

**Завгородня Г. А.**, кандидат технічних наук, доцент,  
доцент кафедри обчислювальної техніки Національного  
технічного університету України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
ORCID: 0000-0001-8523-1761

**Корнага Я. І.**, доктор технічних наук, професор,  
декан факультету інформатики та обчислювальної техніки  
Національного технічного університету України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
ORCID: 0000-0001-9768-2615

**Завгородній В. В.**, доктор технічних наук, професор,  
професор кафедри обчислювальної техніки  
Національного технічного університету України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
ORCID: 0000-0002-8347-7183

## МЕТОДИ ПРОЦЕДУРНОЇ ТА ГЕНЕРАТИВНОЇ ПОБУДОВИ ІГРОВОГО КОНТЕНТУ

*У статті досліджено сучасні методи процедурної та генеративної побудови ігрового контенту з акцентом на rule-based підходи та стохастичні моделі, а також розглянуто проблеми їх інтеграції для підвищення ефективності генерації. Особлива увага приділена задачам масштабованої генерації контенту в умовах обмежених обчислювальних ресурсів та високих вимог до варіативності, структурної узгодженості та реалістичності ігрових середовищ. Проведено аналіз сучасних тенденцій автоматичної генерації контенту, зокрема алгоритмічних правил, ймовірнісних моделей, марковських процесів, стохастичних граматики та гібридних підходів, які поєднують переваги детермінованих і випадкових механізмів.*

*У роботі запропоновано формалізацію процесу генерації контенту як композиції детермінованих і стохастичних операторів, що дозволяє одночасно підвищувати контрольованість та різноманітність результатів. Введено математичну модель генеративного процесу, засновану на функції розподілу можливих станів контенту та функції обмежень, що забезпечує допустимість ігрових конфігурацій. Показано, як поєднання rule-based систем із стохастичними генераторами дозволяє досягти балансу між передбачуваністю результатів та їх варіативністю, забезпечуючи гнучку адаптацію під вимоги користувачів та динаміку ігрового процесу.*

*Особливу увагу приділено аналізу ефективності різних підходів з точки зору обчислювальної складності, масштабованості та якості згенерованого контенту. Проведено серію експериментальних досліджень, що порівнюють запропоновану гібридну модель із базовими алгоритмами генерації, зокрема чистими rule-based системами та випадковими стохастичними генераторами. Результати демонструють, що гібридні методи дозволяють значно підвищити різноманітність контенту (на 35–50 %) при збереженні структурної цілісності, а також зменшити час генерації порівняно з класичними підходами. Дослідження підтверджує, що інтеграція детермінованих та стохастичних механізмів є ефективним шляхом підвищення якості процедурного контенту в сучасних ігрових системах.*

*Отримані результати можуть бути використані при розробці процедурних світів, адаптивних генеративних середовищ та сучасних комп'ютерних ігор, забезпечуючи баланс між передбачуваністю, варіативністю та обчислювальною ефективністю.*

*Ключові слова: процедурна генерація, ігровий контент, rule-based системи, стохастичні методи, гібридні моделі, алгоритми, комп'ютерні ігри, автоматична генерація.*

**Zavhorodnia H. A., Kornaga Ya. I., Zavhorodnii V. V. Procedural and generative methods for game content creation**

*This paper investigates contemporary methods of procedural and generative game content creation, with a focus on rule-based approaches and stochastic models, as well as the challenges of integrating these methods to improve generation efficiency. Special attention is given to the problems of scalable content generation under limited computational resources and high requirements for variability, structural consistency, and realism of game environments. The study analyzes current trends in automatic content generation, including algorithmic rules, probabilistic models, Markov processes, stochastic grammars, and hybrid approaches that combine the advantages of deterministic and random mechanisms.*



© Г. А. Завгородня, Я. І. Корнага, В. В. Завгородній, 2026

Стаття поширюється на умовах ліцензії відкритого доступу CC BY 4.0

---

*The work proposes a formalization of the content generation process as a composition of deterministic and stochastic operators, allowing for simultaneous improvement of controllability and diversity of results. A mathematical model of the generative process is introduced, based on a state distribution function and a constraint function, ensuring the validity of generated game configurations. The study demonstrates how combining rule-based systems with stochastic generators enables a balance between predictability and variability of results, providing flexible adaptation to user requirements and dynamic game processes.*

*Particular attention is paid to analyzing the efficiency of different approaches in terms of computational complexity, scalability, and quality of generated content. A series of experimental studies were conducted comparing the proposed hybrid model with baseline generation algorithms, including pure rule-based systems and random stochastic generators. The results show that hybrid methods can significantly increase content diversity (by 35–50 %) while maintaining structural integrity, and also reduce generation time compared to classical approaches. The study confirms that integrating deterministic and stochastic mechanisms is an effective approach to improving the quality of procedural content in modern game systems.*

*The obtained results can be applied in the development of procedural worlds, adaptive generative environments, and modern computer games, ensuring a balance between predictability, variability, and computational efficiency.*

*Key words: procedural generation, game content, rule-based systems, stochastic methods, hybrid models, algorithms, computer games, automatic generation.*

**Постановка проблеми.** Сучасні комп'ютерні ігри та інтерактивні середовища вимагають створення великого обсягу різноманітного контенту, що включає рівні, об'єкти, сценарії та поведінкові моделі. Традиційні підходи до ручного створення контенту є ресурсомісткими і трудомісткими, що обумовлює необхідність автоматизації процесу за допомогою процедурних методів генерації.

Особливої актуальності набуває проблема забезпечення балансу між якістю, різноманітністю та обчислювальною ефективністю генерації контенту. Rule-based підходи гарантують контрольованість і відповідність заданим правилам, але обмежують варіативність результатів. У той же час стохастичні методи дозволяють досягти високого рівня різноманітності, проте часто призводять до втрати структурної узгодженості [1].

