

А. С. Потєха
В. М. Андрухов

ОГЛЯД МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ ОПТИМІЗАЦІЇ СХЕМ КАРКАСНО-МОНОЛІТНИХ БУДІВЕЛЬ

Вінницький національний технічний університет

У статті розглянуто задачу оптимального розміщення колон на плані будівлі з урахуванням архітектурних обмежень, мінімізації згинальних моментів та виконання вимог щодо рівномірного розподілу навантаження. Основна увага приділяється поєднанню традиційних методів оптимізації (лінійних і нелінійних) з сучасними підходами машинного навчання, такими як евристичні алгоритми (генетичні алгоритми, метод рою частинок), глибокі нейронні мережі та навчання з підкріпленням. Для кожного алгоритму демонструється простий приклад реалізації на Python, порівнюються критерії часу виконання, точності розв'язку, стійкості до шуму та схильності до потрапляння в локальні мінімуми.

Дослідження передбачає формулювання вихідних даних, які відображають геометрію будівлі, механічні характеристики матеріалів та розрахункові навантаження. Для ілюстрації різних підходів до мінімізації цільової функції за наявності певних обмежень (допустимі координатні кроки, мінімальна відстань між опорами колон, вимоги до рівномірності навантаження) запропоновано спрощений тестовий приклад. На основі порівняльного аналізу складається таблиця з обчислювальною складністю, середнім часом виконання, відсотком помилок та індексом стійкості. Для наочності введено інтегральну оцінку, яка узагальнює кілька критеріїв в єдину метрику продуктивності.

Результати демонструють, що у відносно простих прикладах локальні методи (наприклад, SLSQP) здатні швидко і точно наближатися до оптимуму, тоді як евристичні алгоритми (GA, PSO), хоча і вимагають більше часу, часто уникають локальних мінімумів. Посилене навчання і глибокі нейронні мережі стають більш конкурентоспроможними в складних багатовимірних умовах, де велика кількість обмежень і змінних робить переваги моделей глобального пошуку і апроксимації очевидними. Приклади і таблиці, представлені в цій статті, можуть слугувати відправною точкою для подальшого впровадження методів машинного навчання в реальних задачах структурного проектування.

Ключові слова: несучі конструкції, машинне навчання, евристичні алгоритми, генетичний алгоритм, метод рою частинок, підсилене навчання, глибокі нейронні мережі

Вступ

Оптимізація розміщення вертикальних несучих елементів (наприклад, колон) у будівлях є надзвичайно важливою у сучасному будівництві. Емпіричні підходи або досвід проектувальників не йдуть ні в яке порівняння зі штучним інтелектом (ШІ) та машинним навчанням (МН). Ці підходи значно прискорюють оптимізацію розташування колон та підвищують точність прогнозів [1], [2]. Дослідження з проектування залізобетонних плит, стін та колон постійно показують, що правильний вибір кількості, розмірів та розташування несучих елементів безпосередньо впливає на технічну та економічну ефективність конструкції [3], [4].

Сучасна практика будівництва потребує врахування низки факторів: від зменшення використання матеріалів до забезпечення міцності конструкції та енергоефективності будівель [5]. Алгоритми машинного навчання можуть обробляти надзвичайно велику кількість параметрів (форма будівлі, властивості матеріалів, умови навантаження тощо) і миттєво пропонувати сценарії оптимального розміщення колон у заданих стінах.

Методи штучного інтелекту значною мірою застосовуються в оцінці вартості будівництва [6], прогнозуванні часу виконання робіт [7] та пошуку оптимального рівня ресурсів [8]. Існує велика кількість наукових праць, присвячених прогнозуванню кошторисної вартості та економічної ефективності будівельної діяльності, серед яких домінують нейронні мережі (ANN), машини опорних векторів (SVM) та дерева рішень (DT) [9]-[11].

Між тим, огляд сучасних досліджень показує, що питання оптимізації позиціонування колон з урахуванням структурних обмежень у будівельному проекті досліджено недостатньо. У більшості робіт йдеться лише про певне застосування алгоритмів оптимізації (наприклад, генетичних алгоритмів або методу рою частинок) до структурної схеми будівлі [12]. Однак відсутня загальна стратегія, яка стосується мінімізації матеріалів при розробці та компоновці схем.

Метою статті є теоретична перевірка існуючих методів машинного навчання, їх порівняння та визначення найбільш ефективного алгоритму для задачі підбору розташування колон в плані будівлі.

