

Programowanie 1R – Ćwiczenia 10

Równania różniczkowe zwyczajne (ODE)

Mikołaj Marcinkowski

19 maja 2026

1 Metody numerycznego rozwiązywania równań różniczkowych [1]

Często spotykamy się równaniami różniczkowymi typu:

$$y'(t) = f(t, y(t)), \quad (1)$$

które są zbyt złożone, by rozwiązać je analitycznie. Rozwiązywanie numeryczne polega na oszacowaniu wartości funkcji $y(t)$ w wybranych punktach t_n , zwykle rozstawionych co krok czasowy Δt . Zwykle oznaczamy $y_n \equiv y(t_n)$. Zakładamy też, że znamy wzór $f(t, y)$ dla dowolnych y i t . Istnieje wiele sposobów na podejście do tego problemu. Poniżej opiszę dwa.

1.1 Metoda Eulera

Metoda Eulera jest najprostszym podejściem:

$$y_{n+1} = y_n + f(t_n, y_n) \cdot \Delta t. \quad (2)$$

Szacujemy, że wartość funkcji y po kroku czasowym Δt , to jej obecna wartość + pochodna razy Δt . Ta metoda jest bardzo prosta, ale szybko akumuluje błędy, bo przybliża y tylko do pierwszego rzędu. Błąd po jednym kroku czasowym jest rzędu $\mathcal{O}(\Delta t^2)$.

1.2 Metoda Runge–Kutty 4. rzędu

Metoda RK4 jest jedną z najpopularniejszych metod numerycznego rozwiązywania równań różniczkowych. Polega na wykonaniu czterech półkroków,

które po dodaniu z odpowiednimi wagami dają oszacowanie y_{n+1} . W tej metodzie:

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \cdot \Delta t, \quad (3)$$

gdzie:

$$\begin{aligned} k_1 &= f(t_n, y_n), \\ k_2 &= f\left(t_n + \frac{\Delta t}{2}, y_n + \frac{k_1}{2}\Delta t\right), \\ k_3 &= f\left(t_n + \frac{\Delta t}{2}, y_n + \frac{k_2}{2}\Delta t\right), \\ k_4 &= f(t_n + \Delta t, y_n + k_3\Delta t). \end{aligned} \quad (4)$$

Polecam przeliczyć sobie w domu, że po podstawieniu tych wszystkich wzorów wyjdzie rozwinięcie Taylora czwartego rzędu:

$$y_{n+1} = y(t_n) + y'(t_n)\Delta t + \frac{1}{2}y''(t_n)\Delta t^2 + \frac{1}{6}y'''(t_n)\Delta t^3 + \frac{1}{24}y''''(t_n)\Delta t^4. \quad (5)$$

Ta metoda pozwala na rozwinięcie funkcji do czwartej pochodnej, znając tylko wzór na pierwszą. Jednocześnie, do obliczenia wartości w następnym kroku czasowym, wystarczy wywołać funkcję $f(y, t)$ tylko cztery razy. Błąd po jednym kroku jest rzędu $\mathcal{O}(\Delta t^5)$.

1.3 Równania wyższego stopnia

W fizyce często spotykamy się z równaniami różniczkowymi zawierającymi wyższe pochodne, na przykład drugą:

$$y''(t) = f(t, y(t), y'(t)). \quad (6)$$

Takie równania da się zawsze sprowadzić do układu równań pierwszego stopnia. Wprowadźmy nową niewiadomą:

$$v(t) \equiv y'(t). \quad (7)$$

Wtedy równanie (6) można rozłożyć na układ dwóch sprzężonych równań pierwszego stopnia:

$$\begin{aligned} y'(t) &= v(t), \\ v'(t) &= f(t, y(t), v(t)), \end{aligned} \quad (8)$$

co można jeszcze wygodniej zapisać na wektorach:

$$\mathbf{u}'(t) = \mathbf{F}(t, \mathbf{u}(t)), \quad (9)$$

gdzie $\mathbf{u}(t) = [y(t), v(t)]$. Takie równania można już rozwiązywać powyższymi metodami.

2 Zadania

Zadania możesz wykonywać jako oddzielne skrypty `.py` lub w notatniku jupytera `.ipynb`, chyba że treść zadania wskazuje inaczej.

Zadania wzorowane na Serii 11 zadań autorstwa Bartłomieja Zglinickiego (<https://www.fuw.edu.pl/~bzglinicki/teaching/p1r/zadania/seria-11/>), udostępnionej na licencji CC BY 4.0..

Zadanie 1. Wahadło matematyczne

Wahadło matematyczne (punktowa masa na końcu nieważkiego pręta) opisuje równanie:

$$\frac{d^2\theta}{dt^2} + \frac{g}{l} \sin(\theta) = 0, \quad (10)$$

gdzie θ to wychylenie wahadła od pionu, g to przyspieszenie grawitacyjne, a l to długość ramienia wahadła.

