

УДК 517.938:004.9

**КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ЭЛЕМЕНТОВ ДИНАМИЧЕСКИХ
СИСТЕМ В PYTHON**

Т.В. Алферова, Т.А. Трохова

Гомельский государственный технический университет имени П.О. Сухого

Аннотация.** ЦЕЛЬ.* Исследование возможности применения системы программирования Python при анализе процессов в линейных и нелинейных динамических системах. При создании и исследовании компьютерных моделей часто приходится решать задачу выбора инструментария моделирования. ***МЕТОДЫ. При решении поставленных задач использовался метод катастроф, учитывающий скачкообразные изменения, возникающие в виде внезапного ответа системы на плавное изменение внешних условий. ***РЕЗУЛЬТАТЫ*** В работе представлены результаты апробации системы программирования Python в качестве основы для разработки компьютерных моделей линейных и нелинейных динамических систем. Дана краткая характеристика библиотек и модулей Python, применяемых при разработке систем, описанных дифференциальными уравнениями и передаточными функциями. Дан теоретический обзор возникновения хаотических процессов в нелинейных динамических системах, продемонстрирован механизм перехода к хаосу на примере построения модели, основанной на уравнении Ферхюльста. Реализована математическая модель уравнений Лоренца, которая описывает многообразие хаотических изменений нелинейной динамической системы, построены фазовые портреты выходных параметров модели в виде трехмерных графиков. Сделан вывод о том, что система программирования Python может быть использована в качестве инструментария при разработке компьютерных моделей линейных и нелинейных динамических систем. Апробация данного подхода проводилась при разработке моделей динамических систем в курсе математического моделирования сложных систем.

Ключевые слова: динамические модели, компьютерное моделирование, нелинейные динамические системы, инструментарий компьютерного моделирования, система программирования Python.

Введение. При компьютерном моделировании технических устройств и процессов особое внимание уделяется исследованию и анализу динамических систем, одной из актуальных задач при этом является задача выбора инструментария моделирования. Развитые системы компьютерной математики, такие как Matlab, Mathcad, Maple обладают обширным набором функций для построения и исследования как линейных, так и нелинейных динамических систем, однако, это лицензионные, дорогостоящие системы, поэтому необходимо искать компромиссные решения при выборе инструментария. Авторами проведена апробация применения системы программирования Python в качестве инструментария моделирования динамических систем. Python – бесплатная, удобная и простая в освоении система программирования, в которой можно не только создать и

исследовать модели динамических систем, но и разработать удобный для анализа пользовательский интерфейс, позволяющий экспериментировать с моделью. Апробация инструментария проводилась при разработке моделей динамических систем в курсе математического моделирования сложных систем. В настоящее время одним из перспективных направлений расширения данного курса является моделирование нелинейных динамических систем. Анализ таких моделей неразрывно связан с исследованием хаотических процессов в динамических системах и получением в результате набора точек бифуркации.

Цель работы – исследовать возможность применения системы программирования Python при анализе процессов в линейных и нелинейных динамических системах.

Материалы и методы. Язык Python применяется для программирования задач из широкого диапазона предметных областей, подходит для различных платформ (iOS, Android), применяется в серверной разработке [1]. Для исследования динамических моделей Python содержит ряд библиотек и модулей, основными из которых являются следующие:

`numpy` – позволяет работать с массивами, случайными величинами, полиномами, выполнять быстрое преобразование Фурье и т.д.;

`matplotlib` - содержит как функции, так и модули, позволяющие строить графики, редактировать их, выполнять сервисные функции для работы с графическими данными;

`scipy` - библиотека предназначена для выполнения научных и инженерных расчетов и содержит модули и функции для оптимизации; интегрирования; обработки сигналов; решения обыкновенных дифференциальных уравнений и т.д.;

`sympy` – библиотека, предназначенная для выполнения вычислений в аналитическом виде, когда результат исследований сначала получается в виде формулы, а затем, при необходимости, приводится к числовому виду.

`control.matlab` – модуль предназначен для исследования динамических систем методами теории автоматического управления. В модуле реализованы: полный набор средств для анализа одномерных и многомерных стационарных систем; временные характеристики: передаточная и переходная функции, реакция на произвольные входные воздействия; частотные характеристики и т.д.

При исследовании динамических моделей, заданных обыкновенными дифференциальными уравнениями (ОДУ) или системами ОДУ качественные результаты дает применение таких функций модуля `scipy.integrate` Python, как `ode()` и `odeint()`. Например, выходные параметры модели механической динамической системы, описанной ОДУ второго порядка, можно получить с помощью следующей программы (рисунок 1).

