

10 Persamaan Differensial Biasa (PDB) Dengan Nilai Batas

10.1 PENGENALAN TOPIK

Pada persoalan engineering lebih sering dijumpai PDB tingkat 2 dengan kondisi batas yang diberikan pada dua titik. Umumnya kedua titik ini ada pada batas-batas domain permasalahan. Karena solusi yang dicari berada pada dua batas yang tertutup, maka problem ini dikenal sebagai problem domain tertutup atau PDB dengan nilai batas. Bentuk umum dari PDB tingkat 2 dengan nilai batas adalah

$$\frac{d^2 y}{dx^2} + p(x, y) \frac{dy}{dx} + q(x, y)y = f(x) \quad \text{antara } x_0 \leq x \leq x_n \quad (10-1)$$

Dengan nilai-nilai batas

$$A_1 y(x_0) + B_1 \frac{dy}{dx}(x_0) = \alpha \quad (10-2a)$$

$$A_2 y(x_n) + B_2 \frac{dy}{dx}(x_n) = \beta \quad (10-2b)$$

dimana

$$|A_1| + |B_1| \neq 0 \quad \text{dan} \quad |A_2| + |B_2| \neq 0 \quad (10-3)$$

Dari kondisi batas (10-2) ada 3 kemungkinan jenis kondisi batas yang mungkin diterapkan pada PDB ini:

(i) Nilai batas konstan (tipe Dirichlet)

Nilai batas diberikan sebagai sebuah konstan, contoh jika $A_1 = 1$ dan $B_1 = 0$ maka $y(x_0) = \alpha$

(ii) Nilai batas derivatif (tipe Neumann)

Nilai batas diberikan sebagai sebuah nilai derivatif, contoh jika $A_1 = 0$ dan $B_1 = 1$ maka $y'(x_0) = \alpha$

(iii) Nilai batas campuran (tipe Robin)

Nilai batas terdiri dari nilai konstan dan derivatif, contoh jika $A_1 = 1$ dan $B_1 = 1$ maka $y(x_0) + y'(x_0) = \alpha$

Tergantung dari koefisien-koefisien $p(x,y)$ dan $q(x,y)$, PDB (10-1) dapat diklasifikasikan sebagai:

1. PDB Linier, jika $p(x,y)$ dan $q(x,y)$ berupa fungsi dari x saja atau berupa sebuah bilangan konstan.

$$p(x,y) = p(x) \quad (10-4)$$

atau

$$p(x,y) = \text{konstan} \quad (10-5)$$

2. PDB Non-Linier, jika $p(x,y)$ dan $q(x,y)$ merupakan fungsi dari x dan y .

Pada bab ini kita akan bahas beberapa teknik untuk memecahkan PDB linier maupun non-linier yang dibatasi oleh kondisi batas tipe Dirichlet, tipe Neumann maupun tipe Robin.

10.2 METODE LINIER TEMBAK

Metode ini sangat efektif dan mudah digunakan untuk memecahkan PDB linier dengan kondisi batas tipe Dirichlet. Secara umum problem yang dapat dipecahkan dengan metode ini adalah

$$\frac{d^2 y}{dx^2} + p(x) \frac{dy}{dx} + q(x)y = f(x) \quad (10-6)$$

Dengan nilai batas

$$y(x_0) = \alpha \quad (10-7a)$$

$$y(x_n) = \beta \quad (10-7b)$$

Tanpa mengurangi artinya persamaan (10-6) dapat juga dituliskan

$$y'' = p(x)y' + q(x)y + f(x) \quad (10-8)$$

Langkah utama dari metode tembak adalah merubah problem (10-8) menjadi problem PDB dengan nilai awal. Dua PDB nilai awal akan didapat sebagai berikut

$$y' = z \quad (10-9a)$$

$$z' = p(x)z + q(x)y + f(x) \quad (10-9b)$$

Sistim persamaan (10-9) memerlukan nilai-nilai awal. Nilai awal untuk (10.9a) adalah

$$y(x_0) = \alpha \quad (10-10)$$

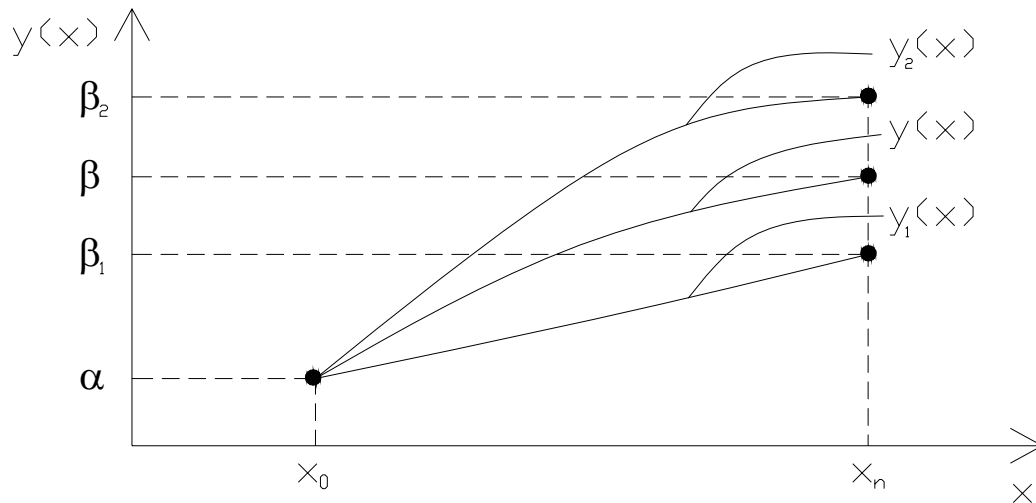
Sedangkan nilai awal (10.9b) tidak diketahui sehingga kita asumsikan

$$y'(x_0) = z(x_0) = \xi_1 \quad (10-11)$$

Dengan kedua nilai batas (10-10) dan (10-11) sistim PDB nilai awal (10-9) dapat dipecahkan dengan salah satu teknik pemecahan PDB nilai awal yang telah dibahas pada bab 9 seperti Runge-Kutta. Dengan assumisi ξ_1 , solusinya $y_1(x)$ yang mempunyai nilai $y_1(x_n) = \beta_1$. Karena β_1 masih berbeda dari nilai $y(x_n)$ sebenarnya β , maka kita gunakan sebuah assumisi lain.

$$y'(x_0) = z(x_0) = \xi_2 \quad (10-12)$$

Dengan asumsi ini kita dapatkan solusi $y_2(x)$ dengan nilai $y_2(x_n) = \beta_2$. Kedua solusi $y_1(x)$ dan $y_2(x)$ tidak menghasilkan $y_1(x_n)$ atau $y_2(x_n) = \beta$.



Gambar 10.1 Metode tembak linier

Tetapi karena PDB linier maka solusi sebenarnya, $y(x)$ dapat diberikan oleh superposisi dari $y_1(x)$ dan $y_2(x)$.

$$y(x) = C_1 y_1(x) + C_2 y_2(x) \quad (10-13)$$

Nilai C_1 dan C_2 dapat dicari dengan menggunakan nilai-nilai batas $y(x_0) = \alpha$ dan $y(x_n) = \beta$. Nilai $y(x_0)$ dihitung dengan (10-13) yang menghasilkan

$$\alpha = C_1 \alpha + C_2 \alpha \quad (10-14)$$

atau

$$C_1 + C_2 = 1 \quad (10-15)$$

Sedangkan nilai $y(x_n)$ menghasilkan

$$\beta = C_1 \beta_1 + C_2 \beta_2 \quad (10-16)$$

Dengan mensubstitusikan nilai C_1 yang diperoleh dari (10-15) ke (10-16).

$$\beta = (1 - C_2) \beta_1 + C_2 \beta_2 \quad (10-17)$$

Maka C_2 diperoleh

$$C_2 = \frac{\beta - \beta_1}{\beta_2 - \beta_1} \quad (10-18)$$

dan,

$$C_1 = 1 - \frac{\beta - \beta_1}{\beta_2 - \beta_1} \quad (10-19)$$

Dengan mensubstitusikan (10-18) dan (10-19) ke (10-13) didapat

$$y(x) = \left[1 - \frac{\beta - \beta_1}{\beta_2 - \beta_1} \right] y_1(x) + \frac{\beta - \beta_1}{\beta_2 - \beta_1} y_2 \quad (10-20)$$

Selanjutnya dengan mendifferensiasikan (10-20) kita peroleh

$$y'(x) = \left[1 - \frac{\beta - \beta_1}{\beta_2 - \beta_1} \right] y_1'(x) + \frac{\beta - \beta_1}{\beta_2 - \beta_1} y_2' \quad (10-21)$$

Sekarang nilai $y'(x_0)$ dapat diperoleh

$$y'(x_0) = \left[1 - \frac{\beta - \beta_1}{\beta_2 - \beta_1} \right] \xi_1 + \frac{\beta - \beta_1}{\beta_2 - \beta_1} \xi_2 \quad (10-22)$$

Setelah diatur kita bisa dapatkan

$$\xi = \xi_1 + \frac{(\xi_2 - \xi_1)}{(\beta_2 - \beta_1)} (\beta - \beta_1) \quad (10-23)$$

Setelah $y'(x_0) = \xi$ yang tepat didapat, selanjutnya sistim PDB nilai awal (10-9) kita pecahkan dengan $z(x_0) = \xi$.