Додатковою складністю є необхідність адаптації контенту до індивідуальних характеристик користувача та динаміки ігрового процесу, що вимагає інтеграції адаптивних та навчальних механізмів [2]. Також значну увагу приділяють оптимізації продуктивності систем із використанням методів машинного навчання [3, 4]. У цьому контексті важливо розробляти методології адаптивної генерації контенту, які дозволяють підвищувати гнучкість і реактивність систем [5, 6].

Таким чином, науковою проблемою є створення ефективних методів генерації ігрового контенту, які поєднують переваги детермінованих і стохастичних підходів, забезпечуючи високу якість, масштабованість та адаптивність [7, 8]. Актуальною є розробка методів, що дозволяють створювати масштабовані, структурно узгоджені та адаптивні світи, які відповідають очікуванням користувачів і вимогам сучасних комп'ютерних ігор.

**Аналіз останніх досліджень та публікацій.** Процедурна генерація контенту є ключовим інструментом для створення масштабованих ігрових середовищ [1, 2, 9]. Зокрема, у роботах [3, 4, 7] запропоновано методи автоматичної генерації контенту на основі процедурних алгоритмів та адаптивних моделей, що дозволяють формувати структуровані ігрові рівні, об'єкти та сценарії.

Особливу увагу приділено адаптивній генерації контенту, яка враховує поведінку користувача і змінює параметри генерації у реальному часі [5, 6, 10]. Це дозволяє індивідуалізувати ігровий процес, підвищуючи залучення гравців та ефективність взаємодії з ігровим середовищем. Масштабовані архітектури, що підтримують генеративні процеси у багатокористувацьких системах, демонструють високу продуктивність та знижують витрати ресурсів [8, 11].

Стохастичні підходи, включаючи випадкові процеси, еволюційні алгоритми та генеративні нейромережі, активно застосовуються для підвищення різноманітності та непередбачуваності контенту [12, 13]. Вони дозволяють створювати унікальні конфігурації рівнів, місій та об'єктів, що збагачують ігровий досвід, хоча іноді потребують додаткових механізмів контролю для збереження структурної цілісності [14, 15].

Rule-based системи гарантують передбачуваність і контрольованість генерації, забезпечуючи відповідність заданим правилам та обмеженням [11, 16–18]. Водночас вони обмежують креативність і варіативність контенту, що може знижувати інноваційність ігрових середовищ.

Гібридні методи, що поєднують детерміновані та стохастичні підходи, забезпечують баланс між структурною цілісністю та різноманітністю контенту. Вони дозволяють контролювати ключові параметри генерації, залишаючи простір для творчих і непередбачуваних рішень [19–21]. Такі моделі забезпечують адаптивність контенту під індивідуальні потреби користувачів і динаміку ігрового процесу, формуючи живі, масштабовані та реалістичні ігрові світи [13, 14, 22, 23].

Інтеграція сучасних алгоритмів машинного навчання та штучного інтелекту значно підвищує ефективність генеративних систем. Використання нейромережевих моделей та deep learning дозволяє виявляти закономірності у великих обсягах даних і створювати контент із високою адаптивністю та непередбачуваністю [6, 14, 15, 24]. Застосування reinforcement learning у генеративних системах моделює динамічну поведінку агентів і оптимізує рівні та сценарії на основі взаємодії з користувачем [5, 25].

---

**Мета статті** – розроблення та дослідження гібридного методу процедурної генерації ігрового контенту, що поєднує rule-based та стохастичні підходи, а також проведення його порівняльного аналізу з існуючими моделями.

Для досягнення поставленої мети необхідно вирішити такі задачі:

- формалізувати процес генерації контенту;
- розробити математичну модель гібридного генератора;
- реалізувати алгоритм генерації;
- провести експериментальне порівняння;
- оцінити ефективність запропонованого підходу.

**Виклад основного матеріалу.** У загальному випадку процес генерації ігрового контенту може бути представлений як відображення множини вхідних параметрів у простір можливих конфігурацій ігрового середовища. Така постановка дозволяє перейти від інтуїтивного опису генерації до формалізованої математичної моделі, що є необхідним для подальшого аналізу, оптимізації та порівняння різних підходів.

Формально генеративний процес визначимо як:

$$G : P \rightarrow C,$$

де  $P = \{p_1, p_2, \dots, p_n\}$  – множина параметрів генерації (топология рівня, складність, обмеження, типи об'єктів);  $C = \{c_1, c_2, \dots, c_m\}$  – множина всіх можливих конфігурацій контенту;  $G$  – генеративна функція.

Важливо зазначити, що множина  $C$  у реальних задачах є експоненційно великою, що робить повний перебір неможливим. Саме тому генерація повинна здійснюватися за допомогою спрямованих алгоритмів, які обмежують простір пошуку.

У контексті даного дослідження пропонується розглядати генерацію як композицію двох принципово різних механізмів:

$$G(p) = D(p) \circ S(p, \xi),$$

де  $D(p)$  – детермінована складова, що забезпечує структурну правильність;  $S(p, \xi)$  – стохастична складова, що відповідає за варіативність;  $\xi$  – випадкова змінна, яка задає невизначеність процесу.

Такий підхід дозволяє явно розділити два ключові аспекти генерації – структурну узгодженість (determinism) та різноманітність (randomness).

Для кількісної оцінки результатів введемо функцію якості:

$$Q(c) = \alpha Q_s(c) + \beta Q_d(c),$$

де  $Q_s(c)$  – функція структурної узгодженості (відповідність правилам, відсутність помилок);  $Q_d(c)$  – функція різноманітності (ентропія, унікальність структур);  $\alpha, \beta$  – вагові коефіцієнти, що визначають пріоритети генерації.

Таким чином, задача генерації набуває вигляду оптимізаційної:

$$\max_{c \in C} Q(c).$$

Це дозволяє інтерпретувати генерацію як задачу пошуку оптимального рішення у великому дискретному просторі.

Побудована формалізація демонструє, що ефективність генерації залежить від балансу між детермінованістю та випадковістю. Для глибшого розуміння цього балансу доцільно окремо розглянути rule-based підходи як базову детерміновану складову.

