LAPORAN

PRAKTIKUM KOMPUTASI BIOMEDIS

Chapter 13 : ODE: Runge-Kutta's Method

Pelaksanaan Praktikum:

Hari: Selasa Tanggal: 26 November 2019 Jam ke: 9-10

Oleh:

Nama : M. Thoriqul Aziz E

NIM : 081711733002

Dosen Pembimbing : Osmalina Nur Rahma S.T., M.Si

LABORATORIUM KOMPUTER

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS AIRLANGGA

SURABAYA

2019

A. TUJUAN

Mahasiswa dapat menentukan solusi dari persamaan deferensial biasa menggunakan metode Runge-Kuttas dan membandingkan kelebihan dan kekurangan dengan metode sebelumnya.

B. DASAR TEORI

Persamaan diferensial merupakan persamaan yang sering digunakan untuk mendefinisikan suatu keaadan berdasarkan perubahan tiap limit satuan waktunya. Secara umum persamaan bisa sangat sulit untuk disederhanakan dalam penjabaran koefisiennya, sehingga dapat digunakan metode numerik untuk menebak nilai dari hasil persamaan diferensial tersebut. Beberapa Metode yang digunakan adalam Metode Runge-Kutta's (Capra, Steven C and Canale,1991).

Metode Runge-Kutta's adalah sebuah metode yang dikembangkan dar metode sebelumnya yaitu metode Heuns. Metode ini juga memilki nilai pengoreksi yang jumlahnya bergantung dari orde metode yang ingin digunakan. Secara konsep mirip dengan metode Heuns, kaan tetapi karena jumlah pengoreksi lebih banyak maka mampu memberikan nilai tebakan persamaan diferensial lebih baik baik dari metode sebelumnya

adalah persamaan Runge-Kutta's orde 4:

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)h \tag{25.40}$$

where

$$k_1 = f(x_i, y_i) \tag{25.40a}$$

$$k_2 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1h\right) \tag{25.40b}$$

$$k_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2h\right) \tag{25.40c}$$

$$k_4 = f(x_i + h, y_i + k_3h) \tag{25.40d}$$

C. TUGAS

1. Task 1:

Task

1. By using the same problem as shown in Chapter 12, please calculate the load (q) at t=2 by using the fourth order Runge-Kutta method and plot the curve of loading process (charging) towards t between t = 0s and t = 2s. Compare the result!

2. Task 2:

2. A spherical cell or particle of radius a and uniform density ρ_s, is falling under gravity in a liquid of density ρ_f and viscosity μ. The motion of the sphere, when wall effects are minimal, can be described by the equation:

$$\frac{4}{3}\pi a^3 \rho_s \frac{dv}{dt} + 6\mu\pi a v - \frac{4}{3}\pi a^3 (\rho_s - \rho_f) g = 0$$

With initial boundary conditions

$$z = 0; \quad v = \frac{dz}{dt} = 0,$$

Where z is the displacement distance of the sphere in the downward direction. Downward motion v is along the positive z direction. On the left-hand side of the

equation, the first term describes the acceleration, the second describes viscous drag, which is proportional to the sphere velocity, and the third is the force due to gravity

acting downwards from which the buoyancy force has been subtracted. Two initial conditions are included to specify the problem completely. This initial value problem can be converted from a single second-order ODE into a set of two coupled first-order ODEs. Set $y_1 = z$ and $y_2 = dz/dt$. Then:

$$\frac{dy_1}{dt} = y_2, \qquad y_1 = 0$$

$$\frac{dy_2}{dt} = \frac{(\rho_s - \rho_f)}{\rho_s} g - \frac{9\mu}{2a^2\rho_s} y2, \qquad y_2 = 0$$

The constants are set to:

$a = 10^{-4}$ cm; $\rho_s = 1.1$ g/cm^3; $\rho_f = 1$ g/cm^3; $g = 981$ cm/s^2; $\mu = 3.5 \times 10^{-2}$ g/cm.s

Plotting the numerically calculated displacement z and the velocity dz/dt with time interval 0 to 0.001 s and find the minimum step size to obtain the steady state velocity (which it does not increase or decrease).