----- CONTOH 10.1 -----

Pecahkan PDB nilai batas berikut

$$y'' = 2y' - y + xe^x - x$$

Antara $[0,2]$ dengan nilai-nilai batas $y(0) = 0$ dan $y(2) = -4$. Gunakan metode Runge Kutta dengan $h = 0,2$ dan bandingkan hasilnya dengan solusi analitik

$$y(x) = \frac{1}{6}x^3e^x - \frac{5}{3}xe^x + 2e^x - x - 2$$

Pemecahan contoh ini sama dengan pemecahan sistim PDB dengan nilai awal (10-9),

$$y' = z$$

$$z' = 2z - y + xe^x - x$$

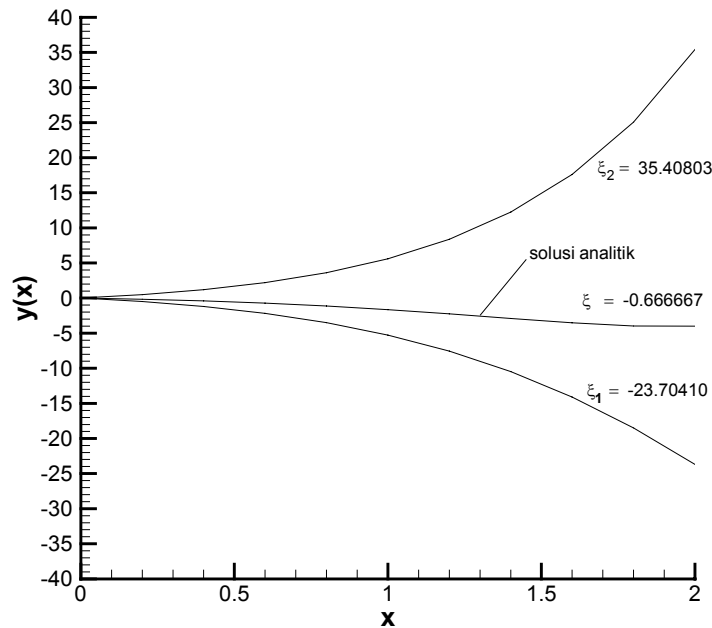
Sekarang kita asumsikan suatu nilai ξ_1 untuk persamaan (10-11). Tidak ada rumusan khusus untuk menghitungnya tetapi perkiraan dapat kita mulai dengan,

$$\xi_1 = \frac{y(x_n) - y(x_0)}{x_n - x_0} = \frac{-4 - 0}{2 - 0} = -2$$

Dengan nilai awal $y(0) = 0$ dan $z(0) = -2$. Sistim PDB nilai awal pada contoh ini dipecahkan dengan metode Runge Kutta 4 dengan interval $h = 0,1$. Dengan menggunakan program FOR9_9 kita peroleh $y(2) = -23,70410$. Dengan nilai awal $z(0) = 2$ atau $\xi_2 = 2$, kita peroleh $y(2) = 35,40803$, guna memperkirakan ξ yang tepat kita gunakan (10-23)

$$\xi = \xi_1 + \frac{(\xi_2 - \xi_1)}{(\beta_2 - \beta_1)}(\beta - \beta_1) = -2 + \frac{2 - (-2)}{35,40803 - (-23,70410)}(-4 - (-23,70410)) = -0,666667$$

Dengan $z(0) = -0,666667$ kita peroleh $y(2) = -4,000063$ yang mendekati nilai batas $y(2) = -4$. Plot perbandingan antara hasil dengan $\xi_1 = -2$, $\xi_2 = 2$ dan $\xi = -0,666667$ diberikan pada gambar 10.2 di bawah ini.



Gambar 10.2 Perbandingan antara $\xi_1 = -2$, $\xi_2 = 2$, $\xi = -0,666667$ dan solusi analitik

10.3 METODE TEMBAK NON-LINIER

Untuk PDB non-linier, superposisi tidak dapat kita gunakan sehingga cara lain harus digunakan. Langkah pertama disini sama dengan cara pemecahan PDB linier yaitu merubah persoalan menjadi PDB nilai awal dan mengasumsikan $y'(x_0) = \xi_i$. Dengan menggunakan $y(x_0) = \alpha$ dan $y'(x_0) = \xi_i$ akan diperoleh nilai $y(x_n) = \beta_i$. Jika nilai $y'(x_0) = \xi$ dapat kita tebak dengan benar maka $y(x_n) = \beta$ akan kita dapatkan dalam batas akurasi yang memungkinkan dalam teknik numerik. Jika tidak benar atau perbedaan terlalu jauh maka kita akan coba ξ yang baru, ξ_{i+1} . Jelas disini bahwa kita bereksperimen dengan nilai ξ sampai target $y(x_n) = \beta$ didekati dalam batas akurasi yang telah ditentukan. Problem mencari ξ yang tepat dapat diartikan secara matematika mencari akar dari persamaan non-linier berikut

$$g(\xi) = y_{x_n}(\xi) - \beta = 0 \quad (10-24)$$

dimana $y_{x_n}(\xi)$ adalah nilai $y(x_n)$ dengan $y'(x_0) = \xi$ dan akar dari persamaan (10-24) dapat dicari dengan salah satu metode pada bab 3. Metode yang umum dipakai adalah metode Secant yang diberikan oleh

$$\xi_{i+1} = \xi_i - \frac{g(\xi_i)(\xi_i - \xi_{i-1})}{g(\xi_i) - g(\xi_{i-1})} \quad (10-25)$$

Contoh 10.2 di bawah memperjelas penerapan teknik ini.

----- CONTOH 10.2 -----

Pecahkan PDB nilai batas berikut

$$y'' = -(y')^2 - y + \ln(x)$$

Antara [1,2] dengan nilai-nilai batas $y(1) = 0$ dan $y(2) = \ln(2)$ dengan menggunakan metode Runge Kutta dengan $h = 0,1$ dan metode secant. Bandingkan hasilnya dengan solusi analitik

$$y(x) = \ln(x)$$

Pemecahan contoh ini sama dengan memecahkan sistim PDB dengan nilai awal (10-9),

$$y' = z$$

$$z' = -z^2 - y + \ln(x)$$

Dengan nilai awal $y(1) = 0$ dan asumsi $z(1) = \xi_0$ (sebagai aproksimasi pertama) dan ξ_1 sebagai aproksimasi kedua. ξ_0 ini didapat dengan

$$\xi_0 = \frac{y(x_n) - y(x_0)}{x_n - x_0} = \frac{\ln(2) - 0}{2 - 1} = 0,693147$$

ξ_1 kita assumisikan 1 sedangkan ξ_2 dihitung dengan menggunakan persamaan (10-25).

$$\xi_2 = \xi_1 - \frac{g(\xi_1)(\xi_1 - \xi_0)}{g(\xi_1) - g(\xi_0)} = 1 - \frac{g(\xi_1)(1 - 0,693147)}{g(\xi_1) - g(\xi_0)} \quad (C1)$$

$$g(\xi_0) = 0,5536475 - \ln(2) = -0,13949968 \quad (C2)$$

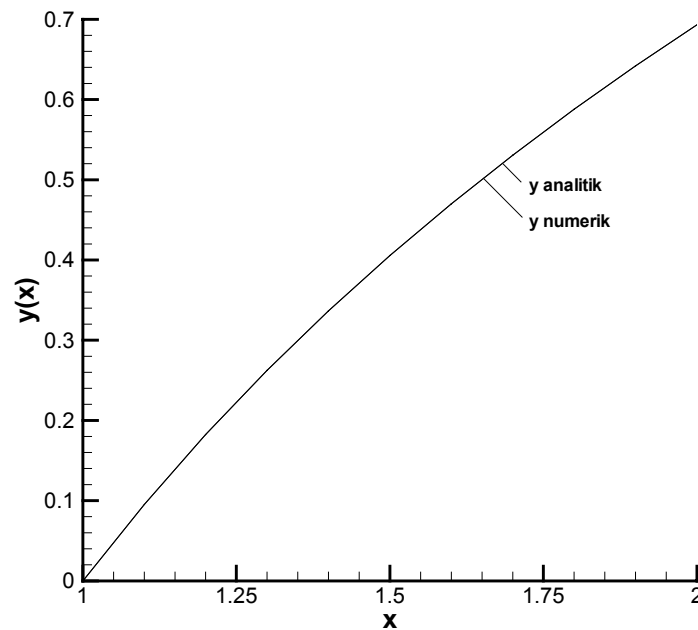
$$g(\xi_1) = 0,693146 - \ln(2) = -0,00000118 \quad (C3)$$

Dengan mensubstitusikan (C2) dan (C3) ke (C1) kita peroleh

$$\xi_2 = 1 - \frac{-0,00000118(1 - 0,693147)}{-0,00000118 - (-0,13949968)} = 1,000002$$

Dalam iterasi $\beta = \ln(2)$, dengan menggunakan program FOR10_1 konvergensi diperoleh dalam 1 iterasi saja. Interval $h = 0,1$ dan konvergensi criteria = 0,0001.

i	ξ_i	ξ_{i+1}	$y_{xn}(\xi_i)$	$y_{xn}(\xi_{i+1})$	$g(\xi_i) = y_{xn} - \beta$
0	0,693147	1,0	0,5536475	0,693146	-0,139499
1	1,0	1,000002	0,693146	0,693146	-0,000001



Gambar 10.3 Perbandingan antara solusi numerik dan analitik

Hasil numerik dan analitik tidak berbeda jauh. Hal ini menunjukkan bahwa dengan metode secant akurasi yang didapat cukup tinggi.

