УДК 519.6:004.8

О.Г. РИБАЛЬЧЕНКО, Н.Н. ШАПОВАЛОВА, старші викладачі,
С.В. БІЛАШЕНКО, асистент, Криворізький національний університет

БАГАТОПОТОКОВІ ОБЧИСЛЕННЯ В ОПТИМІЗАЦІЇ ФУНКЦІОНАЛУ ЯКОСТІ МОДЕЛЕЙ МАШИННОГО НАВЧАННЯ МЕТОДОМ ІМІТАЦІЇ ВІДПАЛУ

Мета. Вивчення можливостей реалізації багатопотокових обчислень для алгоритму знаходження глобального мінімуму багатовимірної цільової функції, заснованого на стохастичному методі імітації відпалу, з метою оптимізації функціоналу якості моделей машинного навчання, вибір оптимальних значень параметрів застосування алгоритму для різних наявних обчислювальних потужностей, розроблення рекомендацій до застосування багатопотокових обчислень за певних початкових умов задачі багатовимірної оптимізації.

Методи. Використано числовий експеримент задачі паралельної оптимізації оціночного функціоналу якості моделей машинного навчання з розподіленням навчальної вибірки при різному розмірі пулу потоків, на вибірках різного розміру, для гладкої та негладкої оптимізаційних функцій.

Наукова новизна. Досліджена можливість використання багатопотокових обчислень з розподіленням навчальної вибірки при реалізації алгоритму знаходження глобального мінімуму багатовимірної цільової функції, заснованого на стохастичному методі імітації відпалу, та доцільність їх застосування до різних типів оптимізаційних функцій у машинному навчанні, вивчена закономірність зміни показника прискорення паралельних алгоритмів для різних обчислювальних потужностей.

Практична значимість виконаної роботи полягає в обґрунтуванні доцільності застосування багатопотокових обчислень з розподіленням навчальної вибірки при реалізації алгоритму імітації відпалу для оптимізації негладких цільових функцій. Емпірично знайдено оптимальні значення параметрів проведення експерименту для наявних обчислювальних ресурсів, що дозволяє значно збільшити швидкість виконання завдання мінімізації функції помилок регресійної моделі за критеріями середньоквадратичного відхилення.

Результати. Розроблено бібліотеку opti_methods методів багатовимірної оптимізації оціночного функціоналу якості моделей задач машинного навчання для мови програмування Python 3, реалізоване розподілення навчальної вибірки для різного розміру пулу потоків, знайдено оптимальні значення параметрів проведення експерименту для наявних обчислювальних ресурсів. Показано, що запропонований паралельний варіант алгоритму оптимізації методом імітації відпалу за схемою больцманівського гасіння може ефективно застосовуватися для вирішення завдань пошуку глобального мінімуму багатовимірної цільової функції.

Ключові слова: багатопотокові обчислення, показник прискорення, цільова функція, оптимізація, машинне навчання, функціонал якості, алгоритм імітації відпалу, градієнтний метод.

doi: 10.31721/2306-5435-2018-1-103-59-65

Проблема та її зв'язок з науковими та практичними задачами. Протягом останнього десятиріччя спостерігається невпинне підвищення інтересу до машинного навчання (МН) та аналізу даних як у фундаментальній науці, так і в прикладних дослідженнях. Це є результатом того, що стали доступними великі обчислювальні потужності та помітно збільшилися обсяги та складність накопичених даних. Ця галузь інформаційних технологій потребує визначення найбільш придатних, гнучких алгоритмів, що за умов впливу зовнішніх параметрів не вимагатимуть перегляду або заміни всієї моделі.

Як відомо, завдання оптимізації функціоналу якості моделей машинного навчання приводять до пошуку екстремумів цільової функції (ЦФ) різними методами. Наявність різноманітних обмежень на параметри, що оптимізуються, та багатоекстремальність ЦФ, як правило, ведуть до залучення великих обчислювальних потужностей і, відповідно, до неможливості знаходження рішення за прийнятний час при використанні одного комп'ютера.

