

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра математики факультету інформатики

Курсова робота
за спеціальністю „Прикладна математика” 113

Послідовності динамічного хаосу

Керівник курсової роботи

Щестюк Наталія Юріївна

Доцент кандидат наук

(підпис)

Виконав студент 3-го курсу

спеціальності «Прикладна математика»

Шелемей Ярослав Олегович

«__» _____ 2023 р.

Київ – 2023

Міністерство освіти і науки України

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра інформатики факультету інформатики

ЗАТВЕРДЖУЮ

Зав.кафедри математики,

проф., д.ф.-м.н.

_____ Р. К. Чорней

(підпис)

« ____ » _____ 2023 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу

студенту Шелемею Ярославу Олеговичу факультету інформатики 3 курсу

ТЕМА Послідовності динамічного хаосу

Зміст ТЧ до курсової роботи:

Індивідуальне завдання

Анотація

Вступ

Розділ 1. Поняття систем динамічного хаосу

Розділ 2. Послідовності динамічного хаосу на прикладах

Розділ 3. Проблематика динамічних хаотичних послідовностей

Висновки

Список літератури

Додатки

Дата видачі „26” жовтня 2022 р. Керівник _____

(підпис)

Завдання отримав _____

(підпис)

Тема: Послідовності динамічного хаосу

Календарний план виконання роботи:

№ п/п	Назва етапу курсової роботи	Термін виконання етапу	Примітка
1.	Отримання завдання на курсову роботу.	26.10.2022	
2.	Обговорення деталей теми на розробка плану виконання	10.11.2022	
3.	Огляд технічної літератури за темою роботи та опрацювання матеріалів	8.01.2023	
4.	Написання теоретичної частини роботи	Квітень 2023	
5.	Розробка практичної частини та фіналізація практичної	30.04.2023	
6.	Аналіз отриманих результатів, обговорення остаточної доповіді з керівником	6.05.2023	
7.	Здача курсової роботи	23.05.2023	

Студент: Шелемей Я.О.

Керівник: Щестюк Н.Ю.

“26” жовтня 2022

Зміст

Анотація	5
Вступ.....	6
Розділ 1. Поняття систем динамічного хаосу.....	8
1.1. Детерміновані системи	8
1.2. Поняття хаосу	8
1.3. Означення систем динамічного хаосу.....	9
Розділ 2. Послідовності динамічного хаосу на прикладах	11
2.1. Логістичне відображення $x \rightarrow Tx \equiv \lambda x(1 - x)$	11
2.2. Модель Бернуллі $x_n = 2x_{n-1} \pmod{1}$	15
2.3. Модель $x_n = 1 - 1 - 2x_{n-1} - 1$	17
2.4. Модель $x_n = 1 - 2x_{n-1} - 1$	18
Розділ 3. Проблематика динамічних хаотичних послідовностей	21
3.1. Ідея фрактальності (самоподібності) у бінарних моделях.....	21
3.2. Складнощі розпізнавання хаотичних та стохастичних послідовностей ..	23
3.3. Прогнозування за допомогою динамічних хаотичних послідовностей ...	26
Висновки	27
Список використаної літератури	28
Додатки.....	29

Анотація

Дана наукова робота присвячена дослідженню послідовностей динамічного хаосу та аналізу чотирьох моделей, що демонструють хаотичну поведінку систем. У роботі розглянута проблематика формування послідовностей з кількістю точок, яка відрізняється в залежності від обраної моделі, а також відображення цих послідовностей на фазових просторах. Кожна з моделей динамічного хаосу була детально проаналізована з використанням математичних методів та чисельного моделювання. Було показано, що зміна початкових умов може призвести до значних розходжень в поведінці системи, що свідчить про хаотичну поведінку.

Ключові слова: динамічний хаос, динамічні системи, детермінована система, нелінійні динамічні системи, стохастичність .

Вступ

В сучасному світі все більше уваги приділяється вивченню складних систем, що мають динамічну та нелінійну природу. Особливо актуальним є дослідження динамічного хаосу, який зустрічається в різних областях науки та техніки, включаючи фізику, біологію, економіку, інформатику та інші. В рамках динамічного хаосу вивчаються послідовності, що породжуються нелінійними детерміністичними системами, тобто системами, в яких майбутнє повністю визначається початковими умовами та параметрами.

Коли людство хотіло спрогнозувати рух ризикованих активів, акцій тощо, то для цього підбиралися стохастичні моделі, а можна використати і інший підхід – хаотичні динамічні моделі, або моделі динамічного хаосу. Однак, на практиці виявилось, що навіть прості детерміністичні системи типу $x_{n+1} = f(x_n; \lambda)$ можуть породжувати послідовності, які мають властивості схожі на стохастичні, тобто випадкові послідовності. З точки зору прогнозу майбутнього руху цін значний інтерес представляє питання про те, наскільки прогнозовані нелінійні хаотичні моделі. Ми побачимо далі, що ситуація тут не дуже оптимістична, і причина полягає в тому, що незважаючи на детермінований характер, поведінка траєкторій хаотичних систем може значно змінюватися залежно від точності визначення значень початкових даних та значень деякого параметра λ [10, 11].

Метою нашої роботи є з'ясування, наскільки ефективно можна прогнозувати поведінку послідовностей, що породжуються різними моделями динамічного хаосу та розглянути проблеми, що можуть виникнути у таких послідовностей.

Об'єктом дослідження є послідовності динамічного хаосу утворені різними моделями.

Предметом даного дослідження є послідовності утворені чотирма різними моделями динамічного хаосу, їх властивості, поведінка траєкторій, проблематика та ймовірності.

Робота складається із трьох розділів. У першому розглядаються загальні поняття із теорії хаосу та детермінованих систем. Другий розділ присвячений аналізу чотирьох моделей динамічного хаосу та їх властивостям. У третьому розділі розглядають проблеми таких послідовностей, наприклад, ідея фрактальності.

Розділ 1. Поняття систем динамічного хаосу

1.1. Детерміновані системи

Детермінована система – це така система, яка задається однозначно визначеними функціями, а поведінка завжди передбачена та прогнозована[1]. На відміну від стохастичних систем – поведінка яких несе ймовірнісний характер та описується розподілами випадкових величин, детерміновані не мають невизначеності, оскільки з відомим початковим станом та програмою обробки інформації можна точно передбачити її наступний стан.

Якщо у нас існує такий простір всіх можливих станів Ω певної системи, який допускає міру ймовірності $prob(\cdot)$, то ми можемо охарактеризувати поняття детермінованої системи. Нехай система визначена як $f_t: \mathcal{P}(\Omega) \rightarrow \mathcal{P}(\Omega)$, де $\mathcal{P}(\Omega)$ - частини всіх можливих станів в момент часу t . Тоді дана система буде вважатись детермінованою якщо

$$\forall x \in \Omega : \forall A \subset \Omega : (x \in A) \Leftrightarrow \exists! z \in \Omega : prob(f_t(A) = \{z\}) = 1 \quad (1)$$

– для деякої системи, що в визначена в просторі Ω при виконанні певних умов існує один єдиний елемент z , який відповідає значенню f_t на будь-якій підмножині A з Ω із ймовірністю 1. Іншими словами, система є детермінованою, коли результат її функції на будь-якому елементі можна передбачити[2].

Проста детермінована система - це система, яка містить невелику кількість елементів, має визначену поведінку та малу кількість внутрішніх зв'язків. Прикладом може бути система розміщення складів, яка вирішує задачу забезпечення оптимального переміщення продукції від складу до місця призначення[3].