Постановка задачі

Одною із важливих задач, яка чітко визначена як план впровадження штучного інтелекту в межах

моделювання несучих конструкцій – є оптимізація витрат будівництва. Для вирішення задачі по зменшенню витрат розглянута методика по оптимізації розташування колон в будівлі. Дана методика використовує машинні алгоритми для навчання програми розташовувати несучі елементи будівлі.

Оптимізація розташування колон здійснюється в існуючому планувальному рішенні будівлі, де геометрію стін та габарити перекриттів визначено заздалегідь. Для використання такого методу необхідно формувати ряд параметрів, які будуть впливати на розташування колон в плані.

Планувальні вимоги: Товщина плити, розміри панелей, розташування капітальних стін і перегородок, а також допустима ширина стін, у яких можна “сховати” колони.

Конструктивні вимоги: Жорсткість і несуча здатність плити, діапазони розмірів перерізу колон, межі можливого відхилення вертикальних сил у колонах (не більше 15%), забезпечення балансу за об'ємом залізобетону.

Експлуатаційні вимоги: Архітектурні вимоги (невидимість колон), умови зручності планувань, евакуаційні коридори тощо.

Навантаження: Власна вага плити, корисне навантаження (люди, меблі, обладнання), додаткові вітрові чи сейсмічні навантаження в залежності від регіону будівництва.

Враховуючи ці критерії, необхідно знайти схему розташування колон, яка мінімізує згинальні моменти перекриття, мінімізує кількість і розмір колон, забезпечує рівномірний розподіл вертикальних навантажень і має мінімальну кількість бетону, задовольняючи при цьому всі проектні вимоги.

Задача пошуку оптимального розташування колон може бути описана через вектор змінних за виразом (1):

$$X = \{x_i\} \quad (1)$$

де: x_i – відповідають координатам колон

Для даної задачі необхідно сформулювати ряд обмежень, які будуть впливати на розташування колон, та контролювати мінімальні необхідні параметри. Набір обмежень формується так:

$$M_{max}(X) \leq M_{don}$$

де: M_{don} – гранично допустимий згинальний момент у плиті перекриття.

$$\Delta N_{col}(X) \leq \delta$$

де: N – різниця у вертикальних навантаженнях (між найнавантаженішою та найменш навантаженою колоною) не перевищує 15%.

$$n_{col}(X) \leq n_{don}$$

де: кількість колон не перевищує встановленого максимуму (або ж додатково вводиться вимога до мінімальної кількості для забезпечення несучої здатності).

Крім того, розглядається також мінімізація об'єму бетону колон і пов'язаних з ними конструктивних елементів. Таким чином, цільова функція (або система цільових функцій) може бути виражена у вигляді багатокритеріальної залежності (2):

$$f(X) = \alpha \cdot M_{max}(X) + \beta \cdot n_{col}(X) + \gamma \cdot \Delta N_{col}(X) + \zeta \cdot V_c(X) \quad (2)$$

де: $M_{max}(X)$ - найбільший згинальний момент у плиті при заданому розташуванні колон;

$n_{col}(X)$ – загальна кількість колон;

$\Delta N_{col}(X)$ – відносна різниця у вертикальних зусиллях між найнавантаженішою та найменш навантаженою колоною;

$V_c(X)$ – сукупний обсяг бетону в колонах (за потреби може включати також обсяг додаткових конструктивних елементів);

$\alpha, \beta, \gamma, \zeta$ – вагові коефіцієнти, які встановлюються з урахуванням техніко-економічних пріоритетів.

Мета полягає у визначенні набору змінних X , які зменшують значення $f(x)$ за умови дотримання всіх обмежень, наведених вище. У класичній задачі багатокритеріальної оптимізації функції можуть бути представлені у векторній формі для реалізації методів Парето-оптимальності або методів зі «згорнутими» критеріями.

За аналогією з підходами до прогнозування вартості чи строків будівництва застосування методів машинного навчання до завдання оптимізації розташування колон передбачає деякі кроки рис. 1.

