振り子のシミュレーション
（常微分方程式）

講師：高安 亮紀

第6回

目的

- 常微分方程式の解を求めるODEソルバを用いて単振り子運動のシミュレーションを行う。

- 【演習】ODEソルバを用いて、２重振り子のシミュレーションプログラムを作成する

参考文献:「最新　使える！MATLAB　青山貴伸／蔵本一峰／森口肇・著」

・Runge-Kutta法

- 常微分方程式の基本的な解法として、
  ４次のRunge-Kutta法を使用。

４次のRunge-Kutta法

$$k_1(n) = f(t_n, y_n)$$

$$k_2(n) = f(t_n + \frac{dt}{2}, y_n + \frac{dt}{2} k_1(n))$$

$$k_3(n) = f(t_n + \frac{dt}{2}, y_n + \frac{dt}{2} k_2(n))$$

$$k_4(n) = f(t_n + dt, y_n + dt k_3(n))$$

$$y_{n+1} = y_n + \frac{dt}{6}(k_1(n) + 2k_2(n) + 2k_3(n) + k_4(n))$$

・ODEソルバ

- MATLABでは、常微分方程式の解を求めるものとして、いくつかのODEソルバが用意されている。

ソルバ	問題のタイプ	精度の度合い	使用法
ode45	ノンスティッフ	中	ほとんどの場合に使います。最初に実行してみるソルバ。
ode23	ノンスティッフ	低	粗い誤差基準を使ったり、わずかにスティッフな問題を解く場合
ode113	ノンスティッフ	低から高	精密な許容誤差を使ったり、計算の集中したODEファイルを解く場合
ode15s	スティッフ	低から中	ode45が遅い(スティッフなシステム)、または大きい質量行列がある場合
ode23s	スティッフ	低	スティッフなシステムを解くために粗い許容誤差を使ったり、または、大きい定数質量行列がある場合
ode23t	少しスティッフ	低	問題がある程度スティッフなだけで、数値減衰が無い解を必要とする場合
ode23tb	スティッフ	低	スティッフなシステムを解くために、大きな誤差の許容量を使うか、または、質量行列がある場合

※ODEソルバで使われるアルゴリズムは、精度と設計されたシステムのタイプ（stiffの度合い）により異なる。
堅い方程式(stiff equation)とは、解の緩やかな変化と急激な変化が混在する方程式で、刻み幅を小さく設定しなければならない場合がある。

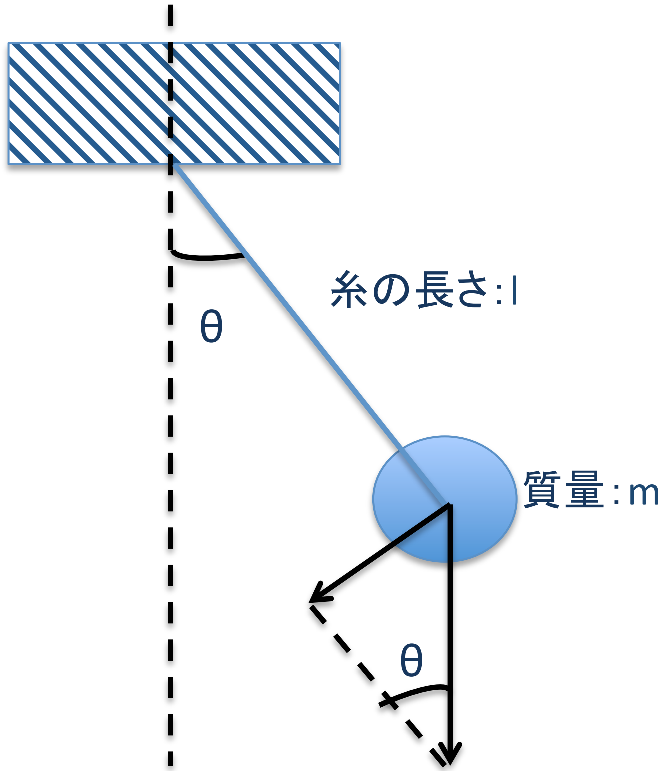
・単振り子の運動方程式

- Lagrange の運動方程式から単振り子の運動方程式が立てられる。
  以下のような2階の微分方程式となる。

$$ml\ddot{\theta} + kl\dot{\theta} + mg\sin\theta = 0$$

$$\dot{\theta} = \frac{d\theta}{dt}, \quad \ddot{\theta} = \frac{d^2\theta}{dt^2}$$

l:糸の長さ
m:質量
θ: 角度
t: 経過時間
g: 重力加速度(9.8)
k: 粘性減衰係数



・単振り子のシミュレーション

・高次の微分方程式を1階の連立微分方程式に変換して、ODEソルバで計算する。

$$\theta = y_1, \quad \dot{\theta} = y_2$$

とおくと、

$$\begin{cases} \dot{y}_1 = y_2 \\ \dot{y}_2 = \dfrac{-kly_2 - mg\sin y_1}{ml} \end{cases}$$

