Metode Numerik

BAB 1 PENDAHULUAN

Metode numerik adalah metode menggunakan komputer untuk mengaproksimasi solusi masalah matematika melalui kinerja dari sejumlah operasi dasar pada angka.

Alasan penggunaan metode numerik:

1. Masalah matematika sulit dicari solusi analitiknya.
   Contoh:
   Hitunglah $\int \cosh(\cos x + e^{2sinx})\, dx$,
   Carilah y sehingga $y \cos x + e^y = 0$.
2. Memerlukan perhitungan dalam skala besar dan banyak.
   Contoh: Mencari solusi dari sistem persamaan linier berukuran besar.

Tahap-tahap penyelesaian masalah secara numerik:

a. Pemodelan, yaitu memformulasikan masalah fisik menjadi masalah matematika.
b. Pemilihan metode numerik untuk menyelesaikan masalah matematika tersebut.
c. Pemrograman: pembuatan algoritma dan proses koding
d. Penafsiran hasil

Contoh:

1. Perhatikan masalah gelombang 2D berikut ini:

Difference Equations for 2D Flow Problems. ***Incompress-ble-Flow Equation.*** The development of the finite-difference :quations for 2D, horizontal reservoirs follows a process similar to hat outlined in Sec. 7.2.1. Again, consider the 2D incompressible-low equation, Eq. 4.56, or

$$\frac{\partial}{\partial x}\left(\beta_c A_x k_x \frac{\partial p}{\partial x}\right)\Delta x + \frac{\partial}{\partial y}\left(\beta_c A_y k_y \frac{\partial p}{\partial y}\right)\Delta y + \mu q_{sc} = 0. \quad \text{.................... (7.22)}$$

Masalah di atas dinyatakan dalam masalah diskrit seperti di bawah ini:

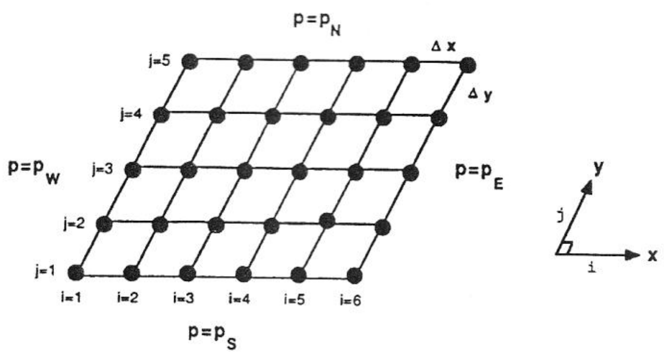


Fig. 7.5—Representation of a 2D reservoir with a mesh-centered grid.

Masalah diskrit dapat diselesaikan menggunakan persamaan matriks vektor berikut ini:

TABLE 7.1—MATRIX EQUATION REPRESENTING EQ. 7.27

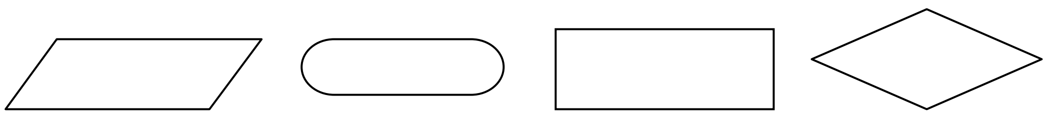
$$
\begin{bmatrix}
-4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & -4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & -4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & -4 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -4 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -4
\end{bmatrix}
\begin{bmatrix}
p_{2,2} \\ p_{3,2} \\ p_{4,2} \\ p_{5,2} \\ p_{2,3} \\ p_{3,3} \\ p_{4,3} \\ p_{5,3} \\ p_{2,4} \\ p_{3,4} \\ p_{4,4} \\ p_{5,5}
\end{bmatrix}
=
\begin{bmatrix}
-p_S - p_W \\ -p_S \\ -p_S \\ -p_S - p_E \\ -p_W \\ 0 \\ 0 \\ -p_E \\ -p_W - p_N \\ -p_N \\ -p_N \\ -p_E - p_N
\end{bmatrix}
$$

Dengan demikian solusi persamaan diferensial untuk 2D flow problems didapat dengan menyelesaikan persamaan di atas, dengan bantuan komputer.

