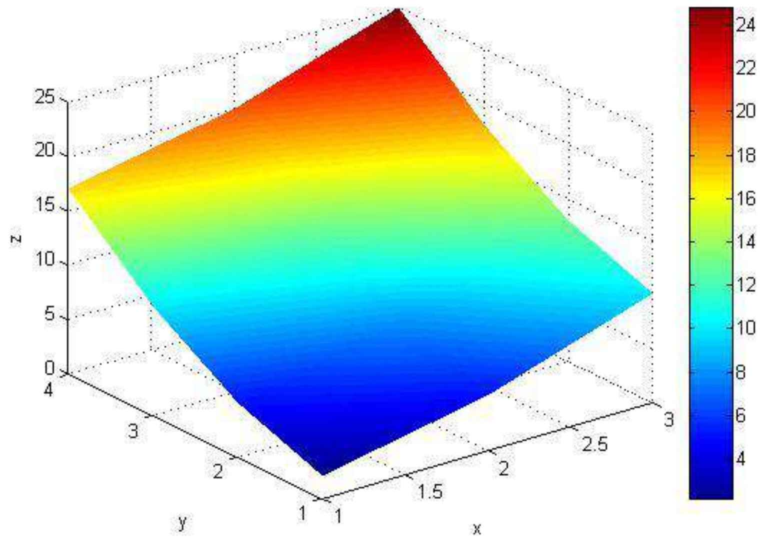
```
>> x=[1 2 3]
x =

     1     2     3

>> y=[1 2 3 4]

y =

     1     2     3     4

>> z=[2 5 10;5 8 13;10 13 18;17 20 25]

z =

     2     5    10
     5     8    13
    10    13    18
    17    20    25

>> surf(x,y,z)
>> xlabel('x')
>> ylabel('y')
>> zlabel('z')
```



Gambar 1.4 Grafik 3 Dimensi

Untuk mempercantik tampilan dan mempermudah penafsiran grafik dengan menambah legenda warna ketikkan perintah berikut ini.

```
>> shading interp
>> colorbar
```

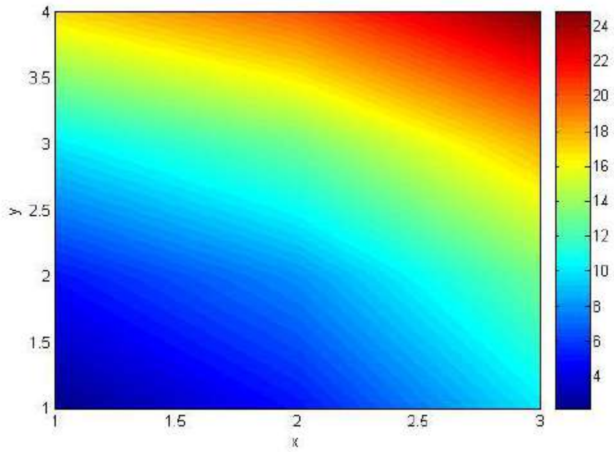


Gambar 1.5 Grafik 3 Dimensi yang diperhalus

Grafik 3 Dimensi Semu

Apabila penafsiran grafik 3D seperti tercetak di muka masih dirasakan sulit, MATLAB telah menyediakan perintah untuk membuat grafik 3D menjadi grafik 2D.

```
>> pcolor(x,y,z)
>> xlabel('x')
>> ylabel('y')
>> zlabel('z')
>> shading interp
>> colorbar
```



Gambar 1.6 Grafik 3 Dimensi semu

Kasus 1 [volume tangki penyimpan]

Senyawa kimia yang mudah menguap pada temperatur kamar biasa disimpan dalam fasa cair pada tekanan uapnya. Dalam kasus ini n-butana (C_4H_{10}) di simpan pada tekanan 2,581 bar dan temperatur 300 K. Penyimpanan skala besar n-butana ,*bulk*>50 m^3, seringkali dilakukan dalam tangki yang berbentuk bola (*spherical*).

Sebuah tangki penyimpan n-butana berbentuk bola. Hitunglah volume tangki berbentuk bola yang memiliki jari-jari 2,3,……9,10 m!.

Jawaban:

$$\boxed{V_{bola} = \frac{4}{3}\pi r^3}$$

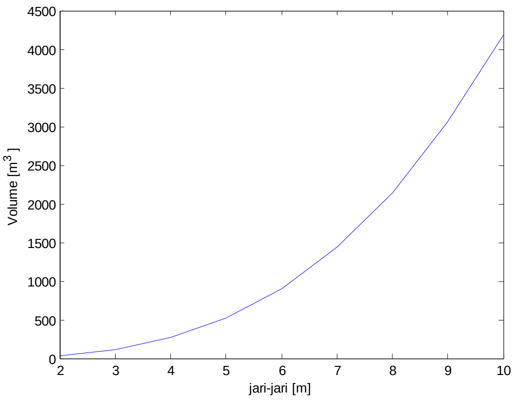
Penulisan program untuk kasus 1 kita lakukan dengan cara, yaitu dalam bentuk skrip dan fungsi .

Penulisan program dalam bentuk skrip

```
% kasus_1.m
clc
clear

r = 2:10
V =4/3*pi*r.^3

% Membuat grafik V terhadap r
plot(r,V)
xlabel('jari-jari [m]')
ylabel('Volume [m^3 ]')
```

Eksekusi kasus_1.m dalam command window

```
>>kasus_1
r =
     2     3     4     5     6     7     8     9    10

V =
  1.0e+003 *
  Columns 1 through 5
    0.0335    0.1131    0.2681    0.5236    0.9048
  Columns 6 through 9
    1.4368    2.1447    3.0536    4.1888
```



Gambar 1.8 Volume vs jari-jari tangki penyimpan [skrip]

Tugas 1: Pengenalan MATLAB dan Membuat Program Sederhana

Nomor 1 (Tutorial MATLAB)

Baca tutorial "Cepat Mahir MATLAB", Bab 1 Memulai Menggunakan MATLAB dan Bab 5 Fungsi M-File.

Nomor 2 (Persamaan Antoine)

Buat sebuah algoritma dan program dalam M-file untuk menghitung tekanan uap murni n-heksana dalam rentang temperatur 25 - 100 ^oC, dengan menggunakan persamaan Antoine sbb:

$$\ln P = A - B/(T + C)$$

Dengan

A = 14.0568 T = Temperatur (K)

B = 2825.42 P = Tekanan uap murni (kPa)

C = -42.7089

Buat pula grafik P terhadap T-nya menggunakan rutin *plot* dalam MATLAB.

Nomor 3 *(Equimolar Counterdiffusion)*

Gas amoniak (A) berdifusi melalui pipa sepanjang 0,10 m yang berisi gas N2 (B) pada tekanan 1,0132 x 105 Pa dan temperatur 298 K. Tekanan pada titik 1 $P_{A,1}$ = 1,013 x 10^4 Pa dan pada titik 2 $P_{A,2}$ = 0,507 x 10^4 Pa. Diffusivitas D_{AB} = 0,230 x 10^{-4} m^2/s. Laju diffusi gas amoniak (A) dapat dievaluasi menggunakan Hukum Fick's berikut ini:

$$J_A^* = \frac{D_{AB}(P_{A1} - P_{A2})}{RT\Delta z}[kmol.A/(s.m^2)]$$

R = 8314 J/(kmol.K)

Buat sebuah algoritma dan program MATLAB berupa suatu **fungsi** dalam M-file untuk menghitung laju diffusi gas amoniak.

Petunjuk : program terdiri atas 2 buah m-file. 1 buah untuk menulis fungsi, 1 buah untuk mengeksekusi fungsi.

________________________________o0o________________________________

Bab 2
Persamaan Linier

4.2 Linierisasi

Seringkali ditemukan persamaan tak linier dalam permasalah real teknk kimia. Tentunya kita tak dapat begitu daja mengalurkan data-data dengan menggunakan pemodelan linier. Agar dapat dimodelkan dengan pemodelan linier, maka persamaan tak linier itu harus dilinierisasi terlebih dahulu. Berikut ini pemaparannya.

$$y = ae^{bx}$$

LINIERISASI

$$\ln y = \ln(ae^{bx})$$
$$\ln y = bx + \ln a$$

Tabel 4.1 Hasil linearisasi persamaan-persamaan tak linier

Tipe persamaan	absis	ordinat	slope	intersep
$y = ax + b$	x	y	a	b
$y = ae^{bx}$	x	ln(y)	b	ln(a)
$y = \dfrac{x}{(ax+b)}$	x	x/y	a	b
$y = a/x + b$	1/x	y	a	b
$y = ax^{b} + c$	ln(x)	ln(y-c)	b	ln(a)

Kasus 1

Harga konduktivitas alumunium pada berbagai temperatur sbb

T (K)	300	400	500	600	800
k (Btu/(h×ft2)(°F/ft)	273	240	237	232	220

Model matematik dapat diwakili dengan menggunakan persamaan linier

$$k = a_0 T + a_1$$

Untuk mencari harga a_0 dan a_1 dapat menggunakan metode jumlah selisih kuadrat terkecil seperti yang telah dijelaskan sebelumnya.

```
%konduktivitas.m
clear
clc
T=[4,5,6,8]*100;          % Absis
k=[240,237,232,220];      % Ordinat
n=length(k);
A = [sum(T.^2),sum(T)
    sum(T), n];
c = [sum(k.*T)
    sum(k)];
a = A\c
kmod = a(1)*T +a(2);
S = sum((k-kmod).^2)
```

Hasil pada *command window*

```
>>konduktivitas
a =
   -0.0511
  261.6571
S =
  3.8857
```

Subrutin MATLAB untuk regresi persamaan linear dan polinomial dapat menggunakan perintah sbb:

[P,S] = polyfit(x,y,n)

data-data

n=orde polinom

Kasus 1 merupakan persamaan linier, maka n yang digunakan dalam subrutin polyfit adalah 1. Berikut ini pemrograman MATLAB-nya.

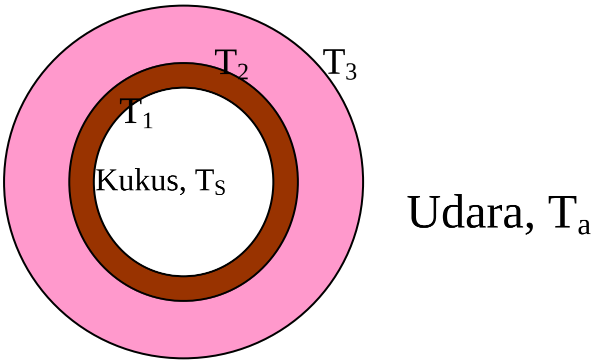
```
%konduktivitas2
T = [4,5,6,8]*100;          % Absis
K = [240,237,232,220];      % Ordinat

[P,S] = polyfit(T,k,1)
```

```
>>konduktivitas2
P =
   -0.0511   261.6571
S =
        R: [2x2 double]
```

```
        df: 2
     normr: 1.9712
```

Kasus1

Kukus lewat jenuh bertemperatur 130 ^{o}C mengalir dalam sebuah pipa yang memiliki diameter dalam 20 mm (D_1), dan diameter luar 25 mm (D_2). Pipa diinsulasi setebal 40 mm [(D3 – D2)/2]. Koefisien konveksi kukus (hi) = 1700 W/m2.K, dan koefisien konveksi udara (ho) = 3 W/m2.K. Konduktivitas termal pipa (ks) = 45 W/m.K, dan insulasi (ki) = 0,064 W/m.K. Temperatur udara di luar insulasi = 25 ^{o}C. Perkirakan temperatur T_1, T_2, dan T_3.



Perpindahan panas dari kukus ke pipa.

$$h_i \pi D_1 (T_S - T_1) = \frac{(T_1 - T_2)}{\ln(D_2 / D_1)/(2\pi k_s)}$$

Perpindahan panas dari pipa ke insulasi

$$\frac{(T_1 - T_2)}{\ln(D_2 / D_1)/(2\pi k_s)} = \frac{(T_2 - T_3)}{\ln(D_3 / D_2)/(2\pi k_i)}$$

Perpindahan panas dari insulasi ke udara

$$\frac{(T_2 - T_3)}{\ln(D_3 / D_2)/(2\pi k_i)} = h_O \pi D_3 (T_3 - T_a)$$

Ada tiga persamaan linier yang berhasil dirumuskan dari peneracaan energi tersebut.