Program FOR10_1 memberikan program metode tembak non-linier secant.

Program FOR10_1 Program Fortran 90 metod tembak non-linier secant

```

1  ! FOR10_1
2  module VARIABEL
3  !----- DEKLARASI MATRIK DAN VARIABEL
4  REAL :: beta,h,t0,tn,yn_lama,yn_baru,y1,y2,y1_0,y2_0_lama,y2_0_baru,eps
5  INTEGER :: n
6  end module VARIABEL
7
8  program PDB_Tinggi_Secant
9  USE VARIABEL
10 implicit none
11 real :: y2_sementara
12 integer :: iter
13 !----- TENTUKAN TITIK-TITIK BATAS & NILAI AWAL
14 PRINT *, "MASUKAN TITIK-TITIK BATAS"
15 READ *, t0, tn
16 PRINT *, "MASUKAN NILAI_NILAI BATAS"
17 READ *, y1_0,beta
18 PRINT *, "MASUKAN NILAI-NILAI AWAL"
19 READ *, y2_0_lama,y2_0_baru
20 !----- TENTUKAN JUMLAH INTERVAL
21 PRINT *, "MASUKAN JUMLAH INTERVAL"
22 READ *, n
23 !----- TENTUKAN KRITERIA ERROR
24 PRINT *, "MASUKAN KRITERIA ERROR UNTUK KONVERGENCE"
25 READ *, eps
26
27 h = (tn-t0)/n
28 do WHILE (ABS(y2_0_baru - y2_0_lama) >= eps)
29   iter = iter + 1

```

```

30      y2 sementara = y2_0 baru
31      y2_0 = y2_0_lama
32      call runge
33      yn_lama = y1
34      y2_0 = y2_0_baru
35      call runge
36      yn_baru = y1
37      print *, "yn lama =", yn_lama, "yn baru =", yn_baru
38      y2_0 baru = y2_0 baru - (yn_baru - beta)*(y2_0 baru - y2_0 lama) / &
39                ((yn_baru - beta) - (yn_lama - beta))
40      y2_0 lama = y2 sementara
41      print *, "y2_lama=", y2_0_lama, "y2_baru =", y2_0_baru
42  end do
43
44  call Runge
45
46  contains
47  subroutine Runge
48  INTEGER :: i
49  REAL :: K1, K2, K3, K4, L1, L2, L3, L4, t
50  y1 = y1_0
51  y2 = y2_0
52  t = t0
53  do i=1,n
54      K1 = f(t, y1, y2)
55      L1 = g(t, y1, y2)
56      K2 = f(t+0.5*h, y1+0.5*K1*h, y2+0.5*L1*h)
57      L2 = g(t+0.5*h, y1+0.5*K1*h, y2+0.5*L1*h)
58      K3 = f(t+0.5*h, y1+0.5*K2*h, y2+0.5*L2*h)
59      L3 = g(t+0.5*h, y1+0.5*K2*h, y2+0.5*L2*h)
60      K4 = f(t+h, y1+K3*h, y2+L3*h)
61      L4 = g(t+h, y1+K3*h, y2+L3*h)
62      y1 = y1 + h * (K1 + 2*K2 + 2*K3 + K4) / 6.0
63      y2 = y2 + h * (L1 + 2*L2 + 2*L3 + L4) / 6.0
64      t = t0 + i * h
65      print *, t, y1
66  end do
67  end subroutine Runge
68
69  real function f(t, y1, y2)
70  REAL, INTENT(IN) :: t, y1, y2
71  f = y2
72  end function f
73
74  real function g(t, y1, y2)
75  REAL, INTENT(IN) :: t, y1, y2
76  g = -y2*y2 - y1 + log(t)
77  end function g
78
79  end program PDB_Tinggi_Secant
80

```

10.4 METODE PERBEDAAN HINGGA (FINITE DIFFERENCE METHOD) LINIER

Metode perbedaan hingga adalah metode yang sangat populer. Pada intinya metode ini merubah problem PDB nilai batas dari sebuah problem kalkulus menjadi sebuah problem aljabar. Dengan metode ini y' dan y'' pada (10-6) kita aproksimasikan dengan menggunakan deret Taylor sebagai berikut,

$$y(x+h) = y(x) + h y'(x) + \frac{h^2}{2} y''(x) + \dots \quad (10-26)$$

$$y(x-h) = y(x) - h y'(x) + \frac{h^2}{2} y''(x) + \dots \quad (10-27)$$

Kalau kita kurangi (10-26) dengan (10-27) dan nilai setelah pangkat 2 diabaikan akan didapat

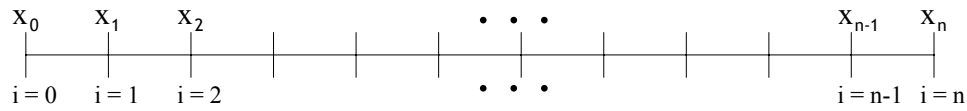
$$y'(x) = \frac{y(x+h) - y(x-h)}{2h} + O(h^2) \quad (10-28)$$

Sedangkan kalau (10-26) ditambah dengan (10-27) , kita akan peroleh

$$y''(x) = \frac{y(x+h) - 2y(x) + y(x-h)}{h^2} + O(h^2) \quad (10-29)$$

Persamaan (10-26) – (10-29) dapat diterapkan dengan membagi $[x_0, x_n]$ (lihat gambar 10.2) menjadi n bagian dengan interval h ,

$$h = \frac{x_n - x_0}{n} \quad (10-30)$$



Gambar 10.2 Pembagian interval antara $[x_0, x_n]$

Dengan metode perbedaan hingga, yang kita cari adalah nilai y pada x tertentu ,

$$x_{i+1} = x_i + h \quad (10-31)$$

Jika $i = 0$, maka $x_1 = x_0 + h$. Dengan menggunakan notasi ini (10-28) dan (10-29) dapat dituliskan

$$y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1}))}{2h} \quad (10-32)$$

$$y''(x_i) = \frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1}))}{h^2} \quad (10-33)$$

Persamaan (10-32) dan (10-33) dikenal dengan aproksimasi perbedaan hingga tiga titik (central three points finite difference approximation, lihat bab 6). Selanjutnya jika kita substitusikan (10-32) dan (10-33) ke (10-6) maka kita dapatkan

$$\left[1 - \frac{1}{2}hp(x_i)\right]y(x_{i-1}) - \left[2 - h^2q(x_i)\right]y(x_i) + \left[1 + \frac{1}{2}hp(x_i)\right]y(x_{i+1}) = h^2F(x_i) \quad (10-34)$$

atau biasa disederhanakan

$$\left[1 - \frac{1}{2}hp(x_i)\right]y_{i-1} - \left[2 - h^2q(x_i)\right]y_i + \left[1 + \frac{1}{2}hp(x_i)\right]y_{i+1} = h^2F(x_i) \quad (10-35)$$

Persamaan (10-35) diterapkan pada setiap titik diskretisasi yaitu $i = 1, 2, \dots, n-1$ sehingga terbentuk SPL dengan bentuk tri-diagonal yang dapat dipecahkan dengan algoritma Thomas (lihat bab 2).

----- CONTOH 10.4 -----

Pecahkan PDB nilai batas berikut

$$y'' + 7y' + 3y = 1$$

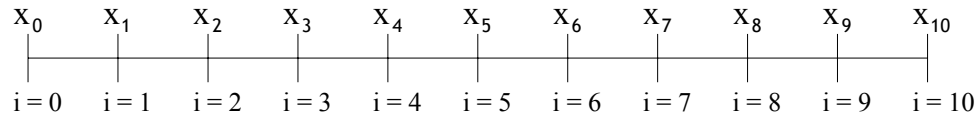
antara $[0, 1]$ dengan nilai batas $y(0) = 0$ dan $y(1) = 1$. Gunakan metode perbedaan hingga dengan $h = 0,1$, pada soal ini $p(x) = 7$, $q(x) = 3$ dan $f(x) = 1$. Persamaan perbedaan hingga untuk contoh ini adalah (10-45),

$$(1 - 3,5h)y_{i-1} - (2 - 3h^2)y_i + (1 + 3,5h)y_{i+1} = h^2$$

dengan $h = 0,1$

$$0,65 y_{i-1} - 1,97 y_i + 1,35 y_{i+1} = 0,01$$

Pembagian interval pada contoh ini adalah



Persamaan perbedaan dapat dibentuk hanya pada $i = 1 - 9$, karena pada $i = 0$ dan 10 nilai y adalah nilai batas.

