自然科学シミュレーション ノート

2006年5月11日（木）実施

前回の続き

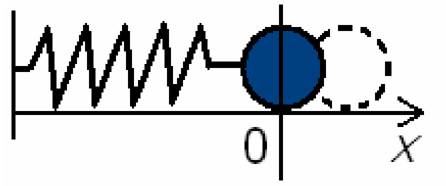
例題

$v(t+\Delta t) \cong v(t) - g\ \Delta t$ と $x(t+\Delta t) \cong x(t) + \dfrac{v(t+\Delta t)+v(t)}{2}\Delta t$ とを組み合わせて，位置座標の時間変化を求める。（$x_0 = 1\,\mathrm{m}$ とする）

ex3-1.c

```
#include <stdio.h>
int main(void)
{
  double dt,t,v,x;
  int i;
  FILE *output;
  dt=0.01;
  t=0.0;
  v=0.0;
  x=1.0;
  output=fopen("ichi.data","w");
  fprintf(output,"%f %f¥n", t, x);
  printf("%f %f¥n", t, x);
  for (i=0;i <50,x>=0.0;i++)
  {
     t=t+dt;
     x=x+v/2*dt;
     v=v-9.8*dt;
     x=x+v/2*dt;
     if (x<0.0) break;
     fprintf(output,"%f %f¥n", t, x);
     printf("%f %f¥n", t, x);
  }
  fclose(output);
  return 0;
}
```

復元力による運動



ばねの先についた重りを，摩擦のない水平面上で自由に運動させることを考える。ばねが伸び縮みしていない平衡位置 $x = 0$ から重りをずらして放すと，重りは平衡位置に戻ろうとする復元力 $f(t) = m\dfrac{dv(t)}{dt} = -kx(t)$ を受けながら運動する。

ここでは，オイラー法よりも精度の良い，4 次のルンゲクッタ法を用いて運動を数値的に解く。（ルンゲクッタ法はテイラー展開から導かれる。）

<u>4 次のルンゲクッタ法</u>

導関数 $\dfrac{dy}{dx} = F(x, y)$ に対して， $x = x_0$ における y の値 y_0 を元に， $x = x_0 + \Delta x$ における y の値 y_1 を次の式で求める。

$$k_0 = F(x_0, y_0) \cdot \Delta x$$

$$k_1 = F(x_0 + \frac{\Delta x}{2}, y_0 + \frac{k_0}{2}) \cdot \Delta x$$

$$k_2 = F(x_0 + \frac{\Delta x}{2}, y_0 + \frac{k_1}{2}) \cdot \Delta x$$

$$k_3 = F(x_0 + \Delta x, y_0 + k_2) \cdot \Delta x$$

$$y_1 = y_0 + \frac{1}{6}(k_0 + 2k_1 + 2k_2 + k_3)$$

<u>例題</u>

4 次のルンゲクッタ法を用いて，復元力によって運動している重りの位置及び速度の時間変化を求める C プログラムを作成し，翻訳編集して実行する。また，ファイルに結果を書き出し，GNUPLOT でグラフ化する。

この例題では， $\dfrac{dx(t)}{dt} = v(t)$ ， $\dfrac{dv(t)}{dt} = -\dfrac{k}{m}x(t)$ の連立微分方程式として，4 次のルンゲクッタ法を適用する。位置と速度の導関数がそれぞれ速度と位置の関数であることから，時刻が $\dfrac{\Delta t}{2}$， Δt だけずれた場合の速度と位置を用いればよいことが分かる。

<u>ex3-2.c</u>

```
/*復元力による運動
  dx/dt=v
  dv/dt=-K/M*x
```

```
  x0=0.1
  v0=0
  をルンゲクッタ法で解く
*/
#include <stdio.h>

#define M 0.1
#define K 0.2
#define TMAX 10.0

double fx(double t,double v);
double fv(double t,double x);

int main(void)
{
    double x,v,t,dt;
    double k0x,k1x,k2x,k3x,k0v,k1v,k2v,k3v;
    FILE *outputx,*outputv;

    t=0.0;
    dt=0.01;
    x=0.1;
    v=0.0;

    outputx=fopen("springx.data","w");
    outputv=fopen("springv.data","w");

    fprintf(outputx,"%f %f¥n",t,x);
    fprintf(outputv,"%f %f¥n",t,v);

    for(t=0.0;t<TMAX;t+=dt) {
        k0x=dt*fx(t,v);
        k0v=dt*fv(t,x);
        k1x=dt*fx(t+dt/2.0,v+k0v/2.0);
        k1v=dt*fv(t+dt/2.0,x+k0x/2.0);
        k2x=dt*fx(t+dt/2.0,v+k1v/2.0);
```

```
        k2v=dt*fv(t+dt/2.0,x+k1x/2.0);
        k3x=dt*fx(t+dt,v+k2v);
        k3v=dt*fv(t+dt,x+k2x);
        x=x+(k0x+2.0*k1x+2.0*k2x+k3x)/6.0;
        v=v+(k0v+2.0*k1v+2.0*k2v+k3v)/6.0;

        fprintf(outputx,"%f %f¥n",t+dt,x);
        fprintf(outputv,"%f %f¥n",t+dt,v);
    }

    fclose(outputx);
    fclose(outputv);

    return 0;
}

double fx(double t,double v)
{
    return v;
}

double fv(double t,double x)
{
    return -K/M*x;
}
```