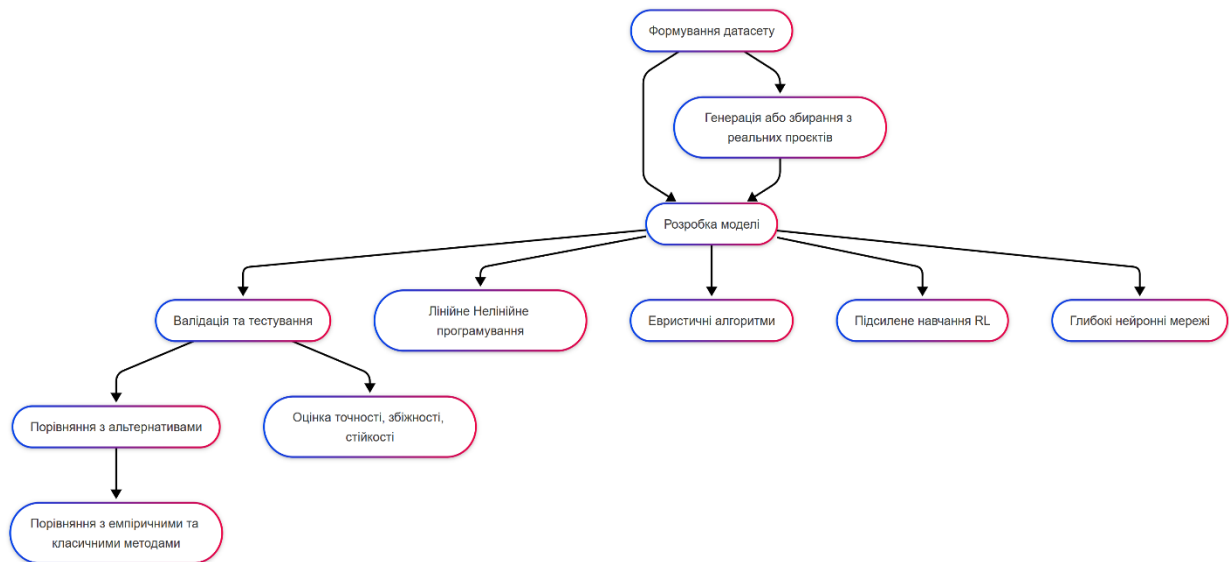


Рисунок 1 – Схема дослідження ефективності методів

Таким чином, у цій роботі пропонується науково обґрунтована та практично орієнтована модель, що поєднує в собі підходи машинного навчання та математичного моделювання, спрямовані на підвищення ефективності будівельного процесу.

Вихідні дані для навчання алгоритмів

При пошуку найкращого розташування колон у заданому проєкті будівлі першочерговим завданням є створення вхідного зразка, який представляє геометричні, матеріальні та експлуатаційні параметри будівлі. Найбільш важливими геометричними параметрами є форма і висота стін, фактичні або проєктні значення перекриття (товщина і довжина прольоту), а також допустимі відстані між стінами і колонами. На основі цієї інформації ми можемо реконструювати ситуацію, за якої колони можуть бути успішно «підігнані» таким чином, щоб мінімізувати їх кількість та згинальні моменти у плиті [4]. Крім того, під час підготовки даних зазвичай вказуються механічні характеристики матеріалів, тобто міцність бетону і клас арматури.

Другим будівельним блоком вхідних даних є визначення прикладених навантажень. Зокрема, розглядається суперпозиція постійних і змінних вертикальних навантажень. Вітрові та сейсмічні навантаження враховуються незалежно, якщо це передбачено місцевими нормами проєктування. У випадку багатоповерхових будівель це доповнюється більш точним розподілом навантажень між поверхами, що має значення для реалістичного розрахунку зусилля на колону (див. [5]). Іноді додаткові заборони пов'язані з архітектурною структурою, наприклад, заборона розміщувати колону в дверних отворах або на перетині інженерних комунікацій.

Таким чином, для навчання алгоритмічних моделей необхідно скласти достатню кількість вихідних даних. До цих даних входять уже готові приклади розташування колон, які можуть бути еталоном для навчання. Кожен з прикладів має включати в себе:

- Габарити плити перекриття;
- Розташування та кількість колон;
- Розміри колон;
- Навантаження;

Отримавши широкий спектр таких параметрів, вчені можуть генерувати штучні або змішані (реальні та синтетичні) набори даних, які включають безліч варіантів планування і можуть бути використані для навчання алгоритмів. Тут важливо збалансувати вибірку в сенсі кількості прикладів з різними конфігураціями колон і правильно задати цільові показники (наприклад, сумарні згинальні моменти в плиті або кількість бетону).

Що стосується машинних підходів, які можуть бути застосовані до описаних завдань, то загалом є кілька стандартних категорій.

Лінійне програмування (ЛП). Воно застосовується в ситуаціях, коли можна привести цільову функцію і всі обмеження до лінійного вигляду. Однак у задачі компонування колон більшість залежностей (наприклад, розподіл моментів у плиті, нелінійна робота залізобетону) містять явно виражені нелінійні частини, що ускладнює застосування чистого ЛП.