- a) Zasymuluj zachowanie wahadła na przedziale czasowym $[0, t_{end}]$ dla wybranego początkowego wychylenia θ_0 i prędkości kątowej ω_0 . Wykorzystaj metody Eulera i RK4 dla takiego samego kroku czasowego.

Dla obu metod zrób wykresy:

1. wychylenia w czasie $\theta(t)$,
2. trajektorii w przestrzeni fazowej $\omega(\theta)$.

- b) Biblioteka `scipy` posiada funkcję do numerycznego rozwiązywania równań różniczkowych: `scipy.integrate.solve_ivp`. Funkcja ta pozwala na rozwiązanie równania kilkoma metodami (standardowo Runge-Kutta 4. rzędu), inteligentnie zmieniając długość kroku czasowego, by oszacowany błąd nie przekroczył założonej granicy. Rozwiąż równanie wahadła matematycznego tą funkcją i dodaj jej wykresy do tych z punktu a).

Zadanie 2. Symplektyczna metoda Eulera

Pewnie rzuciło Ci się w oczy, że rozwiązania metodą Eulera z poprzedniego zadania miały tendencję do zwiększania amplitudy wahań z czasem. Jest tak, ponieważ zwykła metoda Eulera (i RK też, tylko wolniej) **nie zachowuje poprawnie energii układu**. Spróbuj zasymulować ten układ ponownie, tym razem korzystając z lekko zmodyfikowanej metody Eulera (zwanej po angielsku *semi-implicit Euler method* lub *symplectic Euler method*):

$$\begin{aligned} \omega_{n+1} &= \omega_n - \frac{g}{l} \sin(\theta_n) \Delta t, \\ \theta_{n+1} &= \theta_n + \omega_{n+1} \Delta t. \end{aligned} \quad (11)$$

Metoda ta różni się od zwykłej tylko tym, że przy obliczaniu nowego kąta bierzemy nową prędkość (ω_{n+1}), a nie poprzednią (ω_n). Różnica względem oryginalnej jest rzędu $\mathcal{O}(\Delta t^2)$, więc nadal poprawnie przybliża ona rozwiązanie do pierwszego rzędu.

Metody tego typu (tzw. *symplektyczne*) lepiej zachowują stałe ruchu. Można je stosować tylko do rozwiązywania niektórych problemów (tylko do układów Hamiltonowskich, więcej o tym na Mechanice Klasycznej) i zwiększają inne rodzaje błędów. Mimo to opłaca się z nich korzystać w niektórych sytuacjach, na przykład przy symulowaniu orbit ciał niebieskich.

Zadanie 3. Model Lotki–Volterry

Model Lotki–Volterry opisuje ekosystem składający się z dwóch populacji: drapieżników i ich ofiar. Niech x oznacza bieżącą liczebność ofiar, a y – drapieżników. Zmiany tych wartości opisują równania Lotki–Volterry:

$$\begin{aligned}\frac{dx}{dt} &= (a - by)x, \\ \frac{dy}{dt} &= (cx - d)y,\end{aligned}\tag{12}$$

gdzie a , b , c i d to stałe parametry modelu.

Zasymuluj numerycznie ewolucję powyższego układu stosując dowolną/-e z powyższych metod i narysuj wykresy populacji w czasie ($x(t)$ i $y(t)$) oraz zależności liczebności jednej populacji od drugiej ($x(y)$).

Zadanie 4. Oscylator Duffinga

Oscylator Duffinga to układ opisany równaniem:

$$\frac{d^2x}{dt^2} + \delta \frac{dx}{dt} + \beta x + \alpha x^3 = \gamma \cos(\omega t),\tag{13}$$

gdzie $x(t)$ to wychylenie oscylatora w czasie, a α , β , γ , δ i ω to stałe parametry układu. Układ ten jest chaotyczny – niewielkie różnice w warunkach początkowych powodują duże zmiany w jego zachowaniu.

Zasymuluj ewolucję położenia $x(t)$ i prędkości $v(t) = \dot{x}(t)$ układu dla parametrów:

$$\alpha = \omega = 1, \quad \beta = -1, \quad \delta = 0,2, \quad \gamma = 0,3,$$

dla trzech zestawów warunków początkowych:

$$\begin{cases} x_{01} = 0,99 \\ v_{01} = 0, \end{cases} \quad \begin{cases} x_{02} = 1 \\ v_{02} = 0, \end{cases} \quad \begin{cases} x_{03} = 1,01 \\ v_{03} = 0. \end{cases}$$

Wykonaj wykresy położenia od czasu $x(t)$ oraz fazowy $v(x)$ tych przypadków.

Bibliografia

- [1] William H. Press, B. P. Flannery, S. A. Teukolsky i W. T. Vetterling, *Numerical recipes in FORTRAN 77: Volume 1 of FORTRAN numerical recipes volume 1*, 2 wyd. Cambridge, England: Cambridge University Press, wrzesień 1992.