D. PEMBAHASAN

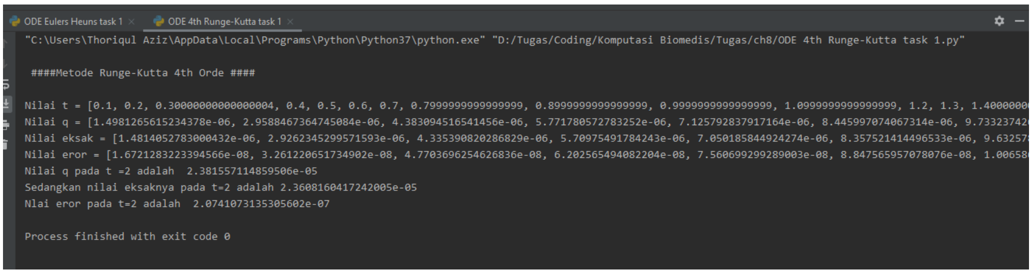
1. Dari permasalahan pertama, yaitu menentukan nilai jumlah electron/ arus (q) yang tejadi dalam rangkaian RLC yang dimulai dengan mendefinisikan terlebih dahulu persamaan fungsi deferensialnyanya. Algoritma yang

disusun pada program kali ini sama dengan metode yang diterapkan sebelumnya, yaitu pada metode Euler dan Heuns. Perbedaanya terletak pada jumlah nilai pengoreksi. Metode Rungkuta orde 4 memiliki 4 nilai yang kemudian akan saling berkorelasi satu sama lain sebagai pengoreksi untuk kemudian menebak nilai yang paling mendekati. Berikut adalah kode pemrograman yang dibuat dalam Bahasa Phyton 3.7 :

```
import math as mt
import matplotlib.pyplot as mtp
def RLC(t,q,e=12,C=5e-6,R=8e5):
    return (e/R)-(q/(R*C))
def Exact(t,C=5e-6,e=12,R=8e5):
    return C*e*(1-mt.exp(-t/(R*C)))
t1=0;t2=2;
h=0.1
t=0;q=0;
er=[];ti=[];qi=[];yi=[]
print('\n ####Metode Runge-Kutta 4th Orde ####\n')
#Runge kuta orde 4
while t1<t2:
    q1=h*RLC(t,q)
    q2=h*RLC(t+0.5*h,q+0.5*q1*h)
    q3=h*RLC(t+0.5*h,q+0.5*q2*h)
    q4=h*RLC(t+h,q+h*q3)
    q=q+(q1+2*q2+2*q3+q4)/6
    t1+=h
    t=t1
    y=Exact(t)
    eror=abs(y-q)
    ti.append(t)
    qi.append(q)
    yi.append(y)
    er.append(eror)
panjang=len(ti)
print('Nilai t =',ti,'\nNilai q =',qi,'\nNilai eksak
=',yi,'\nNilai eror =',er)
print('Nilai q pada t =2 adalah ',qi[panjang-
1],'\nSedangkan nilai eksaknya pada t=2
adalah',yi[panjang-1])
print('Nlai eror pada t=2 adalah ',er[panjang-1])
#ploting
mtp.plot(ti,qi,'g*',ti,yi,'k')
mtp.title('Grafik Hasil Runge-Kutta Orde 4')
mtp.xlabel('time(s)')
```

```
mtp.ylabel('Voltage (v)')
mtp.show()
```

Berikut adalah Hasil dari *command window* :



