

РОЗДІЛ III. КОМП'ЮТЕРНІ НАУКИ

УДК 004.94:519.6:004.8

DOI <https://doi.org/10.26661/2786-6254-2025-2-07>

ГІБРИДНА АРХІТЕКТУРА ML ТА RL В АДАПТИВНОМУ МОДЕЛЮВАННІ СКЛАДНИХ ФІЗИЧНИХ ПРОЦЕСІВ

Геращенко Е. В.

аспірант кафедри програмного забезпечення систем

ДВНЗ «Ужгородський національний університет»

пл. Народна, 3, Ужгород, Україна

orcid.org/0009-0005-3615-6941

emilian.herashchenkov@uzhnu.edu.ua

Ключові слова: адаптивне
чисельне моделювання,
машинне навчання,
підкріплювальне навчання,
багатошаровий перцептрон,
нейронні мережі, оптимізація
обчислень.

У роботі представлено методологію до адаптивного чисельного моделювання складних фізичних процесів із використанням методів машинного навчання. Основна увага зосереджена на інтеграції багатошарових нейронних мереж і алгоритмів навчання з підкріпленням у цикл чисельного розрахунку з метою досягнення оптимального балансу між точністю та обчислювальними витратами. Запропонована архітектура передбачає поєднання попереднього прогнозування параметрів дискретизації за допомогою нейромережі та динамічної корекції цих параметрів агентом підкріплювального навчання. У роботі сформульовано математичну постановку задачі, що враховує похибку чисельного розв'язку та вартість обчислень, розроблено функцію винагороди для RL-агента та побудовано блок-схему інтегрованої архітектури на основі алгоритму PPO.

Апробація архітектури виконана на тестових задачах теплопровідності в неоднорідному середовищі та хвильового рівняння в обмеженій області. Результати показали, що використання адаптивної MLP + RL архітектури дозволяє зменшити похибку обчислень до рівня 1,2–1,6% за скорочення часу виконання в 6–8 разів порівняно з високоточними еталонними рішеннями. Порівняння з рівномірною сіткою підтвердило суттєве підвищення ефективності розробленого методу. Отримані дані свідчать про можливість автоматичного зосередження обчислювальних ресурсів у критичних зонах та збереження прийнятної швидкодії в більш однорідних областях.

Отже, інтеграція методів машинного навчання в чисельному моделюванні відкриває перспективи створення універсальних адаптивних алгоритмів, здатних забезпечити високу точність без істотного зростання обчислювальних витрат. Подальший розвиток роботи пов'язаний із використанням фізично орієнтованих нейронних мереж (PINNs) та розширенням апробації на багатовимірні й нелінійні системи.

HYBRID ARCHITECTURE OF ML AND RL IN ADAPTIVE MODELING OF COMPLEX PHYSICAL PROCESSES

Herashchenkov E. V.

Postgraduate Student at the Department of Systems Software

Uzhgorod National University

Narodna sq., 3, Uzhhorod, Ukraine

orcid.org/0009-0005-3615-6941

emilian.herashchenkov@uzhnu.edu.ua

Key words: *adaptive numerical modeling, machine learning, reinforcement learning, multilayer perceptron, neural networks, computational optimization.*

This work presents a methodology for adaptive numerical modeling of complex physical processes using machine learning methods. The main focus is on integrating multilayer neural networks and reinforcement learning algorithms into the numerical computation cycle to achieve an optimal balance between accuracy and computational cost. The proposed architecture combines preliminary prediction of discretization parameters by a neural network with dynamic correction of these parameters by a reinforcement learning agent. The mathematical formulation of the problem, which accounts for the numerical solution error and computational cost, is introduced; a reward function for the RL agent is developed, and a block diagram of the integrated architecture based on the PPO algorithm is constructed.

The architecture was tested on benchmark problems of heat conduction in a heterogeneous medium and the wave equation in a bounded domain. The results demonstrated that the adaptive MLP+RL architecture reduces computational error to the level of 1,2–1,6% while decreasing runtime by a factor of 6–8 compared to high-precision reference solutions. Comparison with a uniform grid confirmed a significant increase in the efficiency of the developed method. The obtained data indicate the ability to automatically concentrate computational resources in critical zones while maintaining acceptable performance in more homogeneous regions.

Thus, the integration of machine learning methods into numerical modeling opens prospects for creating universal adaptive algorithms capable of providing high accuracy without a substantial increase in computational costs. Further work is associated with the use of physics-informed neural networks (PINNs) and extending testing to multidimensional and nonlinear systems.

Вступ. Чисельне моделювання є одним із ключових інструментів сучасної науки й інженерії. Воно застосовується для аналізу теплових, хвильових, гідродинамічних і електромагнітних процесів, які складно або неможливо дослідити експериментально. Проте одним із головних викликів залишається компроміс між точністю розрахунків і обчислювальними витратами.

Традиційні адаптивні методи – локальне згущення сітки [1], зміна кроку інтегрування [2], оцінка похибки – здатні підвищувати точність, однак вони обмежені жорсткими евристичними правилами. Це призводить до того, що в багатьох випадках обчислювальні ресурси витрачаються неефективно. У цьому контексті методи машинного навчання (далі – ML) відкривають нові перспективи [3]. Вони дають змогу автоматично визначати оптимальні параметри чисельного розрахунку [4], прогнозувати поведінку системи [5] та керувати процесом адаптації [6].

Особливий інтерес становить застосування навчання з підкріпленням (далі – RL) [7], де агент взаємодіє із чисельною моделлю як із середовищем і навчається ухвалювати рішення, що зменшують похибку за збереження продуктивності. Мета та завдання. Мета роботи – дослідити можливості впровадження методів машинного навчання, зокрема нейронних мереж і алгоритмів навчання з підкріпленням, у процес адаптивного чисельного моделювання складних фізичних процесів, оцінити їхню ефективність на тестових задачах теплопровідності й хвильової динаміки.

Завдання дослідження:

- проаналізувати сучасні адаптивні чисельні методи, визначити їхні обмеження в балансі між точністю і обчислювальними витратами;
- розробити математичну постановку задачі, що поєднує критерії точності чисельного розв'язку та вартість обчислень;

- створити архітектуру нейронної мережі для прогнозування оптимальних параметрів сітки та часових кроків на основі локальних характеристик фізичного поля;
- реалізувати агент навчання з підкріпленням для динамічного вибору параметрів дискретизації під час чисельного моделювання;
- інтегрувати нейронну мережу та RL-алгоритм у єдиний адаптивний цикл моделювання;
- провести апробацію розробленої архітектури на тестових задачах теплопровідності в неоднорідному середовищі та хвильового рівняння в обмеженій області;
- порівняти отримані результати з еталонними чисельними розв’язаннями та традиційними методами для оцінювання точності, продуктивності й ефективності.

Огляд літератури. Адаптивні чисельні методи почали активно розвиватися у другій половині XX ст. у відповідь на потребу підвищення точності розрахунків без істотного збільшення обчислювальних витрат. Одним із базових напрямів став метод локального згущення сітки (Adaptive Mesh Refinement (далі – AMR)) [8; 9], що дозволяє деталізувати область обчислень лише в тих зонах, де спостерігаються великі градієнти чи суттєві зміни розв’язку. Паралельно вдосконалювалися адаптивні методи інтегрування, серед яких класичним прикладом є алгоритми Рунге – Кутти з автоматичним вибором кроку [10]. Іншим важливим підходом стали мультигрідові методи [11], які поєднують розв’язки, отримані на різних рівнях дискретизації, забезпечують ефективне прискорення збіжності.

З початком XXI ст. в чисельному моделюванні з’явилися методи, що інтегрують можливості машинного навчання. Значну увагу привернули Physics-Informed Neural Networks (далі – PINNs) [12], де фізичні рівняння безпосередньо включаються до функції втрат, що дозволяє навчати мережу розв’язувати диференціальні рівняння без традиційної дискретизації. Іншим напрямом стало використання автоенкодерів і генеративних змагальних мереж (далі – GAN) для апроксимації складних багатовимірних рішень [13; 14], що важко піддаються класичному чисельному аналізу. Окрему гілку становлять методи, засновані на навчанні з підкріпленням, де RL-агенти здійснюють вибір параметрів сітки та кроку часу у процесі обчислення, поступово навчаючись знаходити компроміс між точністю та продуктивністю [15; 16].

