ментів, що дає змогу зменшити колізії, але збільшує споживання пам'яті. Дерева працюють ефективніше з погляду використання пам'яті, оскільки щоразу збільшуються лише на один вузол без необхідності розширення масиву.

У практичних застосуваннях хеш-таблиці найчастіше використовуються у мовах програмування як реалізація словників і множин (Python dict, Java HashMap, C++ unordered_map). Вони ефективні для кешування, пошуку у таблицях символів та обробки запитів з унікальними ключами, де впорядкованість не потрібна [1]. Збалансовані дерева, навпаки, широко застосовуються у базах даних, файлових системах та індексах, де важливо підтримувати впорядковані ключі. Прикладом є B-дерева і B+-дерева у СУБД, як-от MySQL і PostgreSQL [4]. Також дерева використовують у системах, що обробляють діапазонні запити, нечіткі пошуки, лексикографічну навігацію або завдання зі збереженням порядку.

Підсумовуючи, можна зазначити: вибір між хеш-таблицею та збалансованим деревом пошуку залежить від характеру задачі. Якщо необхідна висока швидкість доступу за ключем і немає вимог до впорядкування, оптимальним вибором є хеш-таблиця. Якщо ж важливо зберігати порядок елементів або гарантувати стабільну продуктивність незалежно від вхідних даних, треба обирати збалансоване дерево. Успішне застосування тієї чи іншої структури залежить від врахування особливостей предметної області, природи вхідних даних та пріоритетів ефективності.

Список використаних джерел

1. Sedgewick R., Wayne K. Algorithms. 4th ed. Boston: Addison-Wesley, 2011. 955 p.
2. Introduction to Algorithms / T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. 3rd ed. Cambridge: MIT Press, 2009. 1312 p.
3. Goodrich M., Tamassia R., Goldwasser M. Data Structures and Algorithms in Java. 6th ed. Wiley, 2014.
4. Silberschatz A., Korth H. F., Sudarshan S. Database System Concepts. 7th ed. New York: McGraw-Hill, 2020.

УДК 004.925.8

Родюк А. І., здобувачка 3 курсу спеціальності 111 Математика,
Комаров В. Ф., канд. техн. наук, старший викладач кафедри прикладної математики та кібербезпеки

ПОРІВНЯЛЬНИЙ АНАЛІЗ МЕТОДІВ ПОБУДОВИ ФРАКТАЛЬНИХ СТРУКТУР

Донецький національний університет імені Василя Стуса, м. Вінниця

Вступ. Фрактальна геометрія є ключовим інструментом моделювання нелінійних процесів. Актуальність теми підтверджується дослідженнями, де розглядаються фрактальні генеративні моделі [1] та оптимізація ітераційних процесів [2]. Проте більшість праць зосереджена на обчисленні розмірностей [3], залишаючи поза увагою комплексне порівняння алгоритмів побудови візуальних образів.

Мета роботи полягає у здійсненні порівняльного аналізу методів MRCM, «Хаотична гра», L-систем та клітинних автоматів для визначення специфіки їх застосування.

Постановка задачі – розглянути математичні основи методів, здійснити програмне моделювання фрактальних структур та порівняти результати за критеріями детермінованості, реалістичності й емерджентності.

Основний текст. Копіювальна машина з багаторазовим зменшенням (MRCM – Multiple Reduction Copy Machine). Метод базується на детермінованих ітераційних функціональних системах (IFS), що працюють як «копіювальна машина» зі зворотним зв'язком. Набір стискаючих відображень («лінз») визначає унікальний атрактор, до якого прагне система.

Для практичної ілюстрації методу MRCM було використано онлайн-інструмент IFS Fractal Generator [4]. На рис. 1 представлено фрактальну структуру, згенеровану шляхом ітерації системи з трьох афінних перетворень. Кожна «лінза» задає коефіцієнт стиснення та кут повороту, а отримане зображення демонструє, як складна самоподібна структура виникає з простих геометричних правил.

