

Ingák
Számítógépes szimulációk
fn1n4i11/1

Csabai István, Stéger József

ELTE

Komplex Rendszerek Fizikája Tanszék

Email: csabai@complex.elte.hu, steger@complex.elte.hu

A harmonikus oszcillátor a rezgő mozgások idealizált változata. Az előző alkalommal bemutatott *Euler-Cromer-algoritmus* a legegyszerűbb numerikus módszer, amellyel ez stabilan szimulálható.

Ezen a gyakorlaton a következő javításokat végezzük el az ingamozgás példáján:

- realiztikusabb inga megvalósítása a csillapítás és a nagy kitérések nemlinearitásának figyelembevételével,
- pontosabb numerikus algoritmus használata a *Runge-Kutta-módszer* és az *adaptív lépéshossz változtatás* bevezetése.
- Vizualizáció OpenGL segítségével.

ODE emlékeztető

- a tavaly tanultak,
- a [Numerical Recipes](#) 16. fejezete,
- rezgőmozgás és káosz.

Az *Euler*- és az *Euler-Cromer*-módszer $O(\tau^2)$ pontosságú, míg a 4-ed rendű *Runge-Kutta*-módszer hibája (τ^5) .

Az ideális inga mozgásegyenlete

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l}\theta,$$

ami csillapítás figyelembe vételével írható

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l}\theta - q\frac{d\theta}{dt},$$

illetve periodikus gerjesztő erő mellett

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l}\theta - q\frac{d\theta}{dt} + F_D \sin(\Omega_D t)$$

alakot ölt.

A paraméterek jelentése

- q : a légellenállási/súrlódási együttható,
- F_D : a külső periodikus gerjesztő erő amplitúdója,
- Ω_D : a gerjesztő erő szögfrekvenciája és
- $\Omega = \sqrt{g/l}$ az inga sajátfrekvenciája.

Kritikus csillapításról beszélünk, ha $\Omega = q/2$, illetve a rendszer túlcillapított, ha $\Omega < q/2$. A csillapítás hatására a következő tranziens viselkedés áll fenn:

$$\theta(t) = \theta_0 e^{-qt/2} \sin\left(\sqrt{\Omega^2 - q^2/4}t + \phi\right).$$

A tranziens lecsengése után a gerjesztő erő vezérli a rendszert. Kis kitérésekre (lineáris közelítés) a modell analitikusan megoldható:

$$\theta(t) = \frac{F_D \sin(\Omega_D t + \phi)}{\sqrt{(\Omega^2 - \Omega_D^2)^2 + (q\Omega_D)^2}}.$$

Fizikai inga (a nagy kitérések esete)

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l} \sin(\theta) - q \frac{d\theta}{dt} + F_D \sin(\Omega_D t)$$

A rendszert kaotikus mozgás jellemzi.

(Pl. $l = 9.8$, $q = 0.2$, $\Omega_D = 0.6666$, $F_D = 1.2$, $\theta(0) = 0.2$, $\omega(0) = 0$.)

Az inga példakódja: [pendulum.cpp](#).

A kód könyvtár elemei

- Fejlécek[♠]: [vector.hpp](#), [odeint.hpp](#).
- Forrásfájlok: [vector.cpp](#), [odeint.cpp](#).
- Könyvtárcsomag[♠]: [libcpl.a](#).

Töltsük le a [♠] jelölésű fájlokat a `./utils/` könyvtárba.

A kód könyvtár használata

- Fordítás:
`g++ -Wall -c -lutils -O2 pendulum.cpp -o pendulum.o`
- Fűzés:
`g++ -Wall pendulum.o -Lutils -lcpl -o pendulum.bin`
- vagy egy menetben:
`g++ -Wall pendulum.cpp -lutils -Lutils -lcpl -o pendulum.bin`

Haladó programozói ismeretekkel rendelkező és érdeklődő hallgatók figyelmébe. Lépünk abba a könyvtárba, ami a forrásfájlokat és a fejléceket tartalmazza. Ha ezek eltérnek, ügyesen válasszuk meg az `-I` kapcsolót.

A forrásfájlok fordítása

```
g++ -Wall -c -O2 -I. vector.cpp -o vector.o  
g++ -Wall -c -O2 -I. odeint.cpp -o odeint.o  
...
```

Linuxos környezetben a könyvtárak neve tipikusan a `lib*.a` formát követi.

Az object fájlok könyvtárba fűzése

```
ar -srl libcpl.a vector.o odeint.o ...
```

- Ingyenes, nagy teljesítményű, cross-platform (win,lin,mac; DirectX-csak win) grafikus könyvtár,
- grafikus kártyákra optimalizálva,
- számos programozási nyelv támogatott: C, C++, Java, Fortran, ...
- GUI (ablakok, menu, stb) nem része, kiterjesztés: GLUT.

Hasznos segédanyagok

- [Az OpenGL hivatalos honlapja](#),
- [Learning guide \("Red Book"\)](#),
- [Reference Manual](#),
- [GLUT: programozási interfész az OpenGLhez](#),
- [OpenGLről magyarul](#).

Az inga kapcsán megosztott [példakód](#) főbb jellemzői:

- Eseményvezérelt program: a program egy végtelen ciklus, a felhasználó akcióit várja.
- Az eseményeket kezelő függvények argumentumként kapják meg az eljárások nevét, amit az esemény bekövetkezésekor meghívunk ("callback funtions").

A program fordítása

```
g++ -Wall -lutils -Lutils pendulum-gl.cpp -lcpl -lglut `pkg-config --libs gl` -o pendulum-gl.bin
```

További játékos példák

- [3D kocka](#)
- [glcube.cpp](#)
- [Forgó négyzet](#)

- 1 A [példakód](#) segítségével és megfelelő átalakításával, tanulmányozzuk az ingamozgás különböző közelítéseit (matematikai, csillapított, gerjesztett, fizikai) és ábrázoljuk a tipikus viselkedések
 - kitérés-idő,
 - sebesség-idő,
 - energia-idő valamint
 - fázistér (kitérés-sebesség)

diagrammjaikat!

- 2 Hasonlítsuk össze, mennyire érzékeny a legegyszerűbb matematikai illetve a legösszetettebb fizikai, csillapított, gerjesztett inga a választott numerikus differenciálegyenlet megoldó módszerekre, úgy mint
 - *Euler-módszer*,
 - *Euler-Cromer-módszer*,
 - *Runge-Kutta-módszer* lépéshossz változtatással, illetve
 - egyszerű *Runge-Kutta-módszer*.
- 3 A [wikipedia cikk](#) alapján alakítsuk át a mintakódot úgy, hogy kettős ingát szimuláljon! Vizsgáljuk a trajektóriákat és fázisteret különböző paraméter-beállítások mellett, illetve próbáljuk meg rekonstruálni az utolsó ábrán látható átbillenési fázisteret!

A jegyzőkönyvet PDF formátumban küldjük be a szamszim@gmail.com címre!