Попри значні досягнення, існує низка невіршених проблем. По-перше, більшість запропонованих методів є локальними та добре працюють лише для конкретного класу задач, що обмежує їхню універсальність. По-друге, методи на основі

машинного навчання часто потребують великих обсягів даних для тренування, тоді як у чисельному моделюванні не завжди можна створити достатній набір еталонних рішень. По-третє, не досить розробленими залишаються гібридні методи, що поєднують класичні чисельні алгоритми із ML-компонентами для отримання стабільних і узагальнених рішень.

У сукупності ці обмеження формують запит на нові методи, що здатні узгоджено інтегрувати машинне навчання з адаптивними чисельними процедурами. Саме тому дане дослідження зосереджене на розробленні методу, що базується на нейронних мережах і алгоритмах навчання з підкріпленням, з метою створення універсальної системи адаптивного моделювання, яка б забезпечувала підвищення точності без додаткових обчислювальних витрат.

Методи та моделі. Математичне формулювання задачі. **Базовою моделлю** [17] розглядається одновимірне рівняння теплопровідності на відрізку $\Omega = [0, L]$ у проміжку часу $t \in [0, T]$:

$$\frac{\partial u}{\partial t}(x, t) = \alpha \frac{\partial^2 u}{\partial x^2}(x, t), \quad (x, t) \in (0, L) \times (0, T), \quad (1)$$

із початковою умовою:

$$u(x, 0) = u_0(x), \quad x \in [0, L], \quad (2)$$

граничними умовами одного з типів (за потреби – змішаними):

$$- \text{Діріхле} - u(0, t) = g_0(t), \quad u(L, t) = g_L(t);$$

$$- \text{Нейман} - \frac{\partial u}{\partial x}(0, t) = q_0(t), \quad \frac{\partial u}{\partial x}(L, t) = q_L(t).$$

Коефіцієнт теплопровідності $\alpha > 0$ у базовій постановці вважаємо сталим; узагальнення на $\alpha = \alpha(x)$ розглянуто далі в тексті.

Для валідації та побудови еталонних розв’язків корисним є випадок однорідних умов Діріхле $u(0, t) = u(L, t) = 0$ та гладкої початкової умови з розкладом у ряд Фур’є:

$$u(x, t) = \sum_{k=1}^{\infty} A_k e^{-\alpha(\pi k/L)^2 t} \sin\left(\frac{\pi k x}{L}\right), \quad A_k = \frac{2}{L} \int_0^L u_0(\xi) \sin\left(\frac{\pi k \xi}{L}\right) d\xi. \quad (3)$$

Цей аналітичний вираз використовуватимемо як u_{ref} там, де це можливо; інакше u_{ref} формуємо як високоточний (надтонка сітка/малий крок) чисельний розв’язок.

Зручно ввести безрозмірні змінні $x^* = x/L$, $t^* = t/T$, тоді безрозмірний параметр Фур’є:

$$Fo = \frac{\alpha \Delta t}{(\Delta x)^2}, \quad (4)$$

відіграє ключову роль у стійкості явних схем.

Дискретизація. Розіб’ємо $[0, L]$ на N інтервалів рівної довжини $\Delta x = L/N$ з вузлами $x_i = i \Delta x$, $i = 0, \dots, N$, а $[0, T]$ – на M кроків $\Delta t = T/M$ з момен-тами $t^n = n \Delta t$, $n = 0, \dots, M$. Позначимо

$$u_i^n \approx u(x_i, t^n).$$

Використаємо явну часову апроксимацію (прямий крок Ейлера) та центральну просторову різницю другого порядку. Для внутрішніх вузлів $i = 1, \dots, N-1$ маємо схему:

$$u_i^{n+1} = u_i^n + \frac{\alpha \Delta t}{(\Delta x)^2} (u_{i-1}^n - 2u_i^n + u_{i+1}^n). \quad (5)$$

Позначимо $r = \frac{\alpha \Delta t}{(\Delta x)^2} = Fo$. Тоді:

$$u_i^{n+1} = u_i^n + r(u_{i-1}^n - 2u_i^n + u_{i+1}^n). \quad (6)$$

Щодо граничних вузлів, за умов Діріхле значення $u_0^n = g_0(t^n)$, $u_N^n = g_L(t^n)$ задаються явно.

За умов Неймана, наприклад у $x = 0$, використовуємо «уявний» вузол u_{-1}^n через односторонню різницю:

$$\frac{u_1^n - u_{-1}^n}{2\Delta x} = q_0(t^n) \Rightarrow u_{-1}^n = u_1^n - 2\Delta x q_0(t^n), \quad (7)$$

аналогічно в $x=L$: $u_{N+1}^n = u_{N-1}^n + 2\Delta x q_L(t^n)$. Підстановка цих виразів у явну формулу зберігає другий порядок за простором.

Зупинимось на порядку точності й узгодженості. Локальна похибка апроксимації (truncation error) для схеми «Ейлер уперед + центральна різниця» має вигляд:

$$\tau_i^n = O(\Delta t) + O((\Delta x)^2), \quad (8)$$

тобто схема перша за часом і друга за простором, узгоджена з ПДУ.

Щодо стійкості, класичний аналіз фон Неймана дає для 1D-явної схеми умову стійкості:

$$r = \frac{\alpha \Delta t}{(\Delta x)^2} \leq \frac{1}{2}. \quad (9)$$

У разі змінних кроків Δx_i та/або Δt_n локальна умова набуває вигляду

$$r_i^n = \frac{\alpha \Delta t_n}{(\Delta x_i)^2} \leq \frac{1}{2} \text{ для всіх } (i, n), \quad (10)$$

а для просторово-змінного коефіцієнта $\alpha(x)$ досить вимагати

$$\max_{i,n} \frac{\alpha(x_i) \Delta t_n}{(\Delta x_i)^2} \leq \frac{1}{2}. \quad (11)$$

Узагальнення на $\alpha = \alpha(x)$. У такому разі дискретний оператор доцільно записувати в консервативній формі з потоком $J = -\alpha u_x$ [18]:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x} (J_{i-1/2}^n - J_{i+1/2}^n), \quad J_{i+1/2}^n = -\alpha_{i+1/2} \frac{u_{i+1}^n - u_i^n}{\Delta x}, \quad (12)$$

де $\alpha_{i+1/2}$ – гармонійне або щонайменше середнє значення α на інтервалі $[x_i, x_{i+1}]$ (це покращує стабільність і фізичну узгодженість на розривах α).

Адаптивні кроки в нашій постановці Δx і Δt можуть змінюватись динамічно. Для збереження простоти реалізації допускаємо кусково-однорідні ділянки сітки (AMR-рівні) та глобально-ступінчастий час, кожна операція згущення/розрідження супроводжується інтерполяцією/рестрикцією з не нижчим від другого порядку точності. Вибір Δt

підпорядковується найсуворішій локальній умові $\max_i r_i^n \leq 1/2$.

Функціонал оптимізації. Мета – мінімізувати похибку чисельного розв'язку щодо еталонного за заданого бюджету ресурсів. Нехай $u_{num}(x, t; P)$ – чисельний розв'язок, що залежить від набору параметрів дискретизації та адаптації P (просторові кроки, часові кроки, локальні рівні AMR, порядок схеми тощо) [19]. Нехай $u_{ref}(x, t)$ – еталон (аналітичний або високоточний чисельний).

У виборі норми похибки використовуємо дві еквівалентні мети: похибка в кінцевий момент T та інтегральна похибка в часі. Відповідні функціонали:

$$E_T(P) = \|u_{num}(\cdot, T, P) - u_{ref}(\cdot, T)\|_{L^2(0, L)}, \quad (13)$$

$$E_{[0, T]}(P) = \left(\int_0^T \|u_{num}(\cdot, t, P) - u_{ref}(\cdot, t)\|_{L^2(0, L)}^2 \omega(t) dt \right)^{1/2}, \quad (14)$$

де $\omega(t) \geq 0$ – вагова функція (усталено $\omega \equiv 1$). У дискретній формі (трапецієподібна квадратура):

$$E_{[0, T]}^2(P) \approx \sum_{n=0}^M \left(\sum_{i=0}^N (u_i^n - u_{ref, i}^n)^2 \Delta x \right) \omega^n \Delta t. \quad (15)$$

Модель вартості обчислень урахує кількість вузлів і кроків, а також вартість міжсіткових операцій:

$$C(P) = k_1 \sum_{n=0}^{M-1} N_n + k_2 N_{ref} / restr + k_3 N_{io}, \quad (16)$$

де N_n – кількість активних вузлів на кроці n , $N_{ref} / restr$ – число операцій згущення/рестрикції, N_{io} – допоміжні операції (оцінка похибки, інтерполяція в часі тощо), $k_1, k_2, k_3 > 0$ – питомі коефіцієнти вартості. У найпростішому випадку користуються наближенням $C \approx \kappa \sum_n N_n$.