MATLAB &Pengantar Pemrograman

$$\left[\frac{2k_s}{\ln\left(D_2/D_1\right)}+h_iD_1\right]T_1-\left[\frac{2k_s}{\ln\left(D_2/D_1\right)}\right]T_2=h_iD_1T_S$$

$$\left[\frac{k_s}{\ln\left(D_2/D_1\right)}\right]T_1-\left[\frac{k_s}{\ln\left(D_2/D_1\right)}+\frac{k_i}{\ln\left(D_3/D_2\right)}\right]T_2+\left[\frac{k_i}{\ln\left(D_3/D_2\right)}\right]T_3=0$$

$$\left[\frac{2k_i}{\ln\left(D_3/D_2\right)}\right]T_2-\left[\frac{2k_i}{\ln\left(D_3/D_2\right)}+h_OD_3\right]T_3=-h_OD_3T_a$$

Ubah sistem persamaan linier menjadi bentuk matrik Ax = c, menjadi sbb:

$$\begin{bmatrix} \left[\frac{2k_s}{\ln\left(D_2/D_1\right)}+h_iD_1\right] & -\left[\frac{2k_s}{\ln\left(D_2/D_1\right)}\right] & 0 \\ \left[\frac{k_s}{\ln\left(D_2/D_1\right)}\right] & -\left[\frac{k_s}{\ln\left(D_2/D_1\right)}+\frac{k_i}{\ln(D_3/D_2)}\right] & \left[\frac{k_i}{\ln\left(D_3/D_2\right)}\right] \\ 0 & \left[\frac{2k_i}{\ln\left(D_3/D_2\right)}\right] & -\left[\frac{2k_i}{\ln\left(D_3/D_2\right)}+h_OD_3\right] \end{bmatrix}\begin{bmatrix}T_1\\T_2\\T_3\end{bmatrix}=\begin{bmatrix}h_iD_1T_S\\0\\-h_OD_3T_a\end{bmatrix}$$

Berikut ini pemrograman MATLAB-nya.

```
%kasus2.m
clc
clear
% Input data
Ts = 130;           % oC
Ta = 25;            % oC
D1 = 20e-3;         % Diameter dalam pipa, m
D2 = 25e-3;         % Diameter luar pipa, m
Ith = 40e-3;        % Tebal insulasi, m
D3 = (D2 + 2*Ith);      % Diameter pipa + insulasi
hi = 1700;  % Koefisien transfer panas bagian dalam (W/m2.K)
ho = 3 ;            % koefisien transfer panas bagian luar (W/m2.K)
ks = 45;            % Konduktivitas panas baja (W/m.K)
ki = 0.064;         % Konduktivitas panas insulasi (W/m.K)

% Matriks koefisien variabel
A = [2*ks/log(D2/D1)+hi*D1 , -2*ks/log(D2/D1) , 0
   ks/log(D2/D1) , -(ks/log(D2/D1)+ki/log(D3/D2)) , ki/log(D3/D2)
   0 , 2*ki/log(D3/D2) , -(2*ki/log(D3/D2)+ho*D3)];

% Matriks konstanta
c = [hi*D1*Ts ; 0 ; -ho*D3*Ta];
% Menyelesaikan sis pers. linier dengan fungsi invers MATLAB
T = inv(A)*c
```

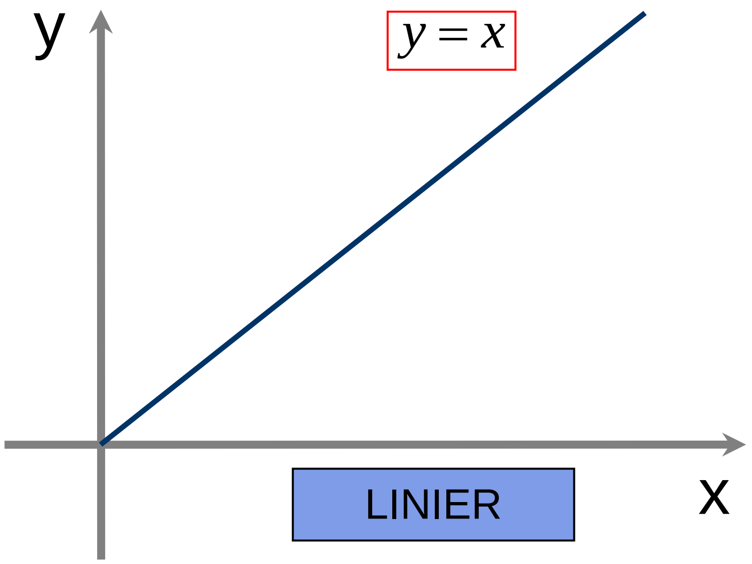
Eksekusi persamaan di *command window*

```
>>kasus2
T =

  129.7858
  129.7678
   48.1191
```

o0o

Bab 3
Persamaan Tak Linier

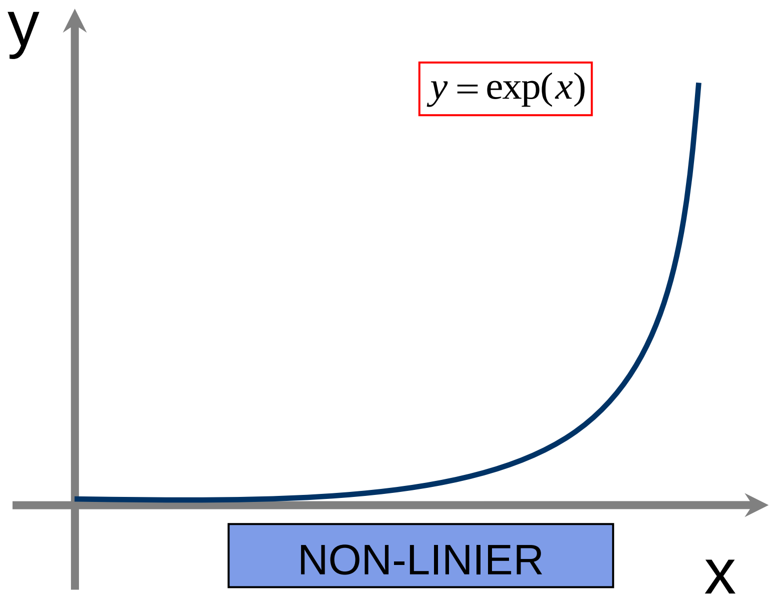
Persamaan Linier $\boxed{y = mx + c}$



Gambar 3.1 Kurva linier

Persamaan Tak Linier

Contoh:



Gambar 3.2 Kurva tak linier

Berikut ini beberapa contoh Persamaan Tak Linier

Tabel 3.1 Contoh Persamaan Tak linier

Jenis Pers. Tak Linier	**Contoh**
Persamaan Kuadrat	$x^2 - 4x + 3 = 0$
Persamaan Polinomial	$x^4 + 6x^3 + 7x^2 - 6x - 8 = 0$
Persamaan Transenden	$\sin x - 2\exp(-x^2) = 0$
Persamaan Logaritmik	$\ln(1 + x^2) - 2\exp(-x^2) = 0$

Dalam aplikasinya di bidang teknik kimia, persamaan tak linier memiliki peranan yang sangat penting.

Tabel 3.2 Aplikasi Persamaan Tak Linier dalam bidang teknik kimia

Aplikasi Pers. Tak Linier	**Contoh**
Neraca Massa dan Energi,	$\varepsilon\Delta H_0^o + N^{out}\int_{To}^{T_{out}} C_{P,i}^{out}\,dT - N^{in}\int_{To}^{T_{in}} C_{P,i}^{in} = 0$
Termodinamika Persamaan gas nyata/kubik, Kesetimbangan reaksi kimia,	$P = \frac{RT}{V-b} - \frac{a}{V^2}$ (1 $\ln K + \frac{\Delta G_0^o - \Delta H_0^o}{RT_0} + \frac{\Delta H_0^o}{RT} + \frac{1}{T}\int_{T_0}^{T} \frac{\Delta C_p^o}{R} dT - \int_{T_0}^{T} \frac{\Delta C_p^o}{R}\frac{dT}{T} = 0$
Operasi Teknik Kimia, dll.	$\left(\sum_{j=1}^{n} \frac{\alpha_j z_{jF} F}{\alpha_j - \phi}\right) - F(1-q) = 0$ (2

1) Persamaan kubik tersebut diusulkan oleh Johannes Diderik van der Waals (1873), Fisikawan Belanda, peraih nobel Fisika pada tahun 1910.
2) Persamaan Underwood untuk distilasi multikomponen.

Contoh:

Carilah akar-akar persamaan kuadrat $x^2+4x+3=0$ dengan menggunakan metode penyetengahan interval!.

```
%kuadrat.m
function y = kuadrat(x)

y = x^2+4*x+3;
```

Perintah pada *command window* sbb:

```
>>biseksi('kuadrat',-2,1,1e-6)
ans =
   -1.0000
>> biseksi('kuadrat',-2,-4,1e-6)
ans =
  -3.0000
```

Dari perhitungan menggunakan metode bisection diperoleh akar-akar dari persamaan kuadrat adalah [-1,-3].

Metode Newton-Raphson

Tabel 4.3 Karakteristik metode penyetengahan interval

No	Keunggulan	Kelemahan
1.	Hanya butuh satu tebakan awal.	Kekonvergenan adakalanya gagal dicapai.
2.	Laju konvergensi cepat	

Kita gunakan contoh kasus yang sama dengan contoh pada metode bisection.

$$y=x^2+4x+3=0$$
$$\frac{dy}{dx}=2x+4$$

```
>> [x iter]=NewRap('kuadrat','dkuadrat',2,1e-6)
x =
   -1.0000
```

```
iter =
     6
>> [x iter]=NewRap('kuadrat','dkuadrat',-4,1e-6)
x =
   -3.0000
iter =
     5
```

Dari perhitungan menggunakan metode Newton Raphson diperoleh akar-akar dari persamaan kuadrat adalah [-1,-3].

Subrutin dalam MATLAB untuk persamaan tak linier tunggal

MATLAB telah menyediakan program untuk menyelesaikan persamaan linier tunggal yang telah menyatu dengan program MATLAB itu sendiri. Ada dua subrutin yang umum digunakan, yaitu roots dan fzero.

Tabel 4.4 Perbandingan subrutin roots terhadap fzero

Rutin	Keunggulan	Kelemahan
roots.m	1. Seluruh akar dapat diketahui dengan hanya sekali menjalankan rutin. 2. Tidak membutuhkan tebakan mula.	1. Hanya untuk pers. kuadrat dan polinomial.
fzero.m	1. Solusi bagi segala jenis pers tak linier.	1. Hanya satu buah akar yang dapat diketahui sekali menjalankan rutin. 2. Membutuhkan tebakan mula.

Penggunaan roots:

Penulisan perintah roots di *Command window* MATLAB

C(1)*X^N + ... + C(N)*X + C(N+1)

C = [C(1)   C(2)........C(N) C(N+1)

roots(C)

Contoh persamaan kuadrat $x^2 + 4x - 5 = 0$ maka C(1)=1, C(2)=4, C(3)= -5.

Carilah akar-akar persamaan kuadrat di bawah ini.

$$x^2 + 4x - 5 = 0$$

MATLAB *Command window*

```
>> C=[1 4 -5]

C =

     1     4    -5

>> roots(C)

ans =

    -5
     1
```

Penggunaan fzero:

Penulisan fzero di MATLAB *Command window*

x =fzero('fungsi',x0)

Contoh penggunaan fzero:

$$x^2 + 4x + 3 = 0$$

Penulisan contoh di MATLAB *Command window*

```
>> fzero('x^2+4*x+3',0)

ans =

    -1
```

Untuk keteraturan dan kemudahan pemanggilan akan lebih baik mendefinisikan fungsi pada m-file.

```
%kuadrat.m
function y = kuadrat(x)
y = x^2+4*x+3
```

Baru kemudian kita panggil fungsi dari MATLAB *Command window*

```
>> x = fzero('kuadrat',0)
x =
    -1
```

Untuk mencari akar lainnya, ubah tebakan awalnya.

```
>> x = fzero('kuadrat',-4)
x =
   -3.0000
```

Kasus 3 Aplikasi subrutin roots

Kasus 3
Tekanan uap n-butana pada temperatur 350 K adalah 9.4573 bar.
Hitunglah volume molar uap jenuh dan cair jenuh n-butana pada
Kondisi tersebut dengan menggunakan persamaan gas Van der Waals.
(R=8.314j/mol.K ;Tc=425.1 K; Pc=37.96 bar)

Jawaban:

Persamaan Van der Waals

$$P = \frac{RT}{V-b} - \frac{a}{V^2}$$

$$a = \frac{27}{64}\frac{R^2T_c^2}{P_c} \quad \text{dan} \quad b = \frac{1}{8}\frac{RT_c}{P_c}$$

Transformasi ke dalam bentuk umum pers.polinomial

$$P(V-b)(V^2)=RTV^2-a(V-b)$$
$$P(V^3-bV^2)=RTV^2-aV+ab$$
$$PV^3-(Pb+RT)V^2+aV-ab=0$$

```
% kasus3.m

clear
clc
% Masukan kondisi operasi
P = input('masukan tekanan, Pa = ');
T = input('masukan temperatur, K = ');
R = 8314 ;        %J/(kmol.K)
Pc = 37.96e5;     %Pa
Tc = 425.1;       %K
% Hitung konstanta a & b
a = (27/64)*R^2*Tc^2/Pc;
b = (1/8)*R*Tc/Pc;
% Definisikan koefisien polinomial
VdW=[P, -(P*b + R*T), a, -a*b];
vol = roots(VdW)          %liter/mol
```

Eksekusi program kasus3.m. Masukan dan hasil di *Command Window*

```
>>kasus3
masukan tekanan, Pa = 9.4573e5
masukan temperatur, K = 350

vol =

   2.6669
   0.3354
   0.1910
```

Kasus 4 Aplikasi subrutin fzero

Diketahui sebuah persamaan kapasitas panas sbb.

$$Cp = 0.716 - 4.257E^{-6}T + \frac{15.04}{\sqrt{T}} \qquad \left[\frac{kJ}{kg.K}\right]$$

Tentukan temperatur pada saat Cp = 1 kJ/kg.K !