$$\begin{array}{lcl}
 i = 1 \rightarrow & -1,97y_1 & +1,35y_2 & = 0,01 - 0,65y_0 \\
 i = 2 \rightarrow & 0,65y_1 & -1,97y_2 & +1,35y_3 & = 0,01 \\
 i = 3 \rightarrow & & 0,65y_2 & -1,97y_3 & +1,35y_4 & = 0,01 \\
 i = 4 \rightarrow & & & 0,65y_3 & -1,97y_4 & +1,35y_5 & = 0,01 \\
 i = 5 \rightarrow & & & & 0,65y_4 & -1,97y_5 & +1,35y_6 & = 0,01 \\
 i = 6 \rightarrow & & & & & 0,65y_5 & -1,97y_6 & +1,35y_7 & = 0,01 \\
 i = 7 \rightarrow & & & & & & 0,65y_6 & -1,97y_7 & +1,35y_8 & = 0,01 \\
 i = 8 \rightarrow & & & & & & & 0,65y_7 & -1,97y_8 & +1,35y_9 & = 0,01 \\
 i = 9 \rightarrow & & & & & & & & 0,65y_8 & -1,97y_9 & = 0,01 - 1,35y_{10}
 \end{array}$$

Dalam bentuk matrik, vektor y merupakan vektor solusi yang dapat dicari dengan menggunakan algoritma Thomas.

$$\begin{bmatrix} -1,97 & 1,35 & 0 & \dots & & & & & 0 \\ 0,65 & -1,97 & 1,35 & 0 & \dots & & & & 0 \\ 0 & 0,65 & -1,97 & 1,35 & 0 & \dots & & & 0 \\ 0 & 0 & 0,65 & -1,97 & 1,35 & 0 & \dots & & 0 \\ 0 & \dots & 0 & 0,65 & -1,97 & 1,35 & 0 & \dots & 0 \\ 0 & & \dots & 0 & 0,65 & -1,97 & 1,35 & 0 & 0 \\ 0 & & & \dots & 0 & 0,65 & -1,97 & 1,35 & 0 \\ 0 & & & & \dots & 0 & 0,65 & -1,97 & 1,35 \\ 0 & & & & & \dots & 0 & 0,65 & -1,97 \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \end{pmatrix} = \begin{pmatrix} 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ -1,34 \end{pmatrix}$$

Hasil yang diperoleh adalah

$$\mathbf{y}^T = [0,6420, 0,9443, 1,0762, 1,1233, 1,1284, 1,1131, 1,0885, 1,0598, 1,0299]$$

Pada contoh 10.4, nilai batas yang dipecahkan adalah tipe Dirichlet. Dengan tipe ini, terlihat bahwa persamaan diskretisasi (10-35) pada $i = 1$ dan $i = n-1$ mengalami sedikit modifikasi yaitu

$$i = 1 \quad - \left[2 - h^2 q(x_1) \right] y_1 + \left[1 + \frac{1}{2} h p(x_1) \right] y_2 = h^2 F(x_1) - \left[1 - \frac{1}{2} h p(x_1) \right] y_0 \quad (10-36)$$

$$i = n-1 \quad \left[1 - \frac{1}{2} h p(x_{n-2}) \right] y_{n-2} - \left[2 - h^2 q(x_{n-1}) \right] y_{n-1} = h^2 F(x_{n-1}) - \left[1 + \frac{1}{2} h p(x_{n-1}) \right] y_n \quad (10-37)$$

Program MAT10_1 memberikan program MATLAB metode perbedaan hingga dengan nilai batas Dirichlet untuk contoh 10.4.

Program MAT10_1 Program MATLAB metode perbedaan hingga dengan nilai batas Dirichlet

```

1 % MAT10_1
2 % PDB linier dengan Perbedaan Hingga
3 % Program ini memecahkan PDB nilai batas linier dengan batas Dirichlet
4
5 % Definisi problem
6 x0 = 0;
7 xn = 1;
8 n = 10;
9 p = 7.0*ones(1,n-1);
10 q = 3.0*ones(1,n-1);
11 f = ones(1,n-1);
12
13 % Kondisi batas
14 y0 = 0.;
15 yn = 1.0;
16
17 h = (xn - x0)/n;
18 h2 = 0.5*h;
19 hh = h*h;
20 x = linspace(x0+h,xn,n);
21
22 % Elemen-elemen dari matrik tri-diagonal
23 a = zeros(1,n-1);
24 b = a;
25 c = a;
26 a(2:n-1) = 1 - p(1,2:n-1)*h2;
27 d = -(2. - hh*q);
28 c(1:n-2) = 1 + p(1,1:n-2)*h2;
```

```

29 b(1) = hh*f(1) - (1 - p(1)*h2) * y0;
30 b(2:n-2) = hh*f(2:n-2);
31 b(n-1) = hh*f(n-1) - (1 + p(n-1)*h2) * yn;
32 y = tri_diag(a,d,c,b);
33
34 xx = [x0 x];
35 yy = [y0 y yn];
36 out = [xx' yy'];
37 plot(xx,yy)

```

10.5 KONDISI BATAS DERIVATIF (TIPE NEUMANN)

Pada bagian 10.2, 10.3 dan 10.4 baik metode tembak maupun metode perbedaan hingga, kondisi batas yang kita gunakan untuk memecahkan problem semuanya adalah kondisi batas tipe Dirichlet. Pada bagian ini kita akan bahas bagaimana memecahkan problem dengan kondisi batas Neumann. Kita akan pecahkan persamaan (10-6) dengan kondisi batas:

$$y(x_0) = \alpha \quad (10-38)$$

$$y'(x_n) = \beta' \quad (10-39)$$

Dengan menggunakan metode tembak linier (bagian 10.2) target saja yang berubah. Kita tetap mulai dengan estimasi $y'(x_0) = \xi_1$ kemudian $y'(x_0) = \xi_2$ kemudian nilai ξ yang tepat ditentukan oleh

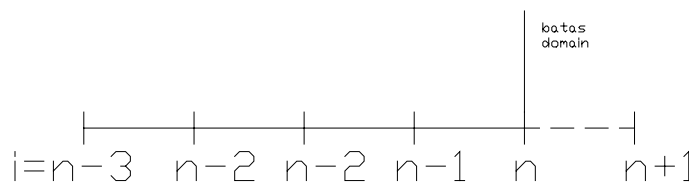
$$\xi = \xi_1 + \frac{(\xi_2 - \xi_1)}{(\beta'_2 - \beta'_1)} (\beta'_1 - \beta'_1) \quad (10-40)$$

Pada persamaan (10-40), β'_1 dan β'_2 adalah $y'(x_n)$ apabila $y'(x_0) = \xi_1$ dan ξ_2 . Untuk PDB non-linier, kita gunakan suatu variable $g(\xi)$ yang didefinisikan sebagai,

$$g(\xi) = y'_{x_n}(\xi) - \beta' \quad (10-41)$$

Nilai ξ yang benar ditentukan dengan metode secant (10-25). Kalau kita gunakan metode perbedaan hingga, persamaan (10-33) perlu dimodifikasi pada $i = n$.

$$y''(x_n) = \frac{y(x_n + h) - 2y(x_n) + y(x_n - h))}{h^2} + O(h^2) \quad (10-42)$$



Gambar 10.3 Titik $n+1$ di luar domain

Pada (10-42) kita perlu menghitung $y(x_n+h)$, tetapi karena titik ini berada diluar domain (lihat gambar 10.3) maka kita perlu mengaproksimasikan nilai $y(x_n+h)$ atau y_{n+1} . y_{n+1} kita gunakan informasi mengenai $y'(x_n) = \beta'$. Sesuai (10-28)

$$y'_n = \frac{y_{n+1} - y_{n-1}}{2h} + O(h^2) \quad (10-43)$$

Dari (10-43) kita dapatkan

$$y_{n+1} = y_{n-1} + 2hy'_n + O(h^2) \quad (10-44)$$

Dengan mensubstitusikan (10-44) ke (10-42) kita peroleh

$$y''_n = \frac{y_{n-1} + 2hy'_n - 2y_n + y_{n+1}}{h^2} + O(h^2) = \frac{2hy'_n - 2y_n + 2y_{n-1}}{h^2} + O(h) \quad (10-45)$$

Dengan mensubstitusikan (10-55) dan $y'_n = \beta$ ke (10-6) didapat

$$2y_{n-1} - \left(2 - h^2 q(x_n)\right)y_n = h^2 f(x_n) - 2h\beta - h^2 p(x_n)\beta \quad (10-46)$$

Karena diskretisasi y''_n hanya mempunyai akurasi sampai $O(h)$ maka ada kemungkinan terjadinya penurunan akurasi dari metode ini jika dibandingkan dengan metode tembak.