В ході проведених досліджень методів оптимізації функціоналу якості моделей машинного навчання [1], було визначено, що обчислювальна складність стохастичних методів імітації відпалу та генетичного алгоритму різко зростає із збільшенням обсягу навчальної вибірки. Наслідком цього стає незадовільний час роботи цих алгоритмів і для гладких, і для негладких оптимізаційних функцій. Для оптимізації гладких функцій у задачах навчання за прецедентами градієнтний метод і метод Нелдера-Міда демонструють хороші результати – незалежно від обсягу вибірки, їх швидкість є прийнятною. Але градієнтні методи не використовуються для оптимізації негладких функції, оскільки такі функції є недиференційованими [2]. Тому задача поліпшення часу пошуку рішення для методу імітації відпалу є актуальною.

Дана проблема може бути вирішена застосуванням сучасних паралельних і розподілених

обчислювальних систем у поєднанні з використанням ефективних розпаралелених алгоритмів оптимізації [3].

У даній роботі була поставлена задача вивчення можливостей реалізації багатопотокових обчислень для алгоритму знаходження глобального мінімуму багатовимірної ЦФ з явними обмеженнями типу рівностей, заснованого на стохастичному методі імітації відпалу, з метою оптимізації функціоналу якості моделей машинного навчання.

Постановка завдання. Пошук глобального мінімуму функції $f : Rn$ при наявності явних обмежень здійснюється на деякій власній підмножині Ω метричного простору Rn

$$f(x) = f(x_1, \ldots, x_n) \to \min,\ x \in \Omega,\quad \Omega \subset R^n, \tag{1}$$

де підмножина Ω визначається обмеженнями типу рівностей $q(x) = 0$, де $q : Rn$.

Функція може мати багато локальних мінімумів. Якщо виконується нерівність $f(x^*) < f(x)$, $x \in X$, де $x \neq x^*$ — будь-яка точка множини X, то говорять про глобальний мінімум функції $f(x)$.

Існує проблема оптимізації параметрів $w_1, ..., w_N$ деякого алгоритму машинного навчання. Завдання оптимізації полягає в тому, щоб підібрати ці параметри таким чином, щоб алгоритм давав найкращий результат [4]. Зокрема, якщо якість роботи алгоритму описувати функцією якості $Q(w_1, ..., w_N)$ від його параметрів – вагових коефіцієнтів моделі, то задача оптимізації набуває вигляду $Q(a_1, ..., a_N) \to \max_{a_1,...,a_N}$. Розглянемо задачу навчання за прецедентами – задачу лінійної регресії

$$a(x) = w_0 + \sum_{j=1}^{d} w_j x^j,$$

де w_0 – вільний коефіцієнт, x – ознаки, w_j – вага x^j-ї ознаки, d – кількість ознак у вибірці.

Якщо додати (d+1)-у ознаку, яка на кожному об'єкті приймає значення 1, то лінійний алгоритм можна буде записати у більш компактному вигляді

$$a(x) = \sum_{j=1}^{d+1} w_j x^j = \langle w, x \rangle,$$

де використовується позначення $\langle w, x \rangle$ для скалярного добутку двох векторів.

Якість алгоритму оцінюється тим, наскільки точно отримана модель описує залежності даних у вибірці, тобто чим менша помилка (відхилення) на кожному об'єкті, тим вище якість алгоритму [5]. В якості міри помилки не може бути вибрано відхилення від прогнозу (y) $Q(a, y) = a(x) - y$, оскільки в цьому випадку мінімум функціоналу не буде досягнутий при правильній відповіді $a(x) = y$. Найпростіший спосіб – розрахувати модуль відхилення $|a(x) - y|$. Тоді функціонал якості для регресійної моделі набуває вигляду $Q(a, x) = \frac{1}{l}\sum_{i=1}^{l} |a(x_i) - y_i|$. Запишемо цей функціонал у вигляді функції від вектора вагових коефіцієнтів

$$Q(w, x) = \frac{1}{l}\sum_{i=1}^{l} |\langle w, x_i \rangle - y_i| \cdot$$

Необхідно підібрати коефіцієнти w такими, щоб помилка алгоритму була найменшою

$$Q(w, x) = \frac{1}{l}\sum_{i=1}^{l} |\langle w, x_i \rangle - y_i| \to \min_w. \tag{2}$$

Оскільки функція (2) негладка, то використання градієнтних методів для оптимізації функціоналу якості, заданого таким чином, стає неможливим. Тому доцільно використовувати не модуль відхилення $|a(x) - y|$, а квадрат відхилення прогнозу $(a(x) - y)^2$. Запишемо функцію якості як середньоквадратичний критерій відхилення через вектор вагових коефіцієнтів

$$Q(w, x) = \frac{1}{l}\sum_{i=1}^{l} (\langle w, x_i \rangle - y_i)^2 \to \min_w. \tag{3}$$

Таку цільову функцію (3), що має неперервну похідну на всій множині визначення, можливо мінімізувати градієнтними методами [6].

