Ćwiczenia V
Droga do chaosu

Jakub Tworzydło

Katedra Teorii Materii Skondensowanej
Instytut Fizyki Teoretycznej
telefon: (022)5532-314, pokój 344

31/10/2012 Hoża, Warszawa

Plan

1. Przejście do chaosu w “odwróconym wahadle”

2. Wskazówki

Plan

1. Przejście do chaosu w “odwróconym wahadle”

2. Wskazówki

Zadanie 1

Rozwiązać układ równań różniczkowych (ang. ODE) opisujących ewolucję cząstki w potencjale z podwójnym minimum, siłą wymuszającą i tłumieniem (tzw. równanie Duffinga)

$$\begin{cases} \dot{x} = v \\ \dot{v} = bx - ax^3 - cv + f\cos(\omega t) \end{cases}$$

przy ustalonych następujących parametrach:
$a = 1$, $b = 1$, $c = 0.2$, $\omega/2\pi = 0.2$.

Należy użyć metody całkowania `sc.integrate.odeint` z modułu SciPy, podając tej procedurze także (opcjonalną) macierz Jakobiego.

Przy pomocy `matplotlib` wykreślić ewolucję układu w przestrzeni fazowej z warunkiem początkowym $x(t=0) = 0, v(t=0) = 0.1$ i przy amplitudzie pobudzania $f = 0.2$.

Zadanie 2

W tym zadaniu wykreślimy portrety fazowe trajektorii $(x(t), v(t))$ otrzymywane w zagadnieniu Duffinga, ale tylko dla odpowiednio długich czasów ewolucji (np. $t > 200$).

Należy poeksperymentować z wartością f tak, aby znaleźć charakterystyczne rozwiązania:

- z pojedynczym okresem
- z podwójnym okresem
- a nawet z poczwórnym okresem
- zaproponować wartość f, dla której pojawia się już chaos.

Wyniki zilustrować na wykresach w przestrzeni fazowej (dodatkowo też np. jawnej zależności $x(t)$ lub przecięcia Poincare). Format wykresów (pojednyczy, kilka itp.) pozostawiam inwencji słuchaczy.

Zadanie 3

Wykonać wykres (pikselowy) przecięcia Poincare dla możliwie długiej ewolucji układu Duffinga w obszarze chaotycznym. Zaznaczać punkty (x_n, v_n) otrzymane w dyskretnych chwilach czasu, odpowiadających pełnemu okresowi siły pobudzającej $t_n = \frac{2\pi}{\omega} n$ (dla naturalnego n). Powstaje w ten sposób portret dziwnego atraktora.

Przykładowy kod

Warto, przystępując do ćwiczeń, najpierw skopiować i uruchomić pożyteczny przykład (w części dotyczącej wykreślania trajektorii ODE):
`http://www.scipy.org/Cookbook/LoktaVolterraTutorial`
Następnie można ten kod zmodyfikować tak, aby rozwiązywał równanie podane w Zadaniu 1.

Szerszą dokumentację procedury całkującej można znaleźć np. w `http://docs.scipy.org/doc/scipy/reference/` w podrozdziałach Tutorial i Reference

W celu przyspieszenia rachunków można też obniżyć wymaganą dokładność, podając np. `atol=1e-4,rtol=1e-4`

Składnia Pythona pozwala łatwo “obcinać” tablice

```
pl.plot(x[t>200],v[t>200],...)
```

Wykres pikselowy

Najprostszą metodą umieszczenia pikseli jest użycie modułu `pyplot`:

```
pl.scatter(xsolve, ysolve,
           s=3, facecolor='0.', lw=0, label='Punkty')
```

gdzie `xsolve, ysolve` są tablicami rozwiązań.

W pewnych sytuacjach bardziej efektywne jest bezpośrednie towrzenie obrazków pikselowych. Nie pracujemy na tablicach, umieszczamy (w pętli) punkt po punkcie na regularnej siatce, co ilustruje poniższy zbiór komend

```
from PIL import Image
Nimg = 512
image = Image.new("L", (Nimg, Nimg), 255)
image.putpixel((jx, jy), 0) # a pixel at jx, jy
image.save("ifs.png", "PNG")
```