Библиотека символьных вычислений `sympy` позволяет не только получать решение ОДУ в аналитическом виде при исследовании параметров динамических систем, но и выполнять структурный синтез моделей. Например, в работе [2] рассмотрен синтез системы управления методами символьной регрессии с применением функций библиотеки `sympy`.

Если динамическая модель линейной стационарной системы представлена в виде передаточной функции, то для ее исследования применяются следующие функции модуля `control.matlab`.

`W = tf(nom, den)` – позволяет создать объект в виде передаточной функции, `nom` и `den` – массивы коэффициентов полиномов числителя и знаменателя передаточной функции, расположенных в порядке убывания степеней.

`y, t = step(lti-объект)` – получает массивы значений переходной характеристики динамической системы.

`A,PH,W= bode(lti-объект)` - предназначена для получения диаграммы Боде – графиков логарифмических амплитудно-частотной и фазо-частотной характеристик (АЧХ и ФЧХ).

`pole(lti-объект)` – позволяет получить массив корней характеристического уравнения, которые можно использовать для вычисления полюсов модели при анализе ее на устойчивость.

```
# Описание системы ОДУ
def dempf(y,t):
    return [y[1], -2*n*y[1]-p**2*y[0]]
t = np.linspace(0, 1.5, 500)
y0=[0.05,0] #Задание начальных условий
```

```
#Решение дифференциального уравнения
Y= odeint(dempf, y0, t)
```

```
#Графики функций перемещения и скорости
plt.figure(1) ; plt.plot(t,Y[:,0]); plt.grid()
plt.figure (2) ; plt.plot(t,Y[:,1]); plt.grid()
plt.show()
```

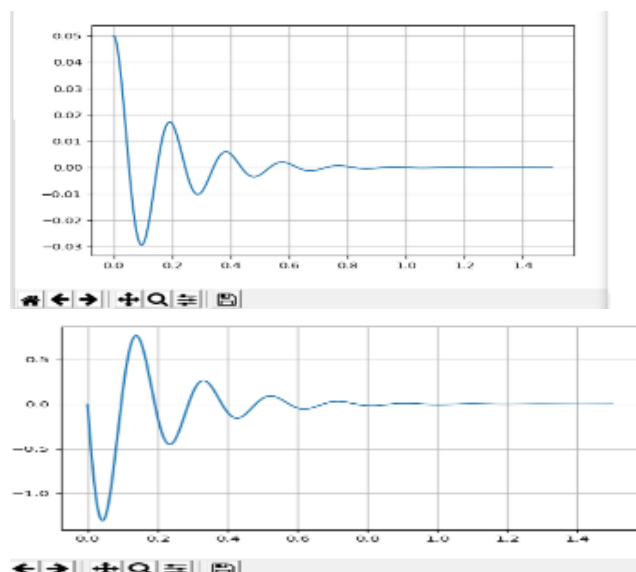


Рисунок 1 – Фрагмент программы и результат моделирования с использованием функции `odeint`

На рисунке 2 приведен пример построение диаграммы Боде в Python.

```
from control.matlab import *
import matplotlib.pyplot as plt
num= [0., 1.]
den= [1., 2., 10.]
w= tf(num, den)
mag, phase, omega = bode(w)
plt.plot()
plt.show()
```

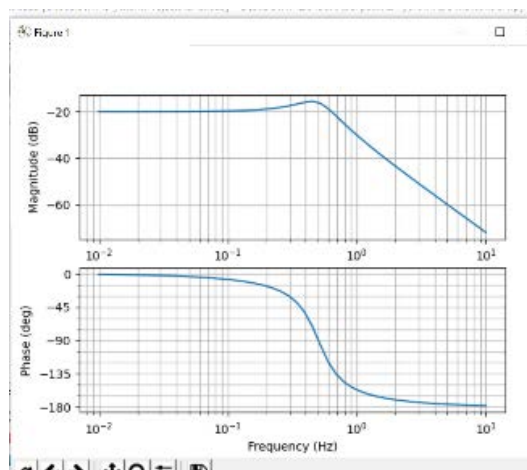


Рисунок 2 – Фрагмент программы и результат построения диаграммы Боде с использованием библиотеки `control`

Результаты и обсуждение. Катастрофами называются скачкообразные изменения, возникающие в виде внезапного ответа системы на плавное изменение внешних условий [3]. При катастрофе под действием управляющих параметров изменяется стационарное состояние системы, т.е. она переходит из одного стационарного состояния в другое. Как проходит переходный процесс - для теории катастроф, в сущности, неважно. Скачкообразное изменение качественного поведения системы при плавном изменении параметра называется бифуркацией [4]. При достижении системой точки бифуркации достаточно сложно предсказать дальнейший путь развития системы, даже слабые воздействия извне могут

привести к резкому изменению не только выходных параметров, но и к качественному изменению внутренних элементов системы.