Langkah 1 Membuat program fungsi yang akan dinolkan.

```
%KapPns.m

function f = KapPns(T,cp)
%Persamaan tak linier yang akan dinolkan
f = cp - 0.716 + 4257e-6*T - 15.04/T^0.5;
```

Langkah 2 Membuat program pengeksekusi

```
% kasus4.m

clear
clc
cp = input('masukan kapasitas panas,kJ/kg.K = ');
T = fzero(@(T) KapPns(T,cp),100)
```

Langkah 3 Eksekusi program kasus4.m

Masukan dan hasil di *Command Window*

```
>> kasus4
masukan harga kapasitas panas,kJ/kg.K = 1

T =

       189.7597
```

Tugas 4

Menyelesaikan persamaan tak linier tunggal dengan menggunakan subrutin MATLAB

Tekanan uap n-butana pada temperatur 350 K adalah 9.4573 bar. Volume molar uap jenuh dan cair jenuh n-butana pada kondisi tersebut dapat dihitung dengan menggunakan persamaan kubik Redlich-Kwong-Soave sebagai berikut:

$$P = \frac{RT}{V-b} - \frac{a\alpha}{V(V+b)}$$

Dalam bentuk persamaan polinomial menjadi sebagai berikut::

$$Z^3 - Z^2 + (A - B - B^2)Z - AB = 0$$

Dengan:

$$B = \frac{bP}{RT} \quad A = \frac{\alpha aP}{R^2T^2} \quad Z = \frac{PV}{RT}$$

$$a = \frac{0.4278R^2T_C^2}{P_C} \;; b = \frac{0.0867RT_C}{P_C} \quad \alpha = \left[1 + S\left(1 - \sqrt{\frac{T}{T_C}}\right)\right]^2;$$

$$S = 0.48508 + 1.55171\omega - 0.15613\omega^2$$

(R=8.314j/mol.K ;Tc=425.1 K; Pc=37.96 bar; ω = 0.1931). **Hitunglah volume molar uap jenuh dan cair jenuh n-butana pada kondisi itu !!.**

Contoh sistem persamaan tak linier.

$$f_1(x,y) = x^3 - 3xy^2 - 1/2 = 0$$

$$f_2(x,y) = 3x^2y - y^3 - \sqrt{3}/2 = 0$$

Langkah 1 Buat terlebih dahulu fungsi sistem persamaan taklinier dalam m-file.

```
%sistem.m
function f = sistem(x)
f=[x(1)^3-3*x(1)*x(2)^2-0.5
    3*x(1)^2*x(2)-x(2)^3-sqrt(3)/2]
```

Langkah 2 Buat program pengeksekusi menggunakan Newton.m pada m-file yang berbeda atau dapat juga langsung di command window.

```
%run_sistem.m
[x iter] = Newton('sistem',[1 2])
```

Langkah 3 Jalankan program pengeksekusi. Klik debug/run

Langkah 2 dapat di loncat dengan menuliskan langsung perintah eksekusi pada Command window

```
>> [x iter] = Newton('sistem',[1 2])
x =
    2.5198
    1.5874
iter =      7
```

Subrutin dalam MATLAB untuk sistem persamaan taklinier

Solusi sistem persamaan taklinier dapat menggunakan fsolve pada MATLAB.

Contoh:

$$x^3 - 3xy^2 = 1/2$$

$$3x^2y - y^3 = \sqrt{3}/2$$

Langkah 1 Buat terlebih dahulu fungsi sistem persamaan taklinier dalam m-file.

Langkah 2 Buat program pengeksekusi menggunakan fsolve pada m-file yang berbeda atau dapat juga langsung di command window.

Langkah 3 Jalankan program pengeksekusi.

```
function f = sistem(x)
f=[x(1)^3-3*x(1)*x(2)^2-0.5
    3*x(1)^2*x(2)-x(2)^3-sqrt(3)/2]
```

```
>>[X,FVAL] = fsolve('sistem',[1 2])
Optimization terminated: first-order optimality is less than
options.TolFun.

X =

    2.5198    1.5874

FVAL =

  1.0e-010 *

    0.1930
    0.0966
```

Kasus 5

Reaksi reformasi kukus berlangsung menurut rangkaian reaksi kesetimbangan berikut:

$$CH_{4(g)} + H_2O_{(g)} \rightleftharpoons CO_{(g)} + 3H_{2(g)} \qquad \text{R-1}$$

$$CO_{(g)} + H_2O_{(g)} \rightleftharpoons CO_{2(g)} + H_2 \qquad \text{R-2}$$

Pada suhu 2000 K harga konstanta kesetimbangan untuk masing-masing reaksi adalah 1,930x10^{-4} dan 5,528. Tentukan komposisi kesetimbangan komponen-komponen apabila Gas umpan berkomposisi 20% $CH_{4(g)}$ dan 80% $H_2O_{(g)}$ berada pada kondisi suhu 2000 K dan tekanan 1 atm.

Jawaban

Misal ditetapkan basis perhitungan 10 mol gas umpan

e_1 = derajat reaksi dari reaksi pertama

e_2 = derajat reaksi dari reaksi kedua

Fraksi mol kesetimbangan setiap komponen dapat dinyatakan sebagai berikut:

$$Y_{CO} = \frac{e_1 - e_2}{10 + 2e_1} \qquad Y_{H_2} = \frac{3e_1 + e_2}{10 + 2e_1} \qquad Y_{H_2O} = \frac{8 - e_1 - e_2}{10 + 2e_1}$$

$$Y_{CO_2} = \frac{e_2}{10 + 2e_1} \qquad Y_{CH_4} = \frac{2 - e_1}{10 + 2e_1}$$

Persamaan konstanta kesetimbangan dinyatakan sebagai berikut:

$$K_1 = \frac{Y_{CO} Y_{H_2}^3 P^2}{Y_{CH_4} Y_{H_2O}} \qquad K_2 = \frac{Y_{CO_2} Y_{H_2}}{Y_{CO} Y_{H_2O}}$$

$$\left.\begin{array}{l} \boxed{\dfrac{(e_1 - e_2)(3e_1 - e_2)^3}{(2 - e_1)(8 - e_1 - e_2)(10 + 2e_1)^2} = K_1} \\[2ex] \boxed{\dfrac{e_2(3e_1 + e_2)}{(e_1 - e_2)(8 - e_1 - e_2)} = K_2} \end{array}\right\} \text{Sistem persamaan tak linier}$$

Berikut ini pemrograman MATLAB-nya.

```
function y = KsT(e,K1,K2)
%Sistem Pers.tak linier yang akan dinolkan
y = [(e(1)-e(2))*(3*e(1)-e(2))^3 /((2-e(1))*(8-e(1)...
      - e(2))*(10+2*e(1))^2) - K1
       e(2)*(3*e(1)+e(2)) / ((e(1)-e(2))*(8-e(1)-e(2))) - K2];
```

```
clear
clc
K1 = input('Masukan konstanta kst. reaksi 1 = ');
K2 = input('Masukan konstanta kst. reaksi 2 = ');

%Pencari nol fungsi KsT.m
e  = fsolve(@(e) KsT(e,K1,K2),[1 0.5])
```

Eksekusi di MATLAB *command window*

```
>>kasus5
Masukan harga konstanta kst. reaksi 1 = 1.93e-4
Masukan harga konstanta kst. reaksi 2 = 5.528
Optimization terminated: first-order optimality is less than options.TolFun.

e =

          0.7480    0.6920
```

Tugas 5

Menyelesaikan sistem persamaan tak linier dengan menggunakan subrutin MATLAB

Suatu reaksi elementer A $\rightarrow$ B + C berlangsung dalam sebuah reaktor tangki berpengaduk kontinu. Laju umpan murni A, 12 mol/s pada temperatur 25 ^{o}C. Reaksi bersifat eksotermik, untuk itu digunakan air pendingin bertemperatur 50 ^{o}C untuk menyerap kalor yang dibebaskan reaksi. Asumsi konstanta kapasitas panas sama baik di sisi reaktan maupun produk, neraca energi untuk sistem ini dirumuskan sebagai berikut:

$$-F_{Ao} X \Delta H_R = F_{Ao} C_{P,A}(T - T_0) + UA(T - T_a)$$

FA0 = laju molar umpan, mol/s.

X = konversi

ΔHR = Kalor reaksi, J/(mol.K)

$C_{P,A}$ = kapasitas panas A, J/(mol.K)

T = temperatur reaktor, ^{o}C

T_0 = temperatur referensi, 25 ^oC

Ta = temperatur air pendingin, ^oC

U = koefisien pindah panas total, W/(m^2.K)

A = luas pindah panas, m^2

Untuk reaksi orde pertama konversi dirumuskan sebagai berikut:

$$X = \frac{\tau k}{1+\tau k}$$

Dengan τ adalah waktu tinggal dalam sekon, dan k adalah laju reaksi spesifik dalam s^{-1} dihitung dengan menggunakan persamaan Arrhenius:

$$k = 650\exp[-3800/(T+273)]$$

Hitunglah harga temperatur reaktor dan konversinya!.

(ΔH_R=-1500 kJ/mol; τ =10 s; CP,A = 4500 J/(mol.K); UA/FA0 =700 W.s/(mol.K).

________________________________o0o________________________________

Bab 4
Regresi Linier dan Non Linier

Kasus

Suatu reaksi berorde n memiliki laju reaksi sbb:

$$r = kC_A^n$$

Apabila volume reaktor partaian (*batch*) konstan, persamaan laju reaksi menjadi sbb:

$$\frac{dC_A}{dt} = -kC_A^n$$

Tentukan orde laju reaksi tersebut jika diketahui data-data eksperimen sbb:

Tabel 4.2 Konsentrasi, waktu, dan laju perubahannya

Waktu (s)	C_A (mol/liter)	dC_A/dt (mol/liter.s)
0	1.0	-0.10000
10	0.50	-0.02500
20	0.33	-0.01110
30	0.25	-0.00625
40	0.20	-0.00400

Jawaban:

$$\frac{dC_A}{dt} = -kC_A^n$$

$$\ln\left(-\frac{dC_A}{dt}\right) = n\ln C_A + \ln k$$

```
%linierisasi
t = [0:10:40];  % waktu
CA = [1,0.50,0.33,0.25,0.20];   %konsentrasi A
dCAdt = -[0.1,0.025,0.0111,0.00625,0.00400]; %Laju
y = log(-dCAdt);
x = log(CA);

[P S] = polyfit(x,y,1);
```

```
n = P(1)          %orde reaksi
k = exp(P(2))     %konstanta laju reaksi
```

Hasil pada command window

```
n =
    1.9982
k =
    0.1002
```

4.3 Regresi Linier Peubah Banyak

$$y(x_1, x_2) = a_0 x_1 + a_1 x_2 + a_2$$

$$S = \sum_{i=1}^{n} (y_i - a_0 x_{1,i} - a_1 x_{2,i} - a_2)^2$$

$$\frac{\partial S}{\partial a_0} = -2\sum_{i=1}^{n} x_{1,i}(y_i - a_0 x_{1,i} - a_1 x_{2,i} - a_2) = 0$$

$$\sum_{i=1}^{n} (x_{1,i} y_i - x_{1,i}^2 a_0 - x_{1,i} x_{2,i} a_1 - x_{1,i} a_2) = 0$$

$$\boxed{\sum_{i=1}^{n} x_{1,i}^2 a_0 + \sum_{i=1}^{n} x_{1,i} x_{2,i} a_1 + \sum_{i=1}^{n} x_{1,i} a_2 = \sum_{i=1}^{n} x_{1,i} y_i}$$

$$\frac{\partial S}{\partial a_1} = -2\sum_{i=1}^{n} x_{2,i}(y_i - a_0 x_{1,i} - a_1 x_{2,i} - a_2) = 0$$

$$\sum_{i=1}^{n} (x_{2,i} y_i - x_{1,i} x_{2,i} a_0 - x_{2,i}^2 a_1 - x_{2,i} a_2) = 0$$

$$\boxed{\sum_{i=1}^{n} x_{1,i} x_{2,i} a_0 + \sum_{i=1}^{n} x_{2,i}^2 a_1 + \sum_{i=1}^{n} x_{2,i} a_2 = \sum_{i=1}^{n} x_{2,i} y_i}$$

$$\frac{\partial S}{\partial a_2} = -2\sum_{i=1}^{n} (y_i - a_0 x_{1,i} - a_1 x_{2,i} - a_2) = 0$$

$$\boxed{\sum_{i=1}^{n} x_{1,i} a_0 + \sum_{i=1}^{n} x_{2,i} a_1 + n a_2 = \sum_{i=1}^{n} y_i}$$

Ketiga buah persamaan linier tersebut dapat dijelmakan dalam matrik sbb:

$$\begin{bmatrix} \sum_{i=1}^{n} x_{1,i}^2 & \sum_{i=1}^{n} x_{1,i}x_{2,i} & \sum_{i=1}^{n} x_{1,i} \\ \sum_{i=1}^{n} x_{1,i}x_{2,i} & \sum_{i=1}^{n} x_{2,i}^2 & \sum_{i=1}^{n} x_{2,i} \\ \sum_{i=1}^{n} x_{1,i} & \sum_{i=1}^{n} x_{2,i} & n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} x_{1,i}y_i \\ \sum_{i=1}^{n} x_{2,i}y_i \\ \sum_{i=1}^{n} y_i \end{bmatrix}$$

Kasus

Perhatikan data-data reaksi non-isotermal suatu reaksi reversibel berikut ini:

$$A \xrightarrow{r} P$$

Tabel 4.3 Data laju reaksi

C_A(mol/liter)	**Temperatur (K)**	**Laju reaksi (mol/liter.s)**
1	373	1.508
0.023	395	2.936
1.15	365	1.293
0.87	400	3.242
1.05	405	4.566
0.75	388	1.899
0.55	410	2.780
0.65	380	1.255

Anggap reaksi tersebut memenuhi model persamaan laju sbb:

$$r = k_0 C^n \exp\left(\frac{-E}{RT}\right)$$

Perkirakan harga k_0, E dan n dari data-data yang tersedia.