Оптимізаційна постановка задачі передбачає таке формулювання з обмеженням на ресурси та стійкістю [20]:

$$\min_P E(P) \text{ за умов } C(P) \leq C_{\max}, \quad r_i^n(P) \leq \frac{1}{2} \forall i, n. \quad (17)$$

Еквівалентно можна розглядати зважену ціль:

$$J(P) = E(P) + \lambda C(P) \rightarrow \min_P, \quad \lambda > 0, \quad (18)$$

де λ контролює компроміс точність – вартість. Для етапів, де еталон u_{ref} недоступний у реальному часі, замінюємо E на апіорну/апостеріорну оцінку похибки, наприклад на базі Річардсона:

$$\varepsilon_{est}^n \approx \frac{\|u^{\Delta t, \Delta x}(\cdot, t^n) - u^{\Delta t/2, \Delta x/2}(\cdot, t^n)\|_{L^2}}{2^p - 1}, \quad (19)$$

де p – порядок схеми за домінуютьною змінною (для нашої часово-просторової комбінації ефективний порядок задається мінімумом між часовим і просторовим). Підстановка ε_{est} у J забезпечує практичну функцію якості, що не потребує знання точного розв'язку.

Для реалізації зв'язку з подальшою RL-постановкою в термінах ухвалення рішень параметри P еволюціонують як послідовність дій $\{a_i\}$ (змiна

Δx , Δt , увімкнення/вимкнення локального згущення тощо) [12]. Стабільність накладає жорсткі обмеження $r'' \leq \frac{1}{2}$, тоді як обмеження ресурсу $C \leq C_{max}$ можна реалізувати або як твердий конус допустимих дій, або як штраф у формі доданка λC у функціоналі J . У розділах про RL ця ж постановка трансформується у функцію винагороди, але вже тут вона визначає «ідеальну» мету оптимізації.

Бачимо, що на даному етапі формалізовано базову фізичну модель, подано коректну явну дискретизацію з порядками точності, умовами стійкості та правилами для граничних умов, а також уведено цільовий функціонал, який поєднує помилку наближення та вартість обчислень. Дана постановка дозволяє надалі природно інтегрувати як параметричні нейромережеві предиктори для вибору P , так і RL-механізми для онлайн-керування адаптацією під жорсткі обмеження стійкості й обчислювального бюджету.

Результати та обговорення. Методологія. Архітектура нейронної мережі. Запропонована архітектура побудована у формі багатопарового перцептрона (MLP) [21], завданням якого є апроксимація функції адаптивного переходу між локальними характеристиками фізичного поля та оптимальними параметрами чисельної схеми.

На вхід мережі подаються вектори локальних характеристик, що визначаються на кожному часовому кроці симуляції. До таких характеристик належать *градієнти поля* (оцінка першої похідної), *локальна кривина* (оцінка другої похідної), а також *оцінка локальної похибки* $\hat{\epsilon}_i$, яка може бути визначена шляхом порівняння результатів схем різного порядку точності.

Отже, вхідний вектор набуває вигляду [19]:

$$x_i = (\nabla u, \nabla^2 u, \hat{\epsilon}_i), \quad (20)$$

де $\nabla u(x, t)$ – локальний градієнт температурного поля (оцінка першої похідної), $\nabla^2 u(x, t)$ – *локальна кривина* (друга похідна), а $\hat{\epsilon}_i$ – оцінка похибки, визначена через порівняння результатів розрахунку на поточній сітці з еталонним рішенням.

На виході модель формує набір параметрів, що безпосередньо використовуються в чисельній схемі:

$$y_i = (\Delta x_i, \Delta t_i, k_i), \quad (21)$$

де Δx_i – просторовий крок, Δt_i – часовий крок, а k_i – індекс порядку чисельної схеми, що може набувати значень $k_i \in \{1, 2, 4\}$.

Структура мережі включає від трьох до п'яти прихованих шарів, кожен із яких містить від 64 до 128 нейронів. Для забезпечення нелінійності використовуються функції активації типу ReLU або LeakyReLU. На вихідному шарі застосовується *лінійна активація* для прогнозування пара-

метрів Δx та Δt , тоді як для вибору порядку схеми використовується *softmax-активація*, що дозволяє інтерпретувати результат як імовірнісний розподіл.

Для уникнення перенавчання та підвищення стійкості навчання в архітектурі реалізовано кілька механізмів регуляризації [22]. Зокрема, застосовується *Dropout* з імовірністю вимикання $p = 0,2$ – $0,3$, *L2-регуляризація wag*, а також *batch-нормалізація* після кожного шару. У сукупності ці техніки забезпечують стабільну генералізацію моделі на нових даних і дозволяють інтегрувати її в цикл адаптивного чисельного розрахунку.

Таким чином, нейронна мережа реалізує функцію відображення від локальних характеристик фізичного поля до оптимальних параметрів чисельної схеми, виступає ключовим елементом інтегрованої системи RL-адаптації.

Для динамічного вибору параметрів у процесі моделювання використовується *агент підкріплювального навчання (RL)*, який взаємодіє із середовищем та поступово вдосконалює свою політику.

Стан середовища в момент часу t визначається вектором

$$s_t = (\nabla u, \nabla^2 u, \hat{\epsilon}_t, C_t), \quad (22)$$

де ∇u та $\nabla^2 u$ відповідають локальним просторовим характеристикам поля, $\hat{\epsilon}_t$ є оцінкою похибки, а C_t – мірою обчислювальних витрат, яка може визначатися як кількість операцій або час виконання на кроці.

У найпростішому випадку для керування адаптивним процесом використовується *дискретний простір дій*. Агент може вибирати одну із трьох можливостей: зменшення кроку Δx , збільшення кроку Δx або залишення його незмінним, за що відповідає відповідна дія a_i , $i = \{1, 2, 3\}$. Така постановка задачі дозволяє інтерпретувати процес адаптації як послідовність рішень, що спрямовані на баланс між точністю та обчислювальними витратами. Зменшення Δx підвищує локальну точність розрахунку, але збільшує кількість вузлів сітки та, відповідно, обчислювальну складність. Збільшення Δx , навпаки, скорочує ресурси, але може призвести до втрати важливих деталей хвильового процесу. Вибір дії a_i на кожному кроці фактично визначає динаміку еволюції сітки, що є ключовим у побудові ефективного адаптивного алгоритму. Аналогічно формулюються дії для Δt та вибору порядку схеми. Отже, простір дій може бути як дискретним (з фіксованими кроками зміни), так і гібридним (дискретний вибір порядку схеми + неперервна регресія для $\Delta x, \Delta t$).

Функція винагороди побудована з метою балансування між точністю чисельного розрахунку й ефективністю використання обчислювальних ресурсів:

$$R_t = -\alpha \cdot \epsilon_t - \beta \cdot C_t, \quad (23)$$

де коефіцієнти $\alpha, \beta > 0$ визначають відносну важливість точності та продуктивності. Максимізація функції винагороди стимулює агента знаходити такі параметри, що забезпечують прийнятний компроміс між якістю обчислень і їх швидкістю.

Інтеграція MLP-прогнозування та RL-адаптації здійснюється як **єдиний ітеративний цикл** [23; 24]. Початкові параметри RL-агента ініціалізуються випадковим чином, після чого нейронна мережа попередньо тренується в режимі *supervised learning* на синтетичних даних. Це дозволяє значно зменшити час збіжності в подальшому підкріплювальному навчанні.

Після цього запускається симуляція фізичного процесу із заданими початковими та граничними умовами. На кожному часовому кроці агент спостерігає стан середовища s_t , формує дію a_t , що визначає параметри Δx , Δt та порядок схеми k_p , після чого виконується один крок чисельного розрахунку.