Обзор бифуркаций динамических систем. Построение фазового портрета динамической системы является одним из удобных инструментов исследования колебательных процессов. Качественное преобразование фазового портрета (бифуркация) происходит при определенных условиях, в зависимости от которых определяется тип бифуркации – седло-узловая; бифуркация Андронова-Хопфа; бифуркация предельного цикла и другие [5]. На тесную связь между теорией катастроф и теорией бифуркаций указывал в своей работе [6] Р. Гилмор. При изучении теории катастроф проще всего иметь компьютерный аппарат исследования простых бифуркаций типа «складка», «камертон», «сборка» и после выявления того, какой из видов бифуркации присущ данной нелинейной системе, можно получать по заданным алгоритмам основные параметры устойчивости системы: область устойчивости, точки устойчивого состояния моделей [4, 7, 8, 9].

Компьютерное моделирование элементов нелинейных динамических систем. При исследовании динамических систем одним из интересных аспектов является анализ изменения поведения системы и возникновения хаоса. Хаотическое поведение означает:

- неустойчивость фазовых траекторий;
- рост малого начального возмущения;
- непредсказуемость поведения системы на больших временах.

Пример возникновения хаоса в нелинейных динамических системах может быть приведен на основе демонстрации влияния коэффициента k на результаты построения модели, основанной на уравнении Ферхюльста [10]. Этот механизм перехода к хаосу был открыт Митчеллом Фейгенбаумом, американским физиком и математиком, одним из пионеров теории хаоса.

Пусть модель динамической системы описывается уравнением вида:

$$x_n = k \cdot x_{n-1} \cdot (1 - x_{n-1}) \quad (1)$$

- Были выявлены следующие особенности поведения модели при различных значениях k :
 - если $1 < k < 2$, то динамическая система быстро приходит в устойчивое стационарное состояние, ее поведение предсказуемо;
 - если $2 < k < 3$, то динамическая система перед тем, как перейти в устойчивое стационарное состояние, совершает ряд колебаний;
 - если $3 < k < 3.45$, то динамическая система совершает ряд колебаний, амплитуда которых ограничена двумя константными значениями;
 - если $3.45 < k < 3.54$, амплитуда колебаний динамической системы ограничена четырьмя значениями;
 - при значениях k , больших 3.54 процесс колебаний носит непредсказуемый характер, наступает хаос.

Рассмотренный пример позволяет сделать заключение о том, что поведение несложных динамических моделей может непредсказуемо изменяться в зависимости от изменения внутренних параметров модели. Графическая иллюстрация возникновения хаотического поведения динамической системы при различных значениях внутреннего параметра приведена на рисунке 3.

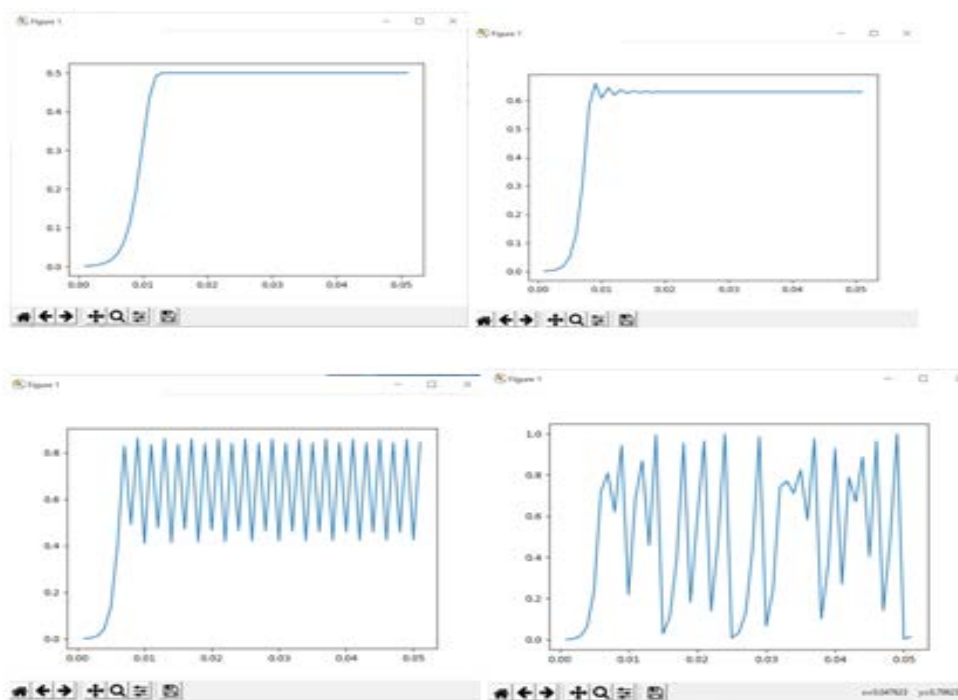


Рисунок 3 – Влияние параметра k на поведение динамической модели (k=2, k=2.7, k=3.45, k=4)

