MODUL 6 : MASALAH SYARAT BATAS

> Modul ini memberikan gambaran penggunaan metode numerik untuk penyelesaian syarat batas yaitu pemecahan persamaan diferensial secara numerik ketika nilai fungsi pada kedua titik batas diketahui.Persamaan differensial ini diselesaikan dengan mengimplementasikan metode beda hingga (*Finite Difference*) yang akan menghasilkan sekumpulan persamaan linear yang diselesaikan secara simultan menggunakan metode numerik Eliminasi Gauss. Mengingat kebanyakan hukum dan masalah fisika diungkapkan dalam bentuk persamaan diferensial seperti uraian tersebut maka setelah menyelesaikan bab ini pengguna diharapkan memiliki kemampuan untuk menyelesaikan dan sekaligus memahami perilaku masalah fisika yang dihadapi.

6.1. Metode Beda Hingga

Ditinjau suatu masalah fisika yang diwakili oleh persamaan diferensial berbentuk

$$\frac{d^2y}{dx^2} = f(x,y), \tag{6.1}$$

dengan $f(x)$ adalah fungsi yang bentuk eksplisitnya diketahui. Apabila penyelesaian yang diinginkan adalah $y(x)$ pada ranah $x_0 \leq x \leq x_N$ dimana nilai atau bentuk penyelesaian pada kedua titik batas yaitu $y(x_0) = y_0$ dan $y(x_N) = y_N$ sudah diketahui maka masalah tersebut dikenal sebagai masalah syarat batas. Salah satu metode untuk menyelesaikan masalah tersebut adalah dengan metode beda hingga yaitu mengganti persamaan (6.1) ke dalam bentuk diskret yaitu mengubah diferesial ndua kali ke bentuk ungkapan beda terpusat sehingga menjadi

$$y_{i-1} - 2y_i + y_{i+1} = h^2 f_i; \tag{6.2}$$

yang seperti biasa diambil kesepakatan simbol $y_i \equiv y(x_i)$, $f_i \equiv f(x_i)$ dan ukuran langkah $h = x_i - x_{i-1}$ untuk $i = 1, 2, 3, \ldots, N$. Berbeda dengan masalah syarat awal, disini nilai $y(x)$ yang akan dicari harus sesuai pada batas x_0 dan x_N. Ini berarti persamaan (6.2) tidak dapat diselesaikan satu persatu untuk tiap i tertentu tetapi harus diselesaikan secara serentak untuk seluruh i yang ada. Apabila nilai N besar maka cacah persamaan menjadi banyak sekali. Salah satu metode paling efektif untuk memecahkan banyak persamaan secara serentak adalah menggunakan cara matriks.

Mudah dibuktikan bahwa bentuk matriks dari persamaan (6.2) adalah

$$\begin{pmatrix} 2 & -1 & 0 & \ldots & 0 & 0 \\ -1 & 2 & -1 & 0 & \ldots & 0 \\ 0 & -1 & 2 & -1 & \vdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & -1 & 2 & -1 \\ 0 & 0 & \cdots & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} y_1 \\ y2 \\ \vdots \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{pmatrix} = \begin{pmatrix} h^2 f_1 - y_0 \\ h^2 f_2 \\ \vdots \\ \vdots \\ h^2 f_{N-2} \\ h^2 f_{N-1} - y_N \end{pmatrix}, \tag{6.3}$$

Jika dilihat secara seksama pada persamaan matriks tersebut, akan terlihat bahwa setiap baris dari matrik pertama pada ruas kiri hanya memiliki unsur bernilai tidak nol paling banyak sebanyak tiga kolom sedang kolom yang lain selalu bernilai nol. Matrik berbentuk seperti ini biasa disebut matrik tridiagonal.

6.2. Penyelesaian Matrik Tridiagonal

Salah satu metode efisien untuk menyelesaikan sistem persamaan simultan yang tersusun atas matriks tridiagonal seperti di atas adalah dengan cara eleminasi unsur-unsur yang terletak persis di bawah diagonal utama. Bentuk umum masalah yang akan dipecahkan adalah persamaan matriks seperti berikut

$$\begin{pmatrix} b_1 & c_1 & 0 & \cdots & 0 & 0 \\ a_2 & b_2 & c_2 & 0 & \cdots & 0 \\ 0 & a3 & b3 & c3 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & a_{N-2} & b_{N-2} & c_{N-2} \\ 0 & 0 & \cdots & 0 & a_{N-1} & b_{N-1} \end{pmatrix} \begin{pmatrix} y_1 \\ y2 \\ \vdots \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{pmatrix} = \begin{pmatrix} h^2 f_1 - y_0 \\ h^2 f_2 \\ \vdots \\ \vdots \\ h^2 f_{N-2} \\ h^2 f_{N-1} - y_N \end{pmatrix} \tag{6.4}$$

