

УДК 004.514

АЛГОРИТМ ВИПАДКОВОЇ ГЕНЕРАЦІЇ РІВНІВ ТА ЇХ ЧАСТИН В UNREAL ENGINE 5 ДЛЯ ROGUELIKE ІГОР

Жиліна К.І., Новіков Ю.С.

e-mail: kseniia.zhylina@nure.ua, yuriy.novikov@nure.ua

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

This study proposes a procedural level generation algorithm for Roguelike games using Unreal Engine 5, leveraging Random Stream for pseudo-randomness with seed-based reproducibility. The algorithm employs a modular room system, enabling diverse, adaptable levels with contextual placement of loot and enemies. Compared to Perlin Noise, BSP, and simple randomization, it balances simplicity and flexibility. Key advantages include ease of integration, customization, and potential for optimization (e.g., collision caching, noise integration). Results demonstrate effective, replayable level design, with future enhancements targeting performance and topological variety.

Випадково згенеровані рівні є ключовою особливістю ігор жанру Roguelike, що забезпечує високу реіграбельність і залученість гравців. Процедурна генерація таких рівнів вимагає поєднання простоти реалізації, різноманітності конфігурацій і контролю над результатом. Метою дослідження є створення алгоритму, який генерує унікальні ігрові простори на основі модульних кімнат із заданими характеристиками, оптимізованого для Unreal Engine 5.

Існуючі методи процедурної генерації мають різні сильні та слабкі сторони. Шумові функції, як-от Perlin Noise, створюють природні ландшафти з плавними переходами, але їхня обчислювальна складність і слабка адаптація до дискретних структур, таких як кімнати, ускладнюють використання в Roguelike. Binary Space Partitioning (BSP) ділить простір на неперетинні зони, що ідеально для зв'язаних приміщень, але потребує складної логіки деталізації. Проста псевдовипадкова генерація без фіксованого сіда є швидкою й легкою, проте не відтворюваною, що суперечить потребам жанру в керованій унікальності. Запропонований підхід використовує псевдовипадкові потоки (Random Stream) [1] в Unreal Engine 5, поєднуючи простоту з можливістю фіксації сіда для повторення рівнів.

У Unreal Engine 5 псевдовипадкова генерація через Random Stream базується на детермінованому алгоритмі, який імітує випадковість у заданому діапазоні чи розподілі. Користувач може ініціалізувати потік із певним сідом – числовим значенням, що визначає послідовність чисел. Наприклад, той самий сід завжди генерує ідентичну конфігурацію рівня, а зміна сіда створює нову.

Розроблений алгоритм призначений для створення модульних рівнів у Unreal Engine 5. Він використовує Random Stream для визначення типу й

позиції кімнат, які формуються на основі батьківського класу з базовими компонентами (двері, стіни, підлога, точки спавну, обмежувальні коробки) та дочірніх класів із варіаціями розмірів і вмісту. Для підвищення контролю над генерацією в майбутньому можна додати підхід із використанням ймовірностей і лічильників [2], де Random Stream визначає базову випадковість, а ймовірності та лічильники регулюють частоту появи певних типів кімнат, наприклад, відпочинку після бойових зон. Щоб уникнути перетинів, алгоритм застосовує перевірку колізій через обмежувальні коробки, замінюючи проблемні елементи, а вільні проходи закриває стінами. Декоративні об'єкти (ящики, стовпи) і лут із монстрами розміщуються залежно від псевдовипадкових значень, що забезпечує різноманітність і цілісність ігрового простору. Ефективність цього підходу підтверджується прикладами згенерованих рівнів із 16 (рис. 1) та 51 кімнатою (рис. 2), які демонструють адаптивність алгоритму до різної складності структур із трьома типами кімнат.