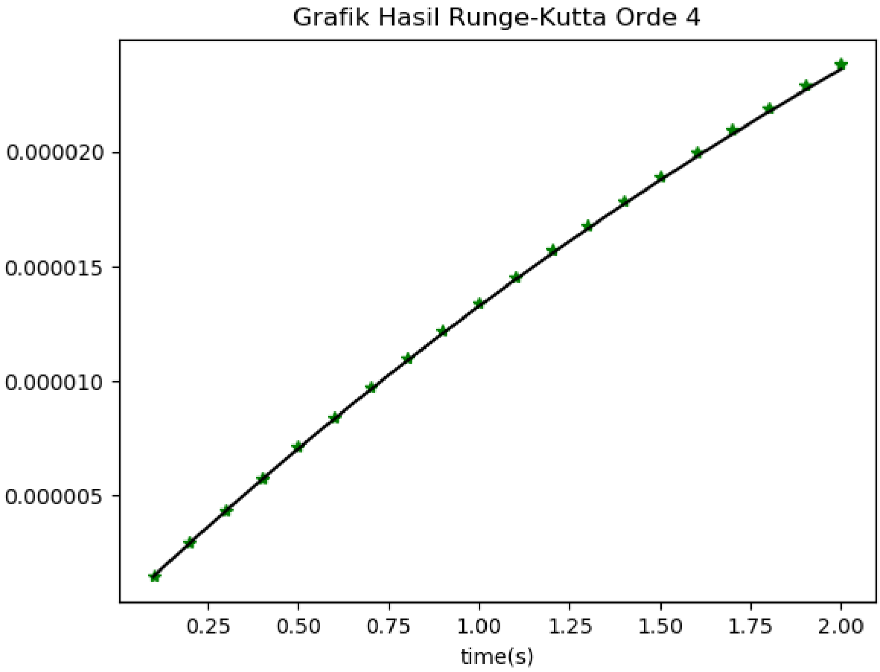
```
ODE Eulers Heuns task 1    ODE 4th Runge-Kutta task 1
"C:\Users\Thoriqul Aziz\AppData\Local\Programs\Python\Python37\python.exe" "D:/Tugas/Coding/Komputasi Biomedis/Tugas/ch8/ODE 4th Runge-Kutta task 1.py"

 ####Metode Runge-Kutta 4th Orde ####

Nilai t = [0.1, 0.2, 0.30000000000000004, 0.4, 0.5, 0.6, 0.7, 0.7999999999999999, 0.8999999999999999, 0.9999999999999999, 1.0999999999999999, 1.2, 1.3, 1.4000000
Nilai q = [1.4981265615234378e-06, 2.9588467364745084e-06, 4.383094516541456e-06, 5.771780572783252e-06, 7.125792837917164e-06, 8.445997074067314e-06, 9.73323742
Nilai eksak = [1.4814052783000432e-06, 2.9262345299571593e-06, 4.335390820286829e-06, 5.70975491784243e-06, 7.050185844924274e-06, 8.357521414496533e-06, 9.63257
Nilai eror = [1.6721283223394566e-08, 3.261220651734902e-08, 4.7703696254626836e-08, 6.202565494082204e-08, 7.560699299289003e-08, 8.847565957078076e-08, 1.00658
Nilai q pada t =2 adalah  2.381557114859506e-05
Sedangkan nilai eksaknya pada t=2 adalah 2.3608160417242005e-05
Nlai eror pada t=2 adalah  2.0741073135305602e-07

Process finished with exit code 0
```

Berikut adalah hasil ploting:



