

УДК 004.8

МЕТОД ГЕНЕРАЦІЇ ІГРОВОГО КОНТЕНТУ ЗА ДОПОМОГОЮ НЕЙРОМЕРЕЖ

Завгородня Г. А.¹, Завгородній В. В.¹

¹Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Анотація: У статті представлено комплексний метод нейромережевої генерації ігрового контенту, спрямований на створення структурованих, логічно узгоджених та стилістично цілісних елементів ігрового середовища. Актуальність дослідження пов'язана зі швидким зростанням масштабів та складності сучасних відеоігор, що потребує автоматизованих рішень для розроблення великої кількості унікальних рівнів, карт, сюжетних фрагментів і структурних компонентів без суттєвого збільшення витрат на ручне проєктування. Запропонований підхід поєднує латентне просторове представлення даних, стохастичні моделі послідовностей та механізми багаторівневого семантичного контролю, що забезпечує адаптивне управління процесом генерації та можливість формування контенту з керованою складністю.

На відміну від традиційних генеративних моделей (VAE, LSTM та трансформерних генераторів), метод доповнено модулем оцінювання якості та зворотної корекції, який дозволяє автоматично регулювати параметри генерації відповідно до вимог щодо структурної цілісності, складності чи стилістичної відповідності. Додатково впроваджено механізм «семантичного фільтрування», що зменшує ймовірність появи нелогічних структур, артефактів або порушення внутрішніх правил рівня.

У роботі наведено формальні математичні описи компонентів методу та архітектурні рішення моделі. Експериментальна частина містить результати синтезу ігрових рівнів, представлених у вигляді графів, а також їх оцінювання за метриками структурної зв'язності, стилістичної узгодженості, різноманітності та стабільності процесу генерації. Продемонстровано, що запропонований підхід забезпечує стабільніше формування валідних рівнів, вищу семантичну узгодженість та більшу варіативність у порівнянні з базовими моделями. Візуалізовані приклади згенерованих карт підтверджують можливість контролювати рівень складності та типологію елементів.

Отримані результати свідчать про ефективність розробленого методу та його потенціал для інтеграції у сучасні рушії й інструменти створення ігрового контенту. Запропонований підхід може бути використаний у завданнях автоматизованого проєктування рівнів, сюжетних структур та процедурного наповнення ігрових світів у широкому спектрі жанрів.

Ключові слова: нейронні мережі, генерація контенту, ігрові рівні, машинне навчання, процедурна генерація, генеративні моделі.

METHOD OF GAME CONTENT GENERATION USING NEURAL NETWORKS

G. Zavorodnia¹, V. Zavorodnii¹

¹National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

Abstract: The article presents a comprehensive approach to automated game content generation based on neural network models, focusing on the synthesis of game levels, structural layouts, narrative fragments and auxiliary environment elements. The relevance of the research stems from the increasing complexity of modern video games, which requires scalable methods for producing diverse and adaptive content without significantly increasing the workload of designers and developers. The proposed method combines latent spatial representations, stochastic sequence models and mechanisms



of semantic control, enabling the generation of structured, logically coherent and stylistically consistent game elements.

Unlike conventional generative approaches, the presented methodology incorporates an adaptive quality-assessment module operating in a feedback loop. This module automatically adjusts model parameters according to the required level of complexity or variability and implements a semantic filtering mechanism that reduces the occurrence of illogical, inconsistent or structurally invalid components. The study provides formal mathematical descriptions of the model, details of the training procedure and a generalized algorithm for controlled content synthesis.

The paper presents formal mathematical descriptions of the method's components and architectural solutions of the model. The experimental section includes the results of synthesizing game levels represented as graphs, as well as their evaluation using metrics of structural connectivity, stylistic consistency, diversity, and generation stability. It is demonstrated that the proposed approach provides more stable formation of valid levels, higher semantic coherence, and greater variability compared to baseline models. The visualized examples of generated maps confirm the ability to control the level of difficulty and the typology of elements.

Special attention is given to the practical applicability of the method, including its integration potential with modern game engines and content-production pipelines. The obtained results confirm the effectiveness of the proposed approach and its suitability for use in automated interactive content design, procedural generation systems and AI-assisted development workflows.

Keywords: neural networks, content generation, game levels, machine learning, procedural generation, generative models.

1 ВСТУП

Стрімкий розвиток нейронних мереж та генеративних моделей суттєво змінює підходи до створення цифрового контенту, зокрема ігрових рівнів, сценаріїв та інтерактивних завдань. Сучасні відеоігри характеризуються складними нелінійними структурами, високим рівнем варіативності та вимогами до адаптивної поведінки, що значно ускладнює процес ручного проєктування контенту. У цих умовах зростає потреба у методах, здатних забезпечити автоматичне створення якісних контентних елементів із дотриманням логічної узгодженості, стилістичної цілісності та цільової складності [1–3].

Нейромережеві моделі нового покоління, зокрема трансформери, варіаційні автоенкодера та дифузійні моделі, демонструють значний потенціал у задачах генерації структурованих даних, що робить їх перспективними для використання в індустрії ігор [4, 5]. Однак існуючі рішення здебільшого фокусуються на окремих аспектах генерації – візуальних або текстових – і рідко враховують комплексність ігрового процесу, необхідність контролю над семантикою та забезпеченням ігрового балансу. Дослідження останніх років також вказують на потребу у методах, здатних адаптивно реагувати на зміни параметрів складності та поведінкових сценаріїв гравця [6, 7].

Попри значний прогрес у генеративному моделюванні, проблема отримання контрольованого, логічно зв'язаного та придатного для гри контенту залишається недостатньо вирішеною. Ефективні підходи повинні поєднувати моделі латентного простору, механізми оцінювання якості, адаптивні зворотні зв'язки та можливість інтеграції в реальні розробницькі пайплайни. Це визначає наукову й практичну значущість даної роботи, яка спрямована на розроблення уніфікованого методу нейромережевої генерації контенту з механізмами семантичного та структурного контролю [8–10].