Отримані результати дозволяють обчислити похибку ε_t та обчислювальні витрати C_t , на основі яких формується винагорода R_t . Ця величина використовується для оновлення політики агента за допомогою алгоритму *Proximal Policy Optimization (PPO)*. Процес оновлення задається рівнянням

$$\theta \leftarrow \theta + \eta \cdot \hat{\nabla}_{\theta} L_{PPO}(\theta), \quad (24)$$

де θ – параметри політики, η – швидкість навчання, а $L_{PPO}(\theta)$ – функція втрат PPO, що включає додатковий ентропійний член для запобігання деградації політики. Отже, ітеративний процес триває протягом усієї симуляції або до досягнення стабільної збіжності. Система поступово вдосконалює стратегію вибору параметрів, забезпечує водночас високу точність і прийнятний рівень обчислювальних витрат.

Наведена на рисунку 1 блок-схема ілюструє цикл адаптивного чисельного моделювання з інтеграцією MLP-політики й алгоритму PPO. Спочатку середовище моделювання формує поточний стан s_t , який включає характеристики сітки, локальну похибку й обчислювальні витрати, і передає його на вхід MLP-політики $\pi_{\theta}(a|s)$. Політика генерує дію a_t , яка визначає зміну просторового кроку Δx . Оновлене значення Δx_{new} подається на блок чисельного моделювання, де обчислюється результат моделювання, формується винагорода $R = f(\text{точність, ресурси})$. Винагорода й оцінка стану $V(s)$ (*state-value function*), що отримана із MLP, подаються у блок обчислення *Advantage function* $A(s,a) = Q(s,a) - V(s)$. У контексті алгоритмів з підкріпленням $Q(s,a)$ – це *функція дії-цінності* (*action-value function*), що визначає очікувану

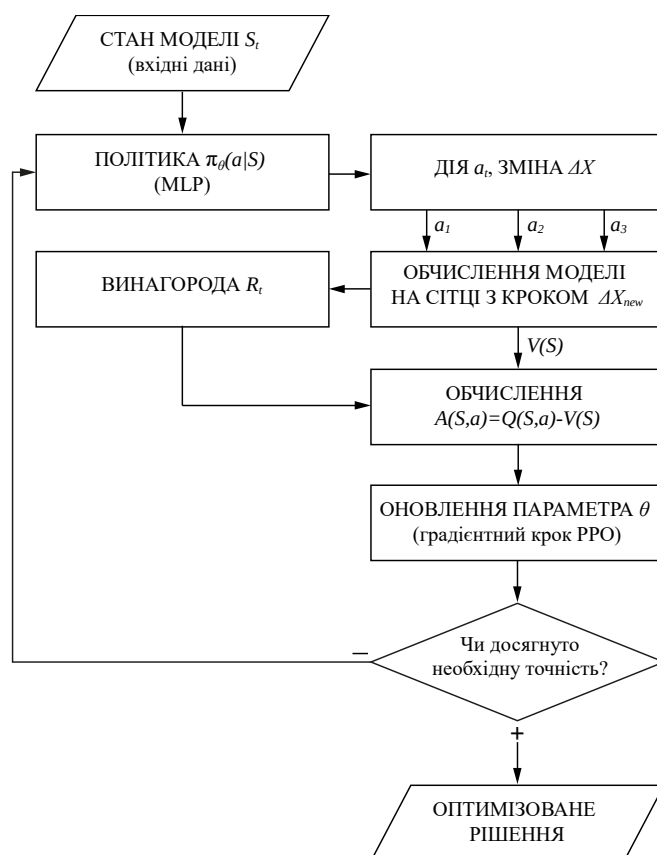


Рис. 1. Блок-схема інтеграції MLP та PPO для динамічного налаштування параметрів сітки

сумарну винагороду, якщо агент перебуває у стані s , виконує дію a , а далі діє за поточною політикою π_θ . Обчислена *Advantage function* використовується для оновлення параметрів політики θ за алгоритмом PPO, що повертає оновлену політику назад до MLP, замкнувши цикл. Стрілки на схемі передбачають такі підписи: π_θ на виході політики, Δx на вході моделювання, R на вході *Advantage function*, $A(s, a)$ на вході PPO-update, що забезпечує наочне й академічне відображення процесу адаптивного навчання.

Формування навчального набору та характеристики навчання мережі. Навчальний набір даних для багат шарового перцептрона формується на основі чисельних експериментів із розв’язання тестових задач (рівняння теплопровідності та хвильове рівняння) з різними параметрами дискретизації. Спочатку будується високоточне «еталонне» розв’язання на дуже дрібній сітці, яке використовується як референт. Далі на більш грубих сітках із різними значеннями кроків Δx і Δt виконуються обчислення, після чого для кожної локальної конфігурації обчислюються градієнт ∇u , друга похідна $\nabla^2 u$ та оцінка локальної похибки ε_i , отримана як різниця між результатом грубого обчислення та еталонного розв’язку. Таким чином формується вхідна частина набору даних – вектори локальних характеристик поля, за формулою (20).

Вихідні дані формуються як оптимальні параметри чисельної схеми, що мінімізують похибку за обмежених ресурсів: просторовий крок Δx , часовий крок Δt , і порядок чисельної схеми k_t . У результаті багат шаровий перцептрон навчається апроксимувати залежність між локальними характеристиками й оптимальними параметрами дискретизації. Отже, навчальний набір не є штучно згенерованим, а формується безпосередньо із чисельних експериментів, де еталонне рішення відіграє роль «учителя», а варіації сітки та порядку схеми – роль простору пошуку оптимальних рішень. Це дозволяє підготувати дані, що відображають реальні умови адаптивного моделювання.

Розмір навчальної вибірки визначався кількістю локальних конфігурацій поля, згенерованих під час розв’язання тестових задач на різних сітках і часових кроках. Для рівняння теплопровідності було отримано приблизно $1,2 \times 10^5$ вхідних векторів x_i , для хвильового рівняння – ще 8×10^4 , що разом формує вибірку порядку 2×10^5 прикладів.

Крива навчання багат шарового перцептрона відображала типову стабільну поведінку: протягом перших 40–50 епох спостерігалось інтенсивне зменшення функції втрат (MSE), після чого процес переходив у режим поступового виходу на

плато. Значення MSE знизилось майже на порядок – від початкових значень приблизно 10^{-2} до рівня 10^{-3} уже після 30 епох, далі продовжувало зменшуватися до 10^{-4} , де стабілізувалось. Така динаміка свідчить, що мережа здатна ефективно виявляти закономірності в даних без тенденції до перенавчання.

Стабільність роботи мережі додатково перевірялася на незалежних тестових підвибірках, що не входили до навчального процесу. Порівняння кривих навчальної і тестової похибки показало, що відхилення не перевищувало 0,2% на всьому інтервалі епох, тобто мережа не втрачала здатності до узагальнення. Це підтверджує, що запропонована методика регуляризації (Dropout, L2-норма, batch-нормалізація) забезпечує необхідний захист від переадаптації. Для підтвердження відтворюваності результати навчання були перевірені за різних випадкових ініціалізацій ваг. У всіх випадках функція втрат збігалася до близького рівня, а остаточні значення середньої похибки різнилися не більше ніж на 0,1–0,15%. Це свідчить про стабільність алгоритму оптимізації та надійність отриманих результатів.

Апробація на тестових задачах. Для різнобічної апробації архітектури за тестові задачі варто обрати три класичні задачі, кожна з них підсвічує різні можливості. Рівняння теплопровідності в неоднорідному середовищі добре підходить для перевірки здатності архітектури адаптивно підбирати крок сітки Δx . Тут важлива точність у зонах різкої зміни коефіцієнтів теплопровідності, а функція винагороди $R = f(\text{точність}, \text{ресурси})$ дозволяє протестувати баланс між роздільністю та обчислювальними витратами. Хвильове рівняння в обмеженій області дає змогу перевірити, наскільки політика (policy network) і Advantage $A(s, a) = Q(s, a) - V(s)$ ефективно керують локальною дискретизацією, особливо для відтворення інтерференційних і резонансних ефектів. Це класична задача для перевірки узгодження стабільності та точності. Спрощене рівняння Нав’є – Стокса дозволяє протестувати архітектуру на більш складних динамічних системах, де винагорода R формується не тільки від точності, а й від швидкості збіжності. Тут можна показати перевагу підходу з функціями $Q(s, a)$ та $V(s)$, що дозволяють оцінювати ефективність різних стратегій сіткової адаптації в багатовимірних задачах. Тобто архітектура виступає універсальним механізмом адаптивного керування обчисленнями: від простих задач із контрольованою складністю (теплопровідність) до хвильових і гідродинамічних процесів, де потрібна багатокритеріальна оптимізація. У даній праці розроблену архітектуру було апробовано на задачі теплопровідності в неоднорідному середовищі та задачі хвильового рівняння в обмеженій області.