----- CONTOH 10.5 -----

Gunakan metode perbedaan hingga untuk memecahkan PDB nilai batas berikut (diambil dari Hoffman, 1992)

$$y'' + 5y' + 4y = 1.0$$

Antara $[0,1]$ dengan nilai-nilai batas $y(0) = 1$ dan $y'(1) = 0$. Untuk perbedaan hingga gunakan $h = 0,1$, pada soal ini $p(x) = 5$, $q(x) = 4$ dan $f(x) = 1,0$. Persamaan perbedaan hingga untuk contoh ini diperoleh dari (10-35)

$$(1 - 2.5h)y_{i-1} - (2 - 4h^2)y_i + (1 + 2.5h)y_{i+1} = h^2$$

Dengan $h = 0,1$

$$0,75 y_{i-1} - 1,96 y_i + 1,25 y_{i+1} = 0,01$$

Dengan distribusi interval sama dengan contoh 10.4. Persamaan perbedaan dapat dibentuk pada $i = 1 - 10$, karena pada $i = 0$ nilai y adalah nilai batas. Untuk $i = 1 - 9$ kita gunakan persamaan perbedaan di atas. Sedangkan untuk $i = 10$ kita gunakan (10-46) dengan $y'_{10} = 0$

$$2y_9 - 1.96y_{10} = 0,01$$

Persamaan-persamaan perbedaan yang terbentuk adalah

$$\begin{array}{rcll}
i = 1 \rightarrow & -1,96y_1 & 1,25y_2 & = 0,01 - 0,75y_0 \\
i = 2 \rightarrow & 0,75y_1 & -1,96y_2 & 1,25y_3 & = 0,01 \\
i = 3 \rightarrow & & 0,75y_2 & -1,96y_3 & 1,25y_4 & = 0,01 \\
i = 4 \rightarrow & & & 0,75y_3 & -1,96y_4 & 1,25y_5 & = 0,01 \\
i = 5 \rightarrow & & & & 0,75y_4 & -1,96y_5 & 1,25y_6 & = 0,01 \\
i = 6 \rightarrow & & & & & 0,75y_5 & -1,96y_6 & 1,25y_7 & = 0,01 \\
i = 7 \rightarrow & & & & & & 0,75y_6 & -1,96y_7 & 1,25y_8 & = 0,01 \\
i = 8 \rightarrow & & & & & & & 0,75y_7 & -1,96y_8 & 1,25y_9 & = 0,01 \\
i = 9 \rightarrow & & & & & & & & 0,75y_8 & -1,96y_9 & 1,25y_{10} & = 0,01 \\
i = 10 \rightarrow & & & & & & & & & 2y_9 & -1,96y_{10} & = 0,01
\end{array}$$

Dalam bentuk matrik vektor y merupakan vektor solusi yang dapat dicari dengan menggunakan salah satu teknik yang telah kita pelajari pada bab 2.

$$\begin{bmatrix}
-1,96 & 1,25 & 0 & \dots & 0 \\
0,75 & -1,96 & 1,25 & 0 & \dots & 0 \\
0 & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 \\
0 & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 \\
0 & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 \\
0 & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 \\
0 & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & 0 \\
0 & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 \\
0 & & & & & & \dots & 0,75 & -1,96 & 1,25 \\
0 & & & & & & & \dots & 2 & -1,96
\end{bmatrix}
\begin{bmatrix}
y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10}
\end{bmatrix}
=
\begin{bmatrix}
-0,74 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01
\end{bmatrix}$$

Dengan menggunakan algoritma Thomas, hasil yang diperoleh,

$$y^T = [0,7051, 0,5136, 0,3903, 0,3118, 0,2628, 0,2329, 0,2155, 0,2062, 0,2020, 0,2011]$$

Program MAT10_2 memberikan program MATLAB metode perbedaan hingga dengan nilai batas Neumann untuk contoh 10.5

Program MAT10_2 Program MATLAB metode perbedaan hingga dengan nilai batas Neumann

```

1 % MAT10_2
2 % PDB linier dengan Perbedaan Hingga
3 % Program ini memecahkan PDB nilai batas linier dengan batas Dirichlet
4
5 % Definisi problem
6 x0 = 0;
7 xn = 1;
8 n = 10;
9 p = 5.0*ones(1,n);
10 q = 4.0*ones(1,n);
11 f = ones(1,n);
12
13 % Kondisi batas
14 y0 = 1.;
15 ydashn = 0.;
16
17 h = (xn - x0)/n;
18 h2 = 0.5*h;
19 hh = h*h;
20 x = linspace(x0+h,xn,n);
21
22 % Elemen-elemen dari matrik tri-diagonal
23 a = zeros(1,n);
24 b = a;

```

```

25 c = a;
26 a(2:n-1) = 1 - p(1,2:n-1)*h2;
27 a(n) = 2.;
28 d = -(2. - hh*q);
29 c(1:n-1) = 1 + p(1,1:n-1)*h2;
30 b(1) = hh*f(1) - (1 - p(1)*h2) * y0;
31 b(2:n-1) = hh*f(2:n-1);
32 b(n) = hh*f(n) - 2*h*ydashn - hh*p(n)*ydashn;
33 y = tri_diag(a,d,c,b);
34 xx = [x0 x];
35 yy = [y0 y];
36 out = [xx' yy'];
37 plot(xx,yy)

```

Untuk mendapatkan akurasi yang lebih tinggi, kita gunakan formula perbedaan mundur

$$y'_i = \frac{1}{h} \left(\nabla y_i + \frac{1}{2} \nabla^2 y_i + \frac{1}{3} \nabla^3 y_i + \dots \right) \quad (10-47)$$

Dengan (10-47) nilai y'_i dengan akurasi tingkat berapapun dapat dicari. Berikut adalah formula perbedaan hingga mundur dengan akurasi $O(h)$, $O(h^2)$ dan $O(h^3)$.

Dengan akurasi $O(h)$,

$$y'_i = \frac{1}{h} \left(\nabla y_i + O(h^2) \right) = \frac{1}{h} (y_i - y_{i-1}) + O(h) \quad (10-48)$$

diperoleh

$$y_i = -y_{i-1} + h y'_i \quad (10-49)$$

Dengan akurasi $O(h^2)$,

$$\begin{aligned} y'_i &= \frac{1}{h} \left(\nabla y_i + \frac{1}{2} \nabla^2 y_i + O(h^3) \right) \\ &= \frac{1}{h} \left[y_i - y_{i-1} + \frac{1}{2} (y_i - 2y_{i-1} + y_{i-2}) \right] + O(h^2) \end{aligned} \quad (10-50)$$

diperoleh

$$y_i = \frac{1}{3} (4 y_{i-1} - y_{i-2} + 2h y'_i) \quad (10-51)$$

Dengan akurasi $O(h^3)$,

$$y'_i = \frac{1}{h} \left(\nabla y_i + \frac{1}{2} \nabla^2 y_i + \frac{1}{3} \nabla^3 y_i + O(h^4) \right)$$

$$= \frac{1}{h} \left[y_i - y_{i-1} + \frac{1}{2}(y_i - 2y_{i-1} + y_{i-2}) + \frac{1}{3}(y_i - 3y_{i-1} + 3y_{i-2} - y_{i-3}) \right] + O(h^3)$$

(10-52)

diperoleh

$$y_i = \frac{1}{11} (18 y_{i-1} - 9 y_{i-2} + 2 y_{i-3} + 6h y'_i)$$

(10-53)

----- CONTOH 10.6 -----

Gunakan ekstrapolasi dari titik di dalam ke $i = n$ tingkat 1,2 dan 3 untuk memecahkan problem pada contoh 10.5

Untuk $i = 1 - 9$ persamaan-persamaan yang didapat sama dengan contoh 10.5 sedangkan untuk $i = 10$ persamaan yang didapat adalah

Tingkat 1, $O(h)$

$$i = 10 \rightarrow -y_9 + y_{10} = 0$$

Sehingga SPL yang harus dipecahkan adalah

$$\begin{bmatrix} -1,96 & 1,25 & 0 & \dots & & & & & & & \\ 0,75 & -1,96 & 1,25 & 0 & \dots & & & & & & \\ 0 & 0,75 & -1,96 & 1,25 & 0 & \dots & & & & & \\ 0 & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & & & \\ 0 & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & & \\ 0 & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & \\ 0 & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & 0 & \\ 0 & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \\ 0 & & & & & & \dots & 0,75 & -1,96 & 1,25 & \\ 0 & & & & & & & \dots & -1 & 1 & \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} -0,74 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,0 \end{pmatrix}$$

Dengan menggunakan algoritma Thomas, hasil yang diperoleh

$$\mathbf{y}^T = [0,6973, 0,5014, 0,3758, 0,2964, 0,2473, 0,2179, 0,2013, 0,1929, 0,1897, 0,1897]$$

Tingkat 2, $O(h^2)$

$$i = 10 \rightarrow 0,333333 y_8 - 1,333333 y_9 + y_{10} = 0$$

Sehingga SPL yang harus dipecahkan adalah

$$\begin{bmatrix}
 -1,96 & 1,25 & 0 & \dots & & & & & & \\
 0,75 & -1,96 & 1,25 & 0 & \dots & & & & & \\
 0 & 0,75 & -1,96 & 1,25 & 0 & \dots & & & & \\
 0 & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & & \\
 0 & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & \\
 0 & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & \\
 0 & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots \\
 0 & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 \\
 0 & & & & & & \dots & 0,75 & -1,96 & 1,25 \\
 0 & & & & & & & \dots & 0,333333 & -1,333333 & 1
 \end{bmatrix}
 \begin{pmatrix}
 y_1 \\
 y_2 \\
 y_3 \\
 y_4 \\
 y_5 \\
 y_6 \\
 y_7 \\
 y_8 \\
 y_9 \\
 y_{10}
 \end{pmatrix}
 =
 \begin{pmatrix}
 -0,74 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,0
 \end{pmatrix}$$

Sistim ini dapat dijadikan tri-diagonal dengan mengurangi lajur 10 dengan $(0,333333/0,75)$ lajur 9.

