

課題提出に関して

- ・ **課題の提出は**
 - 期限: 2月9日(月)正午**
 - 提出先: 物理学部門等事務室**
- ・ 今週は、演習の時間としますので、各人課題のプログラムを作成してください。
 - プログラムの相談にのります
- ・ **2階の微分方程式の数値解法の例題を示します。**
 - テーマが決まっていない人は、この例題でも良い
- ・ **プログラムやレポートの中身は、設定したテーマの難易度に応じて評価**
 - プログラムが容易なものは、議論をしっかり行うこと
 - プログラムが難しいものは、プログラムがきちんとできていればよい
- ・ **プログラムの相談には随時のりますので、TAの西村くんか若狭までコンタクトしてください(メールが確実ですが、居室を訪ねてくれてももちろん構いません)**

数值計算法

第十三回
課題レポート作成(演習)

若狭 智嗣

粒子物理学講座

例(82ページ)

課題：RLC 直列回路のシミュレーション

図 8.1 にコイルが直列に入った RLC 直列回路を考える。コンデンサ両端の電圧 y は 2 階常微分方程式

$$LC \frac{d^2 y}{dx^2} + RC \frac{dy}{dx} + y = 0 \quad (8.19)$$

を満たす。この方程式を、適当な条件下で数値的に解け。 y の時間依存性をグラフに示せ。

- ・ **条件を、以下のようにとる。**

- $LC = 1$

- $RC = 1$

- $E = 1$

- $y(x=0) = 0$

- $dy/dx(x=0) = 0$

$$\longrightarrow \frac{dy^2}{dx^2} + \frac{dy}{dx} + y = 1$$

2階の微分方程式の解法

- ・ **2階の微分方程式も、Runge-Kutta法で解ける。**
- ・ 微分方程式は一般に

$$\frac{d^2 y}{dx^2} = g\left(x, y, \frac{dy}{dx}\right)$$

と書ける。

- ・ $z = dy/dx$ とおくと、

$$\begin{cases} \frac{dy}{dx} = z & \leftarrow y \text{ に対する1階の微分方程式} \\ \frac{dz}{dx} = g(x, y, z) & \leftarrow z = dy/dx \text{ に対する1階の微分方程式} \end{cases}$$

と連立の1階の微分方程式とみなせる。

- ・ **各々を、Runge-Kutta法で解けば、2階の微分方程式が解ける。**

嚴密解

$$\frac{dy^2}{dx^2} + \frac{dy}{dx} + y = 1$$



$$y = 1 - \exp\left(-\frac{1}{2}x\right) \left[\cos\left(\frac{\sqrt{3}}{2}x\right) + \frac{1}{\sqrt{3}} \sin\left(\frac{\sqrt{3}}{2}x\right) \right]$$

課題の例

- ・ RLC回路のプログラムは
 /home/teacher/z6wt01 in/No13
 以下に全てあります。

rc_circuit.f

runge_kutta.f

diff_eq_y.f

diff_eq_z.f

y(電圧)に関する微分方程式

$$\frac{dy}{dx} = z$$

diff_eq_y.f

```
real function diff_eq_y(x, y, z)
```

```
implicit none
```

```
c inputs/outputs:
```

```
real x, y, z
```

```
c begin:
```

```
diff_eq_y = z
```

! z=dy/dx

```
return
```

```
end
```

$z(=dy/dx)$ に関する微分方程式 $(LC)\frac{dz}{dx} + (RC)z + y = E$

```
real function diff_eq_z(x, y, z)
implicit none
```

```
c constant:
```

```
real MAX_VOLT           ! 最大電圧 (電池電圧E)
parameter (MAX_VOLT=1.0)
real LC                 ! LCの値
parameter (LC=1.0)
real RC                 ! RCの値
parameter (RC=1.0)
```

```
c inputs:
```

```
real x, y, z
```

$$\frac{dz}{dx} = (E - (RC)z - y)/(LC)$$

```
c begin:
```

```
diff_eq_z = (MAX_VOLT - y - RC*z)/LC ! dy^2/d^2x
return
end
```

Runge-Kutta法 (2つの1階微分方程式) runge_kutta.f

```
subroutine calc_next_step(x, y, z, step, y_next, z_next) ! Runge-Kutta法
implicit none
```

```
c inputs:
```

```
real x, y, z, step ! 時間x, 電圧y, z=dy/dx, 刻み幅step
```

```
c output:
```

```
real y_next, z_next ! 刻み幅後の電圧yと電圧の勾配z=dy/dx
```

```
c function:
```

```
real diff_eq_y, diff_eq_z
```

```
c local:
```

```
real k1, k2, k3, k4 ← yに関する1階の微分方程式を解く中間変数
```

```
real q1, q2, q3, q4 ← zに関する1階の微分方程式を解く中間変数
```