Аналіз досліджень і публікацій. Пошук глобального мінімуму функції (1) за наявності очевидних обмежень на варійовані параметри може бути здійснений методом імітації відпалу, запропонованим С. Кіркпатриком [4].

Цей метод являє собою алгоритмічний аналог фізичного процесу керованого охолодження і використовує впорядкований випадковий пошук нових станів системи з більш низькою температурою.

В процесі повільного керованого охолодження розплавленого матеріалу, так званого відпалу, кристалізація розплаву супроводжується глобальним зменшенням його енергії E, проте допускаються ситуації, в яких вона може на деякий час зростати (при підігріві розплаву для запобігання його занадто швидкого охолодження). Завдяки допустимості короткочасного підвищення енергетичного рівня, стає можливим вихід з пасток локальних мінімумів енергії, які виникають при реалізації процесу. Тільки зниження температури T до абсолютного нуля унеможливлює будь-яке самостійне підвищення енергетичного рівня розплаву [5].

В цьому випадку елементи x підмножини Ω у функції (1) представляють собою низку станів уявної фізичної системи, а значення функції $f(x)$ у цих точках використовується у якості енергії системи $E = f(x)$ [6]. В кожен момент часу температура T системи, що знаходиться в стані x, вважається заданою, тобто вона зменшується з плином часу за певним законом. Новий стан системи x' вибирається відповідно до заданого породжувального сімейства ймовірнісних розподілів $\zeta(x, T)$, яке при фіксованих x та T задає випадковий елемент $x' = G(x, T)$.

Після генерації нового стану $x' = G(x, T)$ система із ймовірністю $p(\Delta E, T)$ переходить до наступного кроку в стані x', в іншому випадку повторюється процес генерації x'. Тут $\Delta E = f(x) - f(x')$ — приріст енергії. За ймовірність прийняття нового стану $p(\Delta E, T)$ вибирається або точне значення відповідної фізичної величини $p(\Delta E, T) = 1/(1 + \exp(\Delta E / T))$, або наближене значення $p(\Delta E, T) = \exp(\ \Delta E / T)$. Друга формула використовується найчастіше. При цьому $p(\Delta E, T) > 1$ у випадку $\Delta E < 0$, і тоді вважається, що ймовірність прийняття нового стану $p(\Delta E, T)$ дорівнює 1. Відповідно, якщо у новому стані x' функція, що оптимізується, має краще значення, тобто $f(x') < f(x)$, то перехід у цей стан відбудеться в будь—якому випадку [7]. Пошук мінімуму ЦФ закінчується, коли температура T зменшується до заданого рівня T_{end}.

Будь-яка схема відпалу задається наступними параметрами: законом зміни температури $T(k)$, де k — номер кроку; породжувальним сімейством ймовірнісних розподілів $\zeta(x, T)$; функцією ймовірності прийняття нового стану $p(\Delta E, T)$.

Імітація відпалу є універсальним методом пошуку глобального мінімуму цільової функції і має як переваги, так і недоліки. До переваг відносять: можливість пошуку рішень для складних нелінійних задач, можливість роботи з даними з великою кількістю шумів та перешкод; здатність виходу з локальних мінімумів, універсальність методу, відносну легкість модифікації, адаптації та технічної реалізації. Серед недоліків, як правило, відзначають наступні: залежність якості рішення від часу його отримання, необхідність адаптації параметрів для кожного конкретного завдання [8].