Нелінійне програмування (НЛП). Воно більш точно відображає реальну природу будівельних робіт, але при великій кількості змінних (координати кожної колоні та її діаметр) обчислювальна складність дуже швидко зростає, а глобального оптимуму важко досягти [11].

Генетичні алгоритми (ГА). Вони імітують процес еволюції у виборі найкращих рішень відповідно до «генів» (параметрів стовпця). Вони відомі тим, що розв'язують проблеми високої розмірності та здатні оминати локальні мінімуми [7].

Оптимізація рою частинок (PSO). Алгоритм, натхненний природою, розроблений шляхом спостереження за «поведінкою» частинок (рішень) у пошуках оптимального положення. Це алгоритм, що швидко збігається для більшості задач оптимізації, але потребує ретельної ініціалізації та має враховувати високу нелінійність задачі [11].

Підхід бджолиного рою, диференціальна еволюція тощо. Менш популярні, але також здатні знаходити хороші рішення в задачах з численними обмеженнями змішаного типу [14].

Q-навчання, SARSA, Deep-Q-мережі. Нехай агент продовжує «вчитися» на власному досвіді, оцінюючи винагороду (наприклад, мінімізацію загального об'єму бетону або моментів) за кожен крок модифікації розташування колон [5]. Посилене навчання може бути найбільш застосовним там, де є величезна «імітаційна» модель, на якій можна змоделювати поведінку стін і перекриттів при різних варіантах розташування колон, а потім агент коригує свої дії на основі цього.

Глибокі нейронні мережі (DNN). Fully-Connected Networks (FCN) можуть прогнозувати вихідні значення, наприклад, навантаження і моменти, при заданих координатах колон; однак безпосереднє використання глибоких нейронних мереж (ГНМ) для оптимізаційних задач часто вимагає додаткових методів для знаходження оптимального рішення [8].

Ансамблеві методи (Random Forest, XGBoost, LightGBM). У задачах, спрямованих на прогнозування продуктивності певної конфігурації колоні (наприклад, оцінка моменту або кількості необхідного бетону), ансамблеві моделі можуть працювати надзвичайно добре завдяки тому, що вони:

- Дуже стійкі до «зашумлених» вхідних даних.
- Їх легко швидко навчити і можна масштабувати відповідно до зростання набору даних [18].

Тим не менш, ці методи здебільшого застосовуються на етапі «оцінки» проміжних рішень або для побудови деяких сурогатних моделей, які дозволяють спростити складні обчислювальні аналізи.

Критеріальна оцінка методів

Оцінка алгоритмів машинного навчання для оптимізації розміщення колон повинна враховувати безліч аспектів, які охоплюють як технічну продуктивність, так і обчислювальні витрати. Нижче наведено основні категорії показників, які можуть бути використані для висновків щодо придатності та оптимальності підходу.

Час (T) – Чим менше секунд витрачає алгоритм, тим краще. Для нормування беремо значення T_{norm} за виразом (3).

$$T_{norm} = 1 - \frac{T_i}{T_{max}} \quad (3)$$

де: T_{max} – найбільший час серед усіх методів;

T_i – час для i-го методу;

Похибка (ϵ) – Відсоток, на який отримане рішення відхиляється від ідеального (на тестовому прикладі це різниця між знайденим і «еталонним» оптимумом, визначається за виразом (4). Менше значення свідчить про вищу точність.

$$\epsilon_{norm} = 1 - \frac{\epsilon_i}{\epsilon_{max}} \quad (4)$$

де: ϵ_{max} – найбільше відхилення серед усіх методів;

ϵ_i – відхилення для i-го методу;

Стабільність (S). Умовна шкала від 0 до 10, де більше значення означає стійкіший алгоритм, який з меншою ймовірністю «застрягає» у локальних мінімумах або видає нестабільні результати за незначної зміни параметрів (5).

$$S_{norm} = 1 - \frac{S_i}{S_{max}} \quad (5)$$

де: $S_{\max}=10$ або ж це може бути найбільше реальне значення, зафіксоване в досліді;

Методичне порівняння та інтегральна оцінка (E). Підсумкова метрика, що об'єднує вплив часу, точності та стабільності, та визначається за формулою (6). Чим вище інтегральна оцінка (E), тим метод вважається ефективнішим за обраною схемою.

$$E = w_T \cdot T_{norm} + w_E \cdot \varepsilon_{norm} + w_S \cdot S_{norm} \quad (6)$$

де: w_T, w_E, w_S – вагові коефіцієнти, які можна задавати залежно від пріоритетів;

Теоретична складність (Big-O) та O(N). Теоретична складність зазвичай вимірюється при масштабуванні задачі. Визначаємо це значення як алгоритм може поводитись за збільшеної кількості колон або складніших обмежень.