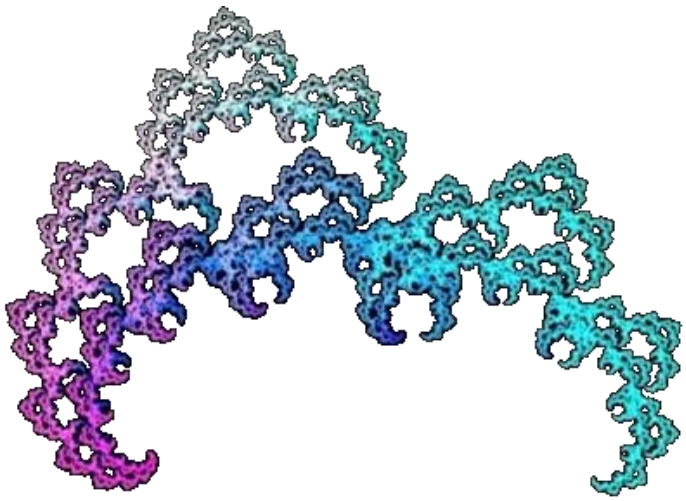


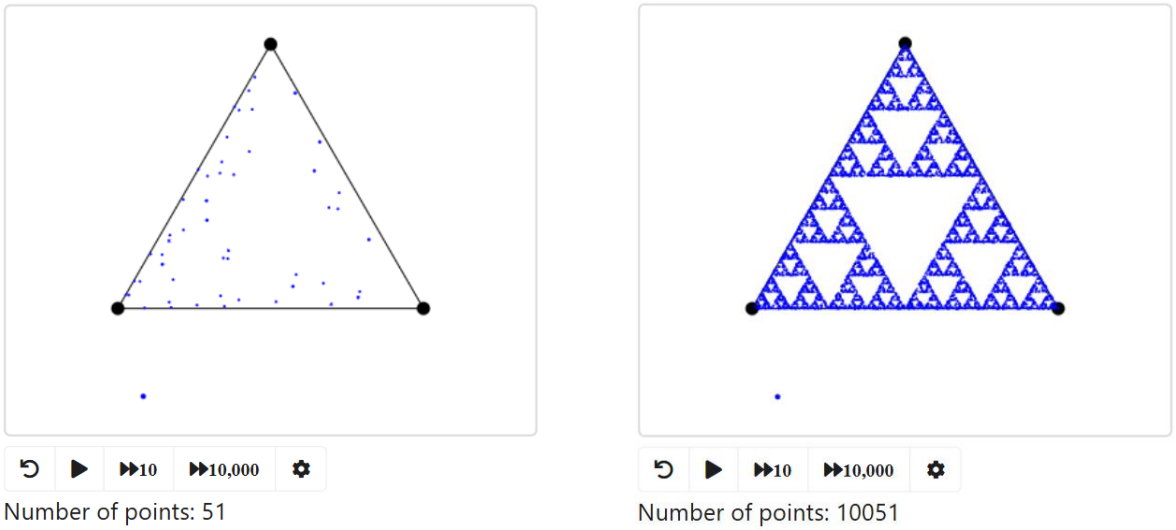
Рисунок 1 – Фрактальне дерево MRCM

Хаотична гра. Цей метод реалізує ймовірнісний підхід до побудови фракталів. Алгоритм генерує послідовність точок, де кожна наступна точка z_{k+1} є серединою відрізка між попередньою точкою z_k та випадково обраним маркером із фіксованого набору вершин. На відміну від детермінованого MRCM, тут зображення формується стохастично.

Моделювання процесу здійснено на платформі Chaos Game Simulation. На рис. 2 наведено порівняння двох етапів роботи алгоритму. Перша частина зображення демонструє початковий стан (51 ітерація), де окремі точки ілюструють випадковий характер вибору маркерів (див. рис. 2а). Друга частина показує результат після 10 000 ітерацій, де множина точок щільно заповнює простір, утворюючи чітку структуру трикутника Серпінського (див. рис. 2б). Це доводить, що хаотичний процес на мікрорівні переходить у впорядковану форму на макрорівні.

L-системи (Системи Ліденмайєра). L-системи – це машини переписування рядків, де правила продукції застосовуються паралельно до всіх символів.

Ця властивість відображає їх біологічне походження та дає змогу моделювати процеси росту. Геометрична інтерпретація здійснюється через «черепашачу графіку», що перетворює текстовий рядок у візуальний образ.



а) б)

Рисунок 2 – Візуалізація методу «Хаотична гра»: а) початковий хаотичний розподіл точок (51 ітерація); б) сформований атрактор (трикутник Серпінського) після 10 000 ітерацій

Для демонстрації біологічної природи L-систем в середовищі L-Systems Turtle Graphics було згенеровано фрактальну рослину (див. рис. 3). Використано аксіому X та правило перезапису $F \to FF$, яке імітує ріст гілок під кутом $25°$. Отримане зображення показує, як послідовна інтерпретація символьного рядка дає змогу моделювати природні об'єкти з високим рівнем самоподібності.