1.2. Поняття хаосу

Теорія хаосу – галузь науки, що займається вивченням систем, поведінка яких залежить від початкових умов та набору даних, що у свою чергу робить довгострокове прогнозування майже неможливим. Хоча динаміка цих систем

характеризується фізичними законам, проте вона також є нерегулярною – такі системи називаються детермінованим хаосом. У своїй науковій статті Е. Лоренц так охарактеризував теорію хаосу: «Chaos: When the present determines the future, but the approximate present does not approximately determine the future.» [4].

У теорії хаосу термін "хаотичний" має конкретне визначення, відмінне від його вживання в повсякденному житті, де він часто асоціюється з безладом. Детермінований хаос - це складна та непередбачувана поведінка детермінованої нелінійної системи, яка має визначені правила та не містить елементів випадковості. З'ясувалося, що іноді прості детерміновані системи можуть проявляти складну та непередбачувану випадковість що в свою чергу і розуміється як хаос[5]. Незважаючи на цю непередбачуваність детермінованого хаосу, при детальнішому дослідженні можна відкрити закономірності в поведінці деякої нелінійної детермінованої системи а також дане явище можна відтворити в лабораторних експериментах, що робить його відмінним від класичних випадкових процесів[6].

1.3. Означення систем динамічного хаосу

Щоб певна динамічна система класифікувалась як хаотична, їй повинні бути притаманні такі властивості[7]:

- 1) Нелінійність;
- 2) Чутливість до початкових умов;
- 3) Щільність періодичних точок;
- 4) Наявність топологічного змішування.

У системі, яка демонструє чутливість до початкових умов, навіть незначні відмінності в початкових умовах можуть призвести до значних відмінностей у подальшій поведінці системи. Таким чином, будь-які точки, що знаходяться як завгодно близько одна до одної через певний момент часу t можуть опинитися як завгодно далеко. Часто в науковій літературі цю властивість називають «ефектом метелика» - таку назву запропонував Едвард Лоренц у своїй науковій роботі для U. Washington Press [8].

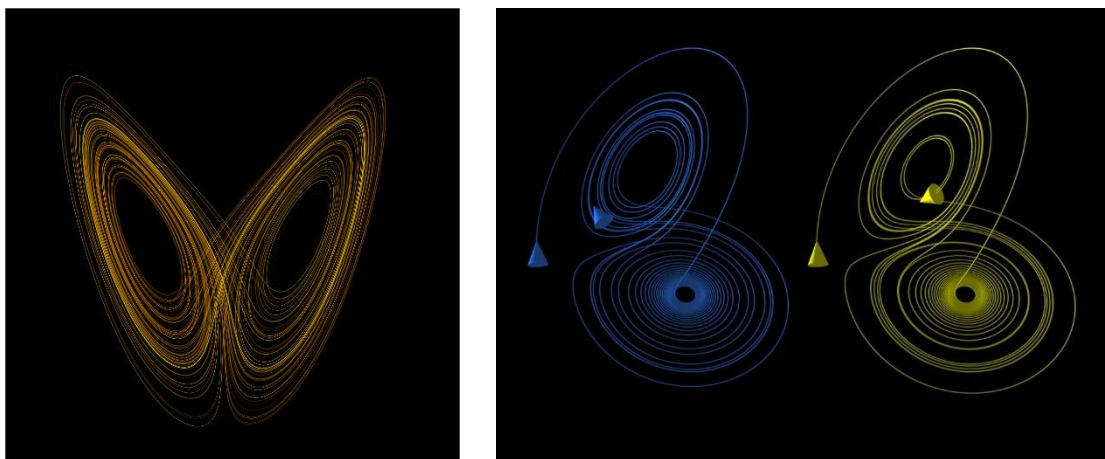


Рис. 1.3. Ефект метелика в динамічних хаотичних системах

Послідовність, що створюється певною моделлю динамічної хаотичної системи і буде називатись послідовністю динамічного хаосу.

Розділ 2. Послідовності динамічного хаосу на прикладах

2.1. Логістичне відображення $x \rightarrow Tx \equiv \lambda x(1 - x)$

Розглянемо нелінійну динамічну послідовність

$$x_n = \lambda x_{n-1}(1 - x_{n-1}), \quad n \geq 1, \quad 0 < x_0 < 1 \quad (1),$$

що утворюється із лінійного відображення $x \rightarrow Tx \equiv \lambda x(1 - x)$ (2).

Для того щоб дослідити дану послідовність напишемо код на Python, який буде за заданою функцією, значенням λ та x_0 знаходити перших n -членів послідовності (1) та будувати графік:

```
def logistic_map(x0, lambd, n):  
    x = [x0]  
    for i in range(n-1):  
        x.append(lambd * x[-1] * (1 - x[-1]))  
    return x  
  
x0 = 0.2 # початкове значення  
lambd = 4 # параметр  $\lambda$   
n = 100 # кількість членів послідовності
```

Тепер проведемо аналіз отриманих графіків для різних λ .

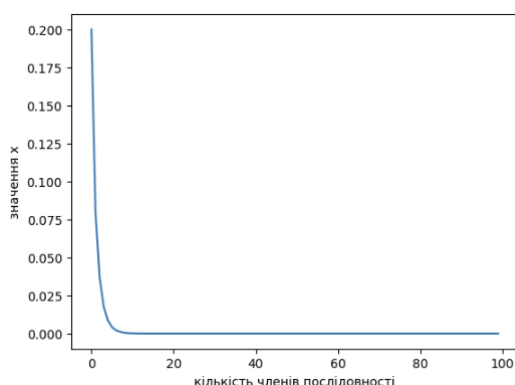


Рис. 2.1.1. $\lambda = 0.5, x_0 = 0.2$

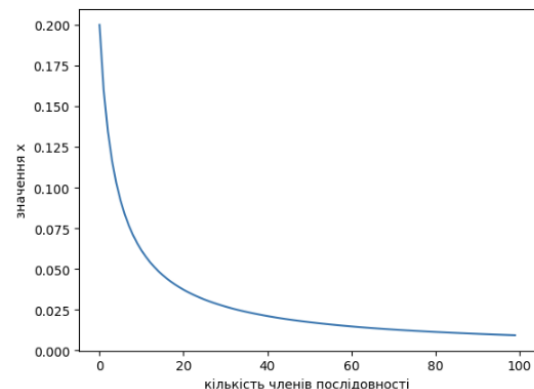


Рис. 2.1.2. $\lambda = 1, x_0 = 0.2$

Як бачимо по графікам (рис. 2.1.1 та рис. 2.1.2.), то для всіх $\lambda \leq 1$ розв'язок нашої послідовності $x_n = \lambda x_{n-1}(1 - x_{n-1}) \rightarrow 0$ при $n \rightarrow \infty$ при всіх $0 < x_0 < 1$. Це

свідчить про те, що ми можемо вважати стан $x_\infty = 0$ як єдиний стійкий розв'язок нашої системи, до якого сходиться значення x_n при $n \rightarrow \infty$.

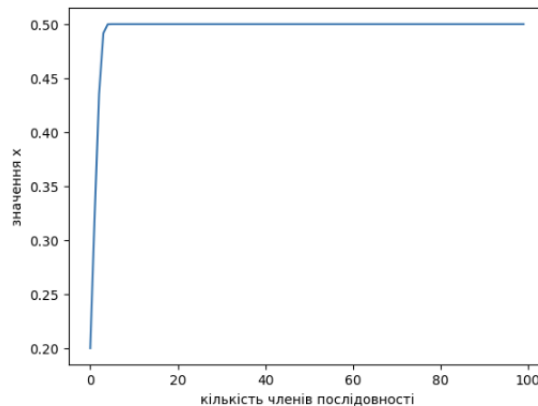


Рис. 2.1.3. $\lambda = 2, x_0 = 0.2$

Так при $\lambda = 2$ (рис. 2.1.3.) розв'язок послідовності $x_n \rightarrow 1/2$ при $n \rightarrow \infty$ при всіх $0 < x_0 < 1$. Відповідно для такого випадку $x_\infty = 1/2$ – єдиний стійкий розв'язок системи, до якого збігається значення x_n при $n \rightarrow \infty$.