Таким чином, процес критеріальної оцінки й відбору алгоритму для оптимізації компонування колон залежить від пріоритетів конкретного проєкту. Вибір оптимальної стратегії базується на комплексному порівнянні методів за низкою критеріїв.

Вибір оптимального методу для даної задачі

Для тестування методів машинного навчання необхідно створити тестову задачу з еталонним значенням. В рамках критеріального оцінювання було створено задачу, в якій буде описаний типовий процес для задач з пошуку оптимального розташування колон.

Геометрична репрезентація:

У цьому дослідженні розглядається планування поверху розміром 8 на 8 метрів. У цьому просторі є внутрішня несуча стіна, розташована вздовж вертикальної лінії $x = 4$, між точками (4.0) і (4.8) рис. 2. На цій стіні необхідно розмістити дві колони в точках (4, y_1) і (4, y_2).

Змінні: y_1, y_2 .

Області визначення: $0 \leq y_1 < y_2 \leq 8$.

Мінімальна відстань між колонами: $y_2 - y_1 \geq 1$

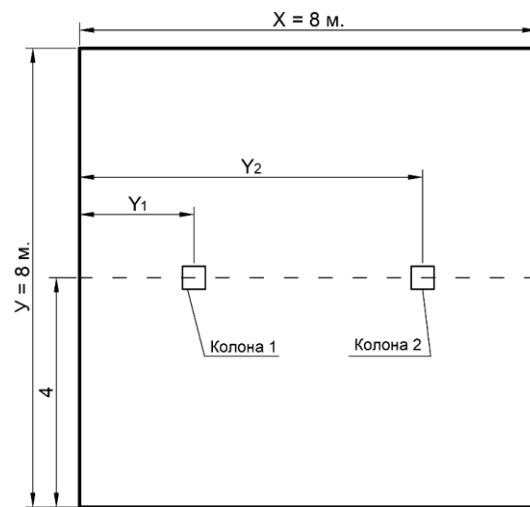


Рисунок 2 – Геометричне зображення деталей задачі

Навантаження в колонах (умовна модель):

Кожна колона i сприймає вертикальне навантаження яке залежить від грузової площі, яка приходить на колону 1 та 2. Плита рівномірно навантажена 10 КН/м^2 . Величина навантаження від колони розраховується за такою формулою:

$$\begin{aligned} L_1 &= (y_1 + \frac{y_2 - y_1}{2}) \cdot 8 \cdot 10 \\ L_2 &= 8 - (y_2 + \frac{y_2 - y_1}{2}) \cdot 8 \cdot 10 \end{aligned} \quad (7)$$

де: y_1 та y_2 – це координати в метрах.;

Для колон 1 та 2 необхідно встановити обмеження їх вертикального навантаження відповідно до умови, що різниця вертикального навантаження не має відрізнитись більше ніж на 15% за виразом (8).

$$\frac{L_1 - L_2}{(L_1 + L_2) \cdot 0.5} \leq 0.15 \quad (8)$$

де: L_1 та L_2 – це навантаження на колони в КН.;

Нехай монолітна плита матиме тенденцію до мінімального моменту, коли колони розташовані ближче до країв перекриття по осі $y \approx 4$. Для спрощення візьмемо цільову функцію (9).

$$M(y_1, y_2) = (4 - y_1)^2 + (4 - y_2)^2 \quad (9)$$

В даній задачі ми маємо еталонне значення $y_1=2$ та $y_2 = 6$. Тому вираз (9) має тяготи до нуля по результатам підстановки у різних алгоритмах.

Нижче наведено приблизні результати одного запуску (на умовних налаштуваннях та з однаковими початковими умовами), аби побачити відмінності між алгоритмами. Зазначимо, що через стохастичність GA, PSO та початкову ініціалізацію у методів можуть бути коливання таблиця 1.

Для оцінювання були взяті такі алгоритми:

- Локальна оптимізація (SLSQP через SciPy);
- Генетичний алгоритм (спрощена версія) GA;
- Метод рою частинок (PSO);
- «Сурогатне» оцінювання глибинною нейронною мережею (DNN);
- Підсилене навчання (RL).