$$\begin{bmatrix}
 -1,96 & 1,25 & 0 & \dots & & & & & & \\
 0,75 & -1,96 & 1,25 & 0 & \dots & & & & & \\
 0 & 0,75 & -1,96 & 1,25 & 0 & \dots & & & & \\
 0 & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & & \\
 0 & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & \\
 0 & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & \\
 0 & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots \\
 0 & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 \\
 0 & & & & & & \dots & 0,75 & -1,96 & 1,25 \\
 0 & & & & & & & \dots & -0,462223 & 0,444445
 \end{bmatrix}
 \begin{pmatrix}
 y_1 \\
 y_2 \\
 y_3 \\
 y_4 \\
 y_5 \\
 y_6 \\
 y_7 \\
 y_8 \\
 y_9 \\
 y_{10}
 \end{pmatrix}
 =
 \begin{pmatrix}
 -0,74 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 -0,00444
 \end{pmatrix}$$

Dengan menggunakan algoritma Thomas, hasil yang diperoleh

$$\mathbf{y}^T = [0,7101, 0,5214, 0,3995, 0,3216, 0,2725, 0,2424, 0,2245, 0,2146, 0,2098, 0,2082]$$

Tingkat 3, $O(h^3)$

$$i = 10 \rightarrow -0,181818 y_7 + 0,818182 y_8 - 1,636464 y_9 + y_{10} = 0$$

Sehingga SPL yang harus dipecahkan adalah

$$\begin{bmatrix}
 -1,96 & 1,25 & 0 & \dots & & & & & & & 0 \\
 0,75 & -1,96 & 1,25 & 0 & \dots & & & & & & 0 \\
 0 & 0,75 & -1,96 & 1,25 & 0 & \dots & & & & & 0 \\
 0 & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & & & 0 \\
 0 & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & & 0 \\
 0 & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & & 0 \\
 0 & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 \\
 0 & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & 0 \\
 0 & & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 \\
 0 & & & & & & & \dots & 0,75 & -1,96 & 1,25 \\
 0 & & & & & & & & -0,18182 & 0,818182 & -1,636364 & 1
 \end{bmatrix}
 \begin{pmatrix}
 y_1 \\
 y_2 \\
 y_3 \\
 y_4 \\
 y_5 \\
 y_6 \\
 y_7 \\
 y_8 \\
 y_9 \\
 y_{10}
 \end{pmatrix}
 =
 \begin{pmatrix}
 -0,74 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,01 \\
 0,0
 \end{pmatrix}$$

Dengan menggunakan metode Gauss Jordan, hasil yang diperoleh

$$\mathbf{y}^T = [0,7047, 0,5130, 0,3896, 0,3110, 0,2620, 0,2321, 0,2148, 0,2055, 0,2014, 0,2005]$$

Untuk $O(h^3)$ matrik tidak diagonal tetapi bisa dijadikan tri-diagonal dengan operasi lajur.

10.6 KONDISI BATAS GABUNG (TIPE ROBIN)

Jika kondisi batas pada batas sebelah kanan diberikan oleh (10-2b), maka kita bisa mengekspresikan dengan menggunakan formula perbedaan hingga mundur. Dengan menggunakan formula dengan akurasi $O(h)$, persamaan (10-2b) pada $i = n$ dapat dituliskan,

$$A_2 y_n + B_2 \left(\frac{y_n - y_{n-1}}{h} \right) = \beta \quad (10-54)$$

atau

$$y_n = \frac{\beta h + B_2 y_{n-1}}{(A_2 h + B_2)} \quad (10-55)$$

Dengan cara yang sama akurasi yang lebih baik dapat diperoleh jika kita menggunakan (10-50) atau (10-52).

CONTOH 10.7

Pecahkan PDB dengan kondisi batas tipe Robin (diambil dari Hoffman, 1992)

$$y'' + 5y' + 4y = 1$$

Antara $[0,1]$ dengan kondisi batas, $y(0) = 0$ dan $y(1) - 0,5 y'(1) = 0,5$. Gunakan metode perbedaan hingga dengan $h = 0,1$. Untuk $i = 1 - 9$ persamaan-persamaan yang didapat sama dengan contoh 10.5 untuk $i = 10$ persamaan yang didapat dari (10-55) dengan $A_2 = 1$, $B_2 = -0,5$ dan $\beta = 0,5$.

$$i = 10 \rightarrow -1,25 y_9 + y_{10} = -0,125$$

Sehingga SPL yang harus dipecahkan adalah,

$$\begin{bmatrix} -1,96 & 1,25 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,75 & -1,96 & 1,25 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 & 0 & 0 & 0 & 0 \\ 0 & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 & 0 & 0 \\ 0 & & & \dots & 0,75 & -1,96 & 1,25 & 0 & \dots & 0 & 0 \\ 0 & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & 0 & 0 \\ 0 & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 & 0 \\ 0 & & & & & & \dots & 0,75 & -1,96 & 1,25 & 0 \\ 0 & & & & & & & \dots & -1,25 & 1 & 0 \end{bmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \end{pmatrix} = \begin{pmatrix} 0,0 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ 0,01 \\ -0,125 \end{pmatrix}$$

Dengan menggunakan algoritma Thomas, hasil yang diperoleh,

$$y^T = [0,2198, 0,3446, 0,4165, 0,4543, 0,4704, 0,473, 0,4675, 0,4572, 0,4444, 0,4305]$$

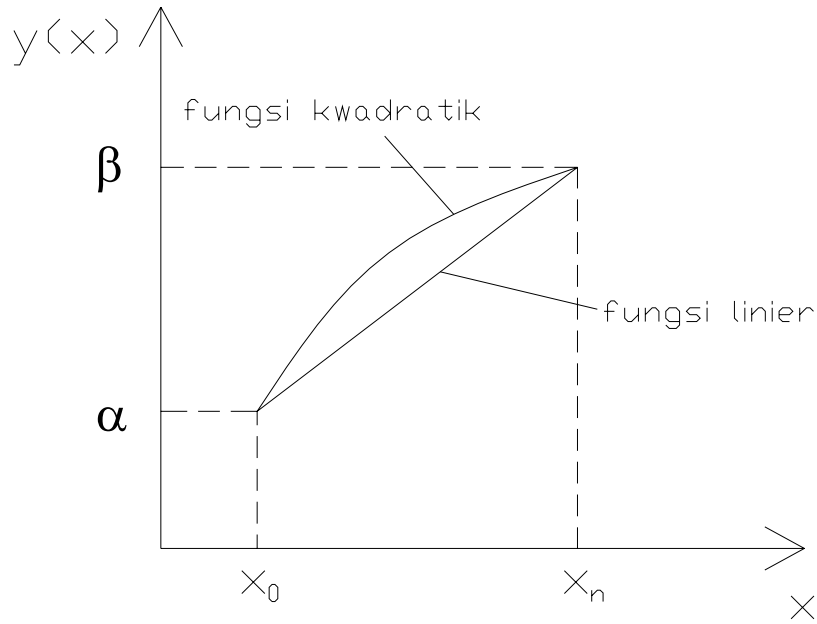
10.7 METODE PERBEDAAN HINGGA NON-LINIER

Pemecahan PDB non-linier dengan menggunakan metode perbedaan hingga sama dengan metode untuk PDB linier, hanya dalam hal ini proses iterasi harus dilakukan. Beberapa hal tentang PDB non-linier perlu kita ketahui: pertama keberadaan solusi untuk persoalan non-linier tidak selalu ada, kedua pada persoalan non-linier ada kemungkinan terdapat solusi lebih dari satu.

Pada problem non-linier karena $p(x,y)$ dan $q(x,y)$ merupakan fungsi dari x dan y , maka pada setiap tahap iterasi k , $p(x,y)$ dan $q(x,y)$ dihitung dengan menggunakan nilai y pada iterasi $k-1$ atau dengan kata lain PDB non-linier (10-1) diubah menjadi PDB linier, dengan bentuk

$$y''^{(k)} + p(x, y^{(k-1)})y'^{(k)} + q(x, y^{(k-1)})y^{(k)} = f(x) \quad (10-56)$$

Jelas guna menerapkan persamaan (10-56) kita memerlukan suatu nilai aproksimasi $y^{(0)}$ awal. Aproksimasi awal menentukan cepatnya konvergensi tercapai. Jika suatu aproksimasi awal yang buruk digunakan maka ada kemungkinan konvergensi tidak akan tercapai. Nilai aproksimasi awal yang baik dapat ditentukan dengan mengikuti bentuk fungsi linier atau kwadratik $y(x)$ antara $[x_0, x_n]$.