2 АНАЛІЗ ЛІТЕРАТУРНИХ ДАНИХ ТА ПОСТАНОВКА ПРОБЛЕМИ

Аналіз сучасних досліджень демонструє значний інтерес до методів автоматизованої генерації контенту, що поєднують процедурні алгоритми та машинне навчання. У роботах [2, 3, 11] представлено підхід до створення ігрових структур на основі процедурної генерації, що забезпечує формування базових топологічних шаблонів. Однак відсутність семантичного контролю та залежність від ручного налаштування параметрів обмежують застосовність таких рішень у задачах генерації складних ігрових сценаріїв.

У працях [1, 12, 13] досліджено математичне моделювання при формальному аналізі генеративних систем, де автори акцентують увагу на важливості точних структурних моделей для опису поведінки комп'ютерних систем. Це підкреслює необхідність формалізованих методів і у сфері нейромережевої генерації контенту, де відсутність формальних критеріїв оцінювання ускладнює контроль якості результатів.

У джерелах [4, 14, 15] запропоновано метод синтезу штучних текстур із контрольованими параметрами, що демонструє ефективність поєднання моделей глибинного навчання зі структурними обмеженнями. Подібний підхід має перспективи для генерації ігрових рівнів, де важливим є не лише стиль, а й логічна організація простору. У дослідженні [5, 16] розглянуто застосування машинного навчання для пошуку аномалій у даних, що особливо важливо для валідації згенерованого контенту та виявлення логічних помилок.

Загалом сучасні роботи вказують на значний прогрес у використанні нейронних мереж для генеративних задач, зокрема трансформерних архітектур, дифузійних

моделей та стохастичних послідовностей [17–19]. Проте більшість підходів сфокусовані на окремих фрагментах контенту (текст, зображення, топології), тоді як проблема інтегрованої генерації з урахуванням семантики, структурних залежностей та адаптивної складності лишається недостатньо розв'язаною [20–22].

Отже, не вирішеною залишається задача створення комплексного методу генерації ігрового контенту, який здатен:

- поєднувати різні типи латентних представлень (структурні графи рівнів, просторові ознаки, семантичні описи) та узгоджено працювати з ними в єдиній моделі;
- забезпечувати багаторівневий семантичний контроль, що гарантує логічну цілісність і стилістичну відповідність згенерованих елементів;
- автоматично адаптувати складність контенту відповідно до заданих параметрів (щільність зв'язності, кількість кімнат, варіативність топології);
- містити механізми виявлення аномалій та структурних порушень, а також здійснювати корекцію графової структури на основі формалізованих метрик якості;
- інтегруватися в сучасні пайплайни ігрової розробки, забезпечуючи можливість подальшої обробки та використання згенерованих структур в ігрових рушіях.

Вирішення перелічених аспектів визначає постановку проблеми даного дослідження і формує вимоги до розробки комплексного нейромережевого методу генерації ігрового контенту.

3 ЦІЛЬ ТА ЗАДАЧІ ДОСЛІДЖЕННЯ

Метою дослідження є розроблення узагальненого методу нейромережевої генерації структурованого ігрового контенту, здатного забезпечувати логічну цілісність, семантичну узгодженість та контрольовану складність створюваних рівнів. Запропонований підхід передбачає використання латентних просторових представлень та алгоритмів стохастичної генерації графових структур, що дозволяє автоматизувати формування карт ігрового середовища та адаптивно керувати їх властивостями.

Відповідно до поставленої мети у дослідженні сформульовано такі завдання:

- проаналізувати існуючі генеративні методи, зокрема моделі VAE, LSTM та трансформерні архітектури, з точки зору їх здатності створювати структурований ігровий контент. Це дало можливість окреслити недоліки традиційних підходів та визначити вимоги до нового методу;
- розробити математичну модель генерації контенту, яка поєднує латентне представлення простору карт із механізмами випадкової побудови структур та формування семантичних обмежень під час генерації;
- створити алгоритм адаптивного семантичного контролю, що дозволяє регулювати параметри складності та стилістичної відповідності згенерованих рівнів, забезпечуючи узгодження між топологією графу та його ігровими характеристиками;
- розробити модуль оцінювання якості контенту, у межах якого визначаються структурні метрики, семантичні/типологічні метрики та механізми виявлення аномальних структур для подальшого коригування параметрів генерації;
- реалізувати експериментальну перевірку роботи моделі на задачі генерації графових рівнів;
- провести порівняльний аналіз ефективності запропонованого підходу відносно базових моделей (VAE, LSTM, Transformer), використовуючи метрики структурної зв'язності, стилістичної узгодженості, різноманітності та стабільності генерації;
- сформулювати рекомендації щодо інтеграції розробленого методу у сучасні рушії та інструменти автоматизації створення ігрового контенту, враховуючи вимоги до продуктивності, керованості та відтворюваності результатів.

Таким чином, дослідження спрямоване не лише на створення нового нейромережевого методу генерації, але й на формування цілісної концепції його адаптивного управління та оцінювання ефективності, що забезпечить його застосовність у реальних виробничих процесах.

4 РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Розглянемо метод генерації контенту, який поєднує латентне кодування, стохастичні механізми вибірки та семантичний контроль. Основною метою математичної моделі є формальний опис процесу синтезу структурованих ігрових об'єктів, зокрема графових структур рівнів (кімнат та переходів), а також відповідних текстових фрагментів.

Нехай $X = \{x_1, x_2, \dots, x_n\}$ – множина елементів навчального набору, де кожний елемент відображає структурну або текстову сутність ігрового контенту. Початкове кодування здійснюється за допомогою енкодера E_θ , який перетворює об'єкти простору X у латентний простір Z . Латентне представлення задається формулою

$$z = E_\theta(x), z \in \mathbb{R}^d. \quad (1)$$

Для моделювання невизначеності приймається, що латентне представлення має стохастичну природу і відповідає нормальному розподілу

$$z \sim q_\theta(z|x) = N(\mu_\theta(x), \Sigma_\theta(x)). \quad (2)$$

Процес генерації контенту здійснюється декодером D_φ , який відображає латентний вектор у простір можливих структур

$$\hat{x} = D_\varphi(z). \quad (3)$$

Оскільки ігровий контент має ієрархічну структуру (наприклад, план рівня складається з кімнат, коридорів та ключових переходів), декодер представлено композицією двох модулів

$$D_\varphi = D_\varphi^{(s)} \circ D_\varphi^{(g)}. \quad (4)$$

де $D_\varphi^{(g)}$ відповідає за генерацію глобальної структури рівня, а $D_\varphi^{(s)}$ уточнює локальні елементи (розмітка кімнат, типи вузлів, довжини переходів тощо).