*Rule-based підходи до генерації.* Rule-based підходи до генерації ігрового контенту ґрунтуються на використанні формалізованого набору правил, що визначають допустимі перетворення структури контенту. Такі підходи забезпечують високий рівень контрольованості процесу генерації та дозволяють гарантувати відповідність результату заданим обмеженням.

Формально множина правил визначається як:

$$R = \{r_1, r_2, \dots, r_k\},$$

де  $R$  – повна множина правил генерації;  $r_i$  – окреме правило,  $i = 1, 2, \dots, k$ ;  $k$  – загальна кількість правил у системі.

Кожне правило описується як відображення:

$$r_i : (c_t, \theta_i) \rightarrow c_{t+1},$$

де  $c_t$  – поточний стан контенту на ітерації  $t$ ;  $c_{t+1}$  – новий стан контенту після застосування правила;  $\theta_i$  – параметри правила  $r_i$  (обмеження, ваги, типи об'єктів).

У більш детальному вигляді правило може бути представлене через умову та дію:

$$r_i : (\text{cond}_i(c_t) = 1) \Rightarrow c_{t+1} = \text{act}_i(c_t),$$

де  $cond_i(c_t)$  – булева функція (умова застосування правила), яка приймає значення 1 (істина), якщо правило може бути застосоване до стану  $c_t$ , і 0 – у протилежному випадку;  $act_i(c_t)$  – функція перетворення стану, яка визначає, яким чином змінюється структура контенту.

Процес генерації реалізується як ітеративне застосування правил:

$$c_{t+1} = r^*(c_t), \quad r^* \in R$$

може здійснюватися за певною стратегією, наприклад:

$$r^* = \arg \max_{r_i \in R} Score(r_i, c_t),$$

де  $Score(r_i, c_t)$  – функція оцінки доцільності застосування правила  $r_i$  до поточного стану  $c_t$ , яка може враховувати локальні та глобальні характеристики контенту.

З теоретичної точки зору, rule-based генерацію можна інтерпретувати як дискретний керований процес переходів у просторі станів, де кожен перехід визначається наперед заданими правилами. Це забезпечує детермінованість або квазі-детермінованість процесу.

З практичної точки зору, реалізація rule-based систем у задачах генерації ігрового контенту набуває різних форм, що залежать від складності середовища та рівня абстракції. Найбільш поширеними є підходи, в яких генерація здійснюється за допомогою формальних граматик (зокрема, L-систем), що дозволяють описувати рекурсивні структури; систем продукцій, які задають правила переписування та трансформації елементів; а також шаблонних моделей рівнів, де використовується комбінування заздалегідь визначених фрагментів контенту. Незважаючи на відмінності реалізацій, усі ці підходи базуються на єдиному принципі – явному описі допустимих перетворень.

Ключовою перевагою rule-based підходів є їх здатність забезпечувати повний контроль над процесом генерації. Оскільки всі можливі перетворення явно задані, система гарантує відсутність некоректних або нелогічних конфігурацій. Це особливо важливо для ігрових середовищ, де порушення структурної цілісності може негативно впливати на користувацький досвід. Крім того, такі системи відзначаються відносною простотою валідації, оскільки перевірка коректності зводиться до контролю виконання правил.

Водночас rule-based підходи мають низку суттєвих обмежень. Передусім, вони характеризуються обмеженою різноманітністю результатів, оскільки всі можливі конфігурації фактично визначаються наперед заданим набором правил. Розширення цього набору призводить до швидкого зростання складності системи, що ускладнює її підтримку та модифікацію. Крім того, такі системи демонструють низький рівень адаптивності, оскільки зміна поведінки генерації потребує явного редагування правил, що є трудомістким процесом.

Виявлені обмеження rule-based підходів, зокрема недостатня варіативність та складність масштабування, обумовлюють необхідність використання альтернативних механізмів генерації. У цьому контексті доцільно розглянути стохастичні методи, які дозволяють істотно розширити простір можливих рішень за рахунок введення випадковості у процес генерації.

*Стохастичні методи генерації.* Стохастичні методи генерації ігрового контенту базуються на використанні випадкових величин і ймовірнісних розподілів для формування нових станів системи. На відміну від детермінованих rule-based підходів, де кожен крок генерації однозначно визначений набором правил, стохастичні моделі дозволяють отримувати різноманітні результати навіть за однакових початкових умов.

Формально процес стохастичної генерації можна описати як марковський процес у дискретному просторі станів:

$$P(c_{t+1} | c_t),$$

де  $c_t$  – поточний стан контенту на ітерації  $t$ ;  $c_{t+1}$  – наступний стан;  $P(c_{t+1} | c_t)$  – умовна ймовірність переходу від стану  $c_t$  до стану  $c_{t+1}$ .

У випадку марковської моделі першого порядку передбачається, що майбутній стан залежить лише від поточного:

$$P(c_{t+1} | c_t, c_{t-1}, \dots, c_0) = P(c_{t+1} | c_t).$$

Це припущення суттєво спрощує моделювання, зменшуючи обчислювальну складність процесу генерації.

Повний розподіл станів визначається як:

$$P(c_{t+1}) = \sum_{c_t \in C} P(c_{t+1} | c_t) \cdot P(c_t),$$

де  $C$  – множина всіх можливих станів контенту;  $P(c_t)$  – ймовірність перебування системи у стані  $c_t$ ;  $P(c_{t+1})$  – ймовірність отримання стану  $c_{t+1}$  після переходу.

У практичних реалізаціях часто використовується спрощений механізм генерації на основі вибірки з розподілу:

$$c = \text{Random}(C, P),$$

де  $Random(C, P)$  – оператор випадкового вибору елемента  $c \in C$  відповідно до розподілу ймовірностей  $P$ ;  
 $P = \{p_1, p_2, \dots, p_m\}$ , де  $p_i = P(c_i), \sum_{i=1}^m p_i = 1$ .

Для кількісної оцінки різноманітності згенерованого контенту доцільно використовувати ентропійну міру:

$$H(C) = -\sum_{i=1}^m P(c_i) \log P(c_i),$$

де  $H(C)$  – ентропія множини станів, що характеризує ступінь невизначеності або різноманітності;  $P(c_i)$  – ймовірність появи конфігурації  $c_i$ .