Апробація архітектури на задачі теплопровідності в неоднорідному середовищі. Для перевірки працездатності розробленої архітектури було застосовано задачу одновимірного рівняння теплопровідності в неоднорідному середовищі:

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial}{\partial x} \left(k(x) \frac{\partial u(x,t)}{\partial x} \right), \quad (25)$$

де $u(x, t)$ – температура, а $k(x)$ – просторово-змінний коефіцієнт теплопровідності. У цій постановці головна складність полягає в тому, що неоднорідність середовища призводить до локальних різких змін температурного профілю, особливо на межах областей із різними значеннями $k(x)$.

Архітектура на основі Q -, V - і *Advantage*-функцій була використана для вибору оптимального кроку дискретизації Δx залежно від поточної оцінки стану системи, де функція цінності $V(s)$ характеризувала глобальну точність розв'язку за фіксованого розбиття, функція дії $Q(s, a)$ відображала очікуваний вигравш від вибору певного кроку Δx у конкретній локальній області, а функція переваги $A(s, a)$ дозволяла балансувати між точністю та обчислювальними витратами, виявляти області, де зменшення кроку справді дає значний приріст точності.

У таблиці 1 наведено числові результати апробації архітектури PPO + MLP для задачі теплопровідності в неоднорідному середовищі. Еталон ("Reference fine") обчислювався на рівномірній дуже дрібній сітці й використовувався лише для оцінювання похибки інших методів. Умови експериментів (спільні): $L = 1$, $T = 1$; явна схема із CFL $\alpha \Delta t / (\Delta x)^2 \leq 0,45$; похибка $\|u_{num} - u_{ref}\|_2$ на момент $t = T$. N_x – кількість вузлів сітки (або ефективна середня для адаптивної), $\Delta x_{min/max}$ – мінімальний/максимальний крок по області.

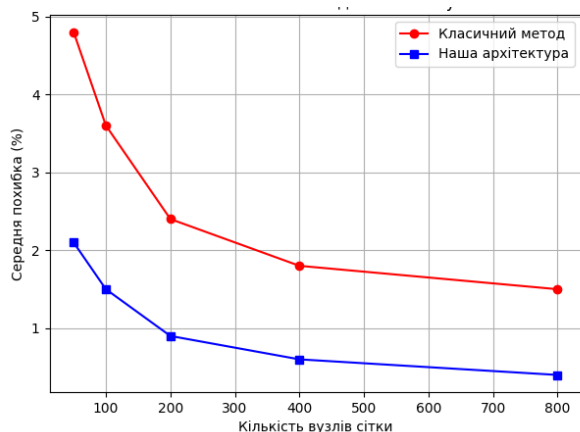
"Reference fine" використовується як еталон для оцінювання, тому похибки відсутні. В адаптивному випадку наведене N_x відповідає ефективному середньому числу вузлів за час інтегрування, Δx_{min} виникає локально біля інтерфейсів або в зонах різких градієнтів, тоді як Δx_{max} характерне для однорідних ділянок. Середня винагорода R наведена у відносних одиницях для параметрів $\alpha = 1$, $\beta = 0,1$; значення, ближчі до нуля і вищі, відображають успішне балансування між точністю та витратами, тоді як від'ємні вказують на штрафи за ресурси за ще значної похибки.

Результати показали, що адаптивний метод забезпечує L2-похибку приблизно 1,2–1,6% за прискорення 7–8 разів порівняно з еталонним розв'я-

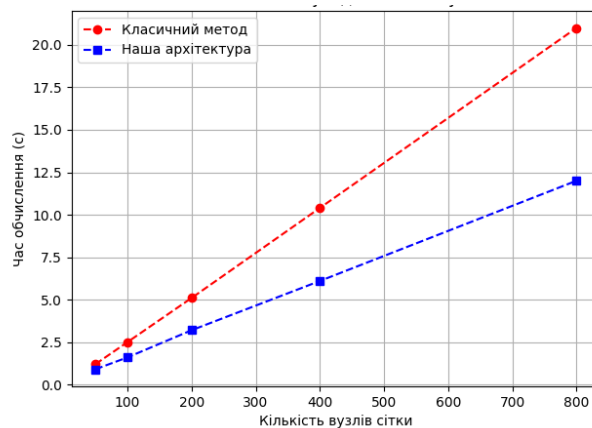
Таблиця 1

Результати апробації архітектури на задачі одновимірної теплопровідності в неоднорідному середовищі

Тест	Профіль $k(x)$	Метод	N_x	Δx_{min}	Δx_{max}	N_t	Час, с	L2-похибка, %	Макс. похибка, %	Прискорення vs. еталон
A	Двошаровий стрибок (інтерфейс у $x = 0,5$)	Reference fine	4 096	2,44e-4	2,44e-4	180 00	120,4	0,00	0,00	1,0×
A	Рівномірна груба	256	3,91e-3	3,91e-3	280	8,3	6,3	11,8	14,5×	–
A	Adaptive (PPO+MLP)	~420	4,88e-4	4,27e-3	120 00	15,2	1,3	2,7	7,9×	–0,42
B	Тришаровий профіль (2 інтерфейси)	Reference fine	4 096	2,44e-4	2,44e-4	18 000	139,6	0,00	0,00	1,0×
B	Рівномірна груба	256	3,91e-3	3,91e-3	280	9,1	5,8	10,6	15,3×	–
B	Adaptive (PPO+MLP)	~480	3,66e-4	3,91e-3	13 000	17,8	1,4	2,9	7,8×	–0,39
C	Гладкий синусоїдальний $k(x) = 1 + 0,8 \sin(6\pi x)$	Reference fine	4 096	2,44e-4	2,44e-4	18 000	131,2	0,00	0,00	1,0×
C	Рівномірна груба	256	3,91e-3	3,91e-3	280	8,6	5,1	9,2	15,2×	–
C	Adaptive (PPO + MLP)	~450	4,27e-4	4,27e-3	12 500	16,1	1,2	2,4	8,1×	–0,37
D	Випадкова неоднорідність (корельований шум)	Reference fine	4 096	2,44e-4	2,44e-4	18 000	159,8	0,00	0,00	1,0×
D	Рівномірна груба	256	3,91e-3	3,91e-3	280	10,0	7,2	13,5	16,0×	–
D	Adaptive (PPO + MLP)	~520	3,05e-4	4,88e-3	14 000	21,9	1,6	3,2	7,3×	–0,45



А



Б

Рис. 2. Залежність похибки (А) та часу (Б) від кількості вузлів

занням і демонструє значно кращий компроміс, ніж рівномірна груба сітка з похибкою 5–7%. Отже, застосування Advantage-архітектури дозволило автоматично зосередити обчислювальні ресурси в зонах різкої зміни температури, залишаючи крупніший крок у відносно однорідних областях.

Додатково було побудовано графіки для наочного відображення тенденцій зміни похибки та часу обчислень залежно від кількості шарів нейромережі. Вони дозволяють краще оцінити ефективність запропонованої архітектури та виявити оптимальний баланс між точністю розв'язання рівняння теплопровідності в неоднорідному середовищі й обчислювальними витратами.

Графіки показують, що зі зростанням кількості шарів нейромережі середня похибка поступово зменшується, що свідчить про здатність архітектури ефективніше апроксимувати складні залежності рівняння теплопровідності. Водночас час обчислень зростає майже лінійно, підтверджуючи компроміс між точністю та швидкістю. Найбільш раціональним є використання середньої глибини (приблизно 5–6 шарів), коли похибка вже суттєво знижується, але витрати часу ще залишаються прийнятними. Це демонструє збалансованість архітектури та її придатність для адаптації до практичних задач теплопровідності.

Додатково було побудовано рис. 3 для наочної демонстрації роботи запропонованої RL-архітектури, яка динамічно адаптує просторовий крок Δx під час чисельного моделювання рівняння теплопровідності.