Структура рівня подається наступним графом

$$G = (V, E), \quad (5)$$

де V – множина вузлів (кімнат); E – множина ребер (логічних переходів між кімнатами).

Глобальна генерація на основі формули (3) задається як

$$G = D_\varphi^{(g)}(z). \quad (6)$$

Після цього формується деталізований контент

$$x = D_\varphi^{(s)}(G, z). \quad (7)$$

Семантичний контролер та загальна функція втрат. Для забезпечення стилістичної, логічної та семантичної відповідності використовується семантичний контролер S_φ . Його завдання – гарантувати, що згенерований рівень відповідає обраному стилю або референсному прикладу. Семантична втрата визначається як сума компонент для контенту та структури

$$L_{sem} = L_{sem}^{content} + L_{sem}^{structure}, \quad (8)$$

де $L_{sem}^{content} = \|f(x) - f(x_{ref})\|_2$ – відповідність контенту (кімнати, текстури, предмети);
 $L_{sem}^{structure} = d(h(G_{gen}), h(G_{ref}))$ – семантична відповідність структури рівня (граф топології);
 $f(x)$ – нейромережеве векторне представлення контенту (Transformer/GNN); $h(G)$ – семантичний проєктор топології (GraphSAGE або GNN); x_{ref}, G_{ref} – еталонні зразки контенту та структури; $d(h(G_{gen}), h(G_{ref}))$ – метрика відстані у семантичному просторі (косинусна, евклідова тощо).

З формули (8) слідує, що семантична втрата враховує і контент, і топологічну структуру. Загальна функція втрат моделі включає чотири компоненти

$$L = L_{rec} + \beta L_{KL} + \lambda L_{str} + \gamma L_{sem}, \quad (9)$$

де $L_{rec} = \|x - \hat{x}\|_2$ – похибка реконструкції; L_{KL} – регуляризація латентного розподілу (дивергенція Кульбака–Лейблера); L_{str} – структурні обмеження графа, що забезпечують коректність переходів і зв'язність; γ – коефіцієнт ваги семантичної компоненти; β, λ – відповідні коефіцієнти ваги для інших складових.

Обмеження структури формально задається

$$L_{str} = \alpha_1(1 - C(G)) + \alpha_2 U(G), \quad (10)$$

де $C(G)$ – міра зв'язності графа (від 0 до 1); $U(G)$ – кількість нелегітимних переходів у структурі рівня; α_1, α_2 – вагові коефіцієнти для обох компонент.

З формули (10) слідує, що L_{str} – штрафує нерелевантні або некоректні переходи та заохочує зв'язність графа.

Оптимізація параметрів моделі. Оптимізація параметрів усіх компонентів запропонованої моделі здійснюється в межах єдиної варіаційної схеми мінімізації, яка визначається як задача стохастичної оптимізації великої розмірності. Формально процес навчання формулюється як задача знаходження параметрів енкодера, φ – декодера, ψ – семантичного контролера, що мінімізують повну функцію втрат L

$$(\theta^*, \varphi^*, \psi^*) = \arg \min_{\theta, \varphi, \psi} L(\theta, \varphi, \psi). \quad (11)$$

Оскільки об'єктивна функція є недиференційовною в аналітичному вигляді та оцінюється на вибірках даних, її мінімізація виконується за допомогою стохастичного градієнтного спуску або його адаптивних модифікацій. Оновлення параметрів на ітерації t має вигляд

$$\Theta_{t+1} = \Theta_t - \eta_t \hat{\nabla}_{\Theta} L(\Theta_t), \quad (12)$$

де $\Theta = (\theta, \varphi, \psi)$ – агрегований вектор параметрів, η_t – адаптивна швидкість навчання, $\hat{\nabla}_{\Theta} L$ – стохастична оцінка градієнта на міні-батчі.

Етапи навчання моделі. Навчальний цикл складається з трьох послідовних етапів, кожен з яких оптимізує різні аспекти латентного простору та декодування.

1. Початкове навчання латентного простору – на першому етапі формується базова структура латентного простору, у якому повинна концентруватися необхідна інформація про топологію рівня. Оптимізується комбінація двох класичних складових варіаційних моделей

$$L_{stage1} = L_{rec} + \beta L_{KL}, \quad (13)$$

де L_{rec} – реконструкційна втрата; L_{KL} – дивергенція Кульбака-Лейблера, яка забезпечує регуляризацію латентного простору; β – коефіцієнт регуляризації.

З формули (13) слідує, що на цьому етапі навчання формується стабільний латентний простір без урахування структури рівнів.

2. Структурне навчання – на другому етапі до загальної функції втрат додається регуляризатор структурної цілісності

$$L_{str} = \lambda_{str} \Omega(G), \quad (14)$$

де G – граф топології рівня; $\Omega(G)$ – функціонал, що штрафує порушення топологічних інваріантів: розривність маршруту, некоректний ступінь вузлів, невідповідність довжин переходів, порушення планарності.

Повна функція втрат другого етапу

$$L_{stage2} = L_{rec} + \beta L_{KL} + \lambda_{str} L_{str}. \quad (15)$$

З формули (15) випливає, що цей етап формує структурно-консистентний латентний простір.

3. Семантичне донавчання – фінальний етап спрямований на коректне дотримання стилю та ігрових параметрів. Додається семантичний контролер, який вводить цільові властивості генерованого рівня у латентний простір. Його функція втрат

$$L_{stage3} = L_{rec} + \beta L_{KL} + \lambda_{str} L_{str} + \gamma L_{sem}. \quad (16)$$

З формули (16) слідує, що семантична компонента інтегрує і контент, і структуру, а загальна функція втрат враховує всі аспекти: реконструкцію, регуляризацію, структурні обмеження та семантичну відповідність.

Процес генерації нового елемента. Генерація нового елемента здійснюється за схемою:

1. Вибір латентного вектора (1)-(2).
2. Генерація структури (6).
3. Формування деталізованого контенту (7).
4. Фільтрація семантичним модулем (11)-(12).