Тепер будемо поступово збільшувати значення λ щоб подивитись на поведінку нашої послідовності. Як бачимо з рис. 2.1.4-2.1.7 при $2 \leq \lambda < 3$ система все ще має один стійкий розв'язок до якого збігається x_n при $n \rightarrow \infty$.

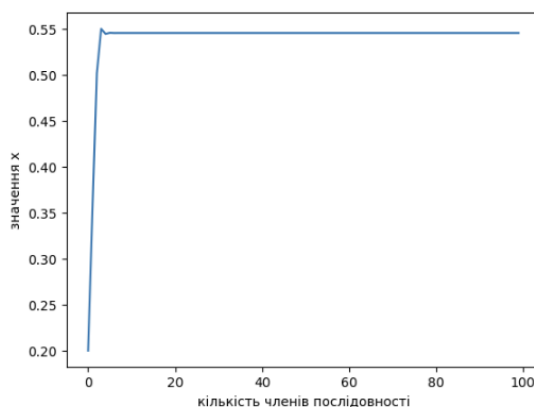


Рис. 2.1.4. $\lambda = 2.2, x_0 = 0.2$

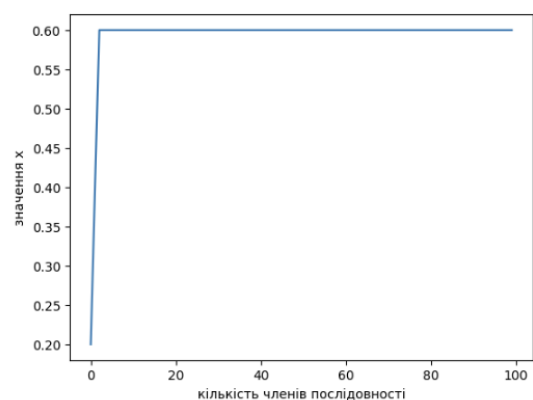


Рис. 2.1.5. $\lambda = 2.5, x_0 = 0.2$

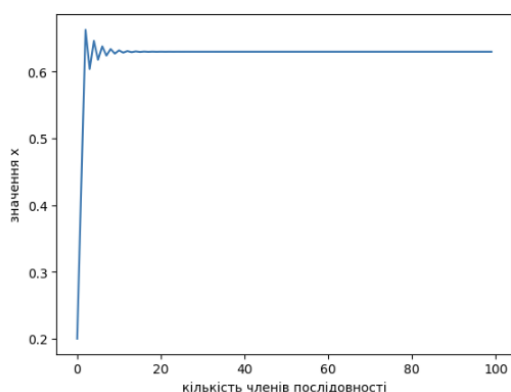


Рис. 2.1.6. $\lambda = 2.7, x_0=0.2$

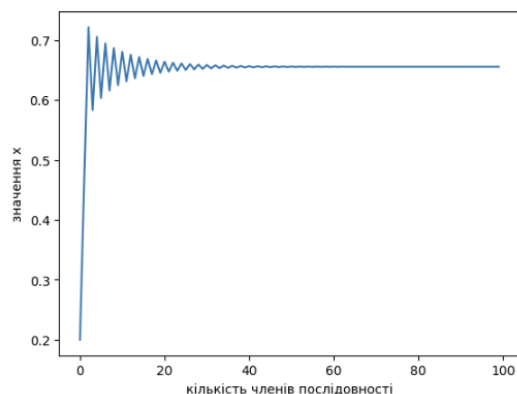


Рис. 2.1.7. $\lambda = 2.9, x_0=0.2$

Однак при $\lambda = 3$ поведінка системи змінюється – з'являється вже 2 стійких розв'язки для x_∞ до яких збігається x_n при $n \rightarrow \infty$ (рис. 2.1.8.). Така ж ситуація буде зберігатись при $3 \leq \lambda < 3.4494$ (рис. 2.1.9.).

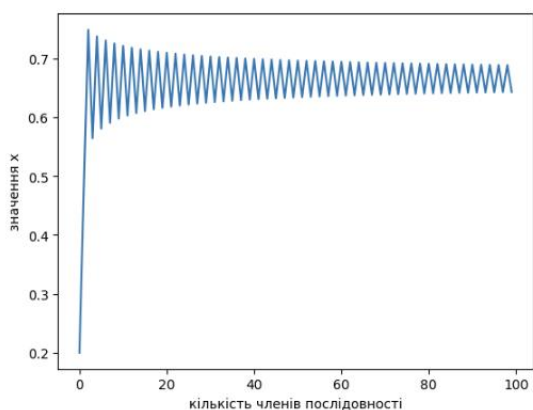


Рис. 2.1.8. $\lambda = 3, x_0=0.2$

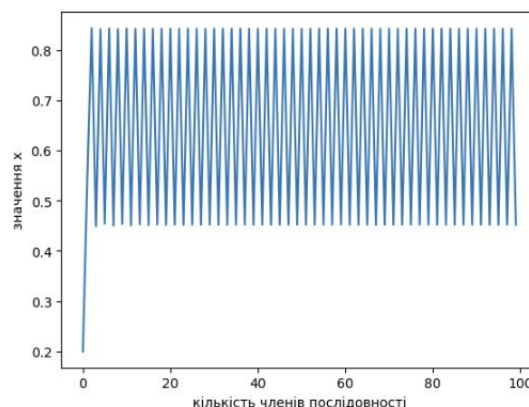


Рис. 2.1.9. $\lambda = 3.4, x_0=0.2$

Починаючи із $\lambda = 3.4494$ поведінка системи знову змінюється – тепер кількість стійких розв'язків стає 4 для x_∞ до яких збігається x_n при $n \rightarrow \infty$ (рис. 2.1.10.).

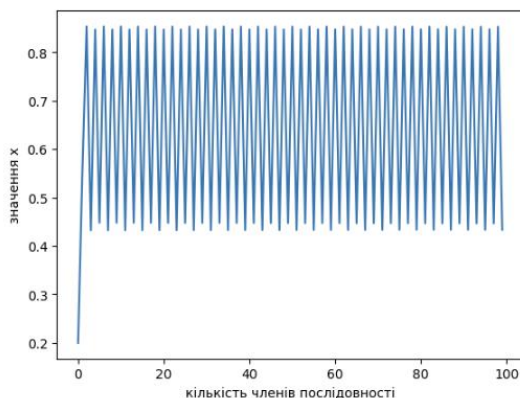


Рис. 2.1.10. $\lambda = 3.4494, x_0=0.2$

При поступовому невеликому збільшенні λ кількість стійких розв'язків буде збільшуватись із швидкістю геометричної прогресії. Так при $\lambda = 3.5644$ будемо мати 16 станів для x_∞ , для $\lambda = 3.5696$ – вже 64 стани. А при $\lambda = 3.6$ кількість таких розв'язків вже буде близькою до ∞ , що в свою чергу означає втрату стійкості системою та перехід в стан хаотичності. На рис. 2.1.11. можемо бачити як стрімко зростає кількість сталих розв'язків для незначного збільшення λ до моменту $\lambda \rightarrow 4$.