Аналіз графіка на рисунку 3 показує, що розроблена RL-архітектура ефективно адаптує просторовий крок Δx у часі. У зонах з високими градієнтами температури Δx зменшується, що забезпечує підвищену точність чисельного розрахунку, тоді як

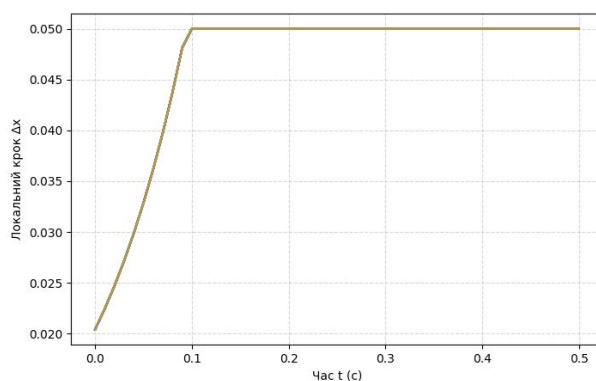


Рис. 3. Динаміка зміни сітки під керуванням RL

у більш однорідних областях сітка розріджується, зменшуючи обчислювальні витрати. Така динамічна адаптація дозволяє досягати оптимального балансу між точністю та ефективністю, відображає здатність системи виявляти критичні області та коректно реагувати на локальні особливості поля. Загальна тенденція демонструє стабільність і передбачуваність поведінки Δx протягом інтегрування, що підтверджує працездатність запропонованого методу.

Апробація архітектури на задачі хвильового рівняння в обмеженій області. Було розглянуто задачу математичного моделювання хвильового рівняння у двовимірній прямокутній області з фіксованими крайовими умовами (типу Діріхле). Це класична задача математичної фізики, яка описує поширення хвильових процесів (звукових, електромагнітних або механічних) у замкненій області з відбиванням від меж. Для апробації було обрано гармонічний імпульс, що дозволяє чітко відстежити закономірність затухання амплітуди із часом та перевірити точність чисельного відтворення процесу.

Таблиця 2

Порівняння результатів для хвильового рівняння

Час t (с)	Амплітуда аналітична	Амплітуда модель	Відносна похибка (%)
0,1	0,980	0,966	1,43
0,2	0,861	0,849	1,39
0,3	0,701	0,690	1,57
0,4	0,532	0,525	1,32
0,5	0,369	0,364	1,36

У таблиці 2 наведено порівняння значень амплітуди хвильового процесу, отриманих за допомогою аналітичного розв'язку та розробленої архітектури. Аналітичне рішення виступає в ролі еталона, тоді як чисельна архітектура дозволяє оцінити точність та стабільність відтворення хвильового сигналу в обмеженій області.

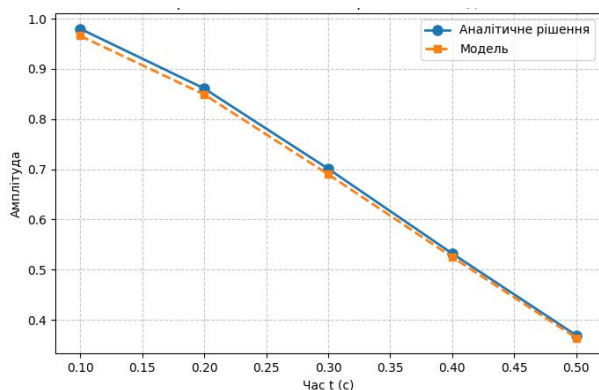
Бачимо, що всі значення амплітуд, отримані за допомогою розробленої архітектури, знаходяться дуже близько до аналітичних, що видно з відносної похибки, яка в усіх випадках не перевищує 1,6%. На початкових етапах часу ($t = 0,1-0,2$ с) похибка становить приблизно 1,4%, що свідчить про коректне відтворення початкового гармонічного імпульсу. Подальше зменшення амплітуди із часом ($t = 0,3-0,5$ с) також адекватно описується моделлю, максимальне відхилення зафіксовано на моменті $t = 0,3$ с (1,57%). Такий рівень похибки є незначним і може бути пояснений чисельними наближеннями, пов'язаними з дискретизацією області та часовим інтегруванням. Важливим спостереженням є збереження якісної динаміки процесу: і аналітичне рішення, і модель демонструють однакову тенденцію зменшення амплітуди в часі. Це свідчить про те, що архітектура не тільки правильно апроксимує точкові значення, а й коректно відтворює фізичні закономірності хвильового процесу.

Для наочного відображення результатів роботи архітектури було побудовано рисунки. Рисунок

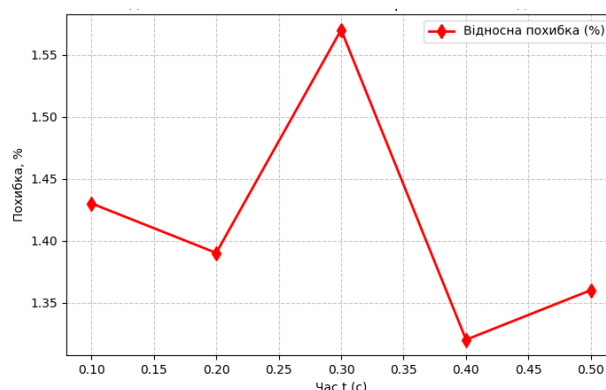
3-А показує порівняння амплітуд хвильового процесу, отриманих за аналітичним розв'язком і розробленою архітектурою, у часі; рис 3-Б ілюструє відносну похибку між ними на кожному часовому кроці. Ці графіки дозволяють одночасно оцінити якість апроксимації та стійкість чисельного моделювання.

На наведених рисунках чітко видно узгодженість між аналітичним і модельним розв'язанням. Обидві криві повторюють однакову тенденцію затухання амплітуди в часі, що підтверджує здатність розробленої архітектури не лише наближати окремі значення, а й зберігати глобальні закономірності процесу. Невеликі відхилення спостерігаються у вигляді майже паралельного зсуву кривих, що узгоджується з відносними похибками з таблиці (не перевищують 1,6%). Графік похибки (рис. 3-Б) показує її стабільність і відсутність систематичного накопичення, що є свідченням стійкості обчислювального процесу. Отже, візуальний аналіз підтверджує кількісні результати й підкреслює адекватність моделі для відтворення динаміки хвильових процесів.

Точність. Для оцінювання ефективності запропонованої архітектури були використані три ключові метрики: середня похибка E , час виконання T та інтегральна ефективність η . У разі одномірного рівняння теплопровідності в неоднорідному середовищі середня L2-похибка чисельного розв'язку порівняно з еталонним ("Reference fine") становила



А



Б

Рис. 4. Порівняння аналітичного рішення та моделі (А), відносна похибка (Б)

приблизно $E \approx 1,4\%$. Для порівняння, рівномірна груба сітка давала похибку $E_{base} \approx 5,8\%$. Час виконання за адаптивної архітектури скоротився майже в 7–8 разів: $T_{ML} \approx 0,13 T_{base}$. Інтегральна ефективність η , обчислена за формулою

$$\eta = \frac{E_{base}}{E_{ML}} \cdot \frac{T_{base}}{T_{ML}}, \quad (26)$$

становила приблизно $\eta \approx 31$, що демонструє значну перевагу адаптивної архітектури як за точністю, так і за обчислювальною ефективністю.

Для хвильового рівняння в обмеженій області середня відносна похибка по п'яти контрольних точках часу становила $E \approx 1,45\%$, тоді як базовий метод із рівномірною сіткою показав $E_{base} \approx 6,2\%$. Час виконання за використання адаптивної MLP + RL архітектури був скорочений приблизно у 6 разів, що дало $T_{ML} \approx 0,16 T_{base}$. Відповідно, ефективність моделі для хвильового рівняння оцінюється як $\eta \approx 26$.

Отже, числові результати демонструють, що запропонована архітектура забезпечує суттєве підвищення точності чисельного моделювання за значного скорочення часу виконання, що підтверджує практичну ефективність інтеграції машинного навчання та RL в адаптивні чисельні методи.

Обговорення. Розроблена математична постановка задачі, що включає функціонал (формули (1) – (4)) з урахуванням похибки розрахунку та вартості обчислень, дозволила створити основу для впровадження методів машинного навчання в адаптивне чисельне моделювання. Використання архітектури багатошарового персептрона для прогнозування параметрів дискретизації (формули (5) – (7)) забезпечило ефективний вибір просторового та часових кроків, тоді як агент навчання з підкріпленням РРО (формули (8) – (12)) динамічно коригував ці параметри у процесі симуляції. На відміну від традиційних адаптивних методів [8; 10], де зміна кроку базується на евристичних правилах, запропонована схема враховує як локальні характеристики поля, так і обчислювальні витрати, що зменшує імовірність нераціонального використання ресурсів.