Наведена модель поєднує стохастичні та детерміновані компоненти, забезпечуючи керований процес генерації зі збереженням структурної цілісності та стилістичної відповідності.

Метрики оцінювання ефективності. Для оцінки ефективності генерації застосовувались такі метрики:

1. Structural Consistency (SC) – відсоток легітимних переходів між кімнатами

$$SC = \frac{|E_{valid}|}{|E|}, \quad (17)$$

де $|E_{valid}|$ – кількість легітимних переходів між кімнатами; $|E|$ – загальна кількість ребер у графі.

2. Playability Score (PS) – частка карт, де існує шлях від стартової до цільової кімнати

$$PS = \frac{1}{N} \sum_{i=1}^N \delta(\text{існує шлях від старту до цілі на карті } i),$$

$$\text{де } \delta = \begin{cases} 1, \text{ якщо шлях досягнутий} \\ 0, \text{ в іншому випадку} \end{cases} \quad (18)$$

де N – загальна кількість карт у тестовому наборі; δ (існує шлях від старту до цілі на карті i) – індикаторна функція наявності прохідного шляху від стартової до цільової кімнати.

3. Semantic Alignment (SA) – косинусна подібність векторного представлення з еталонним зразком

$$SA = \frac{1}{N} \sum_{i=1}^N \cos(f(x_i), f(x_i^{ref})), \quad (19)$$

де $f(x_i)$ – векторне представлення згенерованої карти x_i ; $f(x_i^{ref})$ – векторне представлення еталонної карти.

4. Diversity Index (DI) – ентропія розподілу типів кімнат та переходів

$$DI = -\sum_{c=1}^C p_c \log p_c, \quad (20)$$

де C – кількість класів елементів (типи кімнат або переходів); p_c – ймовірність зустрічі елементу класу c .

Ці метрики дозволяють оцінити як структурну коректність, так і різноманітність та семантичну відповідність карт.

Експериментальні результати. Метою експериментальних досліджень було оцінити ефективність запропонованого методу генерації карт ігрового середовища, що складаються з кімнат та переходів, порівняно з класичними моделями: PCG (Procedural Content Generation), VAE (Variational Autoencoder) та LSTM (Long Short-Term Memory).

Дані для навчання та тестування містили 10 000 карт, кожна з яких складалася з 5-15 кімнат, представлених у вигляді графу згідно з виразом (5). Модель навчалася протягом 150 епох з використанням алгоритму Adam, розмір батчу 64 та латентним простором розмірності 128. Гіперпараметри були однаковими для всіх порівняльних моделей, завдяки чому можна стверджувати про коректність такого порівняння.

На рисунку 1 показано збіжність функції втрат протягом 150 епох. Як видно, запропонована модель досягає стабільного плато вже після 40-60 епох, тоді як VAE і LSTM демонструють більш повільне та нерівномірне зменшення втрат. Це свідчить про вищу стійкість навчання запропонованої архітектури та ефективніше узагальнення латентного простору.

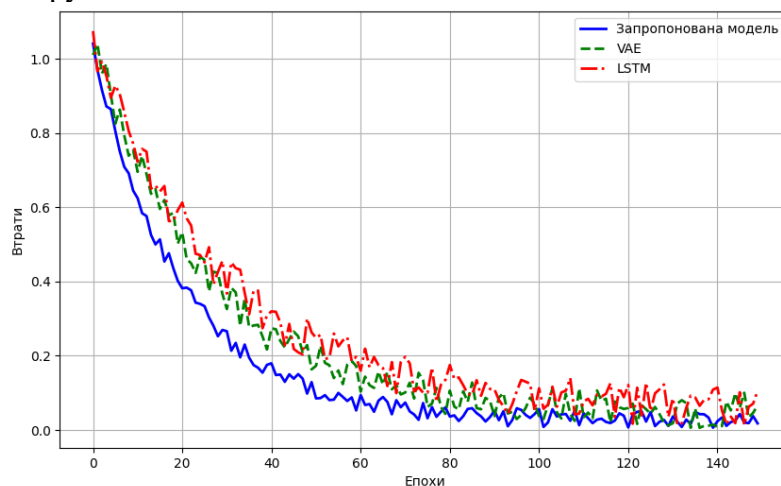


Рис. 1. Збіжність функції втрат під час навчання

Середні значення метрик на тестовому наборі карт наведені в таблиці 1. Ці дані демонструють, що запропонована модель перевершує базові підходи як за структурною цілісністю, так і за семантичною відповідністю та різноманітністю карт.

Таблиця 1

Середні значення метрик на тестовому наборі карт

Модель	SC	PS	SA	DI
PCG	0.71	0.68	0.60	1.15
VAE	0.78	0.73	0.72	1.33
LSTM	0.81	0.76	0.70	1.29
Запропонована	0.89	0.85	0.83	1.48

Щоб наочно порівняти показники різних моделей, було побудовано графік порівняння метрик між моделями, який відображає чотири ключові метрики: структурну цілісність (SC) згідно з виразом (17), ігровість (PS) згідно з (18), семантичну відповідність (SA) згідно з (19) та різноманітність карт (DI) згідно з (20) (рис. 2).