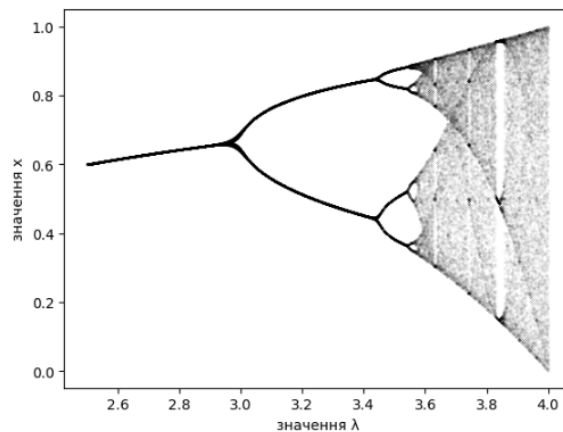


рис. 2.1.11. Діаграма збільшення кількості розв'язків в хаотичній послідовності зі збільшенням λ (біфуркація хаотичної послідовності)

Отже, значення $\lambda = 4$ грає важливу роль для нашої системи (1), оскільки саме при ньому послідовність x_n має схожу поведінку до стохастичної послідовності білого шуму. Таким чином для системи $x_n = 4x_{n-1}(1 - x_{n-1})$, $n \geq 1$, $0 < x_0 < 1$, існує певний інваріантний розподіл P , щільність якого можна задати формулою:

$$p(x) = \frac{1}{\pi[x(1-x)]^{1/2}}, 0 < x < 1. \quad (2)$$

Варто зауважити, що в такому випадку значення x_n будуть мати такий самий розподіл як і x_0 .

Із формули (2) можемо знайти також значення математичного сподівання та дисперсії для послідовності (1): $E(x_0) = 1/2$, $E(x_0^2) = 3/8$, $D(x_0) = 1/8$ [9].

2.2. Модель Бернуллі $x_n = 2x_{n-1} \pmod{1}$

Розглянемо модель Бернуллі $x_n = 2x_{n-1} \pmod{1}, n \geq 1, 0 < x_0 < 1$. (3)

На відміну від попередньої системи, в цій немає параметру λ тому початкові умови та поведінка послідовності буде залежати від x_0 та n . Для дослідження напишемо функцію на Python, що буде обраховувати перших n -членів послідовності (3) залежно від початкового значення x_0 та будувати графік:

```
def bernoulli(x0, n):  
    xn = [x0]  
    for i in range(n):  
        xn.append(2*xn[-1] % 1)  
    return xn  
  
# задаємо значення x0 та кількість членів послідовності  
x0 = 0.0001  
n = 100
```

Зафіксуємо розмір послідовності $n = 100$ та будемо змінювати x_0 щоб знайти певні закономірності. При всіх $0.1 < x_0 < 0.9$ система має схожу поведінку – перших $n \rightarrow 55$ членів система має кілька стійких розв’язків, проте потім значення для $x_\infty \rightarrow 0$ до якого і буде збігатися x_n при $n \rightarrow \infty$ (рис. 2.2.1-2.2.4.).

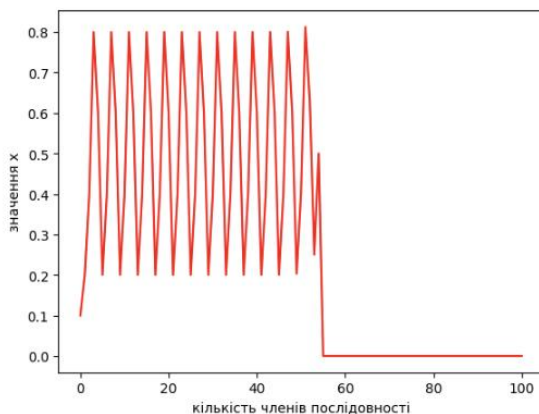


Рис. 2.2.1. $x_0=0.1$

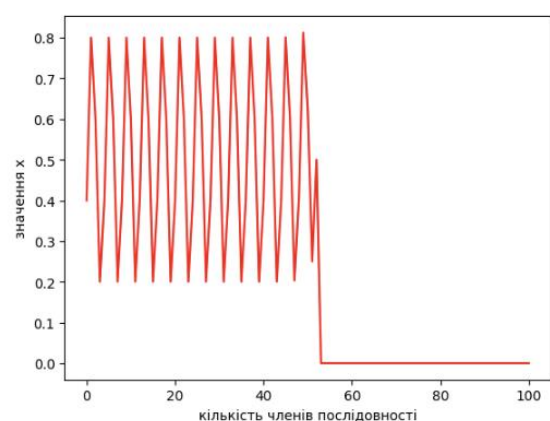


Рис. 2.2.2. $x_0=0.4$

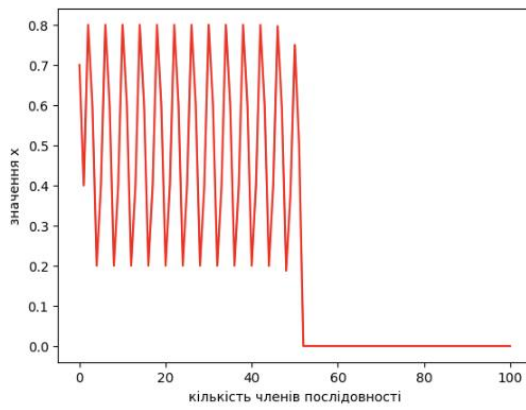


Рис. 2.2.3. $x_0=0.7$

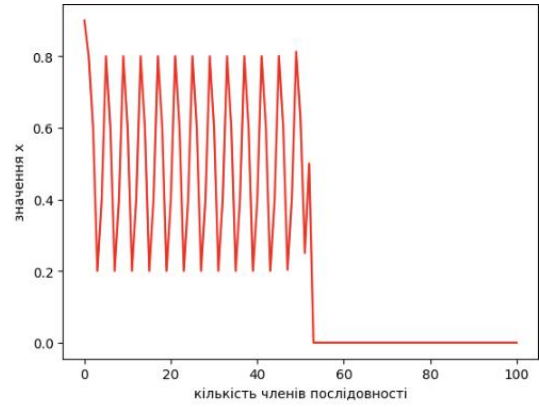


Рис. 2.2.4. $x_0=0.9$

Децю іншою буде поведінка послідовності при $x_0 \rightarrow 0$ та $x_0 \rightarrow 1$. Для таких початкових значення система є набагато більш хаотичною для перших $n \rightarrow 55$ членів (рис. 2.2.5-2.2.8), проте в результаті маємо таку ж ситуацію для x_∞ , що і для всіх інших x_0 з проміжку $(0;1)$ – $x_\infty \rightarrow 0$ до якого збігається x_n при $n \rightarrow \infty$.