Jawaban.

$$\ln r = n \ln C + E\left(\frac{-1}{RT}\right) + \ln k_0$$

$$y_i = \ln r_i \qquad x_{1,i} = \ln C_i \qquad x_{2,i} = -\frac{1}{RT_i}$$

```
% multivariabel
clear
clc
CA = [1,0.023,1.15,0.87,1.05,0.75,0.55,0.65];     %mol/liter
T = [373,395,365,400,405,388,410,380];            %K
r = [1.508,2.936,1.293,3.242,4.566,1.899,2.780,1.255];%mol/liter.s

y = log(r);
x1 = log(CA);
x2 =-1./(0.082*T);

A = [sum(x1.^2),sum(x1.*x2), sum(x1)
    sum(x1.*x2),sum(x2.^2),sum(x2)
    sum(x1),sum(x2),length(r)];
c = [sum(x1.*y)
    sum(x2.*y)
    sum(y)];
a = A\c
```

```
Hasil pada Command window
a =
   -0.0108
  320.9052
   10.8467
```

3.4 Regresi Non Linier

Pada bagian sebelumnya kita telah mempelajari regresi persamaan tak linier dengan terlebih dahulu melakukan linierisasi. Namun tidak semua persamaan tak linier dapat memberikan parameter yang akurat dengan linierisasi. Pada bagian ini kita akan mempelajari regresi persamaan tak linier sehingga kita tidak lagi harus melinierisasikan persamaan tak linier. Perhatikan fungsi tak linier sbb:

$$y = \exp\left(a_0 + \frac{a_1}{(x + a_2)}\right)$$

Persamaan Antoine

a_0, a_1, dan a_2 merupakan parameter.

$$S = \sum_{i=1}^{n}\left(y_i - \exp\left(a_0 + \frac{a_1}{(x_i + a_2)} \right)\right)^2$$

Turunan parsial terhadap a_0

$$\frac{\partial S}{\partial a_0} = -2\sum_{i=1}^{n}\left(y_i - \exp\left(a_0 + \frac{a_1}{(x_i + a_2)} \right)\right)\exp\left(a_0 + \frac{a_1}{(x_i + a_2)} \right) = 0$$

Turunan parsial terhadap a_1

$$\frac{\partial S}{\partial a_1} = -2\sum_{i=1}^{n}\left(\left(y_i - \exp\left(a_0 + \frac{a_1}{(x_i + a_2)} \right)\right)\exp\left(a_0 + \frac{a_1}{(x_i + a_2)} \right)\left(\frac{1}{x_i + a_2} \right)\right) = 0$$

Turunan parsial terhadap a_2

$$\frac{\partial S}{\partial a_2} = -2\sum_{i=1}^{n}\left(\left(y_i - \exp\left(a_0 + \frac{a_1}{(x_i + a_2)} \right)\right)\exp\left(a_0 + \frac{a_1}{x_i + a_2} \right)(-a_1 (x_i + a_2)^{-2})\right) = 0$$

Pada akhirnya diperoleh sistem persamaan tak linier yang terdiri atas 3 buah persamaan tak linier. Sistem persamaan tak linier dapat diselesaikan secara simultan menggunakan metode Newton seperti yang telah dibahas pada bab 3 persamaan tak linier.

3.5 Subrutin MATLAB: nlinfit

[beta,R] = nlinfit(x,y,modelfun,beta0)

Kasus

Tabel 3.3 Tekanan uap dari Benzena (Perry)

Temperatur, T °C	**Tekanan, P (mmHg)**
-36.7	1
-19.6	5
-11.5	10
-2.6	20
7.6	40
15.4	60
26.1	100
42.2	200
60.6	400
80.1	760

Persamaan polynomial

$$P(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + ... + a_nx^n$$

Persamaan Clapeyron

$$\log(P) = A + \frac{B}{T}$$

Persamaan Riedel

$$\log(P) = A + \frac{B}{T} + C\log(T) + DT^{\beta}$$

dengan harga $\beta = 2$.

a. Korelasikan data dengan berbagai orde persamaan polynomial dengan menganggap temperatur absolut (Kelvin) adalah variabel bebas dan P adalah variabel terikat.
b. Korelasikan data dengan menggunakan persamaan Clapeyron
c. Korelasikan data menggunakan persamaan Riedel
d. Diskusikan persamaan manakah yang terbaik mewakili data-data eksperimental tersebut.

Jawab:

a. Polynom orde 2, 3, 4, 5

```
%polynom
clear
clc
T =[-36.7,-19.6,-11.5,-2.6,7.6,15.4,26.1,42.2,60.6,80.1]+273;%K
P = [1 5 10 20 40 60 100 200 400 760];%mmHg
N =length(P);
P2 = polyfit(T,P,2)
Pmod2 = polyval(P2,T); R2 = Pmod2-P; Var2 = sum(R2.^2)/(N-2)
P3 = polyfit(T,P,3)
Pmod3 = polyval(P3,T); R3 = Pmod2-P; Var3 = sum(R3.^2)/(N-3)
P4 = polyfit(T,P,4)
Pmod4 = polyval(P4,T); R4 = Pmod2-P; Var4 = sum(R4.^2)/(N-4)
P5 = polyfit(T,P,5)
Pmod5 = polyval(P5,T); R5 = Pmod2-P; Var5 = sum(R5.^2)/(N-5)
```

Hasil di Command Window

```
P2 =

  1.0e+003 *

    0.0001   -0.0450    5.8560

Var2 =
```

```
  1.0647e+003
Warning: Polynomial is badly conditioned. Remove repeated data
points
         or try centering and scaling as described in HELP
POLYFIT.
> In polyfit at 81
  In polinom at 9
P3 =
  1.0e+004 *
    0.0000   -0.0001    0.0146   -1.2519
Var3 =
  1.2168e+003
Warning: Polynomial is badly conditioned. Remove repeated data
points
         or try centering and scaling as described in HELP
POLYFIT.
> In polyfit at 81
  In polinom at 11
P4 =
  1.0e+004 *
    0.0000   -0.0000    0.0001   -0.0248    1.5881
Var4 =
  1.4196e+003
Warning: Polynomial is badly conditioned. Remove repeated data
points
         or try centering and scaling as described in HELP
POLYFIT.
> In polyfit at 81
  In polinom at 13

P5 =
  1.0e+004 *
   -0.0000    0.0000   -0.0000    0.0002   -0.0339    2.1109
Var5 =
  1.7035e+003
```

Tugas

Nomor 1

Harga viskositas air (centipoise) telah diukur pada berbagai temperatur. Hasil dari eksperimen disajikan dibawah ini. Menggunakan regresi linier ganda (*multiple regresi linier*), carilah konstanta-konstanta yang sesuai dengan persamaan berikut:

$$\frac{1}{\mu} = k_1 + k_2 T + k_3 T^2$$

T(^oC)	10	20	30	40	50	60	70
μ (cp)	1.308	1.005	0.801	0.656	0.549	0.469	0.406

Nomor 2

Sebuah reaksi heterogen diketahui terjadi pada laju yang dapat digambarkan oleh model Langmuir-Hinshelwood berikut ini:

$$r = \frac{k_1 P_A}{(1 + K_A P_A + K_R P_R)^2}$$

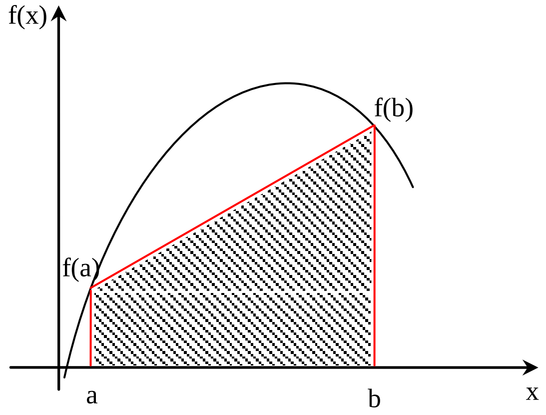
Dari pengukuran laju awal, k1 ditentukan sebagai 0.015 mol/s.g-cat.atm, pada 400 K. Dengan menggunakan data laju reaksi pada 400 K, perkirakan nilai dari K_A dan K_R.

P_A	1	0.9	0.8	0.7	0.6	0.5	0.4
P_R	0	0.1	0.2	0.3	0.4	0.5	0.6
r	$3.4x10^{-5}$	$3.6x10^{-5}$	$3.7x10^{-5}$	$3.9x10^{-5}$	$4.0x10^{-5}$	$4.1x10^{-5}$	$4.2x10^{-5}$

________________________________o0o________________________________

Bab 5
Integrasi
(Under construction)

5.1 Metode Trapesium



Gambar 5.1 Metode trapesium

5.1 Subrutin MATLAB: trapz

Subrutin **trapz** menghitung harga integral dari nilai-nilai diskrit x dan y dengan menggunakan metode Trapezoidal.

Z = trapz(x,y)

Keterangan:

x dan **y** adalah vektor

Kasus 1

Dua buah besaran yang sangat penting dalam pembelajaran proses-proses fermentasi adalan laju pembebasan CO_2 dan laju pengambilan O_2. Hal tersebut dihitung dari analisis eksperimental dari gas masuk dan gas keluar fermentor, dan laju alir, temperatur dan tekanan dari gas-gas ini. Rasio pembebasan CO_2 terhadap pengambilan O_2 menghasilkan RQ (Respiratory Quotient) yang merupakan barometer aktivitas metabolik dari mikroorganisme. Laju di atas dapat dintegrasikan untuk memperoleh jumlah keseluruhan dari CO_2 yang diproduksi dan oksigen yang dikonsumsi selama fermentasi. Tabel berikut menunjukan laju respirasi dihtung dari fermentasi *Penicillium chrysogenum* yang menghasilkan antibiotik penicilin.

Tabel 5.1 Laju pembebasan CO_2 dan laju pengambilan O_2

Waktu fermentasi (jam)	Laju Pembebasan CO_2 (g/jam)	Laju pengambilan O_2 (g/jam)
140	15.72	15.49
141	15.53	16.16
142	15.19	15.35
143	16.56	15.13
144	16.21	14.20
145	17.39	14.23
146	17.36	14.29
147	17.42	12.74
148	17.60	14.74
149	17.75	13.68
150	18.95	14.51

Hitunglah jumlah keseluruhan CO_2 yang dihasilkan dan Oksigen yang dikonsumsi selama fermentasi berlangsung.

```
%Respiratory_Quotient
clear
clc

t = [140:150];          % Waktu fermentasi
dCO2dt = [15.72,15.53,15.19,16.56,16.21,17.39,17.36,17.42,...
    17.60,17.75,18.95];  % Laju pembebasan CO2
dO2dt = [15.49,16.16,15.35,15.13,14.20,14.23,14.29,12.74,...
    14.74,13.68,14.51];  % Laju pengambilan O2

CO2 = trapz(t,dCO2dt)
O2 = trapz(t,dO2dt)
```

```
RQ = CO2/O2
```

5.2 Subrutin MATLAB: quad

Q = quad(fungsi,A,B)

Kasus 2

Harga kapasitas panas suatu material dapat dievaluasi dengan menggunakan persamaan sbb:

$$c_p(T) = 0.132 + 1.56\text{x}10^{-4}T + 2.64\text{x}10^{-7}T^2 \quad \text{cal/g}^o\text{C}$$

Hitunglah besar entalpi material sebanyak 1 gram pada rentang temperatur -100 ^{o}C s.d 200 ^{o}C dengan rumus sbb: $\Delta H = m\int_{T_1}^{T_2} c(T)dT$

```
%kapasitas.m
function cp = kapasitas(T)

cp=0.132+1.56e-4.*T+2.64e-7*T.^2;
```

```
%runkapasitas.m
Q = quad('kapasitas',-100,200)
```

```
>> runkapasitas
Q =
```

42.7320

5.3 Subrutin MATLAB: dblquad

Subrutin dblquad digunakan untuk menghitung integral lipat dua.

q = dblquad(fun,xmin,xmax,ymin,ymax)

```
%integrnd.m
function z = integrnd(x, y)
z = y*sin(x)+x*cos(y);
```

```
%run_integrnd.m
Q = dblquad(@integrnd,pi,2*pi,0,pi);
```

5.4 Subrutin MATLAB: triplequad

Subrutin triplequad digunakan untuk menghitung integral lipat tiga.

triplequad(fun,xmin,xmax,ymin,ymax,zmin,zmax)

```
%intgrnd3
function f = integrnd3(x,y,z)
f = y*sin(x)+z*cos(x);
```

```
%run_integrnd3
Q = triplequad('integrnd3', 0, pi, 0, 1, -1, 1)
```

Tugas

Nomor 1

Lakukan komputasi yang sama seperti pada kasus 1, namun dengan massa material 2000 gram dan temperatur -200 s.d 100 ^{o}C.

o0o

Bab 6
Persamaan Diferensial Biasa (PDB)

Definisi PDB

Persamaan diferensial biasa adalah persamaan diferensial yang terdiri atas fungsi turunan satu buah variabel bebas.

