

МОДЕЛЮВАННЯ ХАОТИЧНИХ СИСТЕМ: АТРАКТОР ЛОРЕНЦА ЗАСОБАМИ PYTHON

Вінницький національний технічний університет, м. Вінниця

Анотація

У роботі досліджено можливості мови програмування Python для чисельного моделювання хаотичних динамічних систем на прикладі системи Лоренца. Реалізовано алгоритм чисельного інтегрування рівнянь Лоренца методом Ейлера та проаналізовано феномен чутливості до початкових умов. Побудовано тривимірний фазовий портрет атратора та фазові траєкторії, що демонструють хаотичну природу системи.

Ключові слова: Python, хаотичні системи, атратор Лоренца, чисельні методи, динамічні системи, візуалізація.

Abstract

The paper investigates the capabilities of the Python programming language for numerical simulation of chaotic dynamical systems using the Lorenz system as an example. An algorithm for numerical integration of the Lorenz equations using Euler's method is implemented, and the phenomenon of sensitivity to initial conditions is analyzed. A three-dimensional phase portrait of the attractor and phase trajectories demonstrating the chaotic nature of the system are constructed.

Keywords: Python, chaotic systems, Lorenz attractor, numerical methods, dynamical systems, data visualization.

Вступ

Теорія хаосу є одним із найбільш захоплюючих розділів сучасної фізики та математики, що вивчає поведінку детермінованих систем, які демонструють непередбачуваність через надзвичайну чутливість до початкових умов [1]. Класичним прикладом такої системи є система диференціальних рівнянь, запропонована метеорологом Едвардом Лоренцом у 1963 році як спрощена модель атмосферної конвекції [2]. Незважаючи на просту математичну форму, ця система породжує надзвичайно складну поведінку, яка отримала назву «ефект метелика». Вивчення хаотичних систем є важливим для розробників програмного забезпечення у сферах симуляції фізичних процесів, комп'ютерної графіки та криптографії. Мова програмування Python з бібліотеками NumPy [5] та Matplotlib [4] забезпечує зручний інструментарій для дослідження таких систем.

Математична модель системи Лоренца

Система Лоренца описується трьома пов'язаними диференціальними рівняннями першого порядку [2]:

$$dx/dt = \sigma(y - x), \quad dy/dt = x(\rho - z) - y, \quad dz/dt = xy - \beta z$$

де x, y, z — змінні стану системи; σ, ρ, β — параметри системи (числа Прандтля, Релея та геометричний коефіцієнт відповідно). При класичних значеннях параметрів $\sigma = 10, \rho = 28, \beta = 8/3$ система демонструє хаотичну поведінку [3]. Розв'язок цієї системи не збігається до стаціонарного стану та не є суворо періодичним, але залишається обмеженим у фазовому просторі, утворюючи так званий «дивний атратор».

Програмна реалізація та результати

Чисельне інтегрування системи реалізовано методом Ейлера з кроком $dt = 0,01$. На кожній ітерації обчислюються похідні та оновлюються координати точки у фазовому просторі. Нижче наведено ключовий фрагмент реалізації:

```
def lorenz(x, y, z, s=10, r=28, b=2.667):  
    dx = s * (y - x)  
    dy = r * x - y - x * z  
    dz = x * y - b * z  
    return dx, dy, dz
```

Результати тривимірної візуалізації атратора Лоренца наведено на рис. 1. На графіку чітко видна характерна форма «метелика» — дві спіральні лопасті, між якими траєкторія безперервно перескакує у непередбачуваному порядку.

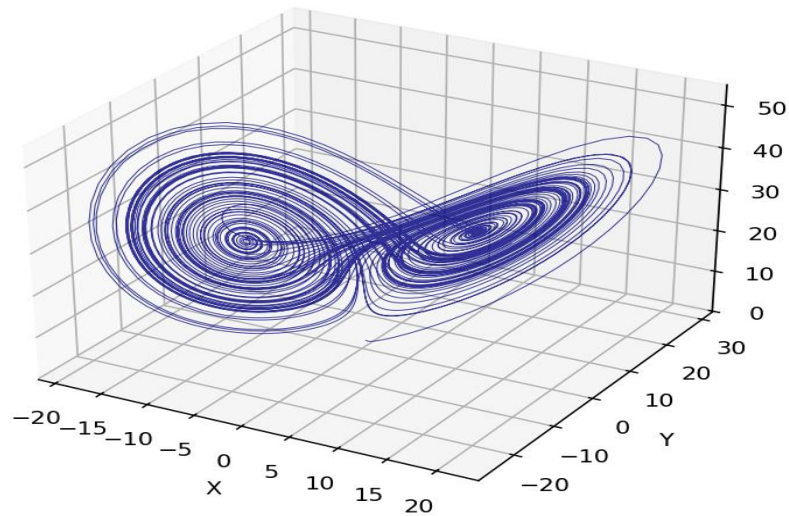


Рис. 1. Тривимірна фазова траєкторія атратора Лоренца ($\sigma=10$, $\rho=28$, $\beta=8/3$), побудована у середовищі Python

Для дослідження чутливості до початкових умов було промодельовано дві траєкторії з початковими умовами, що відрізняються лише на $\Delta x_0 = 0,0001$. Результати та фазовий портрет у площині X–Z наведено на рис. 2.