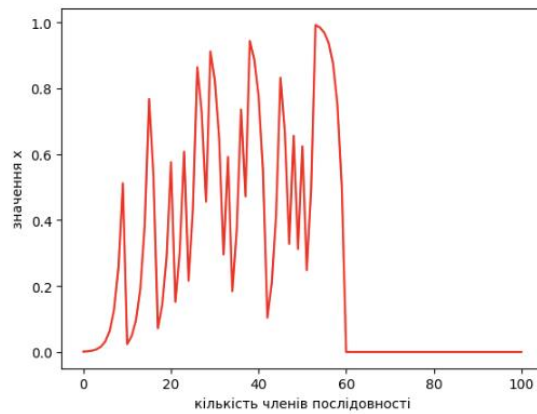


Рис. 2.2.5. $x_0=0.001$

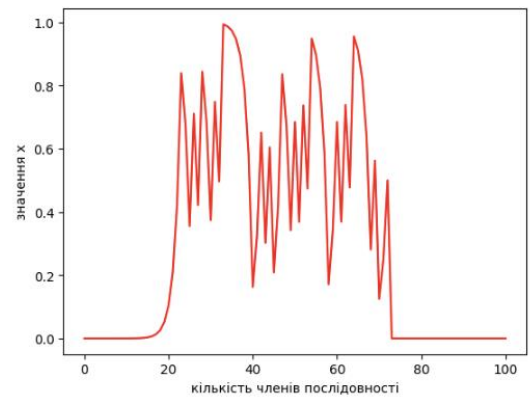


Рис. 2.2.6. $x_0=0.0000001$

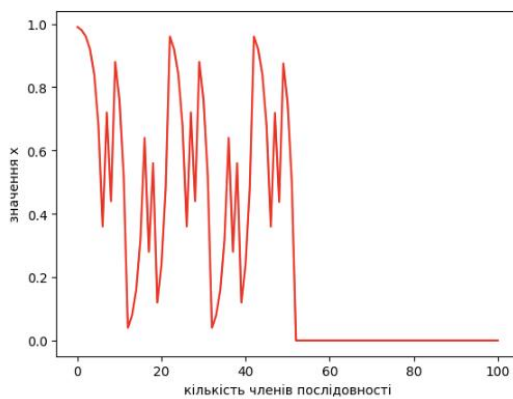


Рис. 2.2.7. $x_0=0.99$

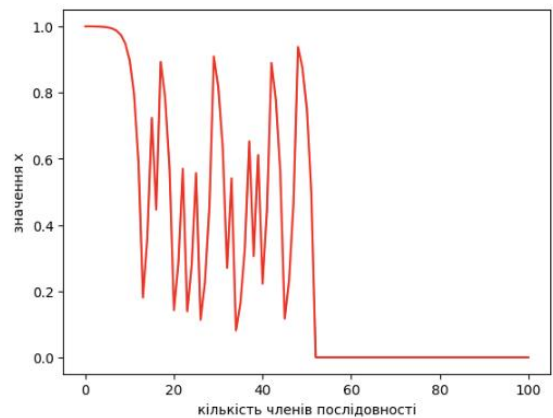


Рис. 2.2.8. $x_0=0.9999$

Для цієї послідовності (3) інваріантним розподілом буде рівномірний розподіл $p(x) = 1, 0 < x < 1$. Обчислені значення математичного сподівання на дисперсії будуть рівними – $E(x_0) = 1/2, E(x_0^2) = 1/3, D(x_0) = 1/12$ [9].

2.3. Модель $x_n = 1 - |1 - 2x_{n-1}|$

Для дослідження поведінки моделі $x_n = 1 - |1 - 2x_{n-1}|, n \geq 1, 0 < x_0 < 1$ (4), напишемо функцію на Python, що буде знаходити перших n - членів послідовності (4) залежно від початкового значення x_0 та будувати графік:

```
def dynamic_map(x0, n):
    xn = x0
    x = [x0]
    for i in range(n):
        xn = 1 - abs(1 - 2*xn)
        x.append(xn)
    return x

# задаємо значення x0 та кількість членів послідовності
x0 = 0.0001
n = 100
```

Поведінка цієї моделі аналогічна до попередньої (3). При всіх $0.1 < x_0 < 0.9$ та $n = 100$ – перших $n \rightarrow 55$ членів послідовності наша система має кілька стійких розв’язків, проте потім значення для $x_\infty \rightarrow 0$ до якого і буде збігатися x_n при $n \rightarrow \infty$ (рис. 2.3.1-2.3.6.). А для $x_0 \rightarrow 0$ та $x_0 \rightarrow 1$ система набагато більш хаотична для перших $n \rightarrow 55$ членів (рис. 2.2.12-2.2.14), але в кінці так само має – $x_\infty \rightarrow 0$ до якого збігається x_n при $n \rightarrow \infty$.

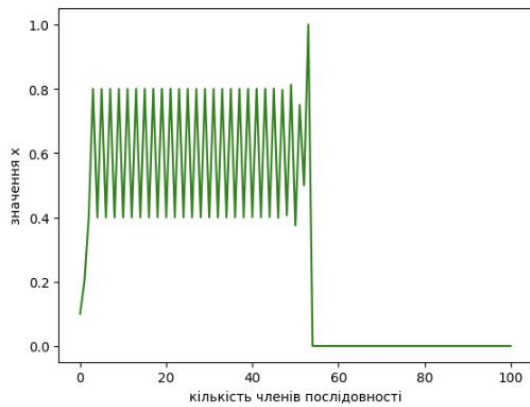


Рис. 2.3.1. $x_0=0.1$

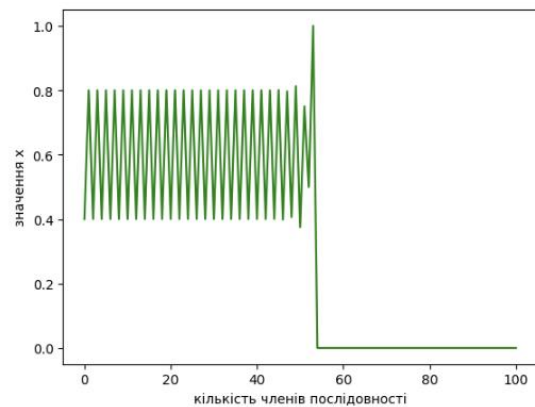


Рис. 2.3.2. $x_0=0.4$

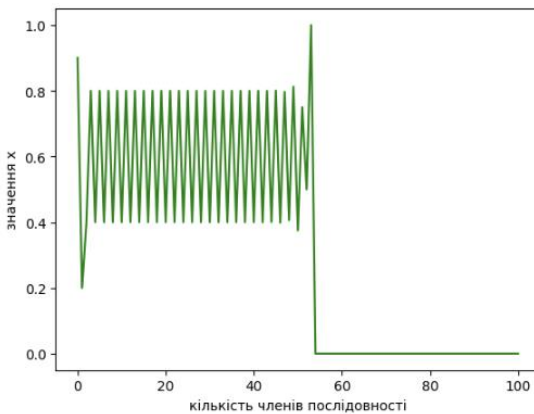


Рис. 2.3.3. $x_0=0.9$

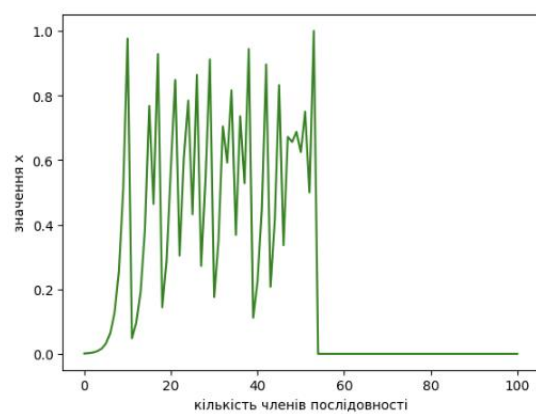


Рис. 2.3.4. $x_0=0.001$

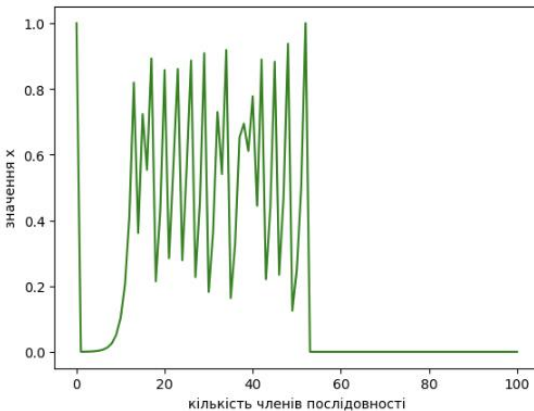


Рис. 2.3.5. $x_0=0.9999$

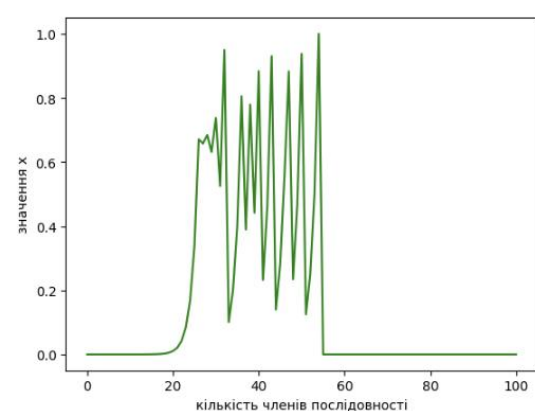


Рис. 2.3.6. $x_0=0.00000001$

Для послідовності (4) інваріантний розподіл - рівномірний розподіл $p(x) = 1$, $0 < x < 1$. Обчислені значення математичного сподівання на дисперсії будуть рівними – $E(x_0) = 1/2$, $E(x_0^2) = 1/3$, $D(x_0) = 1/12$ [9].