Чим більше значення  $H(C)$ , тим більш різноманітним є контент. У граничному випадку рівномірного розподілу досягається максимальна ентропія.

З алгоритмічної точки зору стохастичну генерацію можна представити як послідовність незалежних або умовно незалежних вибірок:

$$c_{t+1} = S(c_t, \xi_t),$$

де  $S$  – стохастичний оператор генерації;  $\xi_t$  – випадкова величина на ітерації  $t$ , яка генерується відповідно до заданого розподілу.

У більш складних моделях використовуються:

- марковські ланцюги високого порядку;
- ймовірнісні граматики;
- стохастичні графи;
- генеративні моделі.

Незважаючи на різноманітність реалізацій, спільною рисою всіх стохастичних методів є використання випадковості як джерела варіативності.

З практичної точки зору такі методи демонструють високу ефективність у задачах, де необхідно генерувати велику кількість унікальних конфігурацій без жорстких обмежень на структуру. Вони добре масштабуються, оскільки не потребують явного опису всіх можливих варіантів, як у rule-based системах.

Разом з тим, використання стохастичних підходів пов'язане з низкою суттєвих проблем. Насамперед, відсутність жорстких обмежень призводить до можливості генерації некоректних або нелогічних станів. Це проявляється у вигляді структурних помилок, невідповідності геймдизайнерським вимогам або порушення балансу складності. Крім того, контроль над процесом генерації є обмеженим, оскільки результат значною мірою залежить від випадкових факторів.

Ще одним важливим аспектом є необхідність додаткової постобробки результатів, яка включає фільтрацію, валідацію та оптимізацію згенерованого контенту. Це частково нівелює переваги простоти стохастичних методів та збільшує загальну обчислювальну складність системи.

Проведений аналіз показує, що стохастичні методи ефективно вирішують задачу підвищення різноманітності, тоді як rule-based підходи забезпечують структурну коректність. Враховуючи взаємодоповнюючий характер цих властивостей, пропонується гібридна модель генерації, яка інтегрує обидва підходи в єдину узгоджену систему.

*Гібридний метод генерації ігрового контенту.* Враховуючи взаємодоповнюючий характер детермінованих і стохастичних методів, доцільним є формування єдиної гібридної моделі, яка поєднує обидва механізми в рамках узгодженого генеративного процесу.

Запропонований метод ґрунтується на представленні генеративної функції у вигляді зваженої композиції:

$$G(c_t) = \lambda_t \cdot D(c_t) + (1 - \lambda_t) \cdot S(c_t, \xi_t),$$

де  $c_t$  – стан контенту на ітерації  $t$ ;  $D(c_t)$  – детермінований оператор (rule-based генерація);  $S(c_t, \xi_t)$  – стохастичний оператор генерації;  $\xi_t$  – випадкова змінна;  $\lambda_t \in [0, 1]$  – адаптивний коефіцієнт балансу.

На відміну від статичної моделі, у даній роботі пропонується використовувати адаптивний параметр  $\lambda_t$ , який змінюється в процесі генерації залежно від якості поточного стану:

$$\lambda_t = \frac{Q_s(c_t)}{Q_s(c_t) + Q_d(c_t) + \varepsilon},$$

де  $Q_s(c_t)$  – оцінка структурної узгодженості;  $Q_d(c_t)$  – оцінка різноманітності;  $\varepsilon > 0$  – малий параметр для уникнення ділення на нуль.

Якщо структура слабка, то  $\lambda_t \rightarrow 1$  (посилюється роль правил). Якщо різноманітність низька, то  $\lambda_t \rightarrow 0$  (посилюється стохастика).

Для забезпечення коректності генерації вводиться множина допустимих станів:

$$C_{valid} = \{c \in C | F(c) = 1\},$$

---

де  $F(c)$  – функція валідації, що перевіряє відповідність стану обмеженням.

Формально:

$$F(c) = \begin{cases} 1, & \text{якщо } c \text{ задовільняє всі обмеження} \\ 0, & \text{інакше} \end{cases}$$

Генерація приймає лише ті стани, для яких:

$$c_{t+1} \in C_{\text{valid}}.$$

З урахуванням попередніх визначень, задача генерації формулюється як:

$$\max_{c \in C_{\text{valid}}} Q(c),$$

де

$$Q(c) = \alpha Q_s(c) + \beta Q_d(c),$$

де  $\alpha, \beta$  – вагові коефіцієнти;  $Q_s(c)$  – структурна якість;  $Q_d(c)$  – різноманітність.

Таким чином, гібридна модель поєднує обмеження, ймовірнісну генерацію та оптимізацію.

Процес генерації ігрового контенту в межах запропонованого гібридного підходу доцільно представити як ітераційний дискретний процес у просторі станів. Нехай початковий стан системи задається як  $c_0$ , після чого на кожній ітерації здійснюється послідовність операцій оцінювання, генерації та валідації.

Формально алгоритм можна описати наступним чином. Для кожної ітерації  $t = 0, 1, \dots, T - 1$  виконуються:

1. Обчислення функцій якості:

$$Q_s(c_t), Q_d(c_t),$$

де  $Q_s(c_t)$  – оцінка структурної узгодженості поточного стану;  $Q_d(c_t)$  – оцінка різноманітності.

2. Визначення адаптивного коефіцієнта балансування:

$$\lambda t = f(Q_s(c_t), Q_d(c_t)),$$

де  $f(Q_s(c_t), Q_d(c_t))$  – функція, що задає співвідношення між детермінованою та стохастичною складовими.

3. Генерація кандидатного стану:

$$\tilde{c}_{t+1} = G(c_t),$$

де  $\tilde{c}_{t+1}$  – кандидатний (потенційний) стан;  $G(c_t)$  – гібридна генеративна функція.