Contoh:

Persamaan gaya geser (*shear stress*) pada aliran fluida dirumuskan sbb.

$$\boxed{\frac{d\tau_{xz}}{dx} = \rho g}$$

Perhatikan PDB hanya memiliki satu buah variabel bebas yaitu *x* dan satu variabel terikat yaitu τ_{xz}.

Aplikasi PDB

PDB banyak ditemukan pada pemodelan-pemodelan teknik reaktor, kinetika reaksi kimia, peristiwa-peristiwa perpindahan dll.

Klasifikasi PDB

Berdasarkan ordenya PDB terdiri atas tiga jenis (paling umum ditemukan dalam permasalahan teknik kimia).

Orde 1 $\quad \frac{dy}{dx} + y = kx$

Orde 2 $\quad \frac{d^2y}{dx^2} + y\frac{dy}{dx} = kx$

Orde 3 $\quad \frac{d^3y}{dx^3} + a\frac{d^2y}{dx^2} + b\left(\frac{dy}{dx}\right)^2 = kx$

Berdasarkan kondisi batasnya PDB terdiri atas dua jenis.

1. PDB bernilai awal

$$\frac{\partial^2 y}{\partial x^2} = -yx$$

$$\underbrace{y(0) = 2, \quad \frac{\partial y}{\partial x}(0)}_{\text{harga } x \text{ sama}} = 1$$

2. PDB bernilai batas

$$\frac{\partial^2 y}{\partial x^2} = -yx$$

$$\underbrace{y(0) = 2, \quad y(1)}_{\text{harga } x \text{ berbeda}} = 1$$

Transformasi ke Dalam Bentuk Kanonikal

Persamaan diferensial biasa linier orde 1 bernilai awal dapat diselesaikan dengan menggunakan metode matrik eksponensial dan metode eigen yang akan dibahas di depan nanti. PDB linier orde 2, 3 bernilai awal dapat pula diselesaikan dengan metode-metode tersebut, asalkan PDB tersebut ditransformsikan terlebih dahulu ke dalam PDB orde 1. Berikut ini penjelasan teknik transformasi dari PDB berorde tinggi menjadi PDB berorde 1.

Contoh 1:

$$\boxed{\frac{d^4 z}{dt^4} + 5\frac{d^3 z}{dt^3} - 2\frac{d^2 z}{dt^2} - 6\frac{dz}{dt} + 3z = 0}$$

Transformasi PDB orde 4 linier tersebut akan menghasilkan 4 buah PDB linier orde 1.

Misalkan:

$$z = y_1$$
$$\frac{dz}{dt} = \frac{dy_1}{dt} = y_2$$
$$\frac{d^2z}{dt^2} = \frac{dy_2}{dt} = y_3$$
$$\frac{d^3z}{dt^3} = \frac{dy_3}{dt} = y_4$$
$$\frac{d^4z}{dt^4} = \frac{dy_4}{dt}$$

Maka PDB orde 4 dapat dituliskan sbb:

$$\frac{dy_1}{dt} = y_2$$
$$\frac{dy_2}{dt} = y_3$$
$$\frac{dy_3}{dt} = y_4$$
$$\frac{dy_4}{dt} = -3y_1 + 6y_2 + 2y_3 - 5y_4$$

Penulisan dalam bentuk matrik sbb:

$$\begin{bmatrix} \frac{dy_1}{dt} \\ \frac{dy_2}{dt} \\ \frac{dy_3}{dt} \\ \frac{dy_4}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 6 & 2 & -5 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

Contoh 2:

$$\boxed{\frac{d^4z}{dt^4} + 5\frac{d^3z}{dt^3} - 2\frac{d^2z}{dt^2} - 6\frac{dz}{dt} + 3z = e^{-t}}$$

Transformasi PDB orde 4 linier tersebut akan menghasilkan 5 buah PDB linier orde 1.

Misalkan:

$$z = y_1$$
$$\frac{dz}{dt} = \frac{dy_1}{dt} = y_2$$
$$\frac{d^2z}{dt^2} = \frac{dy_2}{dt} = y_3$$
$$\frac{d^3z}{dt^3} = \frac{dy_3}{dt} = y_4$$
$$\frac{d^4z}{dt^4} = \frac{dy_4}{dt}$$
$$y_5 = e^{-t}$$
$$\frac{dy_5}{dt} = -e^{-t} = -y_5$$

Maka PDB orde 4 dapat dituliskan sbb:

$$\frac{dy_1}{dt} = y_2$$
$$\frac{dy_2}{dt} = y_3$$
$$\frac{dy_3}{dt} = y_4$$
$$\frac{dy_4}{dt} = -3y_1 + 6y_2 + 2y_3 - 5y_4 + y_5$$
$$\frac{dy_5}{dt} = -y_5$$

Penulisan dalam bentuk matrik sbb:

$$\begin{bmatrix} \frac{dy_1}{dt} \\ \frac{dy_2}{dt} \\ \frac{dy_3}{dt} \\ \frac{dy_4}{dt} \\ \frac{dy_5}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -3 & 6 & 2 & -5 & 1 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}$$

Contoh 3:

$$\boxed{\frac{d^3z}{dx^3} + z^2\frac{d^2z}{dx^2} - \left(\frac{dz}{dx}\right)^3 - 2z = 0}$$

Transformasi PDB orde 3 taklinier.

Misalkan:

$$z = y_1$$

$$\frac{dz}{dx} = \frac{dy_1}{dx} = y_2$$

$$\frac{d^2z}{dx^2} = \frac{dy_2}{dx} = y_3$$

$$\frac{d^3z}{dx^3} = \frac{dy_3}{dx}$$

Maka PDB orde 3 taklinier dituliskan sbb:

$$\frac{dy_1}{dx} = y_2$$

$$\frac{dy_2}{dx} = y_3$$

$$\frac{dy_3}{dx} = 2y_1 - y_1^2 y_3 + y_2^3$$

PDB taklinier tidak dapat dituliskan dalam bentuk matrik.

Contoh 4:

$$\boxed{\frac{d^3z}{dt^3} + t^3\frac{d^2z}{dt^2} - t^2\frac{dz}{dt} + 5z = 0}$$

Transformasi PDB orde 3 taklinier.

Misalkan:

$$z = y_1$$

$$\frac{dz}{dt} = \frac{dy_1}{dt} = y_2$$

$$\frac{d^2z}{dt^2} = \frac{dy_2}{dt} = y_3$$

$$\frac{d^3z}{dt^3} = \frac{dy_3}{dt}$$

$$y_4 = t$$

$$\frac{dy_4}{dt} = 1$$

Maka PDB orde 3 taklinier dituliskan sbb:

$$\frac{dy_1}{dt} = y_2$$

$$\frac{dy_2}{dt} = y_3$$

$$\frac{dy_3}{dt} = -5y_1 + y_4^2 y_2 - y_4^3 y_3$$

$$\frac{dy_4}{dt} = 1$$

PDB taklinier tidak dapat dituliskan dalam bentuk matrik.

Nilai dan Vektor Eigen

$$Aw^{[k]} = \lambda_k w^{[k]}$$
$$Aw^{[k]} - \lambda_k w^{[k]} = 0$$
$$\left(A - \lambda_k I\right) w^{[k]} = 0$$
$$\left(A - \lambda_k I\right) = 0 \text{ atau } w^{[k]} \neq 0$$

Vektor eigen tidak bernilai nol

$$\det\left(A - \lambda_k I\right) = 0$$

Keterangan:

A adalah sebuah matrik kubus

λ_k adalah nilai eigen

$w^{[k]}$ adalah vektor eigen

Berikut ini akan dipaparkan cara menghitung nilai dan vektor eigen secara analitik.

Kasus 6

Tentukanlah vektor dan nilai eigen dari matrik A berikut ini dengan menggunakan cara analitik.

$$A = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 2 & 1 \\ 3 & 1 & -1 \end{bmatrix}$$

Jawaban:

$$\left(A - \lambda_k I\right) w^{[k]} = 0$$
$$\det\left(A - \lambda_k I\right) = 0$$

$$\begin{vmatrix} (1-\lambda) & 0 & 3 \\ 0 & (2-\lambda) & 1 \\ 3 & -1 & (-1-\lambda) \end{vmatrix} = 0$$

$$\left|\begin{array}{ccc|cc} (1-\lambda) & 0 & 3 & (1-\lambda) & 0 \\ 0 & (2-\lambda) & 1 & 0 & (2-\lambda) \\ 3 & -1 & (-1-\lambda) & 3 & -1 \end{array}\right. = 0$$

$$(1-\lambda)(2-\lambda)(-1-\lambda)-(3)(2-\lambda)(3)-(1-\lambda)=0$$

$$(1-\lambda^2)(\lambda-2)-9(2-\lambda)-(1-\lambda)=0$$

$$\lambda-2-\lambda^3+2\lambda^2-18+9\lambda-1+\lambda=0$$

$$-\lambda^3+2\lambda^2+11\lambda-21=0$$

Dengan menggunakan subrutin roots MATLAB diperoleh harga akar-akar polinom pangkat 3 (nilai eigen) tersebut, yaitu:

$\lambda_1=3.4211 \quad \lambda_2=-3.2880 \quad \lambda_3=1.8669$

Kembali ke persamaan awal.

$$\left(A-\lambda_k I\right)w^{[k]}=0$$

$$\begin{bmatrix} (1-\lambda) & 0 & 3 \\ 0 & (2-\lambda) & 1 \\ 3 & 1 & (-1-\lambda) \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}=0$$

Karena vektor eigen (w) tidak bernilai nol, maka kita misalkan harga w_3 sebagai basis bernilai 1.

$$(1-\lambda)w_1+3w_3=0$$

$$(2-\lambda)w_2+w_3=0$$

Misalkan w_3 = 1, maka system persamaan linier menjadi

$$\begin{aligned}(1-\lambda)w_1&=-3\\(2-\lambda)w_2&=-1\end{aligned} \quad\Longrightarrow\quad \begin{aligned}w_1&=\frac{-3}{(1-\lambda)}\\w_2&=\frac{-1}{(2-\lambda)}\\w_3&=1\end{aligned}$$

Masukan harga nilai eigen

Untuk:

$\lambda_1=3.4211$	$\lambda_2=-3.2880$	$\lambda_3=1.8669$
$w^{[1]}=\begin{bmatrix}1.2391\\0.7037\\1\end{bmatrix}$	$w^{[2]}=\begin{bmatrix}-0.6996\\-0.1891\\1\end{bmatrix}$	$w^{[3]}=\begin{bmatrix}3.4607\\-7.5131\\1\end{bmatrix}$

Normalisasi vektor-vektor eigen tersebut dengan menggunakan norma ke-2.

$$\left\| w^{[1]} \right\|_2 = \sqrt{\left(1.2391^2 + 0.7037^2 + 1^2\right)} = 1.741$$

$$w^{[1]} = \begin{bmatrix} 1.2391 / 1.741 \\ 0.7037 / 1.741 \\ 1 / 1.741 \end{bmatrix} = \begin{bmatrix} 0.7117 \\ 0.4042 \\ 0.5744 \end{bmatrix}$$

$$\left\| w^{[2]} \right\|_2 = \sqrt{\left(0.6996^2 + 0.1891^2 + 1^2\right)} = 1.235$$

$$w^{[2]} = \begin{bmatrix} -0.6996 / 1.235 \\ -0.1891 / 1.235 \\ 1 / 1.235 \end{bmatrix} = \begin{bmatrix} -0.5665 \\ -0.1531 \\ 0.8097 \end{bmatrix}$$

$$\left\| w^{[3]} \right\|_2 = \sqrt{\left(3.4607^2 + 7.5131^2 + 1^2\right)} = 8.332$$

$$w^{[3]} = \begin{bmatrix} 3.4607 / 8.332 \\ -7.5131 / 8.332 \\ 1 / 8.332 \end{bmatrix} = \begin{bmatrix} 0.4153 \\ -0.9017 \\ 0.1200 \end{bmatrix}$$

Jadi nilai dan vektor eigen matrik A adalah

$$\lambda = \begin{bmatrix} 3.4211 \\ -3.2880 \\ 1.8669 \end{bmatrix} \qquad w = \begin{bmatrix} 0.7117 & -0.5665 & 0.4153 \\ 0.4042 & -0.1531 & -0.9017 \\ 0.5744 & 0.8097 & 0.1200 \end{bmatrix}$$

Catatan: Perkalian konstanta dengan vektor eigen tidak akan mengubah esensi dari vektor eigen tersebut. Untuk persoalan ini harga vektor eigen yang diperoleh menggunakan MATLAB (sekejap lagi akan dibahas) adalah hasil perkalian antara -1 dengan vektor eigen yang telah diperoleh pada perhitungan secara analitik.