2.4. Модель $x_n = 1 - 2\sqrt{|x_{n-1}|}$

Для дослідження поведінки моделі $x_n = 1 - 2\sqrt{|x_{n-1}|}$, $n \geq 1$, $-1 < x_0 < 1$ (5), напишемо функцію на Python, що буде знаходити перших

n -членів послідовності (5) залежно від початкового значення x_0 та n , а також будувати графік:

```
def f(x):
    return 1 - 2*np.sqrt(np.abs(x))

def logistic_map(x0, n):
    x = np.zeros(n)
    x[0] = x0
    for i in range(1, n):
        x[i] = f(x[i-1])
    return x

# задаємо значення x0 та кількість членів послідовності
x0 = 0.5
n = 100
```

Поведінка цієї послідовності вже відмінна від попередніх трьох – в ній немає прямої залежності від x_0 чи n (рис. 2.4.1-2.4.6.).

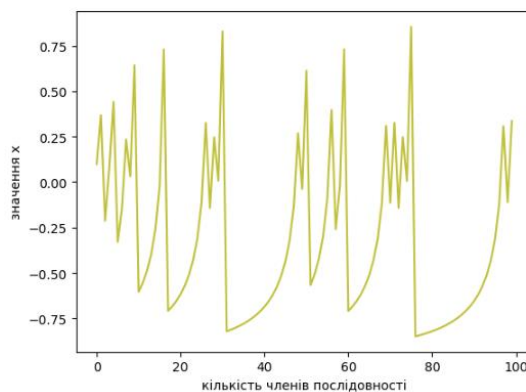


Рис. 2.4.1. $x_0=0.1, n=100$

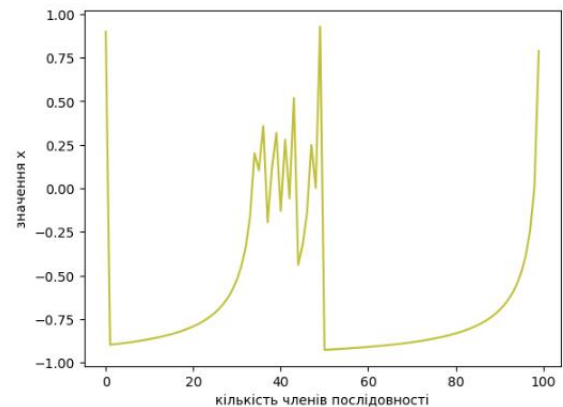


Рис. 2.4.2. $x_0=0.9, n=100$

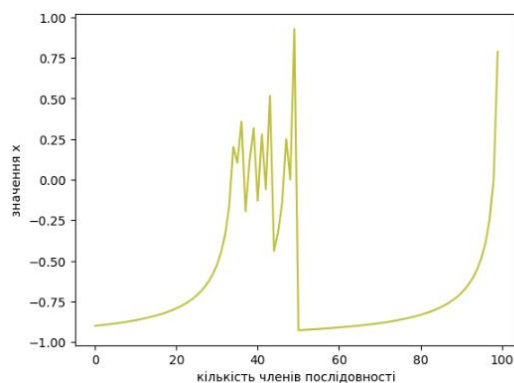


Рис. 2.4.3. $x_0=-0.9, n=100$

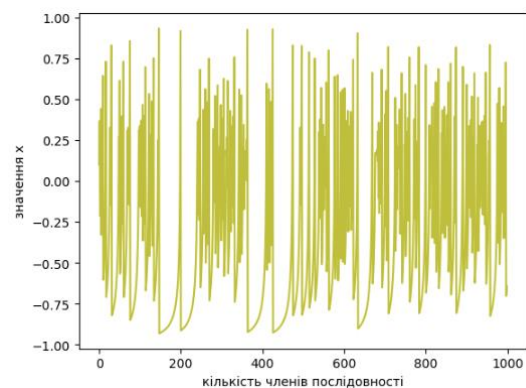


Рис. 2.4.4. $x_0=0.1, n=1000$

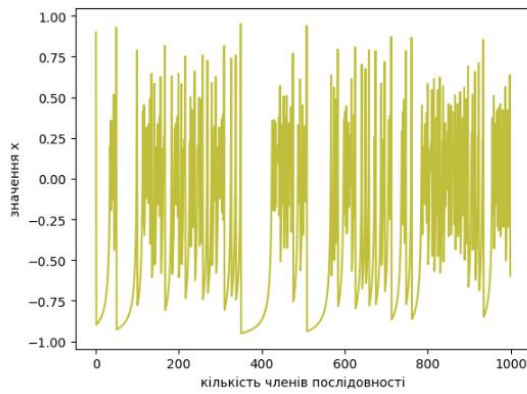


Рис. 2.4.5. $x_0=0.9, n=1000$

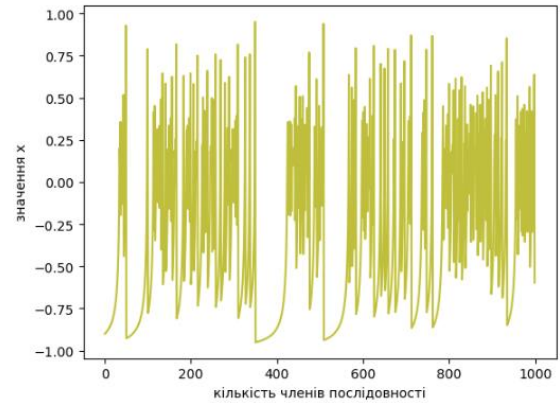


Рис. 2.4.6. $x_0=-0.9, n=1000$

Отже, можна сказати що послідовність (5) має безліч стійких розв'язків, а значить існує безліч значень x_∞ до якого буде збігатися x_n при $n \rightarrow \infty$.

Для послідовності (5) інваріантний розподіл буде мати вигляд $p(x) = (1-x)/2, -1 < x < 1$. Обчислені значення математичного сподівання на дисперсії будуть рівними – $E(x_0) = -1/3, E(x_0^2) = 1/3, D(x_0) = 2/9$ [9].

Розділ 3. Проблематика динамічних хаотичних послідовностей

3.1. Ідея фрактальності (самоподібності) у бінарних моделях

Ідея фрактальності (самоподібності) полягає в тому, що при деяких значеннях параметра λ чи x_0 , динаміка системи стає самоподібною. Це означає, що при збільшенні масштабу зображення чи певних значень, ми отримуємо ту ж саму поведінкову структуру послідовності що і при початкових[12].

Серед послідовностей (1), (3), (4) та (5) найбільше прослідковується ця ідея і першої і останньої. Для логістичного відображення самоподібність буде відрізнятися для різних початкових значень λ – для $\lambda < 3$ будемо мати 1 розв’язок, для при $\lambda = 3$ вже буде 2 розв’язки, при $\lambda = 3.5644$ будемо мати 16 розв’язків, для $\lambda = 3.5696$ - 64 розв’язки, а при $\lambda = 3.6$ кількість таких розв’язків вже буде близькою до ∞ . Детальніше це розглядало в пункті 2.1 даної курсової роботи. Саме те, що для будь-якого n ми будемо мати певну сталу кількість рішень залежно від початкових умов, ілюструє ідею фрактальності в цій моделі. Графічно це можна побачити на рис. 2.1.8-2.1.10 та рис. 3.1.1-3.1.2.