Слід зазначити, що імітація відпалу є послідовним методом, це в деякій мірі ускладнює розпаралелювання, але тим не менш, були розроблені підходи, що дозволяють шляхом розділення обчислень досягти поліпшення часу пошуку рішення [9]. Перший спосіб — паралельний запуск алгоритму імітації відпалу — передбачає обчислення глобального мінімуму цільової функції одночасно на декількох процесорних пристроях з подальшим вибором кращого рішення керуючим вузлом. Другий спосіб — паралельний запуск алгоритму з обміном результатами — припускає після закінчення певної кількості ітерацій обмін між процесорними пристроями і вибір кращого результату для продовження обчислень. Третій спосіб — розбиття простору рішень на області — припускає запуск послідовного алгоритму імітації відпалу в кожній з виділених областей рішень з вибором найкращого рішення після закінчення обчислень.

Також з метою поліпшення часових показників роботи імітації відпалу, були розроблені гібридні методи, які найчастіше представляють собою комбінацію імітації відпалу та генетичних алгоритмів.

Для оцінки ефективності паралельних алгоритмів використовують різні підходи, найбільш поширеним з яких є показник прискорення [10]. Прискорення, що отримується при роботі алгоритму на p процесорах – це відношення часу роботи алгоритму на одному процесорі до часу роботи того ж алгоритму на p процесорах. Лінійне прискорення спостерігається, коли паралельний алгоритм на p процесорах працює в p разів швидше, ніж на одному процесорі. Сублінійне прискорення досягається, коли спостерігається збільшення швидкості розрахунків менше,

ніж у p разів. Суперлінійне прискорення спостерігається, коли збільшення швидкості розрахунків більше, ніж у p разів. Закон Амдала [11] дозволяє обчислити верхню межу прискорення, яке можна очікувати від паралельної реалізації алгоритму.

Викладення матеріалу та результати. В даній роботі ставилося за мету розробити метод оптимізації, що дозволяє за невеликий час надійно знаходити саме область глобального мінімуму. Відповідно на основі аналізу різних схем імітації відпалу, проведеного в [12], був обраний алгоритм пошуку глобального мінімуму ЦФ (1) методом імітації відпалу за схемою больцманівського гасіння. Опишемо основні кроки, які необхідно виконати для реалізації цього алгоритму.

Спочатку випадково обирається початкова точка $x = x_0$, $x \in \Omega$. Поточне значення енергії E встановлюється в значення $f(x_0)$. Встановлюється максимальна початкова температура $T(k)=T_0$, $T_0 > T_{end}$. Кожна k-та ітерація циклу складається з таких кроків. Спочатку генерується нова точка $x' = G(x, T(k))$, щільність породжувального сімейства ймовірнісних розподілів $g(x';x,T) = (2\pi T)^{n/2} \exp(-|x - x|^2/(2T))$. Потім обчислюється значення енергії в новому стані $E' = f(x')$, обчислюється приріст енергії $\Delta E = E' - E$. Якщо приріст енергії від'ємний, то виконується перехід до нового стану $x = x'$, $E = E'$, значення глобального мінімуму змінюється. Якщо ж ні, то виконується обчислення ймовірності переходу до нового стану $p(\Delta E, T(k)) = \exp(-\Delta E / T(k))$ та генерується випадкове число α з інтервалу [0;1]. Якщо $\alpha < p(\Delta E, T(k))$, то встановлюється $x = x'$, $E = E'$ та виконується перехід до наступної ітерації. Інакше повторюється другий крок ітерації, допоки не буде знайдено підходящу точку x'.

В результаті тестування описаного алгоритму, виявлено, що рішення (область мінімуму) дійсно знаходиться за прийнятний час, але недоліком є можливість знаходження локальних мінімумів, а не глобального.

Проблему, що виникла, можна вирішувати шляхом багаторазових повторних запусків процедури пошуку з однакових початкових умов, тому що генерація нових точок у факторному просторі здійснюється спочатку випадково, а потім вибором кращого значення. Наслідком таких дій стає пропорційне збільшення обсягів використання обчислювальних потужностей і, відповідно, витрат часу, що неприпустимо для складних, багатофакторних ЦФ. У такій ситуації підвищення ефективності роботи методу з точки зору мінімізації часу обчислень можна досягти шляхом розпаралелювання алгоритму між декількома процесорними пристроями.

Незручністю методу імітації відпалу з точки зору розпаралелювання є те, що кожен новий стан системи x' генерується на основі її попереднього стану x, і, отже, немає можливості розпаралелювання будь-якої частини алгоритму.