MATLAB telah menyediakan rutin untuk menghitung nilai dan vektor eigen matriks A yaitu *eig*.

Penulisan perintahnya pada MATLAB *command window* sbb:

$$[V, D] = eig(A)$$

Vektor eigen

Nilai eigen

Sebagai contoh berikut ini akan ditampilkan perintah pada *command window* untuk menghitung nilai dan vektor eigen dari matrik A yang telah diselesaikan secara analitik sebelumnya.

```
>> [V,D]=eig(A)

V =

   -0.5665   -0.4153   -0.7118
   -0.1531    0.9018   -0.4042
    0.8097   -0.1200   -0.5744

D =

   -3.2880         0         0
         0    1.8669         0
         0         0    3.4211
```

Tugas 6

Transformasi kanonikal PDB dan analisis eigen

Nomor 1

Hitunglah nilai dan vektor eigen dari matrik A berikut ini. Bandingkan hasilnya dengan menggunakan subrutin *eig* di MATLAB.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 1 \\ 3 & 1 & 4 \end{bmatrix}$$

Nomor 2

Ubahlah persamaan differensial berikut ke dalam bentuk kanonikal.

a. $\frac{d^2x}{dt^2} - 3\frac{dx}{dt} - 10x = 0$

b. $\frac{d^3T}{dt^3} + t^3\frac{d^2T}{dt^2} - t^2\frac{dT}{dt} - 10T = 0$

c. $\frac{d^3y}{dx^3} - y^3\frac{d^2y}{dx^2} - \left(\frac{dy}{dx}\right)^2 + 9y = 0$

Solusi Persamaan Differensial Biasa Linier bernilai awal

1. Metode matriks eksponensial

$$\frac{dy}{dt} = Ay \quad \Rightarrow \quad y(0) = y_0$$

A adalah matriks persegi (m x m) dan y adalah vektor kolom (m x 1)

Integrasikan persamaan diferensial linier tersebut.

$$\int_{y_0}^{y} \frac{dy}{y} = A\int_0^t dt$$

$$\ln\frac{y}{y_0} = At$$

$$\boxed{y = e^{At}\, y_0}$$

Fungsi matriks eksponensial dapat dituliskan sebagai berikut:

$$\boxed{\exp(\mathbf{A}t) = \mathbf{I} + \mathbf{A}t + \frac{\mathbf{A}^2t^2}{2!} + \frac{\mathbf{A}^3t^3}{3!} + \ldots}$$

Contoh soal:

Kasus 7

Berikut ini adalah PDB linier orde 2.

$$\frac{d^2x}{dt^2} - 3\frac{dx}{dt} - 10x = 0$$

Dengan nilai awal pada t = 0, sbb:

$$x(0) = 3$$

$$\frac{dx}{dt}\Big|_0 = 15$$

Selesaikan PDB tercetak menggunakan metode matrik eksponensial dalam interval $0 \le t \le 1.0$ (Langkah integrasi 0.1).

Jawaban :

$$\frac{d^2x}{dt^2} - 3\frac{dx}{dt} - 10x = 0$$

$$x = y_1$$

$$\frac{dx}{dt} = \frac{dy_1}{dt} = y_2$$

$$\frac{d^2x}{dt^2} = \frac{dy_2}{dt} = 3y_2 + y_1$$

$$\begin{bmatrix} \frac{dy_1}{dt} \\ \frac{dy_2}{dt} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

$$\boxed{\frac{dy}{dt}} \qquad \boxed{\mathrm{A}} \quad \boxed{\mathrm{y}}$$

$$\frac{dy}{dt} = Ay$$

Integrasikan.

$$y = \mathrm{e}^{At}\, y_0 \qquad\qquad y_0 = \begin{bmatrix} 3 \\ 15 \end{bmatrix}$$

Rentang integrasi $0 \le t \le 1.0$. Langkah integrasi 0.1

t	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \left(\mathrm{e}^{\begin{bmatrix} 0 & t \\ t & 3t \end{bmatrix}} \right) \begin{bmatrix} 3 \\ 15 \end{bmatrix}$$

MATLAB &Pengantar Pemrograman

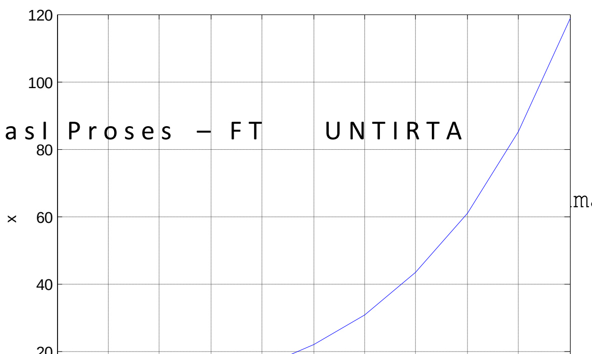
Dengan mensubstitusikan t = 0 s.d 1 (langkah integrasi 0.1) selesailah persoalan ini.

Berikut ini pemrograman MATLAB-nya.

kasus7.m

```
clear
clc

A = [0 1
    1 3];
% Nilai awal
yo = [3;15];
t = [0:0.1:1]';

for i = 1:length(t)
    y(i,:) = expm(A*t(i))*yo;
end

%kurva t-x
x = y(:,1)
plot(t,x)
xlabel('t')
ylabel('x')
grid on
```

```
>>kasus7a
t =

        0
   0.1000
   0.2000
   0.3000
   0.4000
   0.5000
   0.6000
   0.7000
   0.8000
   0.9000
   1.0000
x =

   3.0000
   4.7688
   7.2122
```



10.5945
15.2839
21.7922
30.8319
43.3941
60.8578
85.1416
118.9150

2. Metode nilai-vektor eigen

Harga $e^{\mathbf{A}t}$ dapat dihitung dengan menggunakan bantuan nilai dan vektor eigen.

$$e^{\mathbf{A}t} = \mathrm{V}e^{Dt}\mathrm{V}^{-1}$$

Sehingga solusi PDB linier menjadi.

$$\boxed{y = \left[\mathrm{V}e^{Dt}\mathrm{V}^{-1}\right]y_0}$$

Untuk lebih memahami metode nilai-vektor eigen berikut ini disajikan penyelesaian kasus 7 dengan menggunakan metode nilai-vektor eigen.

Langkah awal sama dengan metode matriks eksponensial.

$$A = \begin{bmatrix} 0 & 1 \\ 1 & 3 \end{bmatrix}$$

Dengan menggunakan rutin *eig* MATLAB diperoleh harga nilai (D) dan vektor eigen (V).

```
>> A=[0 1
1 3]
A =
    0    1
    1    3

>> [V,D]=eig(A)
V =
   -0.9571    0.2898
    0.2898    0.9571
```

$$V = \begin{bmatrix} -0.9571 & 0.2898 \\ 0.2898 & 0.9571 \end{bmatrix} \quad \text{dan} \quad D = \begin{bmatrix} -0.3028 & 0 \\ 0 & 3.3028 \end{bmatrix}$$

Substitusikan matriks V dan D ke dalam persamaan

$$\boxed{y = \left[\mathrm{V} e^{Dt} \mathrm{V}^{-1} \right] y_0}$$

$$\mathbf{y} = \left[\begin{bmatrix} -0.9571 & 0.2898 \\ 0.2898 & 0.9571 \end{bmatrix} \begin{bmatrix} e^{-0.3028t} & 0 \\ 0 & e^{0.3028t} \end{bmatrix} \begin{bmatrix} -0.9571 & 0.2898 \\ 0.2898 & 0.9571 \end{bmatrix}^{-1} \right] \begin{bmatrix} 3 \\ 15 \end{bmatrix}$$

Dengan mensubstitusikan $t = 0$ s.d 1 (langkah integrasi 0.1) selesailah persoalan ini.

Berikut ini pemrograman MATLAB-nya.

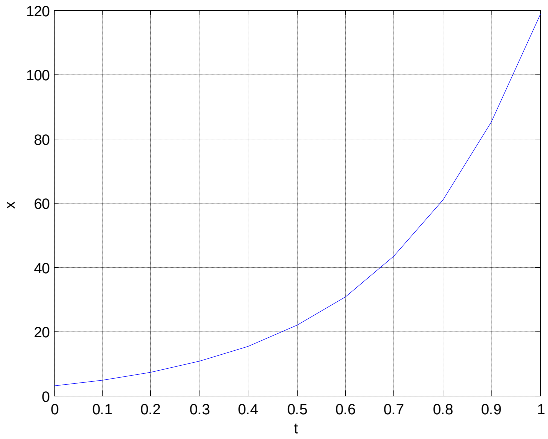
kasus7.m

```
clear
clc
A = [0 1
     1 3];
% Nilai awal
yo = [3;15];
a=length(yo);

% Vektor dan Nilai eigen
[V,D]=eig(A);
% Rentang integrasi
t=[0:0.1:1]'

x =zeros(length(t),a);
for i = 1 : length(t)
y = (V*diag(exp(diag(D)*t(i)))*inv(V))*yo;
x(i,:) = y;
end
x
% kurva t-x
plot(t,x(:,1))
xlabel('t')
ylabel('x')
grid on
```

Eksekusi program kasus7.m (lanjutan)
Hasil di *Command Window* :

```
>>kasus7

t =

     0
    0.1000
    0.2000
    0.3000
    0.4000
    0.5000
    0.6000
    0.7000
    0.8000
    0.9000
    1.0000
```

```
x =

    3.0000   15.0000
    4.7688   20.6902
    7.2122   28.6127
   10.5945   39.6409
   15.2839   54.9901
   21.7922   76.3512
   30.8319  106.0768
   43.3941  147.4403
   60.8578  204.9959
   85.1416  285.0806
  118.9150  396.5110
```



Tugas 7

Metode eigen untuk menyelesaikan sistem persamaan diferensial biasa linier

Suatu bahan radioaktif meluruh berdasarkan mekanisme reaksi berantai sbb:

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$

k_1 dan k_2 adalah konstanta laju reaksi. B adalah produk *intermediate* dan C adalah produk akhir. Persamaan laju reaksinya sbb:

$$\boxed{\begin{aligned} \frac{dC_A}{dt} &= -k_1 C_A \\ \frac{dC_B}{dt} &= k_1 C_A - k_2 C_B \\ \frac{dC_C}{dt} &= k_2 C_B \end{aligned}}$$

C_A, C_B, dan C_C adalah konsentrasi bahan A, B, dan C.

$k_1 = 3\ s^{-1}$, $k_2 = 1\ s^{-1}$.

Konsentrasi mula-mula bahan sbb:

$C_A(0)$=1 mol/m^3 $C_B(0)$=0 $C_C(0)$=0

a) Menggunakan metode matriks eksponensial dan metode eigen tentukan konsentrasi C_A, C_B, dan C_C sebagai fungsi waktu.

b) Hitunglah konsentrasi C_A, C_B, dan C_C saat t = 1 s dan t = 10 s.

c) Buatlah profil konsentrasi A, B, dan C.

Subrutin dalam MATLAB untuk solusi PDB bernilai awal

Pada bagian ini akan dijelaskan subrutin ode23 dalam MATLAB untuk menyelesaikan PDB bernilai awal dengan karakter linier, taklinier, tunggal maupun jamak (sistem).

Cara penulisan sintaks ode23

[t,y] = ode23('fungsiPDB',rentang_t,y0)

Fungsi PDB yang akan dievaluasi	Rentang integrasi	Harga awal

Misal:

$$\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$$ dengan kondisi awal y = 1 pada x = 0 dan rentang integrasi dari x = 0 s.d x = 4.

Berikut ini pemrograman MATLAB-nya.

