

**ДОСЛІДЖЕННЯ МЕТОДІВ ВИКОРИСТАННЯ ІГРОВИХ РУШІЇВ
НА МОВІ ПРОГРАМУВАННЯ JAVASCRIPT.
АЛГОРИТМИ ПРОЦЕДУРНОЇ ГЕНЕРАЦІЇ**

Лабунець В.В.

e-mail: vladyslav.labunets@nure.ua

Науковий керівник – доц. Назаров О.С.

Харківський національний університет радіоелектроніки, каф. ПП
м. Харків, Україна

This work is devoted to study of procedural generation algorithms for 2D game levels implemented using JavaScript-based game engines. The research focuses on structured and stochastic approaches, including Binary Space Partitioning, Cellular Automata, Random Walk, and noise-based methods. A system of quantitative metrics for evaluating generated levels is proposed, including walkable ratio, connectivity of components. The results demonstrate that structured methods provide better connectivity and control, while stochastic methods generate more natural environments but require post-processing. A hybrid approach combining BSP and Cellular Automata is proposed, which achieves a balance between structural integrity and natural appearance.

Процедурна генерація є поширеним підходом створення ігрового контенту, яка дозволяє автоматизувати процес розробки рівнів, підвищити їх різноманітність і забезпечити повторну іграбельність. Також у сучасних умовах актуальність набуває використання процедурної генерації у 2D-іграх, які створюються за допомогою мови програмування JavaScript і ігрових рушіїв, які використовують цю мову [1].

Метою роботи є дослідження алгоритмів процедурної генерації рівнів у 2D-іграх, їх порівняльний аналіз та розробка власного підходу до генерації з урахуванням структурних й ігрових вимог.

У роботі було реалізовано і досліджено декілька алгоритмів генерації, а саме:

- Binary Space Partitioning;
- Cellular Automata;
- Random Walk;
- Perlin Noise;
- гібридний алгоритм на основі BSP та СА.

Алгоритм BSP забезпечує генерацію структурованих рівнів шляхом рекурсивного поділу простору і формування кімнат і коридорів [1]. Cellular Automata дозволяє отримати більш природні форми, які є подібними до печерних структур [2]. Алгоритм Random Walk генерує рівні шляхом випадкового обходу простору, формуючи зв'язні, але менш контрольовані структури. Perlin Noise дозволяє будувати шумові карти, які після обробки перетворюються у ігровий простір [3]. Такий підхід дозволяє формувати

плавні області природної форми, але без додаткової постобробки не завжди забезпечує достатню структурованість і цілісність рівня.

Для оцінювання якості згенерованих рівнів було запропоновано систему метрик, яка дозволяє кількісно оцінити їх іграбельність та структурні властивості, що узгоджується з сучасними підходами до оцінювання програмних систем [4].

Коефіцієнт прохідності визначається як:

$$W = \frac{N_{floor}}{N_{total}}$$

де N_{floor} – кількість прохідних клітин, N_{total} – загальна кількість клітин рівня. Ця метрика характеризує щільність рівня. Низькі значення призводять до обмеженої прохідності, а високі до втрати структури рівня.

Кількість компонент зв'язності визначає число ізольованих прохідних областей. Найкращим значенням є 1. Це означає повну зв'язність рівня та можливість доступу до всіх його частин.

Відносний розмір найбільшої компоненти визначається як:

$$L = \frac{N_{largest}}{N_{floor}}$$

де $N_{largest}$ – кількість клітин у найбільшій зв'язній області. Значення, близьке до 1, свідчить про цілісність рівня навіть у разі наявності деяких ізольованих частин.

Для оцінювання геометричності та штучності рівня було використано метрику прямолінійності меж:

$$S = \frac{L_{straight}}{N_{contour}}$$

де $L_{straight}$ – сумарна довжина довгих прямих сегментів стін, $N_{contour}$ – кількість контурних стін, які безпосередньо межують із прохідними клітинами. Високі значення цієї метрики характерні для структурованих алгоритмів, а низькі для більш природних форм.

Для покращення якості генерації було запропоновано гібридний алгоритм, що поєднує переваги BSP та Cellular Automata. На першому етапі використовується BSP для формування загальної структури рівня та забезпечення його зв'язності. На другому етапі до внутрішніх областей застосовується клітинний автомат, який модифікує форму кімнат, надаючи їм більш природного вигляду. Такий підхід дозволяє зберегти керованість генерації та водночас підвищити візуальну різноманітність рівнів.

Експериментальне дослідження проводилося шляхом генерації серії рівнів одного розміру для кожного алгоритму з подальшим обчисленням зазначених метрик. Узагальнені результати наведено в таблиці 1.

Отримані результати показують, що алгоритм BSP забезпечує найвищу структурованість рівня, але генерує геометрично прості форми. Cellular Automata і Perlin Noise формують більш природні структури,

але призводять до появи ізольованих областей. Random Walk забезпечує зв'язність, але створює менш передбачувані та вузькі структури.

Таблиця 1 – Порівняння алгоритмів

Алгоритм	W	Components	L	S
Binary Space Partitioning	0.31	1	1	0.95
Cellular Automata	0.48	25	0.81	0.06
Perlin Noise	0.49	16	0.63	0.17
Random Walk	0.35	1	1	0.05
Гібридний алгоритм	0.41	1	1	0.22

Запропонований гібридний алгоритм дозволяє досягти компромісу між структурованістю та природністю (див. рисунок 1).

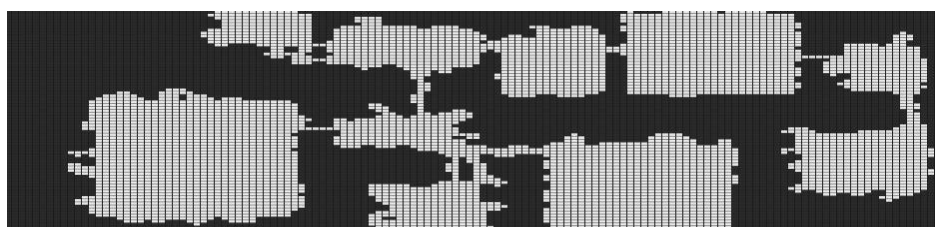


Рисунок 1 – Схематичне зображення згенерованого рівня

Рівень залишається зв'язним, але має менш штучну форму й кращу візуальну різноманітність.

Список використаних джерел:

1. Shaker N., Togelius J., Nelson M. Procedural Content Generation in Games. Cham: Springer, 2016. 237 p.
2. Johnson L., Yannakakis G. N., Togelius J. Cellular automata for real-time generation of infinite cave levels // Proc. FDG. 2010.
3. Perlin K. Improving Noise // ACM Transactions on Graphics. – 2015. – Vol. 21, No. 3. – P. 681–682.
4. Gruzdo I., Kyrychenko I., Tereshchenko G., Shanidze N. Metrics applicable for evaluating software at the design stage // CEUR Workshop Proceedings. – 2021. – P. 916–936.