Для ілюстрації візьмемо результати спрощеного прикладу з двома колонами на прольоті (0-8) з мінімальною довжиною 1 м, вимогою рівномірності навантаження (не більше 15%) і метою мінімізації згинальних моментів. Репрезентативні дані (T, ϵ , S) відносяться до приблизних типових результатів одного випробування (фіксовані гіперпараметри, однакові початкові умови).

Хоча приклад досить штучний, він уже відображає кілька ключових аспектів реальної задачі: (а) існує обмежений «простір для розміщення колон» (у цьому разі, лінія $x=4$), (б) є обмеження відстані та рівномірності навантажень, (в) ціль мінімізації згинальних моментів у плиті, хоча й подано у спрощеній формі $(4-y)^2$.

Таблиця 1

Результати порівняння виконання задачі алгоритмами

Метод	T (с)	ϵ (%)	S (0–9)	Big-O	T _{norm}	ϵ _{norm}	S _{norm}	E
SLSQP	0.01	0.0	8	$O(n^3)$	0.96	1.0	0.89	0.955
GA	0.15	0.5	9	$O(n^2 \cdot G)$	0.40	0.5	1.0	0.62
DNN	0.2	0.3	8	$O(n^3)$	0.20	0.7	0.89	0.607
PSO	0.10	0.8	7	$O(n \cdot \log(n) \cdot l)$	0.60	0.20	0.78	0.494
RL	0.25	1.0	5	$O(n^4)$	0.00	0.00	0.56	0.168

З таблиці бачимо, що для дуже простої задачі SLSQP має найвищу інтегральну оцінку (0.955), оскільки здатен швидко й точно визначити рішення.

Генетичний алгоритм (0.62) хоч і менш швидкий, зате вирізняється стабільністю на різних прикладах ($S=9$) та дає незначну похибку (0.5 %).

DNN (0.607) вимагає навчання (тому час більший), але результати можуть бути досить точними й стійкими.

Метод рою частинок (0.494) трохи програє за точністю, оскільки часто «застрягає», якщо обрана конфігурація параметрів недостатньо оптимальна.

Підсилене навчання (0.168) в такому «спрощеному» варіанті виявляється найменш ефективним: воно потребує великої кількості епізодів і ретельного налаштування винагороди, аби гарантувати вихід на оптимум.

Такий інтегральний підхід дають змогу наочно побачити переваги та недоліки різних методів на спрощеній постановці. У більший (і реалістичніший) задачі розташування колон кількість змінних, обмежень і формул для оцінки «якості» конструктивної схеми будуть значно більшими, але принципи зважування, нормування й підсумкового порівняння залишаться схожими.

Висновки

Огляд та практичне порівняння методів показало, що вибір найкращого методу розв'язання задачі розміщення колон залежить від складності задачі, рівня точності, часу та наявних обчислювальних потужностей. Прості задачі ефективно вирішуються локальними методами (зокрема, SLSQP), але евристичні методи (GA, PSO та навчання з підкріпленням (RL) мають потенціал для пошуку глобальних мінімумів, не зациклюючись на локальних мінімумах. Глибокі