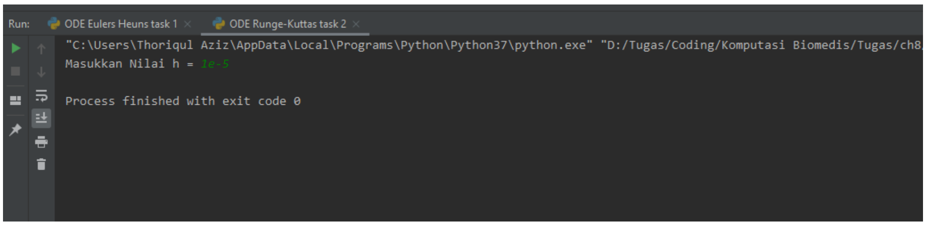
Dari metode tersebut dapat diamati berdasarkan perhitungan metode Runge-Kutta orde 4 untuk nilai q saat t=2 adalah 2.381557114859506 x 10^{-5} dengan nilai eroro yang dihasilkan adalah 2.0741073135305602 x 10^{-7} . sedangkan pada metode sebelumnya yaitu pada metode heun memilki tingkat eror pada orde 10^{-10} ketika nilai t=2 .dari hasil tersebur dapat damati bahwa metode Runge-Kutta dalam penebakan nilai persamaan deferensial tidak lebih baik dari metode Heuns jika hanya diamati dari hasil eror yang yang related lebih besar. Meskipun metode dari Runge-Kutta orde 4 memiliki nilai pengoreksi yang lebih banyak, pada kasus ini tidak bisa memberikan efek yang signifikan dalam penebakan hasil persamaan. Metode Runge-Kutta memiliki bebreapa kelebihan diantaranya terdapat nilai pengoreksi yang lebih banyak sehingga seharusnya didapatkan hasil perhutngan yang lebih presisi dibanding dengan metode yang lain. Hal ini didasarkan dari jumlah pengoreksi yang digunakan dalam perhitungan metode Runge-Kutta orde 4. Akan tetapi, karena jumlah aritmatika yang lebih kompleks dibandingkan dengan metode sebelumnya sehingga dibutuhkan *hardware* yang lebih tangguh dalam pengolaan data serta perhitungannya, dan pada kasus ini jumlah pengoreksi nyatanya tidak memebrikan perbedaan yang signifikan dibanding metode biasa.

2. Permasalahan kedua yaitu diberikan dua persamaan diferensial berpasangan. Persamaan ini digunakan sebagai konsekuensi dari adanya persamaan diferensial orde 2 yang relative sulit untuk kemudian dihitung hasil pendekatan numeriknya secara langsung. Metode yang digunakan adala Runge-Kutta orde 4. Sebelum masuk dalam proses perhitungan, pertama kali yang harus dilakukan yaitu mendefinisikan persaaman Runge-Kuttas agar ketika proses perulangan lebih mudah untuk digunakan. Kemudian juga membuat definisi fungsi untuk 2 persamaan deferensial berpasangan yang diketahui. Dengan algortima yang sama dengan metode sebelumnya, berikut adalah kode program dalam Bahasa Phyton 3.7:

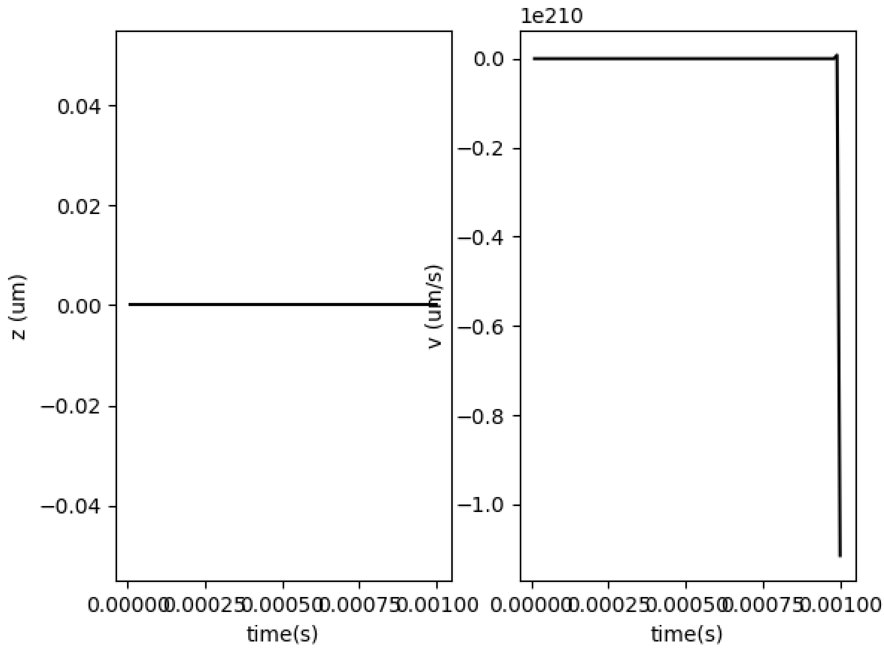
```
import numpy as np
import matplotlib.pyplot as mtp
def Runge4(x,y,func,h):
    w1=h*func(x,y)
    w2=h*func(x+0.5*h,y+0.5*h*w1)
    w3=h*func(x+0.5*h,y+0.5*h*w2)
    w4=h*func(x+h,y+h*w3)
    y=y+(w1+2*w2+2*w3+w4)/6
    return y
def sperical(x,y,a=1e-
4,rhos=1.1,rhof=1.0,g=981.0,u=3.5e-2):
    y1=y[0]
    y2=(rhos-rhof)*g/rhos-(9/2)*u*y[1]/(a**2)/rhos
    return np.array([y1,y2])
####
TimeBegin=0.0;TimeFinal=0.001
```

```
y01=0.0;y02=0.0
y=[y01,y02]
h=eval(input('Masukkan Nilai h = '))
ti=[];y1=[];y2=[]
##
while TimeBegin<TimeFinal:
    y=Runge4(TimeBegin,y,sperical,h)
    TimeBegin+=h
    ti.append(TimeBegin)
    y1.append(y[0])
    y2.append(y[1])
panjang=len(ti)
##Ploting
mtp.subplot(1,2,1)
mtp.plot(ti,y1,'k')
mtp.ylabel('z (um)')
mtp.xlabel('time(s)')
mtp.subplot(1,2,2)
mtp.plot(ti,y2,'k')
mtp.ylabel('v (um/s)')
mtp.xlabel('time(s)')
mtp.show()
```