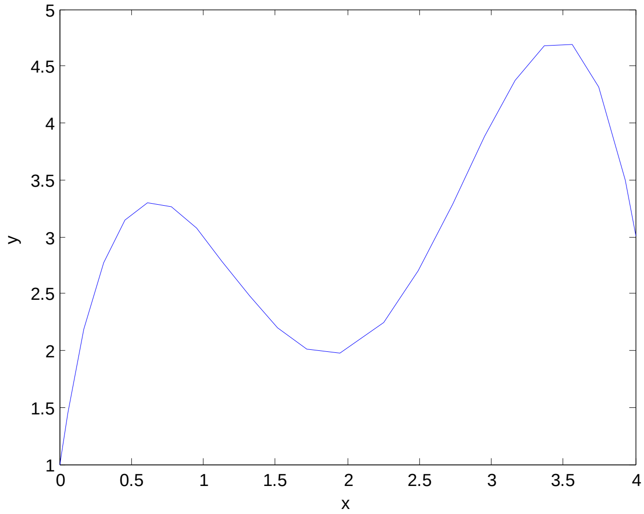
```
%pdb.m
function dydx = pdb(x,y)

dydx = -2*x^3+12*x^2-20*x+8.5;
```

```
%runpdb.m
clear
clc

rentang_x = [0 4];
y0 = 1;

[x,y] = ode23('pdb',rentang_x,y0)

plot(x,y)
xlabel('x')
ylabel('y')
```

Eksekusi di *command window*

```
x =

        0
   0.0094
   0.0565
   0.1712
   0.3046
   0.4524
   0.6111
   0.7777
   0.9511
   1.1319
   1.3196
   1.5150
   1.7217
   1.9512
   2.2503
   2.4902
   2.7301
   2.9516
   3.1622
   3.3655
   3.5614
   3.7483
   3.9277
   4.0000
```

```
y =

   1.0000
   1.0791
   1.4488
   2.1817
   2.7701
   3.1483
   3.3031
   3.2609
   3.0709
   2.7893
   2.4787
   2.2006
   2.0131
   1.9808
   2.2494
   2.6978
   3.2901
   3.8780
   4.3716
   4.6749
   4.6862
   4.3176
   3.4933
   3.0015
```



Kasus 8

Studi terhadap kinetika proses fermentasi berhasil dimodelkan secara matematis sbb:

$$\frac{dy_1}{dt} = k_1 y_1 \left(1 - \frac{y_1}{k_2}\right)$$

$$\frac{dy_2}{dt} = k_3 y_1 - k_4 y_2$$

Dengan k_1 = 0.03120; k_2 = 47.70; k_3 = 3.374 ;k_4 = 0.01268 serta nilai pada t = 0, y_1=5, y_2=0. Evaluasi harga y_1 dan y_2 dalam interval waktu 0 s.d 10 jam setiap jamnya!.

Berikut ini pemrograman MATLAB-nya.

```
%fermen.m
function dydt = fermen(t,y)
k1=0.03120;
k2=47.70;
k3=3.374;
k4=0.01268;

dydt=[k1*y(1)*(1-y(1)/k2)
        k3*y(1)-k4*y(2)];
```

```
%kasus8
clear
clc

tspan = [0:1:10];
yo = [5 0];

[t,y]=ode23('fermen',tspan,yo)
```

Eksekusi di *command window*

```
t =

     0
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
```

```
        y =

    5.0000         0
    5.1414   17.0000
    5.2863   34.2657
    5.4347   51.8056
    5.5868   69.6282
    5.7425   87.7422
    5.9020  106.1564
    6.0652  124.8796
    6.2323  143.9206
    6.4033  163.2886
    6.5783  182.9924
```

Tugas 8

Solusi PDB tak linier menggunakan subrutin MATLAB ode23

Nomor 1

Konversi glukosa menjadi asam glukonik merupakan reaksi oksidasi sederhana dari gugus aldehid gula menjadi gugus karboksil. Enzim glukosa oksidase, terbentuk dalam mikroorganisme untuk mengubah glukosa menjadi glukonolaktona. Kemudian glukonolaktona bereaksi dengan air membentuk asam glukonik. Mekanisme reaksi secara keseluruhan proses fermentasi dapat dituliskan sbb:

Pertumbuhan sel:

Glukosa + sel $\rightarrow$ sel

Oksidasi glukosa:

$$\text{Glukosa} + O_2 \xrightarrow{\text{Glukosa oksidase}} \text{Glukonolactona} + H_2O_2$$

Hidrolisis glukonolactona:

$$\text{Glukonolactona} + H_2O \rightarrow \text{Asam glukonik}$$

Dekomposisi peroksida:

$$H_2O_2 \xrightarrow{\text{Katalis}} H_2O + \frac{1}{2}O_2$$

Model matematika untuk fermentasi bakteri *Pseudomonas ovalis* yang memproduksi asam glukonik telah dirumuskan oleh Rai dan Constantinides. Model yang menggambarkan dinamika pertumbuhan fasa logaritmik ini dapat dituliskan sbb:

Laju pertumbuhan sel:

$$\frac{dy_1}{dt} = b_1 y_1 \left(1 - \frac{y_1}{b_2}\right)$$

Laju pembentukan glukonolaktona:

$$\frac{dy_2}{dt} = \frac{b_3 y_1 y_4}{b_4 + y_4} - 0.9082 b_5 y_2$$

Laju pembentukan asam glukonik:

$$\frac{dy_3}{dt} = b_5 y_2$$

Laju konsumsi glukosa:

$$\frac{dy_4}{dt} = -1.011\left(\frac{b_3 y_1 y_4}{b_4 + y_4}\right)$$

Keterangan:

y_1 = konsentrasi sel

y_2 = konsentrasi glukonolaktona

y_3 = konsentrasi asam glukonik

y_4 = konsentrasi glukosa

b_1 s.d b_5 = parameter, f(T,pH)

Pada kondisi operasi T=30 ^oC dan pH 6.6 harga dari lima parameter yang diperoleh secara eksperimental sbb:

$b_1 = 0.949$

$b_2 = 3.439$

$b_3 = 18.72$

$b_4 = 37.51$

$b_5 = 1.169$

Pada kondisi tersebut buatlah profil y1,y2,y3, dan y4 terhadap waktu selama $0 \leq t \leq 9$ jam!. Nilai-nilai awal (pada saat t=0) adalah sbb:

$y_1(0)$ = 0.5 U.O.D./mL

$y_2(0)$ = 0.0 mg/mL

$y_3(0)$ = 0.0 mg/mL

$y_4(0)$ = 50.0 mg/mL

Petunjuk: soal ini mudah….!!!

________________________________o0o________________________________

Bab 7
Persamaan Diferensial Parsial (PDP)

Definisi PDP

Persamaan diferensial parsial adalah persamaan diferensial yang terdiri atas fungsi turunan lebih dari satu variabel bebas.

Contoh:

Persamaan konduksi tak tunak satu dimensi pada lembaran suatu material dirumuskan dalam PDP sbb.

$$\boxed{\frac{\partial T}{\partial t} = \frac{k}{\rho c_p} \frac{\partial^2 T}{\partial x^2}}$$

PDP tersebut terdiri dari dua buah variabel bebas, yaitu *x* (tebal lembaran) dan *t* (waktu). Sedangkan variabel terikatnya adalah *T* (Temperatur). Jika digambarkan pada koordinat kartesius akan diperoleh gambar 3 dimensi.

Aplikasi PDP

Pemodelan kimia dan fisika atas suatu fenomena dalam bidang teknik kimia seringkali menghasilkan rumusan matematis dalam bentuk PDP khususnya pada berbagai fenomena perpindahan (momentum, massa, dan panas).

Klasifikasi PDP

Berdasarkan ordenya PDP terdiri atas tiga jenis.

Orde 1

$$\frac{\partial u}{\partial x} - \alpha \frac{\partial u}{\partial y} = 0$$

Orde 2

$$\frac{\partial^2 u}{\partial x^2} + u \frac{\partial u}{\partial y} = 0$$

Orde 3

$$\left(\frac{\partial^3 u}{\partial x^3} \right)^2 + \frac{\partial^2 u}{\partial x \partial y} + \frac{\partial u}{\partial y} = 0$$

Umumnya PDP dalam teknik kimia berorde 2 dengan 2 sampai 4 variabel bebas.

Berdasarkan kelinierannya PDP terdiri atas tiga jenis.

1. Linier
2. Quasilinier
3. Taklinier

Kajian PDP dalam diktat ini terbatas hanya untuk PDP orde 2 linier saja. PDP linier orde 2 memiliki persamaan umum sbb:

$$a\frac{\partial^2 u}{\partial x^2} + 2b\frac{\partial^2 u}{\partial x \partial y} + c\frac{\partial^2 u}{\partial y^2} + d\frac{\partial u}{\partial x} + e\frac{\partial u}{\partial y} + fu + g = 0$$

Berdasarkan harga $b^2 - ac$ PDP orde 2 linier terbagi atas tiga bagian, sbb:

1. Eliptik $\quad b^2 - ac < 0$
2. Parabolik $\quad b^2 - ac = 0$
3. hiperbolik $\quad b^2 - ac > 0$

Kondisi Batas

Untuk lebih memahami kondisi batas pada PDP perhatikan contoh persamaan konduksi tak tunak satu dimensi sbb:

$$\boxed{\frac{\partial T}{\partial t} = \frac{k}{\rho c_p}\frac{\partial^2 T}{\partial x^2}}$$

1. Kondisi Dirichlet

Nilai variabel terikat (*T*) diketahui pada nilai variabel bebas (*x*,*t*) tertentu

kondisi awal
$T = f(x)$ pada $t = 0$ dan $0 \le x \le 1$, atau
$T = T_0$ pada $t = 0$ dan $0 \le x \le 1$

kondisi batas
$T = f(t)$ pada $x = 0$ dan $t > 0$, dan
$T = T_1$ pada $x = 0$ dan $t > 0$

Contoh :

$$\alpha \frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t}$$

$$T(0,t) = 120^0 C$$

$$T(1,t) = 120^0 C$$

$$T(x = 0:1,0) = 25^0 C$$

2. Kondisi Neuman

Turunan variabel terikat (*T*) diketahui pada nilai tertentu atau sebagai fungsi dari variabel bebas (x,t).

$$\boxed{\frac{\partial T}{\partial x} = 0 \qquad \text{pada } x = 1 \text{ dan } t \geq 0}$$

3. Kondisi Robbin

$$\boxed{k\frac{\partial T}{\partial x} = h(T - T_f) \qquad \text{pada } x = 0 \text{ dan } t \geq 0}$$

Turunan variabel terikat (*T*) diketahui sebagai fungsi dari variabel terikat itu sendiri.

Solusi PDP

Solusi yang paling sederhana dan gampang untuk diterapkan yaitu dengang metode penghampiran selisih terhingga (*finite difference approximationi*).

Penghampiran Selisih Terhingga

1. Penghampiran selisih terpusat

$$\frac{\partial u_i}{\partial x} = \frac{1}{2\Delta x}\left(u_{i+1} - u_{i-1}\right)$$

$$\frac{\partial^2 u_i}{\partial x^2} = \frac{1}{\Delta x^2}\left(u_{i+1} - 2u_i + u_{i-1}\right)$$

$$\frac{\partial^2 u_{i,j}}{\partial y \partial x} = \frac{1}{4\Delta x \Delta y}\left(u_{i+1,j+1} - u_{i-1,j+1} - u_{i+1,j-1} + u_{i-1,j-1}\right)$$

2. Penghampiran selisih maju

$$\frac{\partial u_i}{\partial x} = \frac{1}{\Delta x}\left(u_{i+1} - u_i\right)$$

$$\frac{\partial^2 u_i}{\partial x^2} = \frac{1}{\Delta x^2}\left(u_{i+2} - 2u_{i+1} + u_i\right)$$

$$\frac{\partial^2 u_{i,j}}{\partial y \partial x} = \frac{1}{4\Delta x \Delta y}\left(u_{i+1,j+1} - u_{i,j+1} - u_{i+1,j} + u_{i,j}\right)$$

3. Penghampiran selisih mundur

$$\frac{\partial u_i}{\partial x} = \frac{1}{\Delta x}\left(u_i - u_{i-1}\right)$$

$$\frac{\partial^2 u_i}{\partial x^2} = \frac{1}{\Delta x^2}\left(u_i - 2u_{i-1} + u_{i-2}\right)$$

$$\frac{\partial^2 u_{i,j}}{\partial y \partial x} = \frac{1}{\Delta x \Delta y}\left(u_{i,j} - u_{i,j-1} - u_{i-1,j} + u_{i-1,j-1}\right)$$

Diskretisasi Persamaan Diferensial

Dalam menyelesaikan persamaan diferensial menggunakan penghampiran selisih terhingga dikenal teknik diskretisasi. Penjelasannya diberikan berdasarkan contoh soal sebagai berikut :

Kasus 9:

Selesaikan persamaan differensial di bawah ini, kemudian petakan harga x dan y pada koordinat kartesius.