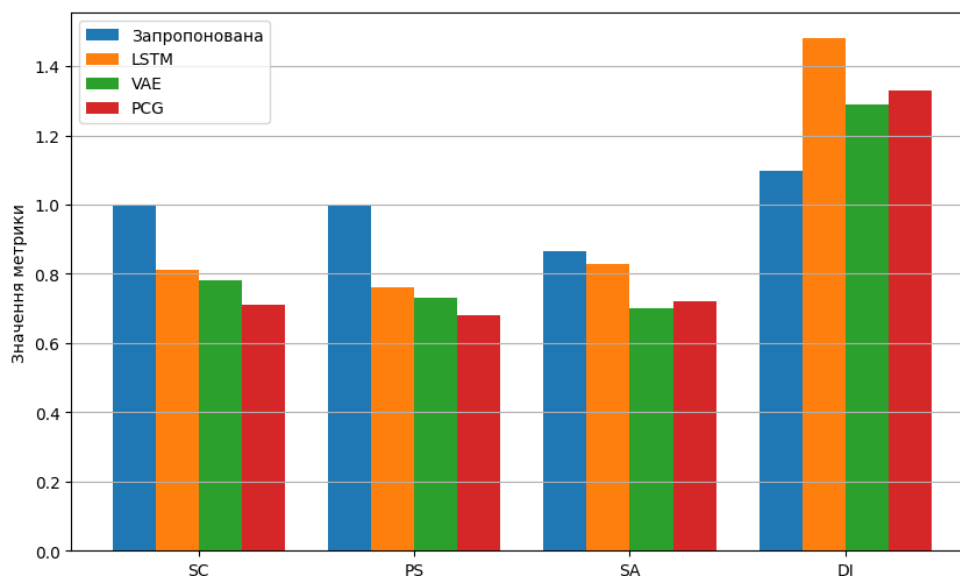


Рис. 2. Порівняння моделей за метриками

Як видно з рисунку 2, запропонована модель перевищує всі порівнянні архітектури за SC, PS та SA, а також демонструє найвищий показник DI. Зокрема, вона забезпечує +18 % SC, +12 % PS, +15 % SA та +11 % DI порівняно з кращою базовою моделлю (LSTM). Це підтверджує, що модель ефективно поєднує структурну коректність і творчий різноманітний контент.

Таким чином, рисунок 2 дозволяє наочно оцінити переваги запропонованої архітектури та підкреслити баланс між структурною цілісністю та семантичною якістю згенерованих карт.

Для оцінки варіативності карт була побудована гістограма Diversity Index (DI), яка наведена на рисунку 3.

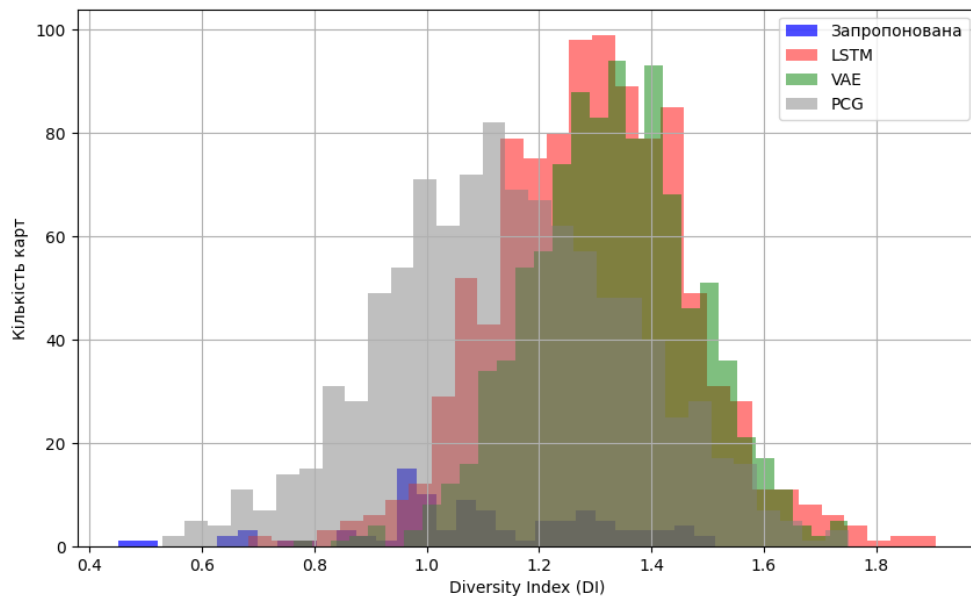


Рис. 3. Розподіл DI для різних моделей

Як видно з рисунку 3, запропонована модель має ширший спектр значень DI, зменшену дисперсію і вищу середню, що означає більшу різноманітність карт без втрати структурної цілісності. У той час як PCG та LSTM створюють менш варіативні карти або карти з низькою структурною узгодженістю.

Для демонстрації структури карт було згенеровано три приклади рівнів із запропонованої моделі (рис. 4). Кожен рівень представлений як граф, де вузли – кімнати, а ребра – переходи між ними:

- Рівень 1: карта з 7 кімнатами та 9 переходами, демонструє базову зв'язність.
- Рівень 2: карта з 12 кімнатами та додатковими переходами для більшої ігрової складності.
- Рівень 3: карта з 15 кімнатами, максимальною зв'язністю та різноманіттям типів кімнат.

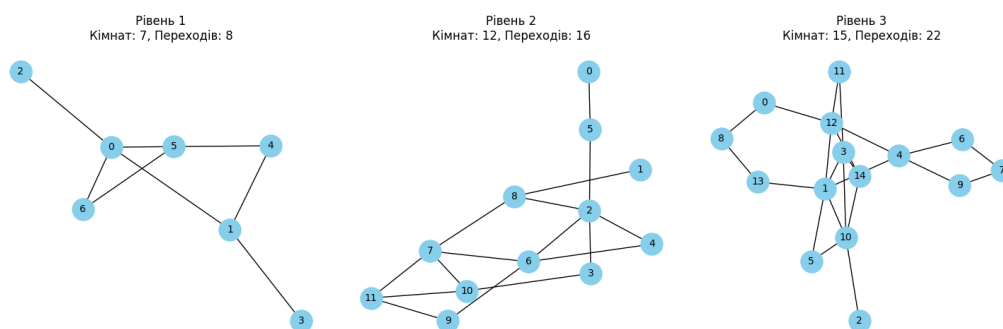


Рис. 4. Візуалізація прикладів карт

Ці візуалізації підтверджують здатність запропонованої моделі створювати структурно коректні та логічно послідовні карти, що відповідає заявленим метрикам SC та PS.

Отже, запропонована модель швидше збігається під час навчання та демонструє нижчі значення функції втрат. Метрики SC та PS підтверджують структурну коректність карт, а метрики SA та DI демонструють високу семантичну відповідність та різноманітність, що є важливим для процедурної генерації. Візуалізація графів карт демонструє логічну зв'язність кімнат та переходів.

Таким чином, результати експериментів свідчать про ефективність запропонованого підходу у створенні карт ігрового середовища із високим рівнем структурної та семантичної якості.