Блок-схема на рисунку 1 демонструє інтеграцію нейронної мережі та RL-агента в єдиний цикл адаптації, що дозволяє автоматично балансувати між точністю та швидкістю. Апробація на задачі теплопровідності в неоднорідному середовищі (табл. 1) показала, що середня L2-похибка адаптивної архітектури становить лише 1,2–1,6 % за прискорення обчислень у 7–8 разів щодо еталонного розв'язку, тоді як рівномірна груба сітка дає похибку 5–7% за значно нижчої ефективності. Це пояснюється автоматичним згущенням сітки в зонах з високими градієнтами (рис. 3), що відсутнє у класичних методах AMR [9].

На задачі хвильового рівняння (табл. 2) відносна похибка не перевищила 1,6% у всіх кон-

трольних точках часу, що свідчить про стійке відтворення динаміки процесу (рис. 4-А, 4-Б). Отримані результати підтверджують, що запропонована архітектура не тільки точно апроксимує окремі значення, а й зберігає глобальні фізичні закономірності хвильових процесів, на відміну від методів [12; 15], які або потребують великих обсягів навчальних даних, або працюють лише для окремих класів задач.

Важливим є також аналіз залежності похибки й часу від конфігурації нейронної мережі (рис. 2). Показано, що збільшення кількості шарів знижує похибку, але підвищує обчислювальні витрати. Оптимальною є середня глибина (5–6 шарів), що забезпечує баланс між точністю та продуктивністю, тоді як занадто глибокі мережі ведуть до перенавантаження системи без суттєвого приросту точності. Отже, результати чисельних експериментів підтвердили ефективність гібридної ML + RL архітектури: досягнуто суттєвого зниження похибки, підвищення швидкодії та узгодження моделі з фізичною природою процесів. Це узгоджується з висновками сучасних робіт [3; 6; 9], але перевершує їх завдяки поєднанню попереднього прогнозування та динамічної онлайн-адаптації. Порівняно із [25], де застосовується строгий аналітичний метод до адаптації чисельних схем, запропонована методологія має перевагу в можливості самонавчання на основі даних і поступового вдосконалення стратегії вибору параметрів.

Робота унікальна, оскільки поєднує класичні адаптивні чисельні методи із сучасними інструментами машинного навчання, зокрема багатошаровими нейронними мережами й алгоритмами навчання з підкріпленням. Запропонована гібридна архітектура є новою тим, що дозволяє не лише попередньо прогнозувати параметри дискретизації, але й динамічно їх коригувати у процесі моделювання, забезпечувати адаптацію під конкретні особливості фізичної задачі. Щодо переваг, запропонована архітектура демонструє значне зменшення похибки розрахунків (до 1,2–1,6%) за суттєвого скорочення часу виконання (у 6–8 разів) порівняно із традиційними методами. Вона автоматично концентрує обчислювальні ресурси у критичних областях із різкими градієнтами, зберігає ефективність в однорідних ділянках. Завдяки гібридному поєднанню ML і RL забезпечуються висока точність, стабільність та універсальність моделювання, що робить метод перспективним для широкого класу складних фізичних процесів.

До основних обмежень архітектури варто віднести потребу значних обчислювальних ресурсів на етапі тренування нейронних мереж та RL-агента, а також наявності якісних еталонних даних для валідації. Окрім того, результати значною мірою залежать від вибору функції винагороди та

параметрів навчання, що може обмежувати універсальність методу.

Основним недоліком є складність реалізації гібридної архітектури та необхідність налаштування великої кількості параметрів, що робить її менш придатною для задач з обмеженим доступом до даних чи ресурсів. Також існує ризик перенавчання нейронної мережі, що може знизити стійкість у разі застосування до нових класів задач.

Подальший розвиток дослідження пов'язаний із застосуванням Physics-Informed Neural Networks для безпосереднього врахування фізичних законів, розширенням апробації на багатовимірні та нелінійні системи, а також створенням більш універсальних адаптивних алгоритмів, здатних працювати в реальному часі для складних інженерних і природничих задач.

Висновки. У роботі запропоновано методику інтеграції машинного навчання з адаптивними чисельними методами для моделювання складних фізичних процесів. Проведені експерименти пока-

зали, що розроблена архітектура на основі багатошарового персептрона й алгоритму підкріплювального навчання PPO здатна ефективно керувати параметрами дискретизації. На задачах теплопровідності було досягнуто зниження середньої похибки розрахунків до рівня приблизно 1,4% за прискорення обчислень у 7–8 разів порівняно з високоточним еталонним розв'язком. Для хвильового рівняння середня відносна похибка становила приблизно 1,45%, а швидкість підвищилася в 6 разів, що підтвердило універсальність і стійкість підходу. Отримані результати доводять, що використання ML та RL забезпечує суттєве покращення балансу між точністю та продуктивністю порівняно із класичними методами. Найбільш ефективною виявилася гібридна архітектура, яка поєднує попереднє прогнозування нейронною мережею та онлайн-корекцію через RL-агента. Подальші дослідження доцільно зосередити на використанні Physics-Informed Neural Networks для прямого врахування фізичних законів і розширенні апробації на багатовимірні нелінійні задачі.

ЛІТЕРАТУРА

1. Ferreira V., Santos L., Simoes R., Franzen M., Ghouati O. Estimating local thickness for finite element analysis. 2010. P. 63–68.
2. Simos T. E. New Variable-Step Procedure for the Numerical Integration of the One-Dimensional Schrödinger Equation. *Journal of Computational Physics*. 1993. Vol. 108. Iss. 1. P. 175–179. <https://doi.org/10.1006/jcph.1993.1172>
3. Faegh M., Ghungrad S., Oliveira J. P., Rao P., Haghighi A. A review on physics-informed machine learning for process-structure-property modeling in additive manufacturing. *Journal of Manufacturing Processes*. 2025. Vol. 133. P. 524–555. <https://doi.org/10.1016/j.jmapro.2024.11.066>
4. Bilak Y. Information system based on a complex model using machine learning for spectral analysis. *Bulletin of Taras Shevchenko National University of Kyiv. Physical and Mathematical Sciences*. 2025. Vol. 80. № 1. P. 104–114. <https://doi.org/10.17721/1812-5409.2025/1.14>
5. Bilak Y. Modeling, optimization and AI-forecasting technology in Raman spectrometry. *Problems of Control and Informatics*. 2025. Vol. 70. № 2. P. 99–112. <https://doi.org/10.34229/1028-0979-2025-2-9>
6. Bilak Y. Y. Development of a combined model for analyzing gas mixtures using machine learning methods. *Applied Aspects of Information Technology*. 2025. Vol. 8, № 1. P. 24–37. <https://doi.org/10.15276/aait.08.2025.2>
7. Martín-Guerrero J., Lamata L. Reinforcement Learning and Physics. *Applied Sciences*. 2021. Vol. 11. Art. 8589. <https://doi.org/10.3390/app11188589>
8. Irigaray O., Ansa Z., Fernandez-Gamiz U., Larrinaga A., García-Fernandez R., Portal-Porras K. Adaptive mesh refinement (AMR) criteria comparison for the DrivAer model. *Heliyon*. 2024. Vol. 10. Iss. 11. P. e31966. <https://doi.org/10.1016/j.heliyon.2024.e31966>
9. Freymuth N., Dahlinger P., Würth T., Reisch S., Kärger L., Neumann G. Swarm reinforcement learning for adaptive mesh refinement. *Advances in Neural Information Processing Systems*. 2023. Vol. 36. P. 73312–73347.
10. Ranocha H., Dalcin L., Parsani M., Ketcheson D. Optimized Runge-Kutta Methods with Automatic Step Size Control for Compressible Computational Fluid Dynamics. *Communications on Applied Mathematics and Computation*. 2021. Vol. 4. <https://doi.org/10.1007/s42967-021-00159-w>
11. Henson V. Multigrid methods for nonlinear problems: An overview. *Proceedings of SPIE – The International Society for Optical Engineering*. 2003. <https://doi.org/10.1117/12.499473>
12. Raissi M., Perdikaris P., Karniadakis G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*. 2019. Vol. 378. P. 686–707. <https://doi.org/10.1016/j.jcp.2018.10.045>
13. Ger S., Jambunath Y. S., Klabjan D. Autoencoders and generative adversarial networks for imbalanced sequence classification. *2023 IEEE International Conference on Big Data (BigData)*. 2023. P. 1101–1108. <https://doi.org/10.48550/arXiv.1901.02514>