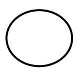
1.1 Pemrograman

Pemrograman terstruktur

Flowchart



Input/Output terminal process decision



off-page connector

Pseudocode

If/then/else/elseif

```
If condition then
        True block
Else
        False block
```

```
If condition1 then
        True block1
Elseif
        True block2
Else
        False block
```

Count-controlled loop

```
Do for i=start, finish
        Block
Enddo
```

Doexit

```
Do
        Block1
        If condition exitdo
        Block2
Loop
```

Case

```
Select case test_expression
Case value_1
        Block1
Case value_2
        Block2
Case value_3
        Block3
End select
```

Pemrograman modular: terbagi dalam modul-modul / prosedur dengan nama: Function dan Subroutine

Main program

....

....

y = Euler(dt, ti, yi)

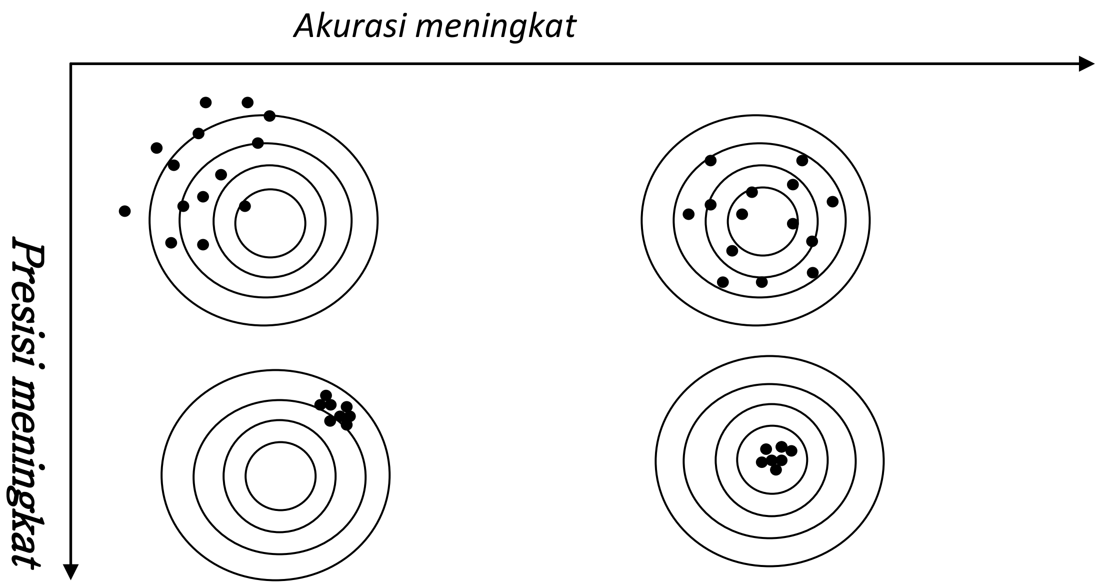
....

End main program

```
Function Euler(dt,ti,yi)
.......................................
End function
```

1.2 Aproksimasi

Dalam mencari aproksimasi dari solusi masalah fisik, perlu dipahami mengenai akurasi dan presisi. Pada gambar dibawah ini perbedaan keduanya dinyatakan dalam grafik.



Bagaimana galat (error) didefinisikan dalam suatu perhitungan? Misal A adalah solusi eksak dan $\bar{A}$ adalah solusi hasil aproksomasi. Terdapat beberapa definisi:

a. Galat : $\varepsilon = \bar{A} - A$
b. Galat mutlak/absolut: $\varepsilon_a = |\bar{A} - A|$
c. Galat relatif: $\varepsilon_r = \frac{|\bar{A}-A|}{|A|}$

Dalam prakteknya, mendapatkan batas atas dari galat mutlak adalah penting $|\bar{A} - A| < E$, dimana E harus kecil. Galat relatif biasanya dinyatakan dalam persentasi. Besar galat yang sama dari solusi eksak yang berbeda tidak menyatakan kualitas aproksimasi yang sama. Perhatikan di bawah ini:

A	$\varepsilon = \bar{A} - A$	$\varepsilon_r = \frac{\lvert\bar{A} - A\rvert}{\lvert A\rvert}$
1000	10^{-5}	10^{-8} (sangat bagus)
1	10^{-5}	10^{-5} (bagus)
10^{-5}	10^{-5}	1 (jelek)

Sumber galat dalam komputasi:

1. Kesalahan manusia (human error)
   Contoh: kesalahan penjumlahan, kesalahan pemrograman
2. Galat pemotongan
3. Galat pembulatan

Pada perhitungan selanjutnya human error diasumsikan tidak ada karena sulit dideteksi kecuali hasil perhitungan jelas salah. Oleh karena itu penting untuk memeriksa kembali perhitungan.