Dengan melakukan metode eliminasi Gauss yaitu mendefinisikan kaitan

$$\beta_j = b_j - \frac{a_j}{\beta_{j-1}} c_{j-1}, \rho_j = r_j - \frac{a_j}{\beta_{j-1}} \rho_{j-1}, j = 2, 3, \ldots, N-2 \tag{6.5}$$

dimana $\beta_1 = b_1$ dan $\rho_1 = r_1$ maka persamaan (6.4) berubah menjadi bentuk persamaan matriks

$$\begin{pmatrix} \beta_1 & c_1 & 0 & \cdots & 0 & 0 \\ 0 & \beta_2 & c_2 & 0 & \cdots & 0 \\ 0 & 0 & \beta_3 & c_3 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \beta_{N-2} & c_{N-2} \\ 0 & 0 & \cdots & 0 & 0 & \beta_{N-1} \end{pmatrix} \begin{pmatrix} y_1 \\ y2 \\ \vdots \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{pmatrix} = \begin{pmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \vdots \\ \rho_{N-2} \\ \rho_{N-1} \end{pmatrix} \tag{6.6}$$

Bentuk persamaan (6.6) sangat menarik karena memungkinkan semua y_i dapat diperoleh dengan cara substitusi balik yaitu pertama dihitung y_{N-1} melalui kaitan

$$y_{N-1} = \frac{\rho_{N-1}}{\beta_{N-1}} \tag{6.7}$$

Setelah y_{N-1} didapatkan maka y_i yang lain diperoleh melalui kaitan

$$y_{N-j} = \frac{\rho_{N-j} - c_{N-j} y_{N-J+1}}{\beta_{N-j}},\ J = 2, 3, \ldots, N-1 \tag{6.8}$$

6.3 Contoh masalah fisika : potensial listrik

Hukum Gauss dalam bentuk diferensial untuk medan listrik **E** dinyatakan sebagai

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0}, \tag{6.9}$$

dengan ϵ_0 adalah permitivitas ruang hampa. Adapun rapat muatan ρ dikaitkan dengan muatan q dan unsur volume dV melalui kaitan

$$q = \int_V \rho dV \tag{6.10}$$

Mengingat medan listrik dikaitkan dengan potensial listrik ϕ oleh ungkapan $E = -\nabla\phi$ maka persamaan (6.9) dapat dinyatakan juga sebagai $\nabla \cdot \nabla\phi = -\rho/\epsilon_0$ yang dalam satu dimensi berbentuk

$$\frac{d^2\phi}{dx^2} = -\frac{\rho}{\epsilon_0} \tag{6.11}$$

Dapat dilihat bahwa persamaan (6.11) berbentuk sama dengan persamaan (6.1) sehingga metode penyelesaian yang dijelaskan di depan dapat digunakan untuk mencari potensial listrik pada syarat batas yang diberikan. Tinjau suatu daerah antara $0 \leq x \leq 3$ yang memiliki rapat muatan berbentuk $\rho(x) = 3\epsilon 0x$ dan adanya syarat batas nilai potensial $\phi(0) = 0$ dan $\phi(3) = 0$. Masalah ini dapat dirumuskan seperti persamaan (6.1) dengan mengambil $y(x) = \phi(x)$ dan $f(x) = -\rho(x)/\epsilon_0 = -3\epsilon_0 x/\epsilon_0 = 3x$.

Di bawah ini adalah contoh program untuk menyelesaikan masalah di atas yang ditulis dengan menggunakan bahasa pemrograman Python.