Другим примером возникновения хаотического поведения является поведение динамической системы из трех нелинейных уравнений - математическая модель уравнений Лоренца, которая описывает многообразие хаотических изменений:

$$\begin{cases} y1' = s(y2 - y1) \\ y2' = -y2 + (r - y3)y1 \\ y3' = -by3 + y1y2 \end{cases} \quad (2)$$

Система дифференциальных уравнений решается в Python в численном виде с использованием функции odeint модуля integrate библиотеки scipy, после проведения исследований по модели результирующие фазовые портреты выводятся в виде трехмерных графиков. К хаотическим изменениям приводит не только изменение параметра r, как указано в работе [11], но и незначительное изменение начальных условий выходных параметров модели. На рисунке 4 приводятся фазовые портреты динамической системы при изменении начальных условий для y2, y3. Если исходные начальные условия имели вид массива $y = [1, -1, 10]$ (рисунок 4,а), то после изменения начального условия для y2 на 0.0001 ($y = [1, -0.9999, 10]$), фазовый портрет имеет вид, приведенный на рисунке 4, б). После изменения начального значения y3 на 0.001 ($y_0 = [1, -1, 10.001]$) фазовый портрет имеет вид, приведенный на рисунке 4, в).

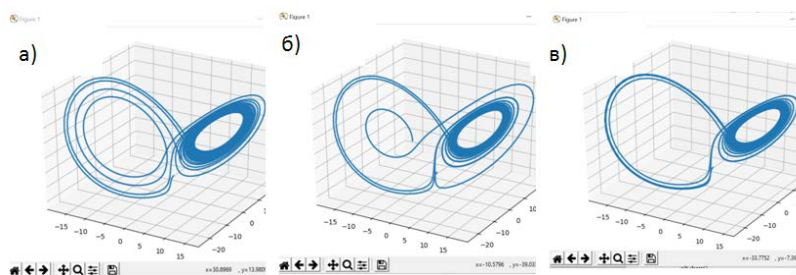


Рисунок 4 – Изменение фазового портрета модели Лоренца при изменении начальных условий

Исследуем бифуркационные процессы, появляющиеся при изменении внутреннего параметра в нелинейной динамической системе, описанной логистическим уравнением (1). Если внутренний параметр k будет плавно изменяться от начального значения с определенным шагом, и будет выполнено n отображений логистической функции, то можно получить бифуркационную диаграмму заданной функции. Например, если параметр k изменять от начального значения, равного 0.01 с шагом 0.01, то бифуркационная диаграмма будет выглядеть так, как приведено на рисунке 5, а). На рисунке 5,б) приведена диаграмма, построенная с шагом 0.1.

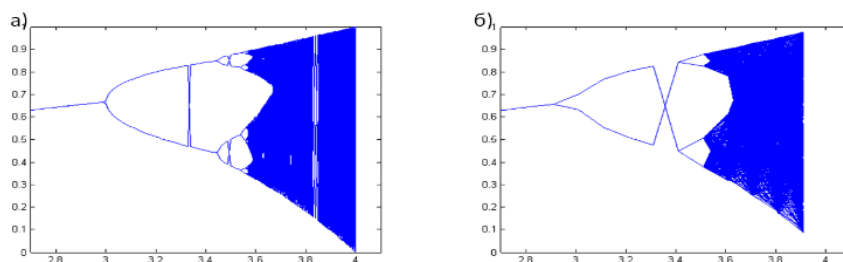


Рисунок 5 – Бифуркационная диаграмма нелинейной динамической модели

Выводы. Проведена апробация применения системы программирования Python в качестве инструментария моделирования динамических систем. Показана возможность использования библиотек Python при моделировании линейных динамических систем как в виде ОДУ, так и в виде передаточных функций. Применение компьютерного моделирования с использованием Python позволило продемонстрировать и исследовать хаотические свойства простых нелинейных систем. Совмещение изучения элементов теории катастроф с компьютерной реализацией моделей нелинейных систем, позволяет повысить наглядность полученных результатов.