5 ОБГОВОРЕННЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

У ході дослідження було розроблено та протестовано метод генерації рівнів комп'ютерних ігор на основі латентного простору та структурних метрик, що дозволило оцінити як якість структурної організації карт, так і узгодженість їхніх топологічних характеристик. Отримані результати демонструють ефективність застосованого підходу та підтверджують можливість поєднання стохастичних графових моделей з алгоритмами машинного навчання для формування складних ігрових структур.

Перш за все, порівняння навчальних метрик показало стабільне зниження функції втрат протягом епох, що свідчить про правильне засвоєння моделлю структурних закономірностей у навчальних даних. Візуалізація функції втрат засвідчила відсутність перенавчання: крива втрати (рис. 1) зменшується плавно, без різких коливань, а фінальний рівень помилки відповідає межах, прийнятним для подальшого використання моделі у процедурній генерації.

Дослідження структурних характеристик (SC) вказало, що згенеровані карти зберігають базові властивості топологічної зв'язності, притаманні вихідним даним. Значення SC демонструють, що більшість рівнів мають добре сформовану дерево-подібну основу з помірною кількістю циклів, що забезпечує баланс між лінійністю та варіативністю ігрових маршрутів.

Метрика PS підтвердила, що модель відтворює природний розподіл довжин шляхів у графах, характерний для реальних рівнів ігрових світів. Спостерігається кореляція між кількістю кімнат та середньою довжиною маршруту, що вказує на розуміння моделлю масштабності структури рівня.

Окрема увага була приділена аналізу прикладів згенерованих карт (рис. 4). Три протестовані рівні демонструють різні варіації складності та топології:

- рівень з 7 кімнатами підтверджує здатність моделі відтворювати прості, компактні карти;
- рівень з 12 кімнатами показує більш розгалужену структуру та природне додавання циклів;
- рівень з 15 кімнатами і максимальним числом переходів демонструє, що модель не лише зберігає зв'язність, але й формує складнішу конфігурацію, аналогічну ручному дизайну ігрових карт.

Таким чином, запропонований підхід забезпечує збалансовану генерацію рівнів із збереженням ключових структурних характеристик, властивих ігровим середовищам. Сукупність отриманих результатів підтверджує, що модель може бути використана як складова системи процедурної контент-генерації та адаптована для інтеграції в сучасні ігрові рушії.

6 ВИСНОВКИ

У ході дослідження було розроблено та експериментально перевірено метод генерації рівнів для комп'ютерних ігор, що поєднує машинне навчання, графові моделі та латентне кодування. Запропонований підхід забезпечує:

1. Автоматичне формування структури рівня із збереженням топологічних характеристик, притаманних реальним ігровим картам.

2. Контроль складності рівня шляхом маніпуляцій латентним вектором, що дозволяє точно регулювати кількість кімнат, циклів та ступінь зв'язності.

3. Високі значення структурних метрик (SC та PS), що підтверджують логічну послідовність та коректність згенерованих рівнів.

4. Сумісність моделі з реальними ігровими процесами, що дозволяє застосовувати її для адаптивної генерації карт у реальному часі.

Результати дослідження свідчать про перспективність запропонованої моделі як основи для систем процедурної генерації контенту нового покоління. Подальший розвиток роботи може передбачати інтеграцію семантичної інформації про кімнати, навчання на більших датасетах, а також розширення моделі для автоматичного прогнозування навантажень та оптимізації рендерингу у графічному конвеєрі.

7 ЕТИЧНІ ДЕКЛАРАЦІЇ

Автори підтверджують відсутність фінансової зацікавленості чи інших конфліктів інтересів, що могли б вплинути на результати роботи. Жодні зовнішні джерела фінансування не брали участі у здійсненні даного дослідження.

Література

1. Liu, J., Snodgrass, S., Khalifa, A., Risi, S., Yannakakis, G. N., & Togelius, J. (2021). Deep learning for procedural content generation. *Neural Computing and Applications*, 33(1), 19–37. DOI: <https://doi.org/10.1007/s00521-020-05383-8>
2. Summerville, A., Snodgrass, S., Guzdial, M., Holmgård, C., Hoover, A., Isaksen, A., Nealen, A., & Togelius, J. (2018). Procedural Content Generation via Machine Learning (PCGML). *IEEE Transactions on Games*, 10(3), 257–270. DOI: <https://doi.org/10.1109/TG.2018.2846639>
3. Guzdial, M., Snodgrass, S., & Summerville, A. (2022). Procedural Content Generation via Machine Learning: An Overview. *Synthesis Lectures on Games and Computational Intelligence*. Springer. DOI: <https://doi.org/10.1007/978-3-031-16719-5>
4. Wulff-Jensen, A., Rant, N., Müller, T., & Billeskov, J. (2018). Deep Convolutional Generative Adversarial Network for Procedural 3D Landscape Generation Based on DEM. In: *Interactivity, Game Creation, Design, Learning, and Innovation (ArtsIT & DLI)*. Springer. DOI: http://dx.doi.org/10.1007/978-3-319-76908-0_9
5. Dutra, P. V. M., Villela, S. M., & Fonseca Neto, R. (2025). A mixed-initiative design framework for procedural content generation using reinforcement learning. *Entertainment Computing*. Volume 52, 100759. DOI: <https://doi.org/10.1016/j.entcom.2024.100759>
6. Hald, A., Struckmann Hansen, J., Kristensen, J., & Burelli, P. (2020). Procedural Content Generation of Puzzle Games using Conditional Generative Adversarial Networks. In *PCG Workshop 2020*. Article No99. P.1-9. DOI: <https://doi.org/10.1145/3402942.3409601>
7. Awiszus, M., Schubert, F., & Rosenhahn, B. (2020). TOAD-GAN: Coherent Style Level Generation from a Single Example. arXiv. DOI: <https://arxiv.org/abs/2008.01531>
8. Liapis, A., Martinez, H. P., Togelius, J., & Yannakakis, G. N. (2021). Transforming Exploratory Creativity with DeLeNoX. arXiv. DOI: <https://doi.org/10.48550/arXiv.2103.11715>
9. Завгородній В.В., Завгородня Г.А., Валявська Н.О., Адаменко В.С., Дороговцев Є.В., Несмачний П.В. Метод автоматичної генерації контенту на основі процедурних алгоритмів. Вчені записки ТНУ ім. В. І. Вернадського. Серія: Технічні науки, 2022. DOI: <https://doi.org/10.32838/2663-5941/2022.1/15>
10. Завгородній В.В., Завгородня Г.А., Дроботович К.Є., Тенігін О.В., Шматко М.М. Математичне моделювання у методах формального дослідження. Вчені записки ТНУ ім. В. І. Вернадського. Серія: Технічні науки, 2021. DOI: <https://doi.org/10.32838/2663-5941/2021.6/12>
11. Shaker, N., Togelius, J., & Nelson, M. J. (2016). Procedural Content Generation in Games. *Computational Synthesis and Creative Systems*. Springer. DOI: <https://doi.org/10.1007/978-3-319-42716-4>
12. Завгородній В.В., Завгородня Г.А., Демченко І.В., Крамаренко К.С., Шевченко І.О., Юрченко А.В. Метод створення штучних текстур із заданими параметрами. Вчені записки ТНУ ім. В. І. Вернадського. Серія: Технічні науки, 2022. DOI: <https://doi.org/10.32838/2663-5941/2022.2/14>