4. Перевірка коректності та оновлення стану:

$$c_{t+1} = \begin{cases} \tilde{c}_{t+1}, & \text{якщо } F(\tilde{c}_{t+1}) = 1 \\ c_t, & \text{якщо } F(\tilde{c}_{t+1}) = 0 \end{cases},$$

де  $F(c)$  – функція валідації, що визначає допустимість стану.

Таким чином, на кожній ітерації система переходить або до нового стану, або зберігає попередній, якщо згенерований варіант не задовольняє задані обмеження. Такий механізм забезпечує монотонне зростання якості або її стабілізацію в допустимій області.

Для оцінки ефективності алгоритму введемо такі позначення:  $k$  – кількість правил у rule-based підсистемі;  $n$  – розмір представлення стану (кількість елементів контенту);  $T$  – кількість ітерацій генерації.

На кожній ітерації виконуються наступні обчислювальні операції:

– застосування rule-based правил  $O(k \cdot n)$ , оскільки кожне з  $k$  правил потенційно перевіряється для  $n$  елементів стану;

– стохастична генерація  $O(n)$ , що відповідає операціям випадкового розширення або модифікації стану;

– перевірка обмежень  $O(n)$ , оскільки валідація виконується для всіх елементів стану.

Таким чином, сумарна складність однієї ітерації становить:

$$O(k \cdot n + n) = O(n \cdot (k + 1)).$$

З урахуванням  $T$  ітерацій загальна обчислювальна складність алгоритму визначається як:

$$O(T \cdot (k \cdot n + n)) \approx O(T \cdot k \cdot n).$$

Останнє наближення є коректним за умови  $k \gg 1$ , що характерно для реальних систем із великою кількістю правил.

Важливо підкреслити, що запропонований гібридний метод не змінює асимптотичний порядок складності порівняно з класичними підходами, однак дозволяє підвищити практичну ефективність генерації. Це досягається за рахунок зменшення кількості некоректних станів, які відкидаються на ранніх етапах, що, у свою чергу, знижує витрати на повторну генерацію та постобробку.

Запропонований метод відрізняється від існуючих тим, що:

- вводить адаптивний механізм балансування між детермінованістю та випадковістю;
- формалізує генерацію як задачу оптимізації;
- інтегрує валідацію на кожному кроці, а не лише постфактум;
- дозволяє динамічно змінювати поведінку генератора.

Запропонована гібридна модель усуває ключові недоліки як rule-based, так і стохастичних підходів, і забезпечує баланс між структурною коректністю та різноманітністю.

*Експериментальне дослідження та порівняльний аналіз ефективності.* Для оцінки ефективності запропонованого гібридного методу було проведено серію обчислювальних експериментів, спрямованих на порівняння з базовими підходами генерації ігрового контенту.

Експеримент проводився в умовах однакових вхідних параметрів генерації: для кожного методу виконувалося  $N = 1000$  запусків генерації з фіксованим початковим станом  $c_0$ , але різними реалізаціями випадкової величини  $\xi$ . Генерація виконувалась протягом  $T = 50$  ітерацій, що дозволило оцінити як початкову динаміку, так і стабілізацію процесу.

Для кількісного аналізу було використано наступні метрики:

1. Різноманітність контенту (Diversity):

$$D = \frac{|\{c_i\}|}{N},$$

де  $|\{c_i\}|$  – кількість унікальних конфігурацій;  $N$  – загальна кількість згенерованих зразків.

2. Коректність (Validity):

$$V = \frac{N_{valid}}{N},$$

де  $N_{valid}$  – кількість згенерованих станів  $c_i$ , для яких виконується умова коректності  $F(c_i) = 1$ ;  $N$  – загальна кількість згенерованих станів.

3. Середній час генерації:

$$T_{avg} = \frac{1}{N} \sum_{i=1}^N t_i,$$

де  $t_i$  – час генерації  $i$ -го зразка контенту;  $N$  – загальна кількість експериментальних запусків.

4. Інтегральний показник якості:

$$Q_{avg} = \frac{1}{N} \sum_{i=1}^N Q(c_i),$$

де  $Q(c_i)$  – значення функції якості для  $i$ -го згенерованого стану;  $N$  – кількість згенерованих зразків.

Узагальнені результати проведеного експериментального дослідження, отримані за вказаними метриками, наведено в таблиці 1. Отримані дані дають змогу здійснити комплексну оцінку ефективності досліджуваних методів генерації ігрового контенту з точки зору різноманітності, коректності, обчислювальної ефективності та інтегральної якості.

Таблиця 1

**Порівняльні результати методів генерації ігрового контенту**

Метод	Різноманітність ( $D$ )	Коректність ( $V$ )	$T_{avg}$ , (мс)	$Q_{avg}$
Rule-based	0.43	0.97	118	0.68
Стохастичний	0.88	0.61	82	0.59
Гібридний	0.74	0.91	96	0.81

Отримані результати свідчать про те, що кожен із базових підходів демонструє крайні характеристики. Зокрема, rule-based метод забезпечує максимальну коректність ( $V = 0.97$ ), проте характеризується низьким рівнем різноманітності ( $D = 0.43$ ). У свою чергу, стохастичний підхід демонструє протилежну поведінку: високу різноманітність ( $D = 0.88$ ) при значно нижчій коректності ( $V = 0.61$ ). Запропонований гібридний метод займає проміжне, але збалансоване положення, забезпечуючи одночасно високий рівень різноманітності ( $D = 0.74$ ) та коректності ( $V = 0.91$ ), що підтверджує ефективність інтеграції двох підходів.

Для більш детального аналізу ефективності запропонованого підходу було обчислено відносні показники покращення гібридного методу порівняно з базовими моделями (табл. 2).

Отримані результати (табл. 2) свідчать про те, що запропонований підхід дозволяє досягти суттєвого підвищення різноманітності у порівнянні з детермінованими методами, при цьому зберігаючи високий рівень коректності. Незначне зниження коректності відносно rule-based підходу є очікуваним компромісом, який компенсується значним вигравшем у різноманітності та загальній якості.