нейронні мережі виявляються корисними в «сурогатному моделюванні», якщо пряме обчислення складної цільової функції є надто трудомістким. Запропонована інтегральна міра (з урахуванням часу, похибки та стабільності) дозволяє універсально порівнювати методи на різних етапах дослідження. Подальші дослідження можуть включати інтеграцію вищезазначених методів з реальними моделями будівель (3D моделі, нелінійна поведінка матеріалу, сейсмічні та інші динамічні навантаження), а також розробку адаптивних та гібридних систем, які дозволять підвищити точність та швидкість розрахунків.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Кишкан А., Кархут, І. (2024). ПОТЕНЦІАЛ АВТОМАТИЗАЦІЇ ТА ПРИСКОРЕННЯ БУДІВЕЛЬНИХ ПРОЦЕСІВ ЗА ДОПОМОГОЮ МАШИННОГО НАВЧАННЯ ТА ШТУЧНОГО ІНТЕЛЕКТУ. Матеріали конференцій МЦНД, (17.05.2024; Ужгород, Україна), 354–358. вилучено із <https://archive.mcnd.org.ua/index.php/conference-proceeding/article/view/1231>
2. <https://archive.mcnd.org.ua/index.php/conference-proceeding/article/view/1231>
3. Сьомко, П., & Левус, Є. (2024). ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ДЛЯ ВИЗНАЧЕННЯ АРХІТЕКТУРНОГО СТИЛЮ БУДІВЕЛЬ. Матеріали конференцій МНЛ, (17 травня 2024 р., м. Київ), 346–347.
4. Вилучено з <https://archive.liga.science/index.php/conference-proceedings/article/view/964>
5. Balaji, G.C.; Vivek, S.S.; Edwin Fernando, P.A.; Balasubramanian, K. Structural Optimization of RC Columns in a Multi-Storeyed Building by Tree-Columns Subjected to Lateral Loads. E3S Web of Conferences 2023, 405, 03001. DOI: <https://doi.org/10.1051/e3sconf/202340503001>.
6. Tunca, O.; Carbas, S. Design Cost Minimization of a Reinforced Concrete Column Section Using Overnew Swarm-Based Optimization Algorithms. Neural Computing and Applications 2024, 36(27). DOI: <https://doi.org/10.1007/s00521-024-09998-z>.
7. Лялюк, О. Г. ., Осипенко, Р. С. ., & Мельник, Д. О. (2024). СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ В БУДІВЕЛЬНИХ ПРОЕКТАХ НА ОСНОВІ МЕРЕЖ БАЙЕСА. Сучасні технології, матеріали і конструкції в будівництві, 36(1), 96–102. <https://doi.org/10.31649/2311-1429-2024-1-96-102>
8. Лялюк, О. Г., & Осипенко, Р. С. . (2023). ОСОБЛИВОСТІ ІМПЛЕМЕНТАЦІЇ ШТУЧНОГО ІНТЕЛЕКТУ В БУДІВНИЦТВІ. Сучасні технології, матеріали і конструкції в будівництві, 35(2), 172–176. <https://doi.org/10.31649/2311-1429-2023-2-172-176>
9. Назіров Е. К. Методи збору та класифікації даних за допомогою саундлетної байєсівської нейронної мережі / Е. К. Назіров, Т. О. Назірова, М. Ю. Карпенко // Вісник Херсонського національного технічного університету . - 2018. - № 3(1). - С. 332-337. - Режим доступу: [http://nbuv.gov.ua/UJRN/Vkhdtu_2018_3\(1\)_49](http://nbuv.gov.ua/UJRN/Vkhdtu_2018_3(1)_49).
10. Методи машинного навчання для підвищення енергоефективності будівель / В. В. Макар, Н. І. Глухова, Р. О. Романів // Вісник Національного університету «Львівська політехніка». Серія: Теплоенергетика. Інженерія довкілля. Автоматизація. – 2024. – № 1. – С. 191–211. – URL: <https://science.lpnu.ua/sites/default/files/journal-paper/2024/jan/33151/231474maket1-191-211.pdf>
11. Потенціал автоматизації та прискорення будівельних процесів за рахунок машинного навчання та штучного інтелекту / О. А. Бойко, А. С. Ковальчук // Modern Construction and Architecture. – 2023. – № 3. – С. 85–92. – Режим доступу: <https://archive.mcnd.org.ua/index.php/conference-proceeding/article/view/1231>
12. Особливості використання штучного інтелекту та машинного навчання в будівельній галузі / Ю. В. Юзвак, О. П. Шпильовий, Л. І. Мороз // Сучасні технології, матеріали і конструкції в будівництві. – 2023. – № 2. – С. 112–119. – Режим доступу: <https://stmkvb.vntu.edu.ua/index.php/stmkvb/article/view/851>
13. <https://stmkvb.vntu.edu.ua/index.php/stmkvb/article/view/851>
14. Поняття про генетичні алгоритми та їх застосування в задачах оптимізації / О. В. Кравченко, П. І. Тимошенко // Комп'ютерні технології обробки даних. – 2023. – № 10. – Режим доступу: <https://jktod.donnu.edu.ua/article/view/16228>
15. Zheng, H.; Moosavi, V.; Akbarzadeh, M. Machine Learning Assisted Evaluations in Structural Design and Construction. Automation in Construction 2020, 114, 103346. DOI: <https://doi.org/10.1016/j.autcon.2020.103346>.