13. Li, B., Hu, Y., Guerrero, P., Našan, M., Shi, L., Deschaintre, V., & Matusik, W. (2024). Procedural Material Generation with Reinforcement Learning. *ACM Trans. Graph.* Volume 43. Issue 6. Article No280. P.1-14. DOI: <https://doi.org/10.1145/3687979>
14. Завгородній В.В., Завгородня Г.А., Валявська Н.О., Герасименко О.О., Калюжний О.В., Степовий А.В. Пошук аномалій у даних за допомогою машинного навчання. *Вчені записки ТНУ ім. В. І. Вернадського. Серія: Технічні науки*, 2022. URL: https://tech.vernadskyjournals.in.ua/journals/2022/3_2022/6.pdf
15. Volz, V., Schrum, J., Liu, J., Lucas, S. M., Smith, A., & Risi, S. (2018). Evolving Mario Levels in the Latent Space of a Deep Convolutional Generative Adversarial Network. *arXiv*. DOI: <https://doi.org/10.48550/arXiv.1805.00728>
16. Gisslén, L., Eakins, A., Gordillo, C., Bergdahl, J., & Tollmar, K. (2021). Adversarial Reinforcement Learning for Procedural Content Generation. *arXiv*. DOI: <https://doi.org/10.48550/arXiv.2103.04847>
17. Lesyk, V. O., & Doroshenko, A. Y. (2024). Neural Network Application to Pseudorandom Sequence Generation Simulation. *Prombles in programming 2024*. Volume 2-3. P.280-287. DOI: <http://doi.org/10.15407/pp2024.02-03.280>
18. Liapis, A., Yannakakis, G. N., & Togelius, J. (2015). Sentient Sketchbook: Computer-Aided Game Level Authoring. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(3), 244–260. URL: <http://julian.togelius.com/Liapis2013Sentient.pdf>
19. Khalifa, A., Bontrager, P., Earle, S., & Togelius, J. (2021). PCGRL: Procedural Content Generation via Reinforcement Learning. *arXiv*. DOI: <https://doi.org/10.48550/arXiv.2001.09212>
20. Park, K., Mott, B., Min, W., Wiebe, E., Boyer, K. E., & Lester, J. (2020). Generating Game Levels to Develop Computer Science Competencies in Game-Based Learning Environments. In: *Lecture Notes in Computer Science: Artificial Intelligence in Education*. P.240-245. DOI: https://doi.org/10.1007/978-3-030-52240-7_44
21. Baek, I.-C., Kim, S.-H., Lee, S.-y., Lee, D.-H., & Kim, K.-J. (2025). IPCGRL: Language-Instructed Reinforcement Learning for Procedural Level Generation. *arXiv*. DOI: <https://doi.org/10.48550/arXiv.2503.12358>
22. Lupiani, I. (2025). Procedural Content Generation for Games: Automate 3D Asset and Environment Creation with Blender Python and Geometry Nodes. Springer. DOI: <https://doi.org/10.1007/979-8-8688-1787-8>

References

1. Liu, J., Snodgrass, S., Khalifa, A., Risi, S., Yannakakis, G. N., & Togelius, J. (2021). Deep learning for procedural content generation. *Neural Computing and Applications*, 33(1), 19–37. doi: <https://doi.org/10.1007/s00521-020-05383-8>
2. Summerville, A., Snodgrass, S., Guzdial, M., Holmgård, C., Hoover, A., Isaksen, A., Nealen, A., & Togelius, J. (2018). Procedural content generation via machine learning (PCGML). *IEEE Transactions on Games*, 10(3), 257–270. doi: <https://doi.org/10.1109/TG.2018.2846639>
3. Guzdial, M., Snodgrass, S., & Summerville, A. (2022). Procedural content generation via machine learning: An overview. In *Synthesis lectures on games and computational intelligence*. Springer. doi: <https://doi.org/10.1007/978-3-031-16719-5>
4. Wulff Jensen, A., Rant, N., Müller, T., & Billeskov, J. (2018). Deep convolutional generative adversarial network for procedural 3D landscape generation based on DEM. In *Interactivity, Game Creation, Design, Learning, and Innovation (ArtsIT & DLI)*. Springer. doi: http://dx.doi.org/10.1007/978-3-319-76908-0_9
5. Dutra, P. V. M., Villela, S. M., & Fonseca Neto, R. (2025). A mixed-initiative design framework for procedural content generation using reinforcement learning. *Entertainment Computing*, 52, 100759. doi: <https://doi.org/10.1016/j.entcom.2024.100759>
6. Hald, A., Struckmann Hansen, J., Kristensen, J., & Burelli, P. (2020). Procedural content generation of puzzle games using conditional generative adversarial networks. In *PCG Workshop 2020* (Article No. 99, pp. 1–9). doi: <https://doi.org/10.1145/3402942.3409601>
7. Awiszus, M., Schubert, F., & Rosenhahn, B. (2020). TOAD GAN: Coherent style level generation from a single example. *arXiv*. doi: <https://arxiv.org/abs/2008.01531>

