2　C言語による数値計算の基礎（2）

以下では，関数 $f(t)$ の t による微分を $\dot{f}$ のように，点を上につけて表す．また $\ddot{f}$ は t による 2 階微分を表す．
($\dot{f} = \dfrac{df}{dt}, \qquad \ddot{f} = \dfrac{d^2 f}{dt}$).

2.1　運動方程式を１階の微分方程式に書き換えること

慣性系において質量 m の質点が位置 $\boldsymbol{r} = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}$ にあり、力 $\boldsymbol{F} = F_x\boldsymbol{i} + F_y\boldsymbol{j} + F_z\boldsymbol{k}$ を受けているとする。ニュートンの運動方程式により、質点の位置 $\boldsymbol{r}$ は以下の微分方程式をみたす。

$$m\ddot{\boldsymbol{r}} = \boldsymbol{F}.$$

この方程式を座標を用いて表せば、

$$\begin{cases} m\ddot{x} = F_x \\ m\ddot{y} = F_y \\ m\ddot{z} = F_z \end{cases}$$

となる。この微分方程式の未知関数は $x(t), y(t), z(t)$ である。また t についての 2 階微分を含んでいるので、2 階常微分方程式である。

ここで新たな変数 $\boldsymbol{v} = v_x\boldsymbol{i} + v_y\boldsymbol{j} + v_z\boldsymbol{k}$ を導入し、速度と関係づける:

$$\boldsymbol{v} = \dot{\boldsymbol{r}}$$

つまり

$$\begin{cases} v_x = \dot{x} \\ v_y = \dot{y} \\ v_z = \dot{z} \end{cases}.$$

加速度は速度の微分なので、$\ddot{\boldsymbol{r}} = \dot{\boldsymbol{v}}$ が成り立つ。

この $\boldsymbol{v}$ を用いると，運動方程式は次のように書き換えられる：

$$\begin{cases} m\dot{\boldsymbol{v}} = \boldsymbol{F} \\ \dot{\boldsymbol{r}} = \boldsymbol{v} \end{cases}$$

座標で表せば

$$\begin{cases} \dot{x} = v_x \\ \dot{y} = v_y \\ \dot{z} = v_z \end{cases} \qquad \begin{cases} m\dot{v}_x = F_x \\ m\dot{v}_y = F_y \\ m\dot{v}_z = F_z \end{cases}$$

となる。この微分方程式の未知関数は $x(t), y(t), z(t)$ および $v_x(t), v_y(t), v_z(t)$ となり、もとの微分方程式よりも増えているが，t についての 1 階微分のみを含んでいる．すなわち，未知関数が 6 個の 1 階常微分方程式になっている。

以上のように、運動方程式は速度を未知関数とすることで、未知関数は増えるがより簡単な 1 階常微分方程式に書き換えられる。よって運動方程式の数値解法は、1 階常微分方程式の数値解法で実現できる。

2.2　運動方程式の数値解法・常微分方程式の数値解法・オイラー法について

先に述べたように，運動方程式は 6 つの未知関数 $x(t), y(t), z(t), v_x(t), v_y(t), v_z(t)$ についての方程式であるが，簡単のため，まず未知関数が 1 つだけ（$\phi(t)$ とする) の場合を考える．

微分方程式

$$\dot{\phi}(t) = F(t, \phi(t)) \tag{1}$$

を満たす $\phi(t)$ について考える．

まず，方程式 (1) の左辺は，微分の定義によると

$$\dot{\phi}(t) = \lim_{h \to 0} \frac{\phi(h+t) - \phi(t)}{h}$$

である．すなわち，非常に小さな h に対しては，近似的に

$$\dot{\phi}(t)h \fallingdotseq \phi(h+t) - \phi(t)$$

になるので，(1) を用いて $\dot{\phi}$ を F で書き換えれば，

$$\phi(t+h) \fallingdotseq \phi(t) + F(t, \phi(t))h \tag{2}$$

という近似式が非常に小さな h に対して成り立つことが分かる．この近似式を用いれば，$\phi(t+h)$ の値が，$\phi(t)$ と h の値によって求めることができる．また，この近似式は h が小さいほど精度が上がることに注意しておく．