同時に解く

```
c begin:
```

```
k1 = step * diff_eq_y(x, y, z)
```

```
q1 = step * diff_eq_z(x, y, z)
```

```
k2 = step * diff_eq_y(x + 0.5*step, y + 0.5*k1, z + 0.5*q1)
```

```
q2 = step * diff_eq_z(x + 0.5*step, y + 0.5*k1, z + 0.5*q1)
```

```
k3 = step * diff_eq_y(x + 0.5*step, y + 0.5*k2, z + 0.5*q2)
```

```
q3 = step * diff_eq_z(x + 0.5*step, y + 0.5*k2, z + 0.5*q2)
```

```
k4 = step * diff_eq_y(x + step, y + k3, z + q3)
```

```
q4 = step * diff_eq_z(x + step, y + k3, z + q3)
```

```
y_next = y + (k1 + 2.0*k2 + 2.0*k3 + k4) / 6.0 ← step後のy
```

```
z_next = z + (q1 + 2.0*q2 + 2.0*q3 + q4) / 6.0 ← step後のz
```

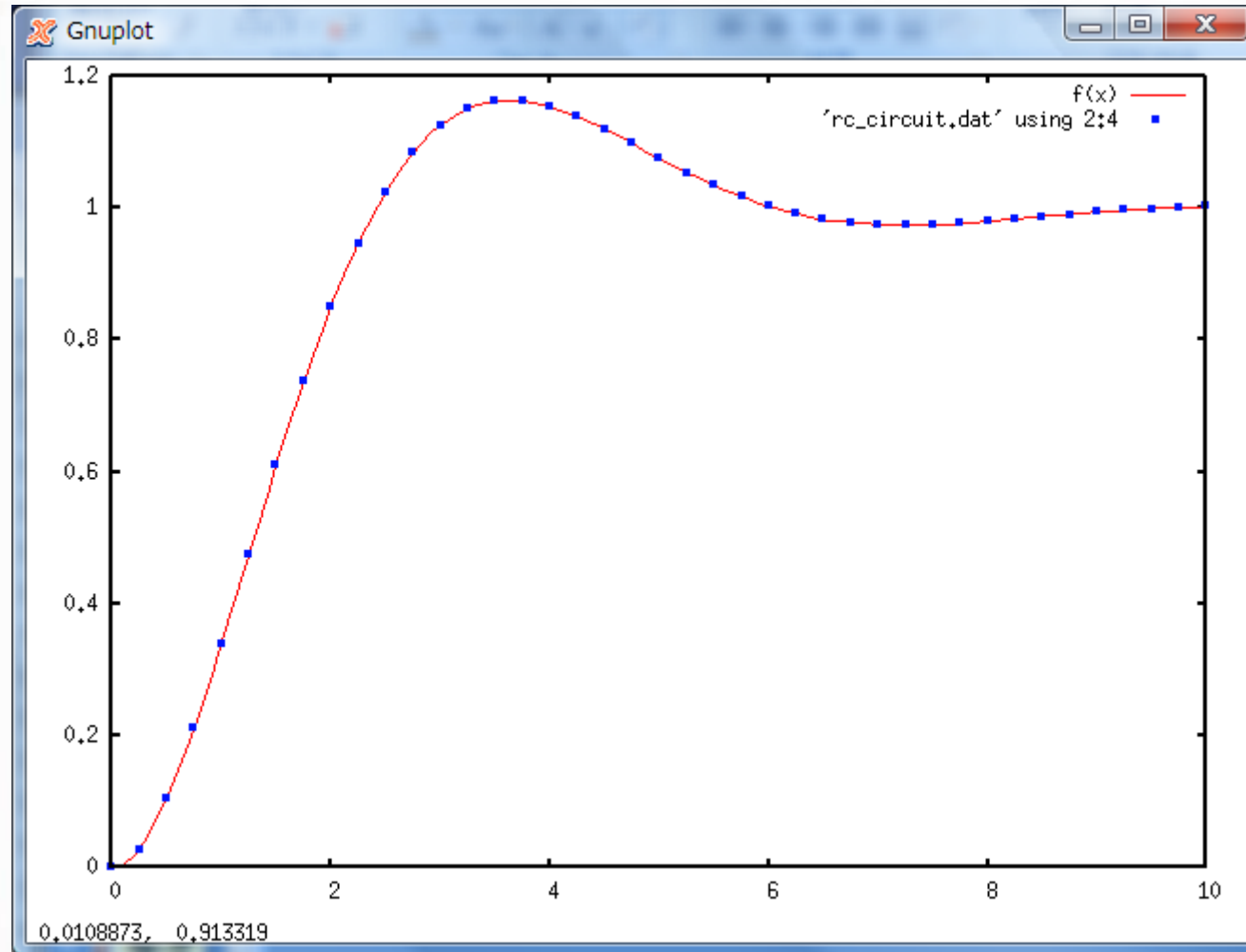
```
return
```

```
end
```

メインプログラム rc_circuit.f

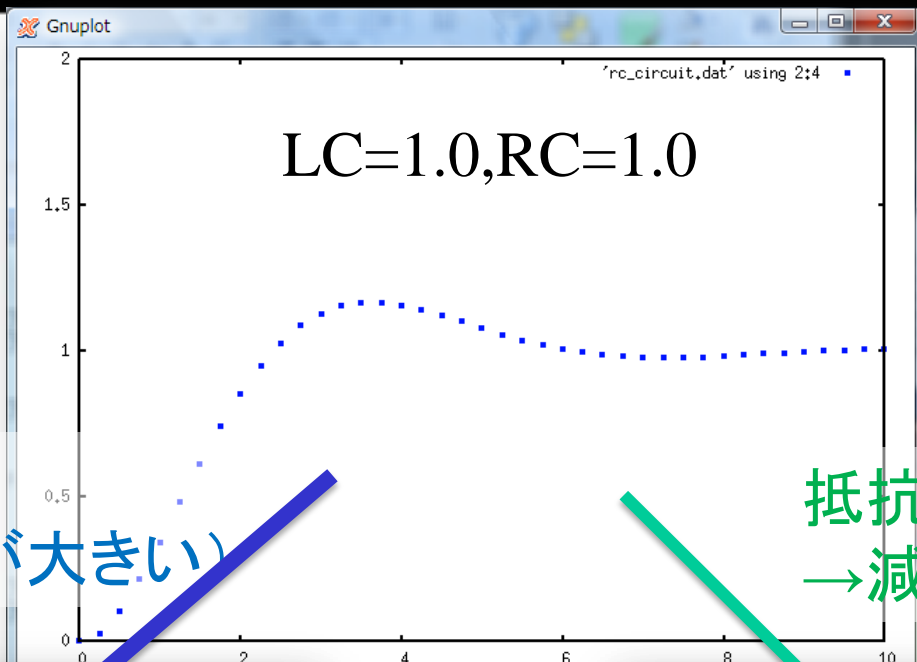
```
program rc_circuit
  implicit none
c constants:
  real TIME_INIT, TIME_LAST           ! 計算する時間範囲
  parameter (TIME_INIT=0. 0, TIME_LAST=10. 0)