Berikut adalah Hasil dari *command window* :



Berikut adalah hasil ploting:



Dari hasil tersebut dengan nilai h yang diberikan adalah 10μs sehingga diperoleh fase *steady state* sesuai dengan yang diinginkan oleh soal. Menemukan nilai lebar h tersebut dilakukan dengan cara mencoba pada rentang interval yaitu 0 hingga 0.001 sekon. Pengerjaan soal ini mengacu pada sumber dari King M.R and Mody N.A (2010) dalam bukunya yang mengatakan bahwa program ini termasuk dalam program *Coupled ODEs* yang menjelaskan solusi dari sebuah penyelesain persamaan deferensial secara numerik jika orde persamaan deferensial tersebut adalah 2, sehingga dilakukan penyederhanaan sedemikian sehingga hanya terdiri dari persamaan orde 1 yang saling berkorelasi sat sama lain. Hasil yang diberikan pada *output* program belum maksimla karena tida dapat diamati kenaikan parameter z pada gambar pertama dan kondisi *damping* dari pegerekan data hingga pada grafik menjadi tahap *steady state*. Hal ini diakibatkna oleh beberapa faktor yaitu fungsi yang dipasangkan dalam program kurang berkorelasi dengan bai sehingga tidak memunculkan hasil gambaran grafik yang sesuai dengan teori.

E. KESIMPULAN

Dari hasil percobaan tersebut dapat disimpulkan bahwa metode Runge-Kuttas orde 4 memiliki 4 nilai pengoreksi salam penentuan solusi persamaan diferensial. Memiliki kelebihan yaitu nilai hasil perhitungan relative baik karena terdapat 4 pengoreksi, tapi disisi lain membutuhkan *hardware* yang mumpuni untuk melakukan perhitungan yang relative lebih rumit dibanding metode sebelumnya. Pada kasus persamaan deferensial orde lebih dari satu dapat digunakan cara *Coupled ODEs* untuk menyederhanakan persamaan.

F. DAFTAR PUSTAKA

Capra, Steven C and Canale.1991. "**Numerical Methods for Engineers with Personal Computers Applications**". MacGraw-Hill Book Company.

King M.R and Mody N.A .2010. "**Numerical and Statical Methods for Bioengineering**".Cambridge University Press. New York.

Munir, Rinaldi.2003."**Metode Numerik**". Didownload dari https://kupdf.net/download/metode-numerik-rinaldi-munir-pdf_58eca95edc0d60f81ada9811_pdf