Gambar 10.4 Penentuan nilai aproksimasi awal

Dengan teknik ini sistim persamaan linier yang berbeda-beda akan dipecahkan beberapa kali. Meskipun teknik ini mudah tetapi teknik iterasi eksplisit lebih sering diterapkan [Fausett, 1999]. Dengan teknik ini persamaan (10-1) kita atur sehingga mempunyai bentuk,

$$y'' = f(x) - p(x, y)y' - q(x, y)y \quad (10-57)$$

Setelah y'' dan y' diaproksimasikan dengan perbedaan hingga kita dapatkan

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = f(x_i) - p(x_i, y_i) \frac{y_{i+1} - y_{i-1}}{2h} - q(x_i, y_i)y_i \quad (10-58)$$

Dengan mengatur (10-58) y_i dapat diekspresikan secara eksplisit

$$y_i = \frac{y_{i+1} + y_{i-1} - h^2 f(x_i) + hp(x_i, y_i) \frac{y_{i+1} - y_{i-1}}{2} + h^2 q(x_i, y_i)y_i}{2} \quad (10-59)$$

Atau jika kita tambahkan ωy_i pada ke dua sisi

$$y_i = \frac{1}{4(1+\omega)} \left[2y_{i+1} + 4\omega y_i + 2y_{i-1} - 2h^2 f(x_i) + hp(x_i, y_i)(y_{i+1} - y_{i-1}) + 2h^2 q(x_i, y_i)y_i \right] \quad (10-60)$$

y_i baru dapat dihitung dari y_i lama pada sebelah kanan (10-60), atau

$$y_i^{(k+1)} = \frac{1}{4(1+\omega)} \left[2y_{i+1}^{(k)} + 4\omega y_i^{(k)} + 2y_{i-1}^{(k)} - 2h^2 f(x_i) + hp(x_i, y_i^{(k)})(y_{i+1}^{(k)} - y_{i-1}^{(k)}) + 2h^2 q(x_i, y_i^{(k)})y_i^{(k)} \right] \quad (10-61)$$

Proses iterasi ini dilanjutkan sampai kriteria konvergensi tercapai. Salah satu kriteria konvergensi yang biasa digunakan adalah

$$\|y_i^{k+1} - y_i^k\|_{\infty} < \varepsilon \quad (10-62)$$

----- CONTOH 10.8 -----

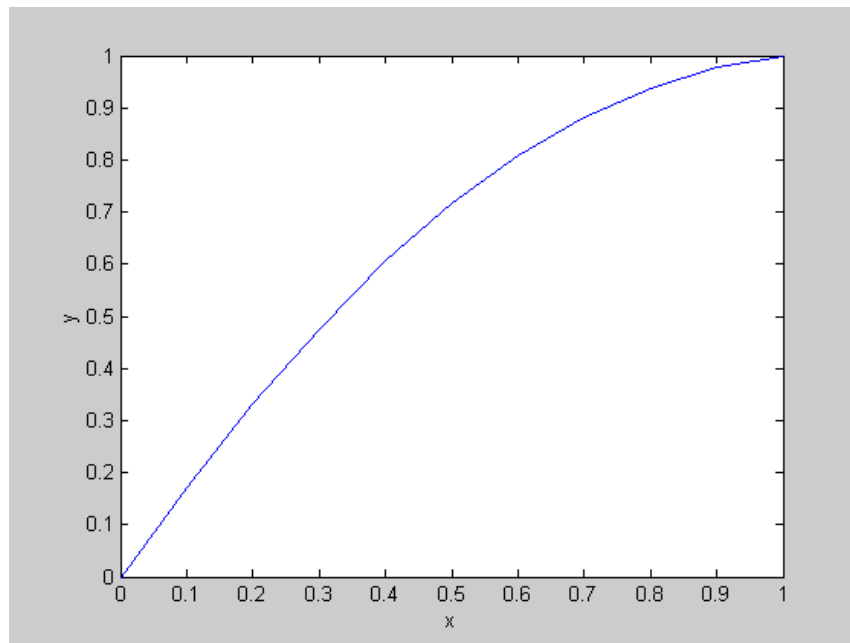
Pecahkan PDB non-linier berikut(diambil dari Hoffman, 1992)

$$y'' + (1+y)y' + (1+y)y = 1$$

Antara $[0,1]$ dengan kondisi batas, $y(0) = 0$ dan $y(1) = 1$. $p(x,y) = q(x,y) = 1 + y$ dan $f(x) = 1$. Gunakan metode perbedaan hingga eksplisit dengan $h = 0,1$ dengan aproksimasi awal y_i^0 untuk $i = 1-9$ yang didapat dalam bentuk linier $y^0 = [0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9]$. y secara eksplisit diberikan oleh (10-58) yang untuk soal in mempunyai bentuk

$$\begin{aligned} y_i^{(k+1)} &= \frac{1}{4(1+\omega)} \left[2y_{i+1}^{(k)} + 4\omega y_i^{(k)} + 2y_{i-1}^{(k)} - 2h^2 + h(1+y_i^{(k)})(y_{i+1}^{(k)} - y_{i-1}^{(k)}) + 2h^2(1+y_i^{(k)})y_i^{(k)} \right] \\ &= \frac{1}{4(1+\omega)} \left[2y_{i+1}^{(k)} + 4\omega y_i^{(k)} + 2y_{i-1}^{(k)} - 0.02 + 0.1(1+y_i^{(k)})(y_{i+1}^{(k)} - y_{i-1}^{(k)}) + 0.02(1+y_i^{(k)})y_i^{(k)} \right] \end{aligned}$$

Hasilnya diberikan pada gambar 10.5.



Gambar 10.5 Solusi PDB non-linier dengan metode eksplisit contoh 10.8

Program MAT10_3 memberikan program MATLAB metode perbedaan hingga non-linier eksplisit untuk contoh 10.8

Program MAT10_3 Program MATLAB metode perbedaan hingga non-linier eksplisit

```

1 % MAT10_3
2 % PDB non-linier dengan Perbedaan Hingga
3 % Program ini memecahkan PDB nilai batas non-linier dengan batas Dirichlet
4 % secara eksplisit
5 % program untuk contoh 10.8
6 %  $p(x,y) = 1 + y$ 
7 %  $q(x,y) = 1 + y$ 
8 % Definisi problem
9 x0 = 0;
10 xn = 1;
11 n = 10;
12 max iter = 100;
13 omega = 0.1;
14 tol = 0.001;
15 f = 1.;
16 % Kondisi batas
17 y0 = 0.;
18 yn = 1.;
19
20 h = (xn - x0)/n;
21 hh = h*h;
22 x = linspace(x0+h,xn-h,n-1);
23 ww = 1/(4.*(1+omega));
24
25 % Aproksimasi y awal secara linier
26 deltay = (yn - y0) / n;
27 ylama(1) = y0;
28 ylama(n+1) = yn;
29 ybaru(1) = ylama(1);
30 ybaru(n+1) = ylama(n+1);
31 for j = 2:n
32     ylama(j) = y0 + deltay * (j-1);
33 end
34
35 iter = 1;
36 % Iterasi y baru

```

```

37 while (iter <= max iter)
38     for i = 2:n
39         p = 1 + ylama(i);
40         q = 1 + ylama(i);
41         ybaru(i) = ww * (2.*ylama(i+1) + 4*omega*ylama(i) + 2.*ylama(i-1)-...
42                     2.*hh*f + h*p*(ylama(i+1) - ylama(i-1)) + 2*hh*q*ylama(i));
43     end
44     if norm(ybaru - ylama) <= tol;
45         disp('Convergence has been reached');
46         xx = [x0 x xn];
47         yy = [ybaru];
48         plot(xx,yy);
49         return;
50     end
51     ylama = ybaru;
52     iter = iter + 1;
53 end

```

10.8 CONTOH-CONTOH APLIKASI

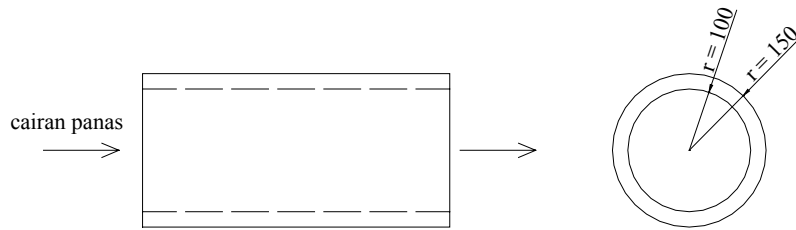
Beberapa contoh problem engineering yang berupa PDB nilai batas akan kita bahas pada bagian ini.

----- CONTOH 10.9 -----

(Hoffman, 1992) Distribusi suhu di dalam dinding pipa (gambar 10.6) yang dialiri oleh cairan panas dapat ditentukan dengan menggunakan PDB

$$\frac{d^2 T}{dr^2} + \frac{1}{r} \frac{dT}{dr} = 0$$

Jika suhu pada $T(100) = 100^\circ\text{C}$ dan $T(150) = 0^\circ\text{C}$ tentukan distribusi suhu di dalam pipa dengan menggunakan $h = 10$.



