Machinery & Energetics. Journal of Rural Production Research. Kyiv. Ukraine. 2020, Vol. 11, No 4, 131-141
ISSN 2663-1334 (print), ISSN 2663-1342 (online), www.journals.nubip.edu.ua/index.php/Tekhnica

http://dx.doi.org/10.31548/machenergy2020.04.131

УДК 621.331.922

МЕТОД ПОШУКУ ЕКСТРЕМУМУ БАГАТОМІРНИХ ФУНКЦІЙ ПРИ ВИРІШЕННІ ІНЖЕНЕРНИХ ЗАДАЧ МАШИН ДЛЯ ЛІСОТЕХНІЧНИХ РОБІТ

Л. Л. Тітова, О. В. Надточій

Національний університет біоресурсів і природокористування України, Україна.

Стаття з спеціальності: 133 – галузеве машинобудування.

Кореспонденція авторів: o.nad@ukr.net.

Історія статті: отримано – червень 2020, акцептовано – жовтень 2020.
Бібл. 22, рис. 5, табл. 3.

Анотація. Дана стаття присвячена аналізу найбільш поширених методів оптимізації, що використовуються в практичних інженерних задачах пошуку екстремуму багатомірних функцій і формування на основі виявлених властивостей рекомендацій щодо вибору кращого на різних наборах даних. В процесі аналізу були розглянуті різні реалізації методів градієнтного спуску, імпульсні методи, адаптивні методи та квазіньютонівські методи, узагальнені переваги та проблеми кожного з методів при їх використанні. Розроблена комп'ютерна програма, яка реалізує всі розглянуті методи. Проведений обчислювальний експеримент за трьома функціями показав, що найефективнішими виявилися методи нульового порядку – Розенброка та нульового порядку – Пауелла.

Ключові слова: методи оптимізації, метод градієнтного спуску, стохастичний градієнт, квазіньютонівські методи, цільова функція.

Постановка проблеми

В лісопромисловому комплексі існує велика кількість задач, які можуть бути вирішені за допомогою комбінаторних методів оптимізації: завдання формування комплексів сільськогосподарських машин, побудови сівозмін, завдання визначення рентабельності сільськогосподарських підприємств і т.п. Це пов'язано з дискретною природою множин, на яких здійснюється оптимізація.

Наприклад, – дискретні безлічі культур, полів, дозволених і неприпустимих сівозмін, безлічі лісогосподарської техніки і технологічних операцій на виконання лісогосподарських робіт, тощо. Основною метою організації технологічного процесу при цьому є пошук поєднання шуканих параметрів, який забезпечує екстремум по одному з критеріїв якості – критерію підвищення біологічного потенціалу рослин, зменшення енерговитрат чи зменшення впливу на екосистему. Вибір швидкого та надійного алгоритму пошуку такого екстремуму є актуальним.

Аналіз останніх досліджень

В процесі проектування інтелектуальних систем управління часто виникає завдання визначення найкращих значень параметрів чи структури об'єктів. Таке завдання називається оптимізацією. Сьогодні оптимізаційні задачі і задачі прийняття рішень моделюються і вирішуються в самих різних областях техніки [1]. До навичок математичного обґрунтування прийняття рішень відносяться навички математичного моделювання оптимізаційних задач, вибору адекватного математичного забезпечення (методу, алгоритму, програмної системи) з необхідним обґрунтуванням, аналізу отриманих результатів та їх інтерпретації в термінах предметної області.

Для оцінки наближення локального екстремуму, отриманого застосуванням методів класичного і стохастичного градієнтного спуску, імпульсних, адаптивних та квазіньютонівських, що звужують околицю і вектор спаду. Визначимо залежність розмірності задачі і тимчасових витрат при пошуку глобального екстремуму функції мети.

Методи мінімізації функції при нелінійних обмеженнях можна розбити на два класи. До першого