REFERENCES

1. Kishkan A., Karkhut I. (2024). Potentsial avtomatyzatsii ta pryskorennia budivelnnykh protsesiv za dopomohoiu mashynnoho navchannia ta shtuchoho intelektu. Materialy konferentsii MTsND, (17.05.2024; Uzhhorod, Ukraina), 354–358.
2. Siomko P., Levus Ye. (2024). Vykorystannia mashynnoho navchannia dlia vyznachennia arkhitekturnoho styliu budivel. Materialy konferentsii MNL, (17 travnia 2024 r., m. Kyiv), 346–347.
3. Lialyuk O. H., Osypenko R. S., Melnyk D. O. (2024). Systemy pidtrymky pryiniattia rishen v budivelnnykh proiektakh na osnovi merezh Baiiesa. Suchasni tekhnolohii, materialy i konstrukttsii v budivnytstvi, 36(1), 96–102.
4. Lialyuk O. H., Osypenko R. S. (2023). Osoblyvosti implemetatsii shtuchoho intelektu v budivnytstvi. Suchasni tekhnolohii, materialy i konstrukttsii v budivnytstvi, 35(2), 172–176.
5. Nazirov Ye. K., Nazirova T. O., Karpenko M. Yu. (2018). Metody zboru ta klasyfikatsii danykh za dopomohoiu saundletnoi baiiesivskoi neironnoi merezhi. Visnyk Khersonskoho natsionalnoho tekhnichnoho universytetu, № 3(1), 332–337.
6. Makar V. V., Hlukhova N. I., Romaniiv R. O. (2024). Metody mashynnoho navchannia dlia pidvyshchennia enerhoefektyvnosti budivel. Visnyk Natsionalnoho universytetu "Lvivska politehnika". Seriya: Teploenerhetyka. Inzheneriia dovkillia. Avtomatyzatsiia, № 1, 191–211.
7. Boiko O. A., Kovalchuk A. S. (2023). Potentsial avtomatyzatsii ta pryskorennia budivelnnykh protsesiv za rakhunok mashynnoho navchannia ta shtuchoho intelektu. Modern Construction and Architecture, № 3, 85–92.
8. Yuzvak Yu. V., Shpylovyi O. P., Moroz L. I. (2023). Osoblyvosti vykorystannia shtuchoho intelektu ta mashynnoho navchannia v budivelnii haluzi. Suchasni tekhnolohii, materialy i konstrukttsii v budivnytstvi, № 2, 112–119.
9. Kravchenko O. V., Tymoshenko P. I. (2023). Poniattia pro henetychni alhorytmy ta yikh zastosuvannia v zadakhakh optymizatsii. Komp'yuterni tekhnolohii obrobky danykh, № 10.

Потєха Андрій Сергійович – аспірант 1 курсу, кафедра БМГА факультет будівництва, цивільної та екологічної інженерії Вінницький національний технічний університет, м. Вінниця.

Андрухов Валерій Михайлович – к.т.н., доцент, Вінницький національний технічний університет, e-mail: vmandruchov@gmail.com.

S Potiekha
V. M. Andrukhov

REVIEW OF MACHINE LEARNING METHODS FOR OPTIMISATION OF FRAME BUILDING SCHEMES

Vinnitsia National Technical University

This paper solves the problem of optimal column placement in a building design while considering architectural constraints, minimizing bending moments, and ensuring uniform load distribution. The main focus is on combining traditional optimization methods (both linear and nonlinear) with modern machine learning techniques, such as heuristic algorithms (genetic algorithms, particle swarm optimization), deep neural networks, and reinforcement learning. For each algorithm, a simple Python implementation is provided, comparing execution time, solution accuracy, noise tolerance, and the tendency to converge to local minima.

The study begins by defining the initial data, including the building's geometry, material mechanical properties, and design loads. A simplified test case is introduced to illustrate how different optimization approaches perform under specific constraints (such as permissible coordinate spacing, minimum distances between column supports, and load uniformity requirements). A comparative analysis is conducted, resulting in a table that outlines computational complexity, average execution time, error rates, and stability indices. To enhance clarity, an integrated performance score is introduced, summarizing multiple criteria into a single metric.

The findings show that for relatively simple cases, local optimization methods (e.g., SLSQP) quickly and accurately converge to an optimal solution. However, heuristic algorithms like genetic algorithms (GA) and particle swarm optimization (PSO), despite taking more time, are better at avoiding local minima. Reinforcement learning and deep neural networks become more advantageous in complex, high-dimensional problems where numerous constraints and variables make global search and approximation models highly effective. The examples and tables presented in this paper serve as a solid foundation for further application of machine learning techniques in real-world structural design challenges.

Keywords: machine learning, heuristic algorithms, genetic algorithm, particle swarm method, reinforcement learning, deep neural networks

Potiekha Andriy – PhD student, Department of Civil and Environmental Engineering Vinnitsia National Technical University, Vinnitsia city.

Andrukhov Valeriy – PhD, Associate Professor, Vinnitsia National Technical University, e-mail: vmandruchov@gmail.com.