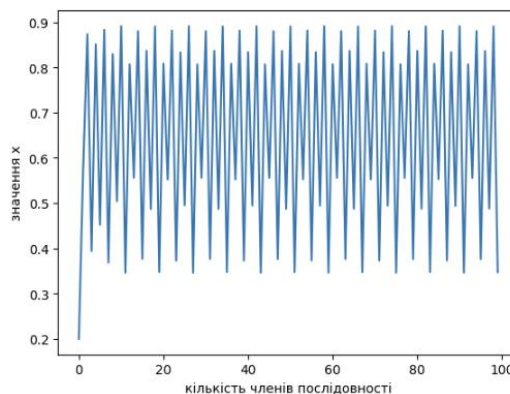


рис. 3.1.1. логістичне відображення із початковими умовами $x_0=0.1$, $n=100$,
 $\lambda = 3.5644$

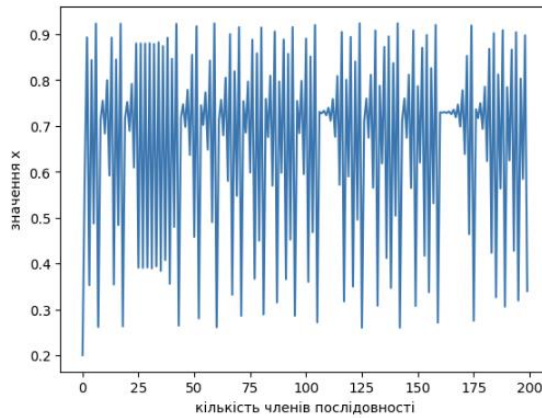


рис. 3.1.1. логістичне відображення із початковими умовами $x_0=0.1$, $n=200$,
 $\lambda = 3.696$

Для послідовності (5) ця ідея дуже гарно прослідковується на рис. 2.4.1-2.4.6 – малі деталі системи детально повторюють загальну структуру.

Якщо говорити про послідовності (3) та (4), то в них гарно прослідковується на графіках ця ідея до $n \rightarrow 55$ для всіх початкових значень x_0 .

Ідея фрактальності (самоподібності) може бути корисною для аналізу хаотичних динамічних моделей в різних галузях науки та техніки. Ось деякі приклади:

- 1) Фінанси: фрактальна структура може виявлятися в поведінці фінансових інструментів, таких як акції, валютні курси тощо. Аналізуючи цю структуру, можна спробувати передбачити майбутні зміни цих інструментів.
- 2) Геологія: фрактальна структура може бути корисною для аналізу природних явищ, таких як землетруси, зсуви ґрунту тощо. Це може допомогти у побудові більш точних моделей для передбачення цих явищ.
- 3) Метеорологія: фрактальна структура може використовуватися для аналізу кліматичних змін, таких як зміна температури, опадів, вітру тощо. Це може допомогти у покращенні точності прогнозування погодних умов.

3.2. Складнощі розпізнавання хаотичних та стохастичних послідовностей

Досить давно відомо, що деякі детерміністичні динамічні системи можуть відтворювати властивості стохастичного білого шуму, що дивно для багатьох. Це ставить питання про те, як розрізнити стохастичні та хаотичні послідовності і чи можливо це взагалі зробити. Розглянемо основний підхід до виявлення відмінностей між стохастичними послідовностями та послідовностями динамічного хаосу та приклад що ілюструє проблематику розпізнавання.

Одним із найбільш відомих способів як можна відрізнити стохастичні послідовності від хаотичних – це метод показнику кореляції. Основна ідея полягає в тому, що кореляційна розмірність для цих послідовностей буде різною – для стохастичних вона має бути більше за послідовності динамічного хаосу. Для цього дослідження визначимо функцію що задає кореляційний інтеграл як

$$C(\varepsilon) = \lim_{N \rightarrow \infty} \frac{\varphi(N, \varepsilon)}{N^2} \quad (6)$$

де $\varphi(N, \varepsilon)$ - число пар (i, j) , $i, j \leq N$ для яких виконується $|x_i - x_j| < \varepsilon$.

Для стохастичних послідовностей виду білого шуму справедливою є така функція

$$C_m(\varepsilon) \sim \varepsilon^{Vm} \quad (7)$$

в якій Vm – швидкість з якою експоненціально зростає C_m , та $Vm = m$. Показник Vm в науковій літературі також іноді називають кореляційною розмірністю.

Згідно роботи [12] для величини Vm прийнято використовувати таку формулу:

$$\widetilde{V}_{m,j} = \frac{\ln C_m(\varepsilon_j) - \ln C_m(\varepsilon_{j+1})}{\ln \varepsilon_j - \ln \varepsilon_{j+1}} \quad (8)$$

де $\varepsilon_j = \varphi^j$, $0 < \varphi < 1$.

Із тієї ж роботи [12] візьмемо значення для кореляційних таблиць стохастичної послідовності та хаотичної із такими змінними $N = 500$, $m = 1, 2, 3, 4, 5, 10$, $j = [16, 20, 25, 30, 35, 40]$.

$j \backslash m$	1	2	3	4	5	10
16	0.78	0.96	1.05	1.00	0.95	0.80
20	0.77	0.90	0.92	1.02	1.12	0.76
25	0.81	0.89	0.94	0.93	0.86	0.50
30	0.74	0.88	1.00	0.98	0.97	1.08
35	0.81	0.85	0.84	0.86	0.97	0.47
40	0.74	0.80	0.81	0.78	0.90	1.94

Табл. 3.2.1. $\widetilde{V}_{m,j}$ для логістичної системи (1)

$j \backslash m$	1	2	3	4	5	10
16	0.78	1.56	2.34	3.13	3.91	7.49
20	0.88	1.75	2.62	3.46	4.39	9.70
25	0.99	1.95	2.91	3.71	4.55	-
30	0.99	1.99	3.04	4.33	5.75	-
35	0.96	1.83	2.46	0.00	-	-
40	1.00	1.71	1.94	0.00	-	-

Табл. 3.2.2. $\widetilde{V}_{m,j}$ для гауссівського білого шуму

Отже, бачимо що стохастичні та хаотичні послідовності схожі при $m = 1$, проте чим більше стає m тим більш помітної є різниця між ними – значення для стохастичної системи набагато більші.

Яскравим прикладом проблематики визначення відмінностей між стохастичними та хаотичними послідовностями є розгляд значень $\widetilde{V}_{m,j}$ для індексів IBM та S&P500 у фінансових рядах.

$j \backslash m$	1	2	3	4	5	10
16	0.20	0.38	0.55	0.72	0.87	1.56
20	0.46	0.90	1.31	1.68	2.05	3.63
25	0.74	1.46	2.14	2.78	3.41	6.27
30	0.89	1.76	2.61	3.44	4.27	8.44
35	0.97	1.93	2.88	3.82	4.79	9.84
40	0.98	1.96	2.94	3.86	4.94	-

Табл. 3.2.3. $\widetilde{V}_{m,j}$ для IBM

$j \backslash m$	1	2	3	4	5	10
16	0.33	0.63	0.89	1.14	1.36	2.27
20	0.58	1.10	1.58	2.03	2.43	3.93
25	0.81	1.57	2.27	2.29	3.51	5.65
30	0.93	1.82	2.67	3.49	4.25	6.93
35	0.98	1.94	2.88	3.79	4.75	11.00
40	0.99	1.98	2.92	3.84	4.81	-

Табл. 3.2.4. $\widetilde{V}_{m,j}$ для S&P500

Із даних табл. 3.2.3-3.2.4. гарно видно що два індекси мають дуже схожу поведінку та значення для кореляційної розмірності. Також ми можемо побачити що обидва індекси мають скоріше властивості гауссівського білого шуму із табл. 3.1.2., ніж динамічних хаотичних послідовностей.