Тому було запропоновано здійснювати обчислення мінімумів ЦФ (1) одночасно на декількох процесорах з розподіленням навчальної вибірки на області (клієнтська частина алгоритму) і наступним вибором кращого рішення (глобального) у серверній частині алгоритму.

У клієнтській частині, яка виконується декількома процесорами в декількох областях простору рішень, здійснюється пошук мінімуму ЦФ (1) методом імітації відпалу за схемою больцманівського гасіння за допомогою описаного вище алгоритму. Потім результати роботи клієнтської частини алгоритму передаються в серверну частину, де й здійснюється вибір рішення з найменшим значенням ЦФ. Такий підхід має забезпечувати глобальність знайденого мінімуму.

В ході дослідження була розроблена бібліотека opti_methods методів багатовимірної оптимізації оціночного функціоналу якості моделей задач машинного навчання мовою Python 3.6. З використанням цієї бібліотеки були проведені числові експерименти паралельної оптимізації оціночного функціоналу якості моделей задач машинного навчання методом імітації відпалу з розподіленням навчальної вибірки при різному розмірі пулу потоків, на вибірках різного розміру, для гладкої та негладкої оптимізаційних функцій.

Таблиця 1

Час роботи алгоритму на різній кількості обчислювальних потоків для конфігурації 1, 2

Розмір пула	1	2	4	8	12
Час роботи алгоритму, с					
Конфігурація 1	42,9	27,09	19,82	27,75	36,8
Конфігурація 2	13,85	12,45	7,23	9,05	14,33

Результати числового експерименту застосування паралельних обчислень мінімізації функції помилок регресійної моделі за критеріями середньоквадратичного відхилення методом імітації відпалу наведено в табл. 1. Експеримент був проведений на двох різних конфігураціях обчислювальних потужностей:

1 – процесор Intel Core i7-7500U CPU 2.70 GHz, ядер: 2, логічних процесорів: 4, RAM 12 Гб; 2 – процесор Xeon E5-1620V2, частота 3.7 GHz, ядер: 4, логічних процесорів: 8, RAM 16 Гб.

Проілюструємо результати роботи алгоритмів графіками (рис. 1, 2).

а) *б)*

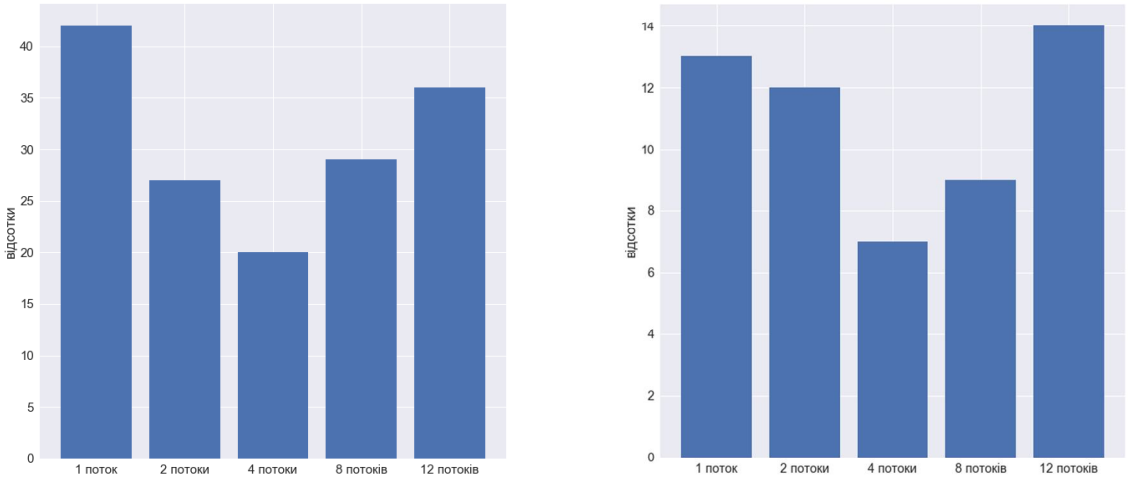

Рис. 1. Час роботи алгоритму на різній кількості потоків для конфігурації 1 – *(а)* і для конфігурації 2 – *(б)*