Клітинні автомати (Cellular Automata). Клітинні автомати – це дискретні динамічні системи, що еволюціонують у часі. Стан кожної клітинки на кроці $t + 1$ визначається станом її сусідів на кроці t за фіксованим алгоритмом. Це дає змогу спостерігати, як локальні взаємодії призводять до глобальних патернів.

Візуалізацію еволюції одновимірного автомата проведено за допомогою алгоритму Rule 30 в WolframAlpha. На рис. 4 представлено 100 кроків еволюції, де стан клітинок визначається логічною операцією над сусідами. Зображення ілюструє феномен емерджентності: з простих початкових умов виникає складна, частково хаотична структура, непередбачувана без покрокової симуляції.

Порівняння розглянутих алгоритмів демонструє, що попри візуальну схожість результатів (властивість самоподібності), механізми їх генерації кардинально відрізняються природою обчислень. Якщо метод MRCM оперує глобальними геометричними трансформаціями площини, то L-системи та клітинні автомати моделюють локальний динамічний ріст (через граматичні правила або еволюцію станів сусідів). Окреме місце посідає «Хаотична гра», яка, на відміну від детермінованих підходів MRCM та L-систем, доводить можливість виникнення суворого геометричного порядку зі стохастичного хаосу. Причому клітинні автомати (зокрема Rule 30) демонструють найвищий рівень емерджентності, де складність структури не закладена в початкових умовах, а виникає спонтанно, що робить їх унікальним інструментом для моделювання непередбачуваних природних процесів.

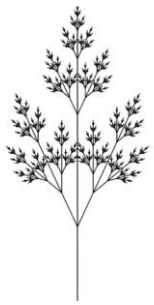


Рисунок 3 – Рослина / Дерево L-системи

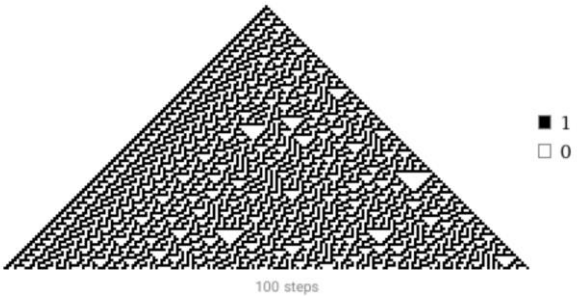


Рисунок 4 – Піраміда Rule 30

Висновки. Проведене порівняння ілюструє, що фрактальні структури можуть формуватися на основі різних математичних принципів: геометричних, ймовірнісних чи алгоритмічних. Результати моделювання свідчать про доцільність застосування методу MRCM у задачах, що вимагають точного відтворення форм, тоді як L-системи демонструють значний потенціал для імітації процесів біологічного росту. Перспективним напрямом подальших досліджень може бути розгляд гібридних підходів, зокрема для можливої процедурної генерації тривимірних (3D) об’єктів.

Список використаних джерел

1. Fractal Generative Models / T. Li et al. New York: arXiv, 2025. 13 p. (Препринт. Cornell University; 2502.17437 (cs)). URL: https://arxiv.org/abs/2502.17437 (дата звернення: 25.11.2025).

2. Fractal generation and analysis using modified fixed-point iteration / A. Tassaddiq et al. *AIMS Mathematics.* 2025. Vol. 10, № 4. P. 9462–9492.

3. A Comparative Study of Fractal Models Applied to Artificial and Natural Data / G. Silva et al. *Fractal and Fractional.* 2025. Vol. 9, № 2. P. 87. DOI: https://doi.org/10.3390/fractalfract9020087 (дата звернення: 25.11.2025).

4. SirXemic. IFS Fractal Generator: інтерактивний генератор фракталів. 2025. URL: https://github.com/SirXemic/jquery-fractals (дата звернення: 03.12.2025).

УДК: 004.738.5

Костюшин М. В., здобувач 2 курсу
ОС «Магістр» спеціальності
F3 Комп’ютерні науки,
Ніколюк П. К., д-р фіз.-мат. наук, професор
кафедри інформаційних технологій

ПОРІВНЯННЯ ПІДХОДІВ ДО ЗЛИТТЯ ДАНИХ У СИСТЕМАХ МУЛЬТИМОДАЛЬНОГО РОЗПІЗНАВАННЯ ОБ’ЄКТІВ

Донецький національний університет імені Василя Стуса, м. Вінниця

Вступ. Сучасні системи штучного інтелекту дедалі частіше працюють не з одним видом даних, а з кількома. Найпоширенішими є зображення, аудіо, текс-