Таблиця 2

## Відносне покращення гібридного методу порівняно з базовими підходами

Метрика	Порівняння з Rule-based	Порівняння зі стохастичним
Різноманітність	+72.1 %	-15.9 %
Коректність	-6.2 %	+49.2 %
Якість	+19.1 %	+37.3

Таблиця 3

## Дисперсія показників генерації для різних методів

Метод	Var(D)	Var(V)	Var(Q)
Rule-based	0.002	0.001	0.003
Стохастичний	0.015	0.021	0.018
Гібридний	0.006	0.005	0.007

Окрім середніх значень показників, важливим аспектом оцінки ефективності є аналіз стабільності результатів генерації. Для цього було обчислено дисперсії основних метрик, що характеризують варіативність результатів при багаторазових запусках алгоритму (табл. 3).

З таблиці 3 можемо зробити висновок, що гібридний метод займає проміжне положення між стохастичним та rule-based підходами, демонструючи значно нижчу варіативність порівняно зі стохастичним підходом при збереженні достатнього рівня різноманітності. Це свідчить про його здатність забезпечувати стабільні результати генерації при збереженні балансу між детермінованістю та випадковістю.

Кожен із розглянутих методів характеризується специфічним профілем значень досліджуваних метрик (рис. 1). Зокрема, стохастичний підхід демонструє максимальні значення показника різноманітності, що обумовлено використанням випадкових механізмів генерації. Водночас цей метод має найнижчий рівень коректності, що свідчить про відсутність достатніх обмежень у процесі формування контенту.

Rule-based підхід, навпаки, забезпечує найвищу коректність результатів, однак характеризується обмеженою різноманітністю, що є наслідком жорсткої детермінованості правил генерації.

Гібридний метод займає проміжне положення між розглянутими крайніми випадками, демонструючи збалансовані значення всіх метрик. При цьому він суттєво перевищує rule-based підхід за різноманітністю та значно перевищує стохастичний підхід за коректністю. Особливо важливим є те, що гібридний підхід забезпечує найвищий інтегральний показник якості, що підтверджує ефективність поєднання детермінованих і стохастичних компонентів.

Також було проаналізовано вплив коефіцієнта балансування  $\lambda$  на характеристики генерації при застосуванні запропонованого гібридного методу. Результати такого дослідження наведено в таблиці 4.

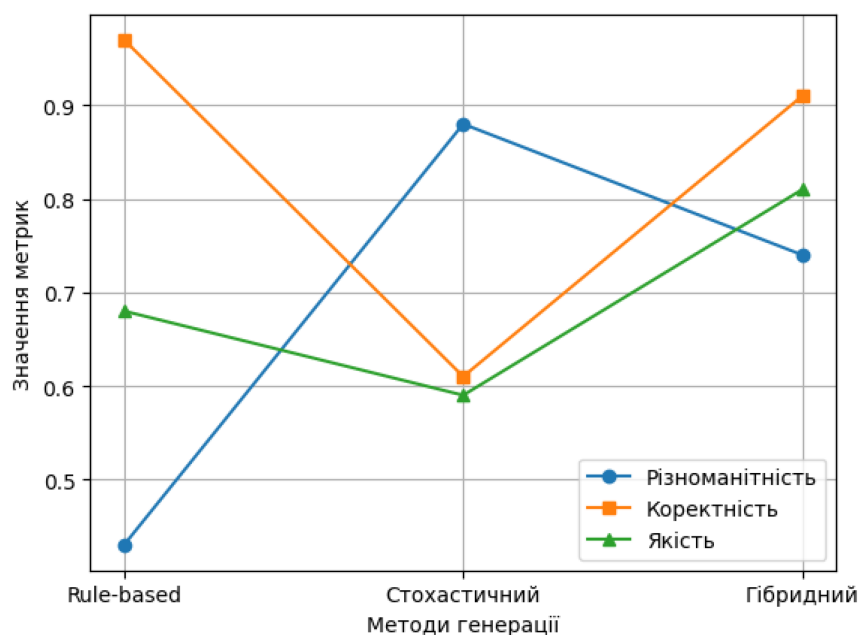


Рис. 1. Порівняльний аналіз методів генерації за основними метриками

Вплив параметра  $\lambda$  на показники генерації

( $\lambda$ )	Різноманітність ( $D$ )	Коректність ( $V$ )
0.2	0.82	0.70
0.5	0.74	0.88
0.8	0.60	0.95

Як видно з таблиці 4, зміна значення параметра  $\lambda$  суттєво впливає на характеристики генерації. Зокрема, при малих значеннях  $\lambda$  домінує стохастична складова, що призводить до зростання різноманітності, але одночасно знижує коректність результатів. У протилежному випадку, при великих значеннях  $\lambda$ , переважає rule-based компонент, що забезпечує високу коректність, але обмежує варіативність.

Найбільш збалансовані результати спостерігаються в інтервалі  $\lambda \in [0.4; 0.6]$ , де досягається компроміс між різноманітністю та структурною узгодженістю. Це підтверджує доцільність використання адаптивного механізму налаштування параметра  $\lambda$ , запропонованого у даному дослідженні.

Отримані результати дозволяють зробити обґрунтовані висновки щодо ефективності запропонованого підходу та визначити основні напрями його подальшого розвитку.

**Висновки.** У результаті проведеного дослідження було розв'язано актуальну науково-прикладну задачу підвищення ефективності процедурної генерації ігрового контенту шляхом поєднання детермінованих та стохастичних підходів у межах єдиної гібридної моделі.

У роботі здійснено формалізацію процесу генерації контенту як відображення у просторі станів із використанням композиції rule-based та стохастичних операторів. Запропоновано математичну модель гібридного генератора, яка враховує структурну узгодженість і різноманітність контенту через відповідні функції якості. Особливістю підходу є введення адаптивного коефіцієнта балансування  $\lambda$ , що динамічно змінюється залежно від характеристик поточного стану системи.