(2) を用いて，方程式 (1) を満たす $\phi(t)$ を以下のように近似的に求めることができる．

$t = t_0$ における ϕ の値が，$\phi(t_0) = x_0$ のように与えられているとき，$t_1 = t_0 + h$ での ϕ の近似値は，(2) で求められる：

$$\begin{aligned} \phi(t_1) = \phi(t_0 + h) &\fallingdotseq \phi(t_0) + F(t_0, \phi(t_0))h \\ &= x_0 + F(t_0, x_0)h \ (= x_1) \end{aligned}$$

この右辺をを x_1 とおくと，x_1 は $\phi(t_1)$ の近似値である．

次に $t_2 = t_0 + 2h$ での ϕ の近似値は，(2) を再び使うことで

$$\begin{aligned} \phi(t_2) = \phi(t_0 + 2h) &= \phi((t_0 + h) + h) \\ &\fallingdotseq \phi(t_1) + F(t_1, \phi(t_1))h \end{aligned}$$

となって，ここで近似値 $\phi(t_1) \fallingdotseq x_1$ を用いると

$$\fallingdotseq x_1 + F(t_1, x_1)h,$$

が得られる．

同様の繰り返しで，$t_{n+1} = t_0 + (n+1)h$ における ϕ の近似値 x_{n+1} が，$t_n = t_0 + nh$ における ϕ の近似値 x_n を用いて計算できる．具体的には

$$\begin{aligned} \phi(t_{n+1}) &= \phi(t_n + h) \\ &\fallingdotseq \phi(t_n) + F(t_n, \phi(t_n, x_n))h \\ &\fallingdotseq x_n + F(t_n, x_n)h, \end{aligned}$$

のようになる．

以上をまとめると，微分の定義の近似式を用いて，このように微分方程式の近似解を求める漸化式「オイラー法」が得られたことになる：

オイラー法による，微分方程式の近似解を求める漸化式

ϕ は方程式 (1) を満たすとする．$t_n = t_0 + nh$ における，ϕ の近似値 x_n が満たす漸化式（関係式）は

$$x_{n+1} = x_n + F(t_n, x_n)h \tag{3}$$

多変数の方程式に対するオイラー法の公式も，同様に考えればできる．結果だけを書き表せば以下のようになる：

オイラー法による，微分方程式の近似解を求める漸化式 (多変数)

N 個の未知関数 $\phi^\mu(t)$, $(\mu = 1, 2, \cdots, N)$ に関する微分方程式の系

$$\dot{\phi}^\mu = F^\mu(t, \phi^1(t), \phi^2(t), \cdots, \phi^N(t)) \qquad (\mu = 1, 2, \cdots, N) \tag{4}$$

の近似解を求めるオイラー法の漸化式は，$t_n = t_0 + nh$ での ϕ の近似値を x_n とおけば，

$$x^\mu_{n+1} = x^\mu_n + F^\mu(t_n, x^1_n, x^2_n, \cdots, x^N_n)h \tag{5}$$

で与えられる．

2.3 運動方程式への応用例

確認のために、実際での使われ方に近い形で書いてみる．

2.3.1 自由落下 (一次元の重力を受ける運動)

m を質量，g を重力加速度 ($\fallingdotseq 9.8\mathrm{m/s^2}$) とし，鉛直下向きに x 軸をとると，運動方程式は $m\ddot{x} = mg$ となる．1 階の微分方程式にするために，$v = \dot{x}$ とおいて書き換えると，

$$\dot{x} = v$$
$$\dot{v} = g$$

となる．(4) における ϕ^1, ϕ^2 が x, v に対応し，F^1, F^2 が v, g に対応する．

近似解を求めるオイラー法による漸化式は，(5) によって

$$x_{n+1} = x_n + v_n h$$
$$v_{n+1} = v_n + gh$$

のようになる．

2.3.2 フックの力 (バネによる力を受ける運動)