  real Y_INIT, Z_INIT                 ! 電圧及びその勾配の初期値 (V)
  parameter (Y_INIT=0. 0, Z_INIT=0. 0)
  real STEP                             ! 時間の刻み
  parameter (STEP=0. 25)
c local:
  real x, y, z, y_next, z_next
c begin:
  x   = TIME_INIT                       ! 時間を初期化
  y   = Y_INIT                           ! 電圧を初期化
  z   = Z_INIT                           ! 電圧の勾配を初期化
  write(*, '(a, F5. 2, a, F8. 5)')      ! 電圧の初期値を出力
&    'Time: ', x, ' Voltage: ', y
  do while (x.lt.TIME_LAST)
    call calc_next_step(x, y, z, STEP, y_next, z_next)
    x   = x + STEP                       ! STEP後の時間
    y   = y_next
    z   = z_next
    write(*, '(a, F5. 2, a, F8. 5)')     ! STEP後の時間と
&    'Time: ', x, ' Voltage: ', y       ! 電圧を出力
  end do
stop
end
```

厳密解との比較の例



RC及びLC依存性

本課題をレポートにする人は、RC及びLC依存性を議論し厳密解と比較する事!!



抵抗(RC)を減らす
→減衰しにくい(振幅が大きい)

抵抗(RC)を増やす
→減衰しやすい(過減衰)