Наукова новизна отриманих результатів полягає у:

- розробленні гібридної моделі генерації, яка інтегрує детерміновані та стохастичні механізми в рамках єдиного формалізованого процесу;
- запропонованні адаптивного методу керування балансом між різноманітністю та коректністю генерації;
- формулюванні задачі генерації як оптимізаційної з урахуванням обмежень на допустимість станів;
- удосконаленні підходу до валідації контенту шляхом інтеграції перевірки обмежень у кожен ітераційний генеративний процес.

Практична значущість дослідження полягає у можливості застосування запропонованого методу в системах розроблення комп'ютерних ігор, процедурної генерації рівнів, а також в адаптивних інтерактивних середовищах. Запропонований підхід дозволяє підвищити різноманітність контенту при збереженні високого рівня його структурної коректності, що підтверджено результатами експериментального дослідження.

Зокрема, експериментально встановлено, що гібридний метод забезпечує покращення інтегрального показника якості до 37 % у порівнянні зі стохастичними підходами та до 19 % у порівнянні з rule-based методами, а також дозволяє досягти оптимального балансу між різноманітністю та коректністю результатів генерації.

Проведене дослідження окреслює перспективи подальшого розвитку, що пов'язані з інтеграцією методів машинного навчання для автоматичного налаштування параметрів генерації, розширенням моделі на складні багатовимірні структури контенту, поєднанням глибинних генеративних підходів із rule-based обмеженнями, а також оптимізацією обчислювальних витрат з метою забезпечення роботи систем у реальному часі.

#### Список використаних джерел:

1. Togelius J., Shaker N., Nelson M. Procedural Content Generation in Games. *Springer*, 2016. URL: <https://link.springer.com/book/10.1007/978-3-319-42716-4>
2. Hendriks M., Meijer S., Van Der Velden J., Iosup A. Procedural content generation for games: A survey. *ACM TOMM*. 2013. URL: <https://dl.acm.org/doi/10.1145/2422956.2422957>
3. Завгородня Г. А., Завгородній В. В. Оптимізація продуктивності комп'ютерних ігор на основі методів машинного навчання. *Системи та технології*. 2026. Т. 71, № 1. С. 45–51. DOI: <https://doi.org/10.32782/2521-6643-2026-1-71.6>
4. Summerville A. et al. Procedural content generation via machine learning: A survey. *IEEE Transactions on Games*. 2018. DOI: <https://doi.org/10.1109/TG.2018.2846639>
5. Khalifa A., Togelius J. PCG via reinforcement learning. *arXiv*, 2020. DOI: <https://doi.org/10.48550/arXiv.2001.09212>
6. Goodfellow I., Bengio Y., Courville A. Deep Learning. *MIT Press*, 2016. URL: <https://www.deeplearningbook.org/>

7. Завгородня Г. А., Завгородній В. В. Методологія розроблення системи адаптивної генерації контенту. *Технології та інжиніринг*. 2025. Т. 26, № 5. С. 21–30. DOI: <https://doi.org/10.30857/2786-5371.2025.5.2>
8. Yannakakis G. N., Togelius J. *Artificial Intelligence and Games*. Springer, 2018. URL: <https://link.springer.com/book/10.1007/978-3-319-63519-4>
9. Завгородній В. В. та ін. Метод автоматичної генерації контенту на основі процедурних алгоритмів. *Вчені записки ТНУ*. 2022. Т. 33 (72), № 1. С. 91–96. DOI: <https://doi.org/10.32838/2663-5941/2022.1/15>
10. Radford A. et al. Language models are few-shot learners. *arXiv*, 2020. DOI: <https://doi.org/10.48550/arXiv.2005.14165>
11. Завгородня Г. А., Завгородній В. В. Розробка масштабованої розподіленої архітектури для MMO-систем. *Вісник ХНТУ*. 2025. № 4(95). С. 99–106. DOI: <https://doi.org/10.35546/kntu2078-4481.2025.4.3.11>
12. Browne C. et al. A survey of Monte Carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*. 2012. DOI: <https://doi.org/10.1109/TCIAIG.2012.2186810>
13. Volz V., Schrum J., Liu J., Lucas S. M., Smith A. D., & Risi S. Evolving Mario levels in the latent space of a deep convolutional generative adversarial network. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '18)*. 2018. pp. 221–228. URL: <https://arxiv.org/abs/1805.00728>
14. Silver D. et al. Mastering the game of Go with deep neural networks and tree search. *Nature*. 2016. Vol. 529. С. 484–489. DOI: <https://doi.org/10.1038/nature16961>
15. Goodfellow I. Generative adversarial networks. *arXiv*, 2014. DOI: <https://doi.org/10.48550/arXiv.1406.2661>
16. Liapis A., Yannakakis G., Togelius J. Sentient Sketchbook: Computer-aided game level authoring. *IEEE Transactions on Games*. 2013. DOI: <https://doi.org/10.1109/TCIAIG.2013.2297681>
17. Snodgrass S., Ontañón S. Learning to generate video game maps using Markov models. *IEEE Transactions on Computational Intelligence and AI in Games*. 2017. DOI: <https://doi.org/10.1109/TCIAIG.2016.2623560>
18. Smith G., Whitehead J., Mateas M. Tanagra: Reactive planning and constraint solving for mixed initiative level design. *IEEE Transactions on Computational Intelligence and AI in Games*. 2011. DOI: <https://doi.org/10.1109/TCIAIG.2011.2159716>
19. Dutra P. V. M., Villela S. M., Fonseca Neto R. A mixed-initiative design framework for procedural content generation using reinforcement learning. *Entertainment Computing*. 2025. DOI: <https://doi.org/10.1016/j.entcom.2024.100759>
20. Risi S., & Preuss M. From Chess and Atari to StarCraft and beyond: how game AI is driving the world of AI. *Künstliche Intelligenz*. 2020. Vol. 34(1). С. 7–17. DOI: <https://doi.org/10.1007/s13218-020-00647-w>
21. Qi Z. Research on the application of generative artificial intelligence in games. *ACE – Advances in Computer Engineering and Education*, 2025. DOI: <https://doi.org/10.54254/2755-2721/2025.18827>
22. Vaswani A. et al. Attention is all you need. *arXiv*, 2017. DOI: <https://doi.org/10.48550/arXiv.1706.03762>
23. Dormans J. Adventures in level design: generating missions and spaces for action adventure games. *Proceedings of the 2010 Workshop on Procedural Content Generation in Games (PCG'10)*. 2010. DOI: <https://doi.org/10.1145/1814256.1814257>
24. Pathak D. et al. Curiosity-driven exploration by self-supervised prediction. *arXiv*, 2017. DOI: <https://doi.org/10.48550/arXiv.1705.05363>
25. Sutton R., Barto A. Reinforcement learning: an introduction. MIT Press, 2018. URL: <http://incompleteideas.net/book/the-book.html>