а) *б)*

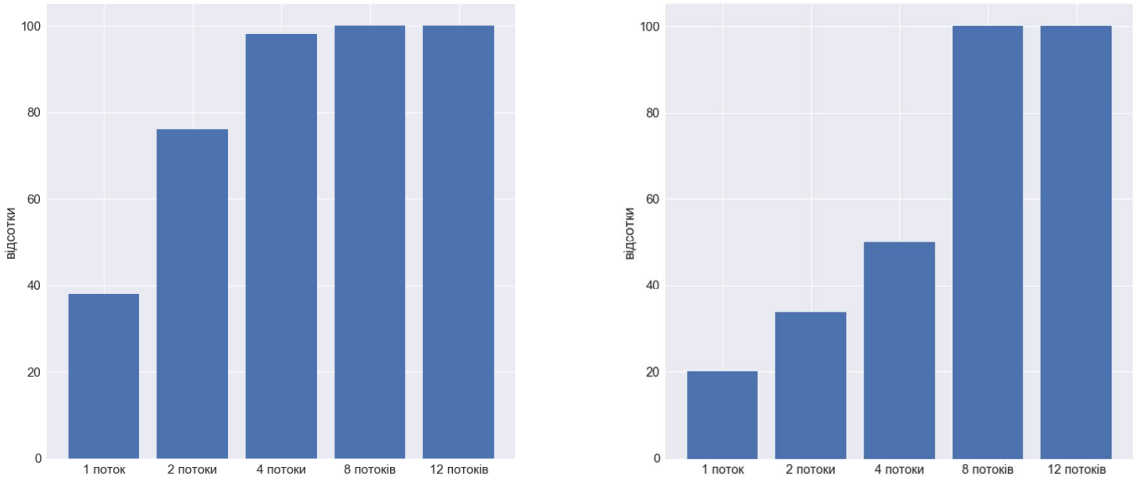

Рис. 2. Навантаження процесора для конфігурації 1 – *(а)* і для конфігурації 2 – *(б)*

Отриманий результат цілком відповідає очікуванням: для конфігурації 2 час виконання алгоритму менший, ніж у конфігурації 1, для всіх розмірів пулу. При цьому слід зазначити, що динаміка часу виконання розрахунків зберігається для будь-якої конфігурації – для 4-ох потоків швидкість реалізації алгоритму найвища.

Розподіл обчислювальних потужностей процесора виконується таким чином, що зазвичай пули мають максимальну кількість потоків. Якщо всі потоки зайняті, то додаткові задачі розміщуються у черзі, де знаходяться до того моменту, поки не з'являться вільні потоки. Це явище спостерігалося під час експерименту при розмірі пулу в 8 та 12 потоків на обох конфігураціях обчислювальних машин. Тому оптимальний розмір пулу складає 4 потоки.

Навантаження на процесор при різному розмірі пулу ілюструє рис. 2.

Використаємо для оцінки ефективності паралельних алгоритмів показник прискорення. Прискорення, що отримується при роботі алгоритму на p процесорах – це відношення часу роботи алгоритму на одному процесорі до часу роботи того ж алгоритму на p процесорах. Розрахуємо і визначимо тип прискорення паралельних обчислень для методу імітації відпалу (рис. 3).

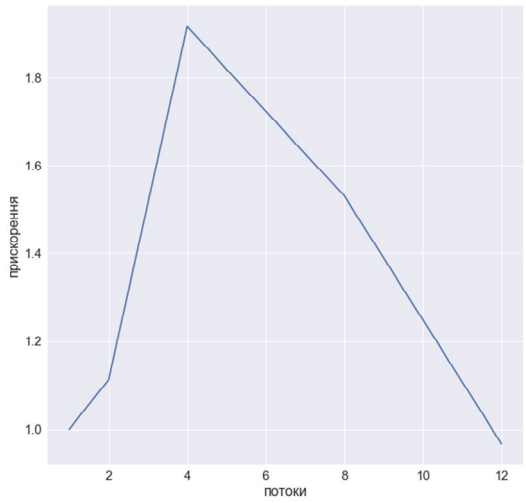

Рис. 3. Прискорення обчислень при розпаралелюванні алгоритма імітації відпалу

Оскільки час розрахунків на піку прискорення (4 потоки) майже вдвічі менше, ніж для роботи алгоритму на одному потоці, можна зробити висновок, що розпаралелювання алгоритму імітації відпалу з розподіленням навчальної вибірки на області дає сублінійне прискорення обчислень, що властиво для стохастичних алгоритмів.