Gambar 10.6 Aliran cairan panas dalam sebuah pipa

Pada contoh ini $p(r) = 1/r$, $q(r) = 0$ dan $f(r) = 0$. Persamaan perbedaan hingga untuk problem ini adalah,

$$\left[1 - \frac{1}{2r_i} h\right] t_{i-1} - 2t_i + \left[1 + \frac{1}{2r_i} h\right] t_{i+1} = 0$$

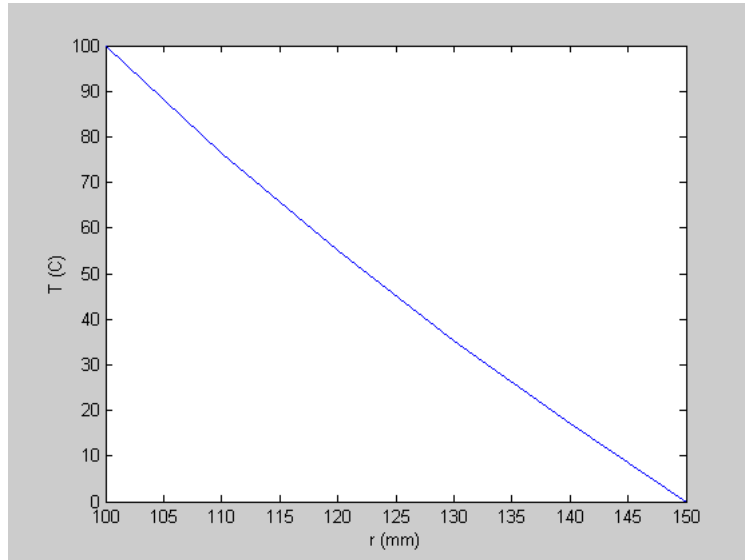
atau

$$\left[1 - \frac{5}{r_i}\right] t_{i-1} - 2t_i + \left[1 + \frac{5}{r_i}\right] t_{i+1} = 0 \quad (\text{C-1})$$

penerapan (C-1) pada $i = 1 - 4$ memberikan SPL berikut

$$\begin{aligned} i = 1 &\rightarrow \begin{bmatrix} -2 & 1,0455 & 0 & 0 \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \end{bmatrix} = \begin{bmatrix} -95,5 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ i = 2 &\rightarrow \begin{bmatrix} 0,958 & -2 & 1,0416 & 0 \end{bmatrix} \\ i = 3 &\rightarrow \begin{bmatrix} 0 & 0,962 & -2 & 1,0385 \end{bmatrix} \\ i = 4 &\rightarrow \begin{bmatrix} 0 & 0 & 0,964 & -2 \end{bmatrix} \end{aligned}$$

Hasil temperatur diplot pada gambar 10.7.



Gambar 10.7 Profil temperatur pada pipa.

----- CONTOH 10.10 -----

Besarnya lengkungan balok yang tertopang secara sederhana (gambar 10.8) diberikan oleh PDB

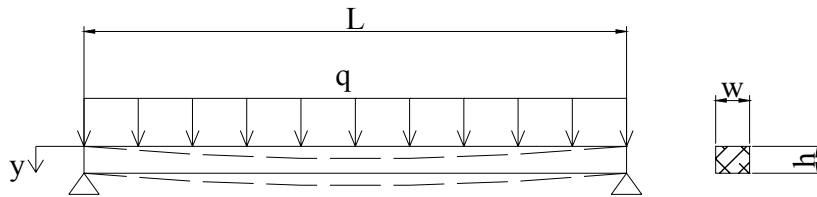
$$EI \frac{d^2 y}{dx^2} = -\frac{qLx}{2} + \frac{qx^2}{2}$$

Dimana q adalah besaran beban seragam, L adalah panjang dari balok, I adalah besarnya momen inertia penampang balok (untuk penampang persegi $I = wh^3/12$ dimana w adalah lebar dan h tinggi balok) dan E adalah elastisitas modulus. Untuk balok dengan $E = 10000 \text{ kN/m}^2$, $L = 2 \text{ m}$, $w = 5 \text{ cm}$, $h = 10 \text{ cm}$ dan beban $q = 1500 \text{ N/m}$. $I = 4,166 \times 10^{-6} \text{ m}^4$, tentukan distribusi besarnya lengkungan $y(x)$.

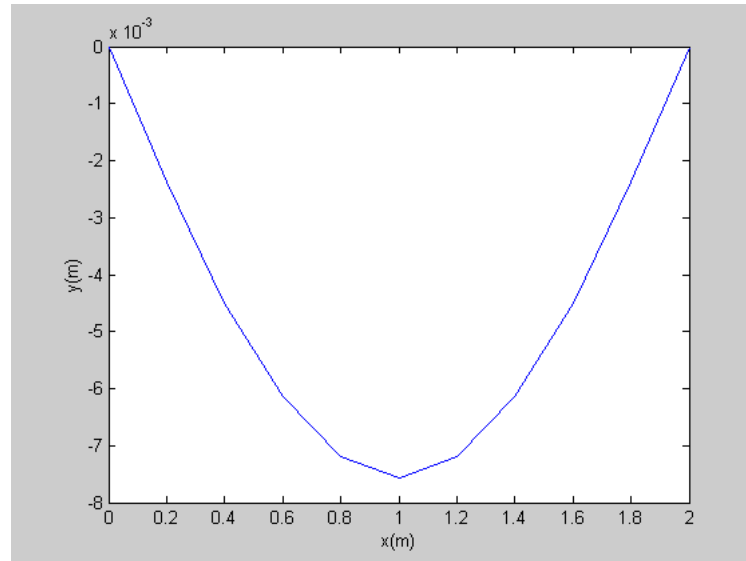
Jelas bahwa kondisi batas problem ini adalah kondisi batas Dirichlet $y(0) = y(2) = 0$. Dengan menggunakan data-data persamaan PDBnya dapat dituliskan

$$y'' = \frac{Vx}{2EI} (x - L) = 0,018x(x - 2) \quad (\text{C-1})$$

Dari (C-1) tampak bahwa $p(x) = 0$, $q(x) = 0$ dan $f(x) = 0,018 x (x - 2)$. Jika balok dibagi dalam 10 interval maka $h = 0,2$. Hasil yang diperoleh digambarkan pada 10.9.



Gambar 10.8 Balok tertopang sederhana



Gambar 10.9 Defleksi balok contoh 10.10

10.9 SOAL-SOAL LATIHAN

10.1 Gunakan metode tembak linier dan perbedaan hingga linier untuk memecahkan soal-soal nilai batas berikut (Faires & Burden, 1993).

PDB	Batas	Nilai batas	solusi analitik	h
$y'' = 4(y - x)$	$[0,1]$	$y(0) = 0$ $y(1) = 2$	$y(x) = \frac{e^2}{e^4 - 1}(e^{2x} - e^{-2x}) + x$	0,1 0,2
$y'' = y' + 2y + \cos x$	$[0, \pi/2]$	$y(0) = -0,3$ $y(\pi/2) = -0,1$	$y(x) = -(1/10)(\sin x + 3 \cos x)$	$\pi/4$ $\pi/6$
$y'' + y = 0$	$[0, \pi/4]$	$y(0) = 1$ $y(\pi/4) = 1$	$y(x) = \cos x + (\sqrt{2} - 1) \sin x$	$\pi/20$
$y'' = -2(2x y' + y - \ln x)/x^2$	$[1,2]$	$y(1) = 0$ $y(2) = \ln 2$	$y(x) = 4x^{-1} - 2x^{-2} + \ln x - 3/2$	0,05
$y'' = 2y' - y + xe^x - x$	$[0,2]$	$y(0) = 0$ $y(2) = -4$	$y(x) = \frac{1}{6}x^3e^x - \frac{5}{3}xe^x + 2e^x - x - 2$	0,2

10.2 Gunakan metode non-linier tembak dan metode perbedaan hingga non-linier untuk memecahkan soal-soal nilai batas berikut (Faires & Burden, 1993).

PDB	Batas	Nilai batas	solusi analitik	h
$y' = y^3 - y y'$	[1,2]	$y(1) = \frac{1}{2}$ $y(2) = \frac{1}{3}$	$y(x) = \frac{1}{x+1}$	0,1
$y'' = 2y^3 - 6y - 2x^3$	[1,2]	$y(1) = 2$ $y(2) = \frac{5}{2}$	$y(x) = x + \frac{1}{x}$	0,1
$y'' = y' + 2(y - \ln x)^3 - \frac{1}{x}$	[1,2]	$y(1) = 1$ $y(2) = 0,5 + \ln(2)$	$y(x) = \frac{1}{x} + \ln x$	0,1
$y'' = 2y' - (y')^2 - 2 + e^y + e^x(\cos x + \sin x)$	[0, $\pi/2$]	$y(0) = 0$ $y(\pi/2) = \pi/2$	$y(x) = \ln(e^x \cos x + e^x \sin x)$	$\pi/20$
$y'' = \frac{1}{3}y^3$	[1,2]	$y(1) = -\frac{2}{3}$ $y(2) = -1$	$y(x) = \frac{2}{x-4}$	0,05

10.3 Pecahkan PDB linier dengan kondisi batas derivatif (type Neumann) berikut (Hoffman, 1992).

PDB	Batas	Nilai batas	h
$y'' = -5y' - 4y + 1$	[0,1]	$y(0) = 1$ $y'(1) = 0$	0,1
$y'' = -4y' - 6,25y + 1$	[0,1]	$y(0) = 1$ $y'(1) = 0$	0,1
$y'' = -5y' - 4y + e^x$	[0,1]	$y(0) = 1$ $y'(1) = 0$	0,1
$y'' = -(1+x)y' - (1+x)y + 1$	[0, 1]	$y(0) = 1$ $y'(1) = 0$	0,1
$y'' = -(1+x)y' - (1+x) + 2e^{x/2} + 1 + x$	[1,2]	$y(0) = 1$ $y'(1) = 0$	0,05

10.4 Pecahkan PDB linier dengan kondisi batas gabung (type Robin) berikut (Hoffman, 1992).

PDB	Batas	Nilai batas	h
$y'' = -4y' - 6,25y + e^x$	[0,1]	$y(0) = 1$ $y(1) - 0,5y'(1) = 0,5$	0,1
$y'' = -5y' - 4y + 1$	[0,1]	$y(0) = 0$ $y(1) - 0,5y'(1) = 0,5$	0,1

10.5 Buat program MATLAB untuk metode tembak linier dan non-linier.