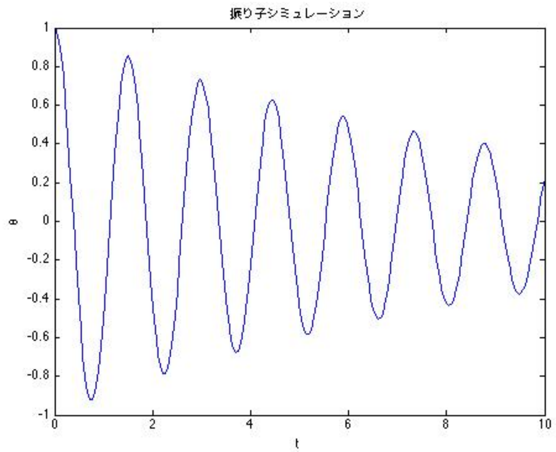
・単振り子のシミュレーション

・以下のような関数Mファイルを作成します。

```
function dy = furiko(t,y,flag,m,l,k)

%時間：t、y(1)=θ、y(2)=dθ/dt
%m:質量、l：糸の長さ、k:粘性減衰

g=9.8;
dy=[y(2);-1/(m*l)*(k*l*y(2)+m*g*sin(y(1)))];

end
```

・初期値を入力して、角度の変化をグラフに表すことができます。

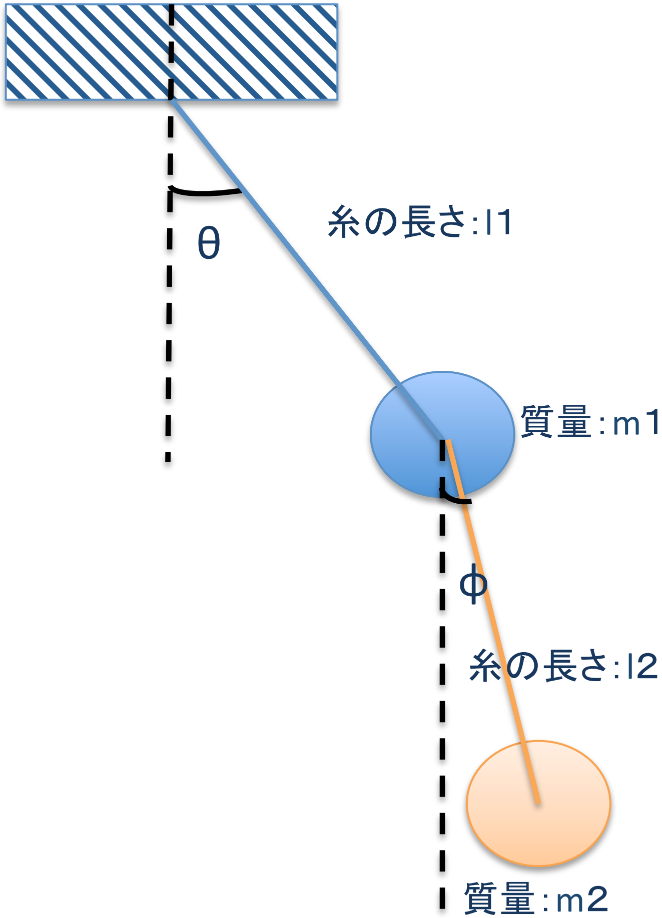
```
>> m=1; l=0.5;k=0.2;
>> [T Y]=ode45('furiko',[0 10],[1;0],[],m,l,k);
>> plot(T,Y(:,1),'-')
```



･2重振り子の運動方程式

- 2重振り子の運動方程式。
- このとき、質量m1のおもりに、糸l2で質量m2のおもりがぶら下がっている状態

$$
\begin{cases}
(m_1+m_2)l_1\ddot{\theta}+m_2l_2\ddot{\varphi}=-(m_1+m_2)g\theta \\
l_1\ddot{\theta}+l_2\ddot{\varphi}=-g\varphi \\
\ddot{\theta}=\dfrac{d^2\theta}{dt^2},\quad \ddot{\varphi}=\dfrac{d^2\varphi}{dt^2}
\end{cases}
$$



l1、l2:糸の長さ
m1、l2:質量
θ、φ:角度
t:経過時間
g:重力加速度(9.8)
k:粘性減衰係数

・２重振り子のシミュレーション

・高次の微分方程式を4元連立１階微分方程式に変換して、ODEソルバで計算する。

$$\theta = x_1, \dot{\theta} = x_2, \varphi = x_3, \dot{\varphi} = x_4$$

とおくと、

$$\begin{cases} \dot{x}_1 = x_2 \\ (m_1 + m_2) l_1 \dot{x}_2 + m_2 l_2 \dot{x}_4 = -(m_1 + m_2) g x_1 \\ \dot{x}_3 = x_4 \\ l_1 \dot{x}_2 + l_2 \dot{x}_4 = -g x_3 \end{cases}$$

・演習　２重振り子のプログラムの作成

ヒント１：dblfuriko.mを作る

1.　４元一次方程式を解く
2.　Θに対する方程式をF、φに対する方程式をGとすると、
　　F＊x‘＝G＊xと計算できるので、x’=(F\G)*xと計算できる

ヒント２：実行する

1,　単振り子と同様に初期値を入力する
2,　[T Y]=ode45(‘dblfuriko’,[時間],x0,[],l1,l2,m1,m2);と関数を実行
3,　θとφの角度の変化の状態をグラフに表す