Проведені дослідження показали, що при однопоточному обчисленні час виконання завдання мінімізації функції помилок регресійної моделі за критеріями середньоквадратичного відхилення достатньо великий за умов значного недовантаження процесорної потужності. Збільшення розміру пулу потоків дозволило емпірично знайти оптимальні значення параметрів проведення експерименту за наявних обчислювальних ресурсів.

Висновки та напрямок подальших досліджень. Проаналізувавши результати числового експерименту мінімізації функції помилок регресійної моделі за критеріями середньоквадратичного відхилення, можна зазначити, що запропонований паралельний варіант алгоритму оптимізації методом імітації відпалу за схемою больцманівського гасіння може ефективно застосовуватися для вирішення завдань пошуку глобального мінімуму багатовимірної ЦФ. Експериментально визначено, що для протестованих конфігурацій обчислювальних пристроїв оптимальний розмір пулу, який одночасно забезпечує високу швидкість розрахунків та доцільне навантаження на процесор, складає 4 потоки. При цьому час виконання розрахунків був майже вдвічі менший, ніж при роботі алгоритму імітації відпалу на одному потоці, тобто запропоноване розпаралелювання алгоритму з розподіленням навчальної вибірки на області дає сублінійне прискорення обчислень. В ході подальших досліджень отримані результати планується перевірити в ході експерименту глибокого навчання багатошарових штучних нейронних мереж.

Список літератури

1. Шаповалова Н.Н., Рибальченко О.Г., Куропятник Д.І. Порівняльний аналіз методів оптимізації функціоналу якості моделей машинного навчання // Вісник Криворізького національного університету, Кривий Ріг. - 2018.

2. **Луис Педро Коэльо**, Вилли Ричарт. Построение систем машинного обучения на языке Python. 2-е издание / пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2016. – 302 с.

3. **Петер Флах**. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. Учебник / пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2015. – 408 с.

4. Kirkpatrick S., Gelatt C. D., Vecchi M. P. Optimizationby simulated annealing // Science. - 1983. - Vol. 220. pp. 671–680.

5.Оптимізаційні методи та моделі. Підручник. / [Л. В. Забуранна, Н. В. Попрозман, Н. А. Клименко, О. І. Попрозман, С. В. Забуранний]. – К.: __, 2014. – 372 с.

6. **Банди Б.** Методы оптимизации. Вводный курс: пер. с англ.Шихеева О.В. - М.: Радио и связь, 1988. - 128 с.

7. **Вітлінський В. В.**, Наконечний С. І., Терещенко Т. О. Математичне програмування: Навч.-метод. посібник для самост. вивч. дисц. – К.: КНЕУ, 2001. – 248 с.

8. Лопатин А.С. Метод отжига // Стохастическая оптимизация в информатике // СПб.: Изд-во СПбГУ – 2005, Вып. 1. - сс. 133–149.

9. Троценко Р.В., Посашенко А.В. Обзор метода имитации отжига и его модификаций в аспекте применимости к решению задачи комплектации вычислительной системы минимальной стоимости в условиях дефицита времени // Наука вчера, сегодня, завтра: сб. ст. по матер. XI междунар. науч.-практ. конф. № 4(11). – Новосибирск: Сиб АК - 2014.

10. Орлянская И. В. Современные подходы к построению методов глобальной оптимизации // [Електронний ресурс] // Исследовано в России. – 2002 - Режим доступу до ресурсу: http://zhurnal.ape.relarn.ru/articles/2002/189.

11. Amdahl G. The Validity of Single Processor Approach to Achieving Large Scale Computing Capabilities // AFIPS Proc. Vol. 30, pp. 483-485, 1967.

12. Ingber L. Simulated Annealing: Practice versus theory // Mathematical and Computer Modelling. - 1993. - Vol. 18(11) - pp. 29–57.

Рукопис подано до редакції 28.11.2017