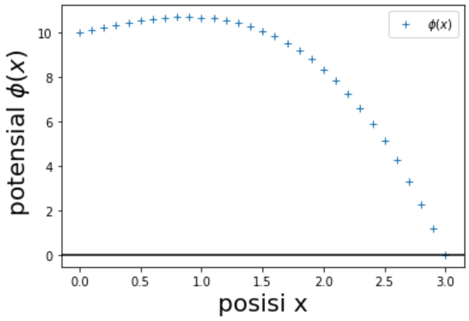
```
# CONTOH PROGRAM UTAMA PENYELESAIAN MASALAH SYARAT BATAS DENGAN MENGGUNAKAN METODE BEDA HINGGA DAN ELIMINASI GAUSS PADA MATRIK TRIDIAGONAL
import numpy as np
import matplotlib.pyplot as plt

def fi(xx): #fungsi untuk menghasilkan f(x)
  fung=-3*xx
  return fung

#---------------------------------------------------------
#Masukan untuk cacah titik n, batas kiri x(0) dan batas
#kanan x(n) daerah yang ditinjau, ukuran langkah h
#serta syarat batas potensial di x0 dan xn
#---------------------------------------------------------
n=30
x0=0
xn=3
h=(xn-x0)/n
y0=10
yn=0

#------------------------------------------------------------------------------------------------------------
# Deklarasi ukuran array/list untuk menyimpan nilai x(i),y(x),a,b,c,r,rho,beta
#------------------------------------------------------------------------------------------------------------

x=[]
y=[]
a=[]
b=[]
c=[]
r=[]
rho=[]
beta=[]

for i in range(0,(n+1)):
  xi=x0+i*h
  yi=1
  bi=-2
  beta1=0
  rho1=0
  if (i==1):
    ri=fi(xi)*h*h-y0
  elif  (i==(n-1)):
    ri=fi(xi)*h*h-yn
  else :
    ri=fi(xi)*h*h
  x.append(xi)
  y.append(yi)
  b.append(bi)
  r.append(ri)
  beta.append(beta1)
  rho.append(rho1)
  #print(ri)
  #print(y)

for i in range(1,(n+1)):
  ci=1
  c.append(ci)

for i in range (1,(n+1)):
  ai=1
  a.append(ai)
```

```
#------------------------------------------------------------------------------
#--- Mengubah matriks tridiagonal ke bentuk matriks atas yang
#--- hanya mengandung dua larik yaitu beta(i) dan c(i) serta r(i) ke rho(i)
#------------------------------------------------------------------------------
  beta[1]=b[1]
  rho[1]=r[1]
for k in range(2,(n)):
  beta[k]=b[k]-(a[k]/beta[k-1])*c[k-1]
  rho[k]=r[k]-(a[k]/beta[k-1])*rho[k-1]
#------------------------------------------------------------------------------
#----Menghitung y(i) dengan substitusi balik yaitu dihitung
#----lebih dahulu y(n-1) dan dilanjutkan ke y(n-2) dan seterusnya sampai y(1)
#------------------------------------------------------------------------------
y[0]=y0
y[n]=yn
y[n-1]=rho[n-1]/beta[n-1]
for j in range(2,n):
  y[n-j]=(rho[n-j]-c[n-j]*y[n-j+1])/beta[n-j]
#------------------------------------------------------------------------------
#----Menampilkan hasil dalam bentuk kolom
#------------------------------------------------------------------------------
print("x_i      y_i     r_i    beta_i   rho_i")
for i in range(n+1):
  print("%2.3f  %4.3f  %4.3f   %4.3f   %4.3f "%(x[i],y[i],r[i],beta[i],rho[i]))
```

```
x_i      y_i     r_i    beta_i   rho_i
0.000  10.000  -0.000   0.000   0.000
0.100  10.116  -10.003   -2.000   -10.003
0.200  10.229  -0.006   -1.500   -5.008
0.300  10.336  -0.009   -1.333   -3.347
0.400  10.435  -0.012   -1.250   -2.522
0.500  10.521  -0.015   -1.200   -2.033
0.600  10.592  -0.018   -1.167   -1.712
0.700  10.645  -0.021   -1.143   -1.489
0.800  10.677  -0.024   -1.125   -1.326
0.900  10.685  -0.027   -1.111   -1.206
1.000  10.667  -0.030   -1.100   -1.115
1.100  10.618  -0.033   -1.091   -1.047
1.200  10.536  -0.036   -1.083   -0.996
1.300  10.418  -0.039   -1.077   -0.958
1.400  10.261  -0.042   -1.071   -0.932
1.500  10.062  -0.045   -1.067   -0.915
1.600  9.819  -0.048   -1.062   -0.905
1.700  9.527  -0.051   -1.059   -0.903
1.800  9.184  -0.054   -1.056   -0.907
1.900  8.787  -0.057   -1.053   -0.916
2.000  8.333  -0.060   -1.050   -0.930
2.100  7.819  -0.063   -1.048   -0.949
2.200  7.243  -0.066   -1.045   -0.972
2.300  6.600  -0.069   -1.043   -0.999
2.400  5.888  -0.072   -1.042   -1.029
2.500  5.104  -0.075   -1.040   -1.063
2.600  4.245  -0.078   -1.038   -1.100
2.700  3.308  -0.081   -1.037   -1.140
2.800  2.291  -0.084   -1.036   -1.184
2.900  1.189  -0.087   -1.034   -1.230
3.000  0.000  -0.090   0.000   0.000
```

Berikut ini adalah perintah untuk menampilkan hasil dalam plot grafik:

```
import numpy as np
import matplotlib.pyplot as plt
plt.plot(x,y,'+')
plt.xlabel("posisi x",fontsize=20)
plt.ylabel(" potensial $\phi (x)$",fontsize=20)
plt.legend(["$\phi(x)$"],loc='best')
plt.axhline(y=0, color='k')
#plt.axvline(x=0, color='k')
plt.show()
```



7.4. TUGAS

1. Modifikasi program tersebut untuk berbagai kemungkinan masukan dan kemudian anda coba untuk menganalisis hasil yang diperoleh.

2. Dalam fisika dasar ditunjukkan bahwa jika jarak antara dua keping kapasitor adalah *a* dan di ruang antara dua keping kapasitor tersebut tidak ada sumber muatan (ruang hampa) maka medan listriknya adalah konstan sebesar ***E*** sedang potensialnya adalah $V = Ex$. Coba anda tunjukkan bahwa hasil analitik ini juga dapat diperoleh dari komputasi di atas.