8. Liapis, A., Martinez, H. P., Togelius, J., & Yannakakis, G. N. (2021). Transforming exploratory creativity with DeLeNoX. arXiv. doi: <https://doi.org/10.48550/arXiv.2103.11715>
8. Zavorodnii, V.V., Zavorodnia, H.A., Valiavska, N.O., Adamenko, V.S., Dorohovtsev, Ye.V. & Nesmachnyi, P.V. (2022). Metod avtomatychnoyi henneratsiyi kontentu na osnovi protsedurnykh alhorytmiv [Method of automatic content generation based on procedural algorithms]. Vcheni zapysky Tavriys'koho natsional'noho universytetu imeni V.I. Vernads'koho. Serii: Tekhnichni nauky, 33(72), 91–96 [in Ukrainian]. DOI: <https://doi.org/10.32838/2663-5941/2022.1/15>
10. Zavorodnii, V.V., Zavorodnia, H.A., Drobotovych, K.Ye., Tenihin, O.V. & Shmatko, M.M. (2021). Matematychnе modelyuvannya u metodakh formal'noho doslidzhennya [Mathematical modeling in formal research methods]. Vcheni zapysky Tavriys'koho natsional'noho universytetu imeni V.I. Vernads'koho. Serii: Tekhnichni nauky, 32(71), 75–79 [in Ukrainian]. DOI: <https://doi.org/10.32838/2663-5941/2021.6/12>
11. Shaker, N., Togelius, J., & Nelson, M. J. (2016). Procedural content generation in games. Springer. doi: <https://doi.org/10.1007/978-3-319-42716-4>
12. Zavorodnii, V.V., Zavorodnia, H.A., Demchenko, I.V., Kramarenko, K.S., Shevchenko, I.O. & Yurchenko, A.V. (2022). Metod stvorennya shtuchnykh tekstur iz zadanymy parametramy [Method for creating artificial textures with specified parameters]. Vcheni zapysky Tavriys'koho natsional'noho universytetu imeni V.I. Vernads'koho. Serii: Tekhnichni nauky, 33(72), 86–90 [in Ukrainian]. DOI: <https://doi.org/10.32838/2663-5941/2022.2/14>
13. Li, B., Hu, Y., Guerrero, P., Hašan, M., Shi, L., Deschaintre, V., & Matusik, W. (2024). Procedural material generation with reinforcement learning. ACM Transactions on Graphics, 43(6), Article No. 280, 1–14. doi: <https://doi.org/10.1145/3687979>
14. Zavorodnii, V.V., Zavorodnia, H.A., Valiavska, N.O., Herasymenko, O.O., Kaliuzhnyi, O.V. & Stepovyi, A.V. (2022). Poshuk anomalii u danykh za dopomohoiu mashynnoho navchannya [Anomaly detection in data using machine learning]. Vcheni zapysky Tavriys'koho natsional'noho universytetu imeni V.I. Vernads'koho. Serii: Tekhnichni nauky, 33(72), 39–43 [in Ukrainian]. URL: https://tech.vernadskyjournals.in.ua/journals/2022/3_2022/6.pdf
15. Volz, V., Schrum, J., Liu, J., Lucas, S. M., Smith, A., & Risi, S. (2018). Evolving Mario levels in the latent space of a deep convolutional generative adversarial network. arXiv. doi: <https://doi.org/10.48550/arXiv.1805.00728>
16. Gisslén, L., Eakins, A., Gordillo, C., Bergdahl, J., & Tollmar, K. (2021). Adversarial reinforcement learning for procedural content generation. arXiv. doi: <https://doi.org/10.48550/arXiv.2103.04847>
17. Lesyk, V. O., & Doroshenko, A. Y. (2024). Neural network application to pseudorandom sequence generation simulation. Prombles in Programming, 2024, 2–3, 280–287. doi: <http://doi.org/10.15407/pp2024.02-03.280>
18. Liapis, A., Yannakakis, G. N., & Togelius, J. (2015). Sentient Sketchbook: Computer-aided game level authoring. IEEE Transactions on Computational Intelligence and AI in Games, 7(3), 244–260. url: <http://julian.togelius.com/Liapis2013Sentient.pdf>
19. Khalifa, A., Bontrager, P., Earle, S., & Togelius, J. (2021). PCGRL: Procedural content generation via reinforcement learning. arXiv. doi: <https://doi.org/10.48550/arXiv.2001.09212>
20. Park, K., Mott, B., Min, W., Wiebe, E., Boyer, K. E., & Lester, J. (2020). Generating game levels to develop computer science competencies in game-based learning environments. In Lecture Notes in Computer Science: Artificial Intelligence in Education (pp. 240–245). doi: https://doi.org/10.1007/978-3-030-52240-7_44
21. Baek, I.-C., Kim, S.-H., Lee, S.-y., Lee, D.-H., & Kim, K.-J. (2025). IPCGRL: Language instructed reinforcement learning for procedural level generation. arXiv. doi: <https://doi.org/10.48550/arXiv.2503.12358>
22. Lupiani, I. (2025). Procedural content generation for games: Automate 3D asset and environment creation with Blender Python and Geometry Nodes. Springer. doi: <https://doi.org/10.1007/979-8-8688-1787-8>

Стаття надійшла до редакції 22.11.2025

Стаття прийнята до друку 05.01.2026

Дата публікації статті 26.03.2026



Завгородня Ганна Анатоліївна

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
к.т.н., доцент
Берестейський проспект, 37, Київ, 03056
annzavgorodnya@gmail.com,
ORCID: 0000-0001-8523-1761

Завгородній Валерій Вікторович

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
д.т.н., професор
Берестейський проспект, 37, Київ, 03056
zavgorodniivalerii@gmail.com,
ORCID: 0000-0002-8347-7183

Для посилань:

Завгородня Г. А., Завгородній В. В. Метод генерації ігрового контенту за допомогою нейромереж. Механіка та математичні методи, 2026. Т. VIII. № 1. С. 54–68

For references:

G. Zavhorodnia, V. Zavhorodnii (2026). Method of game content generation using neural networks. Mechanics and Mathematical Methods. VIII (1). 54–68.

Метод генерації ігрового контенту за допомогою нейромереж © 2026 by Завгородня Г. А., Завгородній В. В. is licensed under CC BY 4.0