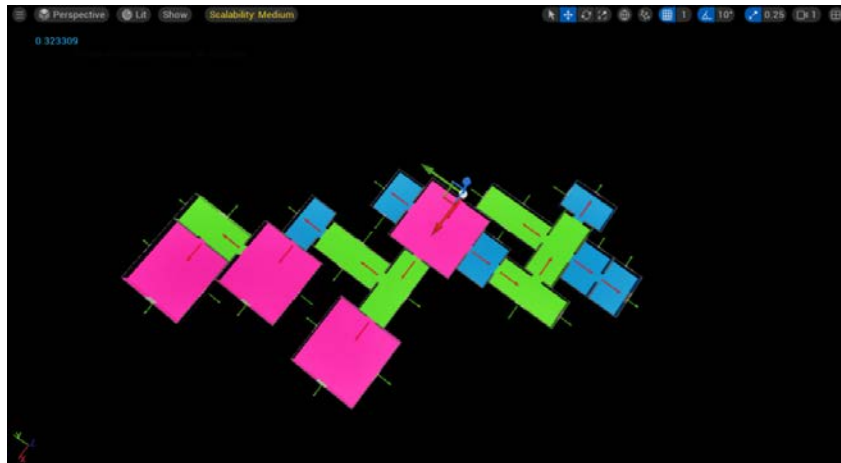


Рисунок 1 – Приклад рівня з 16 кімнатами, згенерованого за 0,3233 с



Рисунок 2 – Приклад рівня з 51 кімнатою, згенерованого за 0,9 с

Алгоритм вирізняється простотою інтеграції в Unreal Engine 5, гнучкістю налаштувань і адаптивністю до ігрових механік. Він перевершує

BSP за легкістю реалізації, але поступається в оптимізації простору й топологічній різноманітності. Перевагою є відтворюваність завдяки сіду, що полегшує тестування. Продуктивність алгоритму підтверджується тестуванням (табл. 1): рівень із 16 кімнатами генерується за 0,3233 секунди, а з 51 кімнатою — за 0,9 секунди, що відповідає середньому часу обробки однієї кімнати 17–20 мс і забезпечує швидке створення реіграбельних рівнів для Roguelike-ігор.

Таблиця 1 – Результати генерації рівнів із різною кількістю кімнат

Кількість кімнат	Типи кімнат	Час генерації (с)	Час на кімнату (мс)
16	3	0,3233	20,2
51	3	0,9	17,6

Перспективи вдосконалення включають кешування колізій для прискорення, інтеграцію шумових функцій для плавності переходів і ієрархічну генерацію кластерів кімнат. Результати показують, що алгоритм ефективно створює різноманітні, відтворювані рівні, відповідаючи потребам Roguelike-ігор. Його простота робить його доступним для початківців, а потенціал оптимізації — привабливим для масштабних проєктів. Подальші дослідження можуть зосередитися на підвищенні продуктивності, розширенні топологічних можливостей і додаванні динамічних елементів для збагачення ігрового досвіду. Крім цього, перспективним напрямком є інтеграція алгоритмів машинного навчання для адаптивної генерації рівнів на основі поведінки гравця. Наприклад, аналіз стилю гри може впливати на частоту появи певних типів кімнат або ворогів, що додасть індивідуалізації ігрового досвіду. Також можливе впровадження системи динамічних подій, де згенеровані рівні реагують на дії гравця в реальному часі, наприклад, створюючи пастки чи нові маршрути залежно від прогресу. Такі вдосконалення потребуватимуть ретельного балансування між продуктивністю та складністю реалізації, але вони здатні значно розширити межі процедурної генерації в Roguelike-іграх.

Список використаних джерел:

1. Random Streams in Unreal Engine. Epic Games Documentation. URL: <https://dev.epicgames.com/documentation/en-us/unreal-engine/random-streams-in-unreal-engine> (дата звернення: 24.02.2025).

2. Kondratiev M. A., Nazarov O. S. Генерація карти рівнів у грі з елементами жанру roguelike. Сучасні комп'ютерні та інформаційні системи і технології: матеріали II Всеукраїнської наук.-практ. інтернет-конф. (01-12 грудня 2021 р.). Мелітополь, 2021. С. 93-94.