1.3 Galat pemotongan

Galat pemotongan terjadi bila sejumlah tak hingga proses diaproksimasi dengan sejumlah hingga proses, karena komputer tidak dapat melakukan proses yang tak hingga jumlahnya. Contoh pada perhitungan nilai $e^{0,1}$ menggunakan ekspansi Taylor :

$$f = e^{0,1} \approx 1 + 0,1 + \frac{(0,1)^2}{2!} + \frac{(0,1)^3}{3!} + \cdots$$

Kapan kita berhenti menghitung? Tidak akan pernah berhenti. Apabila berhenti, maka kita tidak mendapat nilai eksak. Misal perhitungan berhenti pada suku ke-5 sehingga $\bar{f} \approx 1,105$ maka terdapat galat pemotongan (truncation error) yaitu: $\varepsilon_{TE} = \bar{f} - f = -\frac{(0,1)^5}{5!} - \frac{(0,1)^6}{6!} - \cdots$

Biasanya galat ini sulit dihitung. Dalam kasus ini, galat pemotongannya dapat diaproksimasi juga yaitu:

$$|\varepsilon_{TE}| = \frac{(0,1)^5}{5!}\left(1 + \frac{0.1}{6} + \frac{(0,1)^2}{6\times 7} + \frac{(0,1)^3}{6\times 7\times 8} - \cdots\right)$$
$$\leq \frac{(0,1)^5}{5!}\left(1 + 0.1 + \frac{(0,1)^2}{1\times 2} + \frac{(0,1)^3}{1\times 2\times 3} - \cdots\right)$$

$\leq \frac{(0,1)^5}{5!} e^{0.1} \approx \frac{(0,1)^5}{5!}. 1,105 \approx 10^{-7}$

http://personal.fmipa.itb.ac.id/novriana

1.4 Galat pembulatan

Perhatikan perhitungan dari sebelumnya:

$$\bar{f} = 1 + 0,1 + \frac{(0,1)^2}{2!} + \frac{(0,1)^3}{3!} + \frac{(0,1)^4}{4!}$$

$$= 1 + 0,1 + 0,005 + 0,000166\dot{6} + 0,00000416\dot{6} = 1.10517083\dot{3}$$

dimana suku ke-4 dan ke-5 adalah pembulatan sampai 4 angka signifikan. Hasil $\bar{f}$ dibulatkan sampai 6 desimal (6 angka di belakang koma) adalah 1.105171. Perhatikan bila desimal ke-7 adalah 1,2,3 atau 4, maka dibulatkan ke bawah, sedangkan bila 5,6,7,8,dan 9 maka dibulatkan ke atas. Galat pembulatan ε_{RE} sampai 6 desimal dibatasi oleh $\frac{1}{2}10^{-6}$. Galat total yang terjadi adalah:

$$|\varepsilon| \leq |\varepsilon_{RE}| + |\varepsilon_{TE}| \leq \frac{1}{2}10^{-6} + 10^{-7} \approx \frac{1}{2}10^{-6}.$$

Jadi galat yang mendominasi adalah galat pembulatan.

1.5 Perhitungan Langsung dan Tak Langsung

Perhitungan langsung adalah proses pencarian solusi menggunakan rumus-rumus yang ada tanpa memerlukan tebakan awal dan proses iterasi. Biasanya perhitungan langsung menghasilkan solusi eksak. Sedangkan perhitungan tak langsung memerlukan tebakan awal dan proses perhitungan diulang (iterasi) untuk memperbaiki nilai tebakan sehingga galat yang diberikan cukup kecil.

Latihan:

1. Hitung nilai $e^{0,5}$ menggunakan ekspansi Taylor sampai suku ke-4, lalu hitung galat pemotongan dan pembulatan sampai 5 angka di belakang koma.
2. Hitung nilai $\sin 0{,}2$ menggunakan ekspansi Taylor sampai suku ke-5, lalu hitung galat pemotongan dan pembulatan sampai 5 angka di belakang koma.
3. Diketahui SPL berikut:

$$\begin{array}{rcrcrcr} x_1 & - & x_2 & + & 2x_3 & = & 4 \\ -x_1 & + & 4x_2 & + & x_3 & = & -7 \\ 2x_1 & + & x_2 & + & 5x_3 & = & 5 \end{array}$$

   Hitung solusinya menggunakan cara langsung dan tak langsung dengan $x_{awal} = (1,1,1)$.