Metode Numerik untuk Solusi PDB

Heri Purnawan (Teknik Elektro UNISLA)

==

Andaikan diberikan PD tk. 1 sbb:

$y' = f(t, y)$, dengan $y(0) = y_0$

- **Metode Euler**

$y_{j+1} = y_j + (t_{j+1} - t_j) f(t_j, y_j)$

atau

$y_{j+1} = y_j + hf(t_j, y_j)$, dengan $h = t_{j+1} - t_j$

Algoritma Metode Euler:

Langkah 1. definisikan $f(t, y)$

Langkah 2. input nilai awal t_0 dan y_0

Langkah 3. input langkah waktu h and banyaknya iterasi N

Langkah 4. untuk j dari 1 sampai N lakukan

Langkah 5. $k_1 = f(t_j, y_j)$

$y_{j+1} = y_j + h \, \star \, k_1$

$t_{j+1} = t_j + h$

Langkah 6. output t dan y untuk $j = 1, \cdots, N$.

==

Contoh 1:

Diberikan PDB sbb: $y' = 1 - t + 4y$, $y(0) = 1$. Gunakan $h = 0.01$.

Solusi:

Solusi analitik: $y_e = \frac{1}{4}t - \frac{3}{16} + \frac{19}{16}e^{4t}$

Plot solusi analitik untuk y_e pada $t \in [0, 1]$ dengan $h = 0.01$.

```
h = 0.01; N = 1/h;
t = 0:0.01:1;
for j = 1:N + 1
    y_e(j) = (1/4)*t(j) - (3/16) + (19/16)*exp(4*t(j));
end
```

Langkah 1: definisikan $f(t, y) = 1 - t + 4y$

Langkah 2:

```
% input nilai awal t_0 dan y_0
t(1) = 0; y(1) = 1;
```

Langkah 3: Misalkan kita ingin mengetahui waktu akhir adalah $t_f = 1$ dengan $h = 0.01$. Jadi, banyaknya iterasi $N = \frac{1}{0.01} = 100$ (jika di MATLAB indeks dimulai dari 1 bukan dari 0.

```
% tentukan nilai h dan banyak iterasi N
h = 0.01; N = 1/h;
```

Langkah 4: j dimulai dari $1, 2, \cdots, N$

```
for j = 1:N
    % Langkah 5:
    k_1 = 1 - t(j) + 4 * y(j);
    y(j+1) = y(j) + h * k_1;
    %update nilai t
    t(j+1) = t(j) + h;
end
```

Langkah 6: hasil untuk t dan y

```
% hasil dari t
t
```

```
t = 1×101
         0    0.0100    0.0200    0.0300    0.0400    0.0500    0.0600    0.0700 ⋯
```

```
% hasil dari y
y
```

```
y = 1×101
    1.0000    1.0500    1.1019    1.1558    1.2117    1.2698    1.3301    1.3927 ⋯
```

- **Metode Runge-Kutta Orde 4**

$$y_{j+1} = y_j + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

dengan

$$k_1 = f(t_j, y_j)$$

$$k_2 = f\left(t_j + \frac{h}{2}, y_j + \frac{h}{2}k_1\right)$$

$$k_3 = f\left(t_j + \frac{h}{2}, y_j + \frac{h}{2}k_2\right)$$

$$k_4 = f(t_j + h, y_j + hk_3)$$

Algoritma Metode Runge-Kutta Orde 4:

Langkah 1 - 4 dan **Langkah 6** sama seperti metode Euler

Langkah 5. $k_1 = f(t_j, y_j)$

$$k_2 = f\left(t_j + \frac{h}{2}, y_j + \frac{h}{2}k_1\right)$$

$$k_3 = f\left(t_j + \frac{h}{2}, y_j + \frac{h}{2}k_2\right)$$

$$k_4 = f(t_j + h, y_j + hk_3)$$

$$y_{j+1} = y_j + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$t_{j+1} = t_j + h$$

===

Contoh 2:

Lakukan pekerjaan yang sama seperti **Contoh 1**.

Solusi:

Langkah 1 - 3 sama dengan code ***Euler***

```
% input nilai awal y_0
y_1(1) = 1;
```

Langkah 4:

```
for j = 1:N
    % Langkah 5:
    k_1 = 1 - t(j) + 4 * y_1(j);
    k_2 = 1 - (t(j) + h/2) + 4 * (y_1(j) + (h*k_1)/2);
    k_3 = 1 - (t(j) + h/2) + 4 * (y_1(j) + (h*k_2)/2);
    k_4 = 1 - (t(j) + h) + 4 * (y_1(j) + h*k_3);
    % hitung y
    y_1(j+1) = y_1(j) + (h/6) * (k_1 + 2*k_2 + 2*k_3 + k_4);
    %update nilai t
    t(j+1) = t(j) + h;
end
```

Plot perbadingan solusi exact, Euler, dan Runge-Kutta

```
% plot waktu versus y
```

```
plot(t,y_e, '-k' , 'LineWidth',1.5)
hold on
plot(t,y, '--b', 'LineWidth',1.5)
plot(t,y_1,':r', 'LineWidth',1.5)
xlabel('waktu (t)'); ylabel('y');
legend('exact','Euler','RK-4','Location','northwest')
xlim([0 1])
```