m を質量，k をバネ定数とし，バネの伸びる方向に x 軸をとると，運動方程式は $m\ddot{x} = -kx$ となる．1 階の微分方程式にするために，$v = \dot{x}$ とおいて書き換えると，

$$\dot{x} = v$$
$$\dot{v} = -kx$$

となる．(4) における ϕ^1, ϕ^2 が x, v に対応し，F^1, F^2 が v, $-kx$ に対応する．

近似解を求めるオイラー法による漸化式は，(5) によって

$$x_{n+1} = x_n + v_n h$$
$$v_{n+1} = v_n - kx_n h$$

のようになる．

2.3.3 放物運動 (二次元の重力を受ける運動)

m を質量，g を重力加速度とし，水平右向きに x 軸，鉛直上向きに y 軸をとると，運動方程式は $m\ddot{x} = 0$, $m\ddot{y} = -g$ となる．1 階の微分方程式にするために，$v = \dot{x}$, $u = \dot{y}$ とおいて書き換えると，

$$\dot{x} = v$$
$$\dot{v} = 0$$
$$\dot{y} = u$$
$$\dot{u} = -g$$

となる．(4) における ϕ^1, ϕ^2,ϕ^3, ϕ^4 が x, v, y, u に対応し，F^1, F^2, F^3, F^4 が v, 0, u, $-g$ に対応する．

近似解を求めるオイラー法による漸化式は，(5) によって

$$x_{n+1} = x_n + v_n h$$
$$v_{n+1} = v_n$$
$$y_{n+1} = y_n + u_n h$$
$$u_{n+1} = u_n - gh$$

のようになる．

2.4 C 言語による，オイラー法のプログラミング例

オイラー法の漸化式 (5) は，x_n から t が h だけ増加したときの近似値 x_{n+1} を求める形になっている．これは，現在の値 (x) から，h 秒後の次の値 (xa) を計算する公式と考えることができる．つまり t_0 での初期値 x_0 が与えられれば，順次 h 秒後の次の値をこの式で計算することができることになる．

以下のプログラム例は，バネに結び付けられた質点の運動方程式を，オイラー法を用いて近似的に解くものである．この例では，ばね定数を 1，初期時刻 $t_0 = 0$, 初期値 $x_0 = 1.0$, $v_0 = 0.0$ として解いている．

```
/* オイラー法による、バネに結び付けられた質点の運動方程式の数値解を求めるプログラム */
#include <stdio.h>

#define K 1     /* K をバネ定数を表すとしてそれを 1 と定める */

int main(){
  double t;      /* 時間変数 */
  double x,xa;   /* x は現在の x の値、xa は次のステップの x の値 */
  double v,va;   /* v は現在の v の値、va は次のステップの v の値 */
  double dt;     /* 時間の刻み（小さいほど近似の誤差は少なくなるが，処理速度は落ちる）*/

  dt=0.05;       /* 時間の刻みを 0.05 にする */

  x=1.0;         /* 初期条件の設定（初期位置） */
  v=0.0;         /* 初期条件の設定（初速度） */

  for(t=0.0; t<10; t=t+dt){ /* 0 秒から 10 秒まで dt 刻みで計算 */
    printf("%f %f\n",t,x);  /* 時刻 t とそのときの値 x を出力 */

    xa =x +dt * v;       /*現在ステップの値から次のステップの値を計算する*/
    va =v - K * x * dt;

    x = xa;      /* 次のステップの値を，現在のステップの値とする */
    v = va;
  }
  return 0;
}
```

このプログラムでは、dt(=時間の刻み h)= 0.05 とし、$t = 0.0$ から $t = 1.0$ までの範囲で数値解を求め，時刻 t と質点の位置 x を、画面に出力する。この出力結果をファイルにリダイレクトして、gnuplot を用いればグラフ化もできる。