MATLAB &Pengantar Pemrograman

$$\frac{d^2y}{dx^2} = 6x + 4$$
$$x = 0 \rightarrow y = 1$$
$$x = 1 \rightarrow y = 1$$
Rentang Integrasi x = 0 s/d 1

Jawaban:

Rentang integrasi x = 0 s.d 1 didiskretisasikan menjadi 10 bagian (N = 10).

$$N = 10$$
$$\Delta x = \frac{1}{N} = 0.1$$

Δx

1	2	3	4	5	6	7	8	9	10

x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0. 8	0.9	1

Menggunakan penghampiran selisih terpusat.

$$\frac{\partial^2 y_i}{\partial x^2} = \frac{1}{\Delta x^2}\left(y_{i+1} - 2y_i + y_{i-1}\right)$$

Substitusikan penghampiran selisih terpusat itu ke persamaan diferensial.

$$\frac{\partial^2 y}{\partial x^2} - 6x + 4 = 0$$

menghasilkan

$$\frac{1}{\Delta x^2}\left(y_{i+1} - 2y_i + y_{i-1}\right) - 6x_i - 4 = 0$$

Untuk i = 1

$$\frac{1}{\Delta x^2}\left(y_2 - 2y_1 + 1\right) - 6(0.1) - 4 = 0$$

Untuk i = 2 s.d. 8

$$\frac{1}{\Delta x^2}\left(y_{i+1} - 2y_i + y_{i-1}\right) - 6x_i - 4 = 0$$

Untuk i = 9

$$\frac{1}{\Delta x^2}\left(1 - 2y_9 + y_8\right) - 6(0.9) - 4 = 0$$

SISTEM PERSAMAAN LINIER

Transformasi sistem persamaan linier di atas menjadi bentuk matrik sbb:

$$\begin{bmatrix} -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \end{bmatrix} = \begin{bmatrix} \{6(0.1)+4\}\,\Delta x^2 - 1 \\ \{6(0.2)+4\}\,\Delta x^2 \\ \{6(0.3)+4\}\,\Delta x^2 \\ \{6(0.4)+4\}\,\Delta x^2 \\ \{6(0.5)+4\}\,\Delta x^2 \\ \{6(0.6)+4\}\,\Delta x^2 \\ \{6(0.7)+4\}\,\Delta x^2 \\ \{6(0.8)+4\}\,\Delta x^2 \\ \{6(0.9)+4\}\,\Delta x^2 - 1 \end{bmatrix}$$

$\boxed{A}$ $\boxed{y}$ $\boxed{C}$

Harga y dapat dihitung dengan metode yang telah dipelajari pada bagian sistem persamaan linier.

$$\boxed{\begin{aligned} Ay &= C \\ y &= A^{-1}C \end{aligned}}$$

Berikut ini kita hitung harga vektor y dalam m-file MATLAB.

kasus9.m

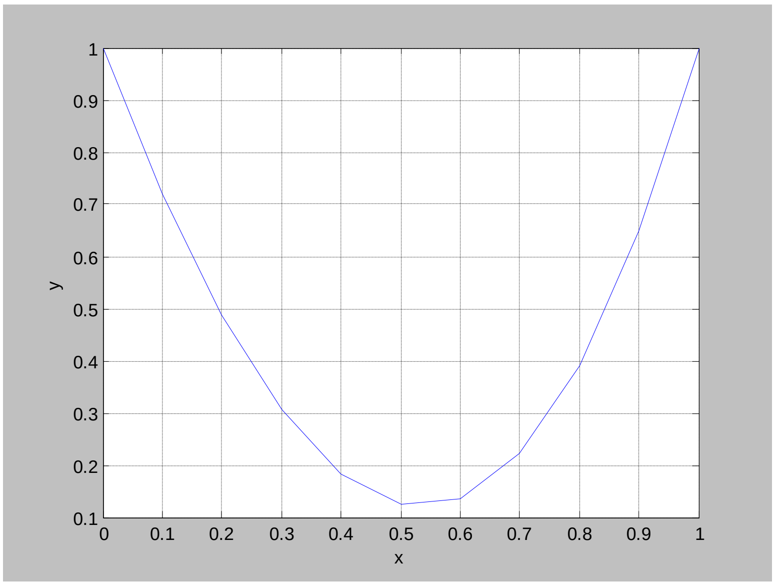
```
clear
clc
%Diskretisasi terhadap x
N=10;
dx=(1-0)/N;
x =[0:dx:1]'
%Membuat matrik A koefisien y
A = diag(-2*ones(N-1,1))+diag(ones(N-2,1),1) + diag(ones(N-2,1),-1)
%Vektor konstanta C
C = (6*[dx:dx:x(end)-dx]+4)*dx^2
C(1)=C(1)-1
C(end)=C(end)-1
%Menghitung harga y
y=A\C'
y=[1;y;1]
%Membuat kurva x-y
plot(x,y)
xlabel('x')
ylabel('y')
grid on
```

Eksekusi program kasus9.m
Masukan dan hasil di *Command Window* :

```
>> kasus9
y =

1.0000
0.7210
0.4880
0.3070
0.1840
0.1250
0.1360
0.2230
0.3920
0.6490
1.0000
```

Dari hasil perhitungan MATLAB di atas dapat dibuat grafik dan hasil lengkap untuk persoalan ini sbb:

x	y
0	1
0.1	0.721
0.2	0.488
0.3	0.307
0.4	0.184
0.5	0.125
0.6	0.136
0.7	0.223
0.8	0.392
0.9	0.649
1	1



Tugas 9

Menyelesaikan persamaan differensial dengan penghampiran selisih terhingga (diskretisasi)

Nomor 1

Dengan menggunakan penghampiran selisih terhingga terpusat selesaikan persamaan diferensial sbb:

$$\frac{d^2y}{dx^2} = y + 2$$

$$y(0) = y(1) = 1$$

Rentang Integrasi = 0 s/d 1

Sertakan pula kurva x,y diagram kartesiusnya.

Semidiskretisasi Persamaan Diferensial Parsial

Di muka telah dipaparkan penjelasan mengenai diskretisasi untuk menyelesaikan persamaan diferensial. PDP dapat diselesaikan juga dengan menggunakan teknik diskretisasi, sayangnya penerapan pada PDP lebih rumit dan melibatkan banyak angka. Sebagai penyederhanaan dapat digunakan teknik semidiskretisasi. Penjelasannya akan diberikan berdasarkan contoh soal sbb:

kasus10

Sebuah lembaran plastik dengan tebal 1 cm mula-mula bertemperatur 25 ^{O}C diletakan diantara pelat yang bertemperatur 120 ^{O}C. Diketahui massa jenis plastik 900 kg/m^3, konduktivitas termal 0.13 W/m^{O}C, dan kalor jenis 1670 J/kg^{O}C . Pemodelan matematis untuk kasus konduksi tak tunak adalah sbb:

$$\alpha \frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t} \qquad\qquad \alpha = \frac{k}{\rho c_p}$$

Dengan metode numeris evaluasi temperatur rata-rata plastik 10 menit kemudian?

Jawaban:

$$\left.\begin{aligned} &\alpha \frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t} \\ &T(0,t) = 120^0 C \\ &T(1,t) = 120^0 C \\ &T(x = 0:1, 0) = 25^0 C \end{aligned}\right\}$$

Kondisi Dirichlet

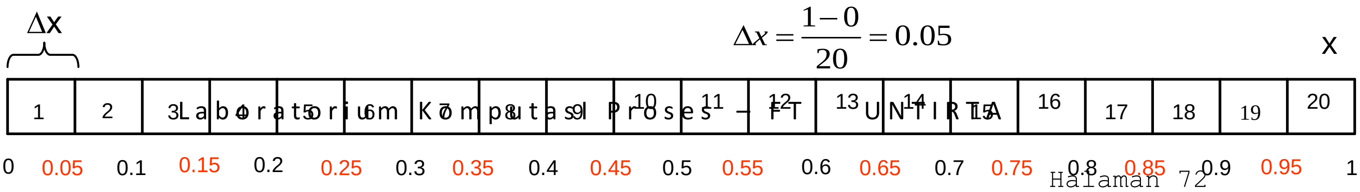
$$\alpha = \frac{k}{\rho c_p} = \frac{(0.13)}{(900)(1670)} = 8.6494\text{x}10^{-8} m^2 / s$$

$$= 5.1896\text{x}10^{-2} cm^2 / menit$$

Diskretisasi dilakukan terhadap *x*.

$$N = 20$$

$$\Delta x = \frac{1-0}{20} = 0.05$$



Menggunakan penghampiran selisih terpusat turunan kedua T terhadap x,

$$\frac{\partial^2 T_i}{\partial x^2} = \frac{1}{\Delta x^2}\left(T_{i+1} - 2T_i + T_{i-1}\right)$$

PDP tersebut akan menjadi.

$$\frac{\partial T}{\partial t} = \frac{\alpha}{\Delta x^2}\left(T_{i+1} - 2T_i + T_{i-1}\right)$$

Untuk i = 1

$$\frac{\partial T}{\partial t} = \frac{\alpha}{\Delta x^2}\left(T_2 - 2T_1 + 120\right)$$

Untuk i = 2 : 18

$$\frac{\partial T}{\partial t} = \frac{\alpha}{\Delta x^2}\left(T_{i+1} - 2T_i + T_{i-1}\right)$$

Untuk i = 19

$$\frac{\partial T}{\partial t} = \frac{\alpha}{\Delta x^2}\left(120 - 2T_{19} + T_{18}\right)$$

SISTEM PERSAMAAN DIFERENSIAL BIASA

Solusi sistem persamaan diferensial biasa telah dibahas pada bagian sebelumnya. Rutin yang akan digunakan untuk sistem PDB di termaksud adalah ode23 MATLAB. Berikut ini pemrogramannya.

taktunak.m	**Pemrograman MATLAB**

```
function dTdt=taktunak(t,T)

N=20;
dx=1/N;
a=5.1896e-2; %diffusivitas termal,cm2/menit

% untuk i = 1
dTdt(1,:) = a/(dx^2)*(T(2)-2*T(1)+120);

% untuk i = 2 s.d 18
for i = 2:18
    dTdt(i,:) = a/(dx^2)*(T(i+1)-2*T(i)+T(i-1));
end
% untuk i = 19
dTdt(19,:) = a/(dx^2)*(120-2*T(19)+T(18));
```

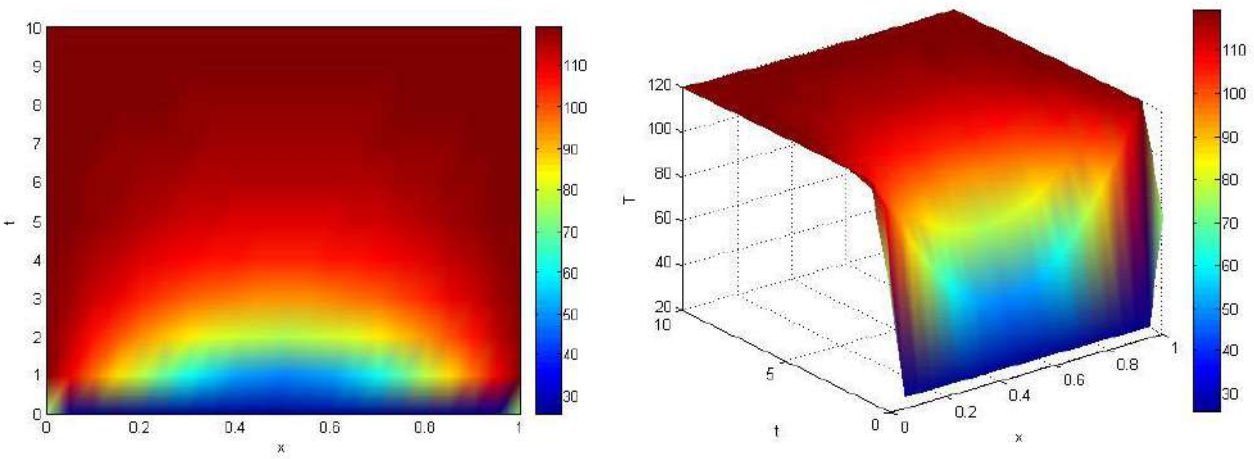
kasus10.m	**Pemrograman MATLAB**

```
Clear
clc

% Nilai awal dan rentang integrasi.
tspan=[0:1:10];
To = 25*ones(1,19);

% Fungsi pengintegrasi.
[t,T] = ode23('taktunak',tspan,To);

% Menampilkan hasil perhitungan.
x=[0:1/20:1]'
t
T0=120*ones(length(tspan),1);
T20=120*ones(length(tspan),1);
T=[T0 T T20]

% Memetakan hasil pd diagram kartesius 3-D.
figure(1)
surf(x,t,T)
xlabel('x')
ylabel('t')
zlabel('T')
colorbar
shading interp

figure(2)
pcolor(x,t,T)
xlabel('x')
ylabel('t')
zlabel('T')
colorbar
shading interp

% Menghitung rata-rata suhu plastik pd menit ke 10
[b k]=size(T);
z=T(b,:);
Tslab=mean(z)
```

Eksekusi program kasus10.m &runkasus10.m. Masukan dan hasil di *Command Window* :

```
>>runkasus10
Tslab =

 119.5598
```

Gambar 3-D.
Profil temperatur plastik sepanjang x pada interval waktu t



Tugas 10: Menyelesaikan PDP dengan penghampiran selisih terhingga terpusat (semi diskretisasi)

Nomor 1

Suatu fenomena difusi-konveksi dapat dideskripsikan dengan PDP berikut ini:

$$\frac{\partial \theta}{\partial t} = \frac{\partial^2 \theta}{\partial x^2} - \lambda \frac{\partial \theta}{\partial x} \qquad 0 < x < 1, \qquad \text{t} > 0$$

$$\theta(0,t) = 1, \qquad t > 0$$

$$\frac{\partial \theta}{\partial x}(1,t) = 0, \qquad t > 0$$

$$\theta(x,0) = 0, \qquad 0 < x < 1$$

Jika harga λ = 25, selesaikan PDP diatas untuk rentang t = 0 s.d 5. Buat pula gambar 3-D θ,t,x pada koordinat kubus (kartesius).

________________________________o0o______________________________