14. Ghojogh B., Ghodsi A., Karray F., Crowley M. Generative Adversarial Networks and Adversarial Autoencoders: Tutorial and Survey. *ArXiv*. 2021. abs/2111.13282. <https://doi.org/10.48550/arXiv.2111.13282>
15. Sutton R. S., Barto A. G. Reinforcement Learning. An Introduction. 2nd ed. Cambridge : A Bradford Book, 2018.
16. Zhang C., Lu Y. Study on artificial intelligence: The state of the art and future prospects. *Journal of Industrial Information Integration*. 2021. Vol. 23. P. 100224. <https://doi.org/10.1016/j.jii.2021.100224>
17. Strikwerda J. C. Finite Difference Schemes and Partial Differential Equations. 2nd ed. Philadelphia: Society for Industrial and Applied Mathematics, 2004. <https://doi.org/10.1137/1.9780898717938>
18. Leveque R. J. Finite Volume Methods for Hyperbolic Problems. Cambridge : Cambridge University Press, 2002. <https://doi.org/10.1017/CBO9780511791253>
19. Berger M. J., Colella P. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*. 1989. Vol. 82. Iss. 1. P. 64–84. [https://doi.org/10.1016/0021-9991\(89\)90035-1](https://doi.org/10.1016/0021-9991(89)90035-1)
20. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. *IEEE Transactions on Neural Networks*. 1998. Vol. 9. № 5. P. 1054–1054. <https://doi.org/10.1109/TNN.1998.712192>
21. Heaton J. Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning: The MIT Press, 2016, 800 p. *Genetic Programming and Evolvable Machines*. 2017. Vol. 19. <https://doi.org/10.1007/s10710-017-9314-z>
22. Ioffe S., Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*. 2015. P. 448–456. <https://doi.org/10.48550/arXiv.1502.03167>
23. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal policy optimization algorithms. *arXiv preprint*. 2017. arXiv : 1707.06347
24. Han J., Jentzen A., Ee W. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*. 2017. Vol. 115. <https://doi.org/10.1073/pnas.1718942115>
25. Білак Ю., Геращенко Е. Багаторівнева декомпозиція для адаптивного чисельного моделювання складних фізичних процесів. *Herald of Khmelnytskyi National University. Technical Sciences*. 2025. Vol. 349. № 2. P. 51–62. <https://doi.org/10.31891/2307-5732-2025-349-7>

REFERENCES

1. Ferreira V., Santos L., Simoes R., Franzen M., Ghouati O. Estimating local thickness for finite element analysis. 2010. P. 63–68.
2. Simos T. E. New Variable-Step Procedure for the Numerical Integration of the One-Dimensional Schrödinger Equation. *Journal of Computational Physics*. 1993. Vol. 108, Iss. 1. P. 175–179. <https://doi.org/10.1006/jcph.1993.1172>
3. Faegh M., Ghungrad S., Oliveira J. P., Rao P., Haghighi A. A review on physics-informed machine learning for process-structure-property modeling in additive manufacturing. *Journal of Manufacturing Processes*. 2025. Vol. 133. P. 524–555. <https://doi.org/10.1016/j.jmapro.2024.11.066>
4. Bilak Y. Information system based on a complex model using machine learning for spectral analysis. *Bulletin of Taras Shevchenko National University of Kyiv. Physical and Mathematical Sciences*. 2025. Vol. 80, № 1. P. 104–114. <https://doi.org/10.17721/1812-5409.2025/1.14>
5. Bilak Y. Modeling, optimization and AI-forecasting technology in Raman spectrometry. *Problems of Control and Informatics*. 2025. Vol. 70, № 2. P. 99–112. <https://doi.org/10.34229/1028-0979-2025-2-9>
6. Bilak Y.Y. Development of a combined model for analyzing gas mixtures using machine learning methods. *Applied Aspects of Information Technology*. 2025. Vol. 8, № 1. P. 24–37. <https://doi.org/10.15276/aait.08.2025.2>
7. Martín-Guerrero J., Lamata L. Reinforcement Learning and Physics. *Applied Sciences*. 2021. Vol. 11. Art. 8589. <https://doi.org/10.3390/app11188589>
8. Irigaray O., Ansa Z., Fernandez-Gamiz U., Larrinaga A., García-Fernandez R., Portal-Porras K. Adaptive mesh refinement (AMR) criteria comparison for the DrivAer model. *Heliyon*. 2024. Vol. 10, Iss. 11. e31966. <https://doi.org/10.1016/j.heliyon.2024.e31966>
9. Freymuth N., Dahlinger P., Würth T., Reisch S., Kärger L., Neumann G. Swarm reinforcement learning for adaptive mesh refinement. *Advances in Neural Information Processing Systems*. 2023. Vol. 36. P. 73312–73347.
10. Ranocha H., Dalcin L., Parsani M., Ketcheson D. Optimized Runge-Kutta Methods with Automatic Step Size Control for Compressible Computational Fluid Dynamics. *Communications on Applied Mathematics and Computation*. 2021. Vol. 4. <https://doi.org/10.1007/s42967-021-00159-w>

11. Henson V. Multigrid methods for nonlinear problems: An overview. *Proceedings of SPIE – The International Society for Optical Engineering*. 2003. <https://doi.org/10.1117/12.499473>
12. Raissi M., Perdikaris P., Karniadakis G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*. 2019. Vol. 378. P. 686–707. <https://doi.org/10.1016/j.jcp.2018.10.045>
13. Ger S., Jambunath Y. S., Klabjan D. Autoencoders and generative adversarial networks for imbalanced sequence classification. *2023 IEEE International Conference on Big Data (BigData)*. 2023. P. 1101–1108. <https://doi.org/10.48550/arXiv.1901.02514>
14. Ghojogh B., Ghodsi A., Karray F., Crowley M. Generative Adversarial Networks and Adversarial Autoencoders: Tutorial and Survey. *ArXiv*. 2021. abs/2111.13282. <https://doi.org/10.48550/arXiv.2111.13282>
15. Sutton R. S., Barto A. G. Reinforcement Learning. An Introduction. 2nd ed. Cambridge: A Bradford Book, 2018.
16. Zhang C., Lu Y. Study on artificial intelligence: The state of the art and future prospects. *Journal of Industrial Information Integration*. 2021. Vol. 23. 100224. <https://doi.org/10.1016/j.jii.2021.100224>
17. Strikwerda J. C. Finite Difference Schemes and Partial Differential Equations. 2nd ed. Philadelphia: Society for Industrial and Applied Mathematics, 2004. <https://doi.org/10.1137/1.9780898717938>
18. Leveque R. J. Finite Volume Methods for Hyperbolic Problems. Cambridge: Cambridge University Press, 2002. <https://doi.org/10.1017/CBO9780511791253>
19. Berger M. J., Colella P. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*. 1989. Vol. 82, Iss. 1. P. 64–84. [https://doi.org/10.1016/0021-9991\(89\)90035-1](https://doi.org/10.1016/0021-9991(89)90035-1)
20. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. *IEEE Transactions on Neural Networks*. 1998. Vol. 9, № 5. P. 1054–1054. <https://doi.org/10.1109/TNN.1998.712192>
21. Heaton J. Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning: The MIT Press, 2016, 800 pp. *Genetic Programming and Evolvable Machines*. 2017. Vol. 19. <https://doi.org/10.1007/s10710-017-9314-z>
22. Ioffe S., Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*. 2015. P. 448–456. <https://doi.org/10.48550/arXiv.1502.03167>
23. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal policy optimization algorithms. *arXiv preprint*. 2017. arXiv: 1707.06347.
24. Han J., Jentzen A., Ee W. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*. 2017. Vol. 115. <https://doi.org/10.1073/pnas.1718942115>
25. Bilak Y., Herashchenkov E. Multilevel decomposition for adaptive numerical modeling of complex physical processes. *Herald of Khmelnytskyi National University. Technical Sciences*. 2025. Vol. 349, № 2. P. 51–62. <https://doi.org/10.31891/2307-5732-2025-349-7>

Дата першого надходження рукопису до видання: 21.08.2025
 Дата прийнятого до друку рукопису після рецензування: 23.09.2025
 Дата публікації: 31.12.2025