Список использованных источников

1. Сысоева, М.В. Программирование для «нормальных» с нуля на языке Python: Учебник. В двух частях. Часть 1./ Сысоева М.В., Сысоев И.В. - М.:БазальтСПО; МАКСПресс, 2018.
2. Дивеев А.И., Доценко А.В. Библиотека Python для синтеза интеллектуальных систем управления // Вестник Российского университета дружбы народов. Серия: Инженерные исследования. 2018. Т. 19. № 2.
3. Бекман, И.Н. Катастрофы. Курс лекций. [Электронный ресурс] / И.Н. Бекман. – Режим доступа <http://beckuniver.ucoz.ru/Katastrofy/Katastrofy.htm>.
4. Бушуев, А.Б. Математическое моделирование процессов технического творчества. / А.Б. Бушуев– СПб: СПбГУ ИТМО, 2010.
5. Антипов, О.И. Бифуркации, катастрофы, синергетика, фракталы и нейронные сети в физических, биологических и экономических системах./ Антипов О.И., Неганов В.А. – Самара: ПГУТИ, 2013.
6. Гилмор, Р. Прикладная теория катастроф. / Р. Гилмор – М.: Мир, 1984.
7. Цветков, В.И. Теория катастроф и фрактальная модель кризисных социально-экономических процессов. / И.В. Цветков // Вестник ТвГУ. Серия: «Прикладная математика». – 2010.

8. Kapanski, A., Hruntovich, N., Bakhur, S., Markaryants, L., & Dolomanyak, L. (2020). Optimize the cost of paying for electricity in the water supply system by using accumulating tanks. In E3S Web of Conferences (Vol. 178, p. 01065). EDP Sciences.
9. Hruntovich, N. V., Kapanski, A. A., Baczynski, D., Vagapov, G. V., & Fedorov, O. V. (2019). Optimization of a variable frequency drive pump working on a water tower. In E3S Web of Conferences (Vol. 124, p. 05060). EDP Sciences.
10. Соколов, С.В. Модели динамики популяций: учеб. пособие/ С.В. Соколов. – СПб:СПбГЭТУ «ЛЭТИ», 2018.
11. Хазова, Ю. А. Элементы теории бифуркаций. Часть 2. Теория хаоса: учебно-методическое пособие / Ю. А. Хазова – ФГАОУ ВО «Крымский федеральный университет имени В. И. Вернадского». — Симферополь, 2019.

*Алферова Тамара Викторовна, кандидат технических наук, доцент,
alferowa.tam@yandex.ru, Республика Беларусь, Гомель, Гомельский технический
 университет имени П.О. Сухого, Сухого*

*Татьяна Анатольевна Трохова, кандидат технических наук, доцент,
trohova77@rambler.ru Республика Беларусь, Гомель, Гомельский технический университет
 имени П.О. Сухого*

COMPUTER SIMULATION OF DYNAMIC SYSTEMS ELEMENTS IN PYTHON

Alferova T.B., Trohova T.A.

Annotation. OBJECTIVE. Investigation of the possibilities of using the Python programming system in the analysis of processes in linear and nonlinear dynamic systems.

t is often necessary to solve the problem of choosing modeling tools when creating and researching computer models METHOD The catastrophe method which takes into account abrupt changes arising in the form of a sudden response of the system to a smooth change in external conditions have been used in solving the set tasks. RESULTS The paper presents the results of testing the Python programming system as a basis for the development of computer models of linear and nonlinear dynamic systems. A brief description of the Python libraries and modules used in the development of systems described by differential equations and transfer functions is given. A theoretical review of the occurrence of chaotic processes in nonlinear dynamical systems is given, the mechanism of transition to chaos is demonstrated on the example of constructing a model based on the Verhulst equation. A mathematical model of the Lorentz equations is implemented, which describes the variety of chaotic changes in a nonlinear dynamic system, phase portraits of the model's output parameters are built in the form of three-dimensional graphs. It is concluded that the Python programming system can be used as a toolkit in the development of computer models of linear and nonlinear dynamic systems. The approbation of this approach have been carried out in the development of models of dynamic systems in the course of mathematical modeling of complex systems.

Keywords: dynamic models, computer modeling, nonlinear dynamic systems, computer modeling tools, Python programming system.

*Alferova T.V. Ph.D., Associate Professor, alferowa.tam@yandex.ru, Republic of Belarus,
 Gomel, Gomel Sukhoi State Technical University*

*Trohova T.A., Ph.D., Associate Professor, trohova77@rambler.ru, Republic of Belarus,
 Gomel, Gomel Sukhoi State Technical University*