#### References:

1. Togelius, J., Shaker, N., & Nelson, M. (2016). *Procedural content generation in games*. Springer. <https://link.springer.com/book/10.1007/978-3-319-42716-4>
2. Hendrikx, M., Meijer, S., Van Der Velden, J., & Iosup, A. (2013). Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications*. <https://dl.acm.org/doi/10.1145/2422956.2422957>
3. Zavorodnia, H. A., & Zavorodnii, V. V. (2026). Optyimizatsiia produktyvnosti komputernykh ihor na osnovi metodiv mashynnoho navchannia. *Systemy ta tekhnologii*, 71(1), 45–51. <https://doi.org/10.32782/2521-6643-2026-1-71.6>
4. Summerville, A., et al. (2018). Procedural content generation via machine learning: A survey. *IEEE Transactions on Games*. <https://doi.org/10.1109/TG.2018.2846639>
5. Khalifa, A., & Togelius, J. (2020). PCG via reinforcement learning. *arXiv preprint arXiv:2001.09212*. <https://doi.org/10.48550/arXiv.2001.09212>
6. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press. <https://www.deeplearningbook.org/>
7. Zavorodnia, H. A., & Zavorodnii, V. V. (2025). Metodolohiia rozroblennia systemy adaptivnoi henearatsii kontentu. *Tekhnologii ta inzhyniring*, 26(5), 21–30. <https://doi.org/10.30857/2786-5371.2025.5.2>
8. Yannakakis, G. N., & Togelius, J. (2018). *Artificial intelligence and games*. Springer. <https://link.springer.com/book/10.1007/978-3-319-63519-4>

- 
9. Zavgorodnii, V. V., Valiavska, N. O., Adamenko, V. S., Dorohovtsev, E. V., Nesmachnyi, P. V. (2022). Method avtomatychnoi henearatsii kontentu na osnovi protseduralnykh alhorytmiv. *Vcheni zapysky TNU*, 33(72), 91–96. <https://doi.org/10.32838/2663-5941/2022.1/15>
  10. Radford, A., et al. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*. <https://doi.org/10.48550/arXiv.2005.14165>
  11. Zavgorodnia, H. A., & Zavgorodnii, V. V. (2025). Rozrobka mashtabovanoi rozpodilenoj arkhitektury dlia MMO-system. *Visnyk KhNTU*, 4(95), 99–106. <https://doi.org/10.35546/kntu2078-4481.2025.4.3.11>
  12. Browne, C., et al. (2012). A survey of Monte Carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*. <https://doi.org/10.1109/TCIAIG.2012.2186810>
  13. Volz, V., Schrum, J., Liu, J., Lucas, S. M., Smith, A. D., & Risi, S. (2018). Evolving Mario levels in the latent space of a deep convolutional generative adversarial network. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '18)*, Kyoto, Japan, 221–228. <https://arxiv.org/abs/1805.00728>
  14. Silver, D., et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529, 484–489. <https://doi.org/10.1038/nature16961>
  15. Goodfellow, I. (2014). Generative adversarial networks. *arXiv preprint arXiv:1406.2661*. <https://doi.org/10.48550/arXiv.1406.2661>
  16. Liapis, A., Yannakakis, G., & Togelius, J. (2013). Sentient Sketchbook: Computer-aided game level authoring. *IEEE Transactions on Games*. <https://doi.org/10.1109/TCIAIG.2013.2297681>
  17. Snodgrass, S., & Ontañón, S. (2017). Learning to generate video game maps using Markov models. *IEEE Transactions on Computational Intelligence and AI in Games*, 9(4), 410–422. <https://doi.org/10.1109/TCIAIG.2016.2623560>
  18. Smith, G., Whitehead, J., & Mateas, M. (2011). Tanagra: Reactive planning and constraint solving for mixed initiative level design. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3), 201–215. <https://doi.org/10.1109/TCIAIG.2011.2159716>
  19. 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*. <https://doi.org/10.1016/j.entcom.2024.100759>
  20. Risi, S., & Preuss, M. (2020). From Chess and Atari to StarCraft and beyond: How game AI is driving the world of AI. *Künstliche Intelligenz*, 34(1), 7–17. <https://doi.org/10.1007/s13218-020-00647-w>
  21. Qi, Z. (2025). Research on the application of generative artificial intelligence in games. *ACE – Advances in Computer Engineering and Education*. <https://doi.org/10.54254/2755-2721/2025.18827>
  22. Vaswani, A., et al. (2017). Attention is all you need. *arXiv preprint arXiv:1706.03762*. <https://doi.org/10.48550/arXiv.1706.03762>
  23. Dormans, J. (2010). Adventures in level design: Generating missions and spaces for action adventure games. In *Proceedings of the 2010 Workshop on Procedural Content Generation in Games (PCG'10)*, 1–8. <https://doi.org/10.1145/1814256.1814257>
  24. Pathak, D., et al. (2017). Curiosity-driven exploration by self-supervised prediction. *arXiv preprint arXiv:1705.05363*. <https://doi.org/10.48550/arXiv.1705.05363>
  25. Sutton, R., & Barto, A. (2018). *Reinforcement learning: An introduction*. MIT Press. <http://incompleteideas.net/book/the-book.html>

Дата першого надходження статті до видання: 26.03.2026

Дата прийняття статті до друку після рецензування: 19.04.2026

Дата публікації (оприлюднення) статті: 30.05.2026