3.3. Прогнозування за допомогою динамічних хаотичних послідовностей

У динамічних хаотичних послідовностях малі зміни початкових умов можуть привести до значних змін у подальшому розвитку послідовності. Однак, хоча хаотичні послідовності і можуть здаватися повністю випадковими, вони мають свої закономірності та структуру.

В окремих випадках можливо використовувати хаотичні послідовності для прогнозування. Наприклад, у застосуваннях, пов'язаних з криптографією, можна використовувати хаотичні послідовності для генерації випадкових ключів шифрування, які не можуть бути передбачені. Також можна використовувати методи аналізу хаотичних послідовностей, такі як фрактальний аналіз, для виявлення певних закономірностей та структури в цих послідовностях. Це може бути корисним для передбачення деяких властивостей хаотичних систем, наприклад, часової залежності чи інших кореляцій в даних.

Існують також різні методи прогнозування за допомогою хаотичних послідовностей. Один з них - це метод нейромережного спектрального аналізу часових послідовностей. Цей метод використовується для виділення ортогональних складових часової послідовності та прогнозування їх майбутнього стану. [14]

Висновки

У даній науковій роботі була проведена детальна аналітична та числова характеристика чотирьох моделей динамічного хаосу. Було досліджено властивості цих моделей та проаналізовано їх поведінку в залежності від початкових умов. Було показано, що динамічний хаос проявляється в системах з нелінійною динамікою та може призвести до значних розходжень в поведінці системи при незначних змінах у початкових умовах.

Результати досліджень вказують на те, що динамічний хаос є важливим явищем в сучасній науці, яке потребує подальшого дослідження та застосування в практичних задачах, таких як передбачення та керування складними системами через проблематику розпізнавання хаотичних та стохастичних послідовностей. Існує багато методів та способів це дослідити, проте всі вони не ідеальні і можуть працювати не для всіх систем що аналізуються.

Список використаної літератури

1. КЛАСИФІКАЦІЯ СИСТЕМ [Електронний ресурс] – Режим доступу до ресурсу: <http://um.co.ua/3/3-14/3-148181.html>.
2. Енциклопедія кібернетики : у 2 т. / за ред. В. М. Глушкова. — Київ : Гол. ред. Української радянської енциклопедії, 1973.
3. Детерміновані і стохастичні системи [Електронний ресурс] – Режим доступу до ресурсу: <http://um.co.ua/3/3-14/3-148182.html>.
4. Lorenz E. Chaos in an Atmosphere Hanging on a Wall / Edward Lorenz. – Danforth. – (Mathematics of Planet Earth 2013).
5. Динамічний (детермінований) хаос [Електронний ресурс] – Режим доступу до ресурсу: <http://um.co.ua/4/4-15/4-153318.html>.
6. Детермінований хаос [Електронний ресурс] – Режим доступу до ресурсу: <http://chaos.kpi.ua/uk/node/5>.
7. Gleick J. Chaos. Making a new science / James Gleick., 1987. – 359 с.
8. Lorenz E. The Essence of Chaos / Edward Lorenz. – Seattle, 1993. – (U. Washington Press).
9. Shiryaev A. Essentials of stochastic finance / Albert N. Shiryaev.. – 852 с.
10. Brock W.A, Hsieh D.A., Le Baron B. Nonlinear Dynamics, Chaos, and Instability: Statistical Theory and Economic Evidence. Cambridge, MA: MIT Press, 1991.
11. Chen P., Day R.H. (Eds). Nonlinear Dynamics and Evolutionary Economics. Cambridge, MA: MIT Press.
12. May, R. Simple mathematical models with very complicated dynamics. Nature 261, 459–467 (1976) – Режим доступу до ресурсу: <https://doi.org/10.1038/261459a0>.
13. Liu T. Using the Correlation Exponent to Decide Whether an Economic Series is Chaotic / T. Liu, W. Heller, C. Granger.. – (John Wiley & Sons).
14. Андрієцький Б. Р. Прогнозування часових послідовностей з елементами невизначеності на основі нейромережевого спектрального аналізу / Андрієцький Богдан Романович.

Додатки

Код програми, що містить всі функції та графіки до курсової роботи.

```
#imports
import matplotlib.pyplot as plt
import numpy as np

#imports model 2.1
def logistic_map(x0, lambd, n):
    x = [x0]
    for i in range(n-1):
        x.append(lambd * x[-1] * (1 - x[-1]))
    return x

x0 = 0.2 # початкове значення
lambd = 3.4494 # параметр  $\lambda$ 
n = 100 # кількість членів послідовності

x = logistic_map(x0, lambd, n)
plt.plot(x)
plt.title('')
plt.xlabel('кількість членів послідовності', fontsize=10)
plt.ylabel(r'значення x', fontsize=10)
plt.show()

#Bifurcation Diagram for model 2.1.
# параметри моделі
n = 100
lambd_values = []
x_values = []

# діапазон значень параметра  $\lambda$ 
lambd_min = 2.5
lambd_max = 4

# крок зміни параметра  $\lambda$ 
step = 0.001

# початкове значення параметра  $\lambda$ 
lambd = lambd_min
```

```

# цикл по всіх значеннях параметра  $\lambda$ 
while lambd < lambd_max:
    # початкове значення x
    x = 0.5

    # генеруємо послідовність з n елементів
    for i in range(n):
        x = lambd * x * (1 - x)
        if i >= n/2:
            # зберігаємо значення для відображення
            lambd_values.append(lambd)
            x_values.append(x)

    # змінюємо параметр  $\lambda$ 
    lambd += step

# візуалізація діаграми біфуркацій
plt.scatter(lambd_values, x_values, s=0.01, marker='.', color='black')
plt.xlabel('значення  $\lambda$ ')
plt.ylabel('значення x')
plt.show()

#model 2.2.
def bernoulli(x0, n):
    xn = [x0]
    for i in range(n):
        xn.append(2*xn[-1] % 1)
    return xn

# задаємо значення x0 та кількість членів послідовності
x0 = 0.9999
n = 100

# рахуємо послідовність та робимо графік
seq = bernoulli(x0, n)
plt.plot(seq, 'r')
plt.xlabel('кількість членів послідовності', fontsize=10)
plt.ylabel(r'значення x', fontsize=10)

```

```

plt.show()

#model 2.3.
def dynamic_map(x0, n):
    xn = x0
    x = [x0]
    for i in range(n):
        xn = 1 - abs(1 - 2*xn)
        x.append(xn)
    return x

# задаємо значення x0 та кількість членів послідовності
x0 = 0.00000001
n = 100

# рахуємо послідовність та робимо графік
x = dynamic_map(x0, n)
plt.plot(x, 'g')
plt.xlabel('кількість членів послідовності', fontsize=10)
plt.ylabel(r'значення x', fontsize=10)
plt.show()

#model 2.4.
def f(x):
    return 1 - 2*np.sqrt(np.abs(x))

def logistic_map(x0, n):
    x = np.zeros(n)
    x[0] = x0
    for i in range(1, n):
        x[i] = f(x[i-1])
    return x

# задаємо значення x0 та кількість членів послідовності
x0 = -0.9
n = 1000

# рахуємо послідовність та робимо графік
x = logistic_map(x0, n)
plt.plot(x, 'y')

```

```
plt.xlabel('кількість членів послідовності', fontsize=10)
plt.ylabel(r'значення  $x$ ', fontsize=10)
plt.show()
```