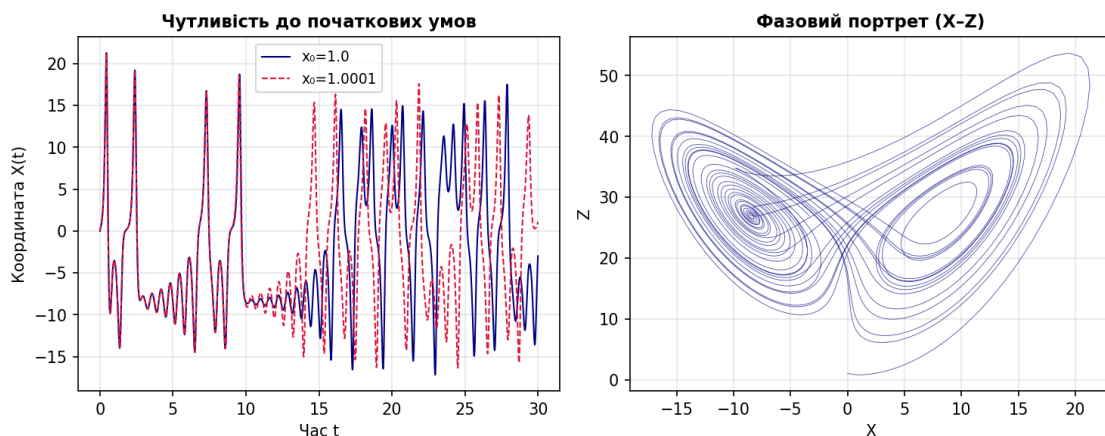


Рис. 2. Чутливість атратора Лоренца до початкових умов (ліворуч) та фазовий портрет у площині X–Z (праворуч)

З рис. 2 видно, що дві траєкторії з практично ідентичними початковими умовами спочатку збігаються, проте вже через час $t \approx 20$ одиниць повністю розходяться. Це явище і є «ефектом метелика» — нескінченно мала різниця у початковому стані призводить до кардинально різних результатів, що робить довгострокове прогнозування принципово неможливим. Фазовий портрет у площині X–Z демонструє обидві «крила» атратора та характерне переплетення траєкторій, що ніколи не перетинаються і не повторюються точно.

Висновки

Використання Python дозволяє ефективно моделювати хаотичні фізичні системи та наочно демонструвати їх ключові властивості — детермінованість та непередбачуваність. Отримані результати підтверджують, що навіть найпростіші нелінійні системи здатні породжувати виключно складну поведінку. Розуміння теорії хаосу є важливим для майбутніх інженерів-програмістів при розробці фізичних симуляторів, систем шифрування та генераторів псевдовипадкових чисел.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Строгац С. Нелінійна динаміка та хаос / С. Строгац. – К.: Наука, 2019. – 480 с.
2. Lorenz E. N. Deterministic Nonperiodic Flow / E. N. Lorenz // Journal of the Atmospheric Sciences. – 1963. – Vol. 20, № 2. – P. 130–141.
3. Кузнецов С. П. Динамічний хаос / С. П. Кузнецов. – М.: Фізматліт, 2006. – 356 с.
4. Документація бібліотеки Matplotlib [Електронний ресурс]. – Режим доступу: <https://matplotlib.org/>
5. Документація бібліотеки NumPy [Електронний ресурс]. – Режим доступу: <https://numpy.org/>
6. Welcome to Python.org [Electronic resource] // Python Software Foundation. – Access mode: <https://www.python.org/>

Гончаренко Дарія Олександрівна - студентка групи 2ПІ-256, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: goncarenkodara506@gmail.com.

Науковий керівник: **Мартинюк Володимир Валерійович** - канд. техн. наук, доцент кафедри загальної фізики, Вінницький національний технічний університет, м. Вінниця.

Goncharenko Daria O. - student of the Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, Ukraine.

Supervisor: Martyniuk Volodymyr V. - Candidate of Technical Sciences, Associate Professor of the Department of General Physics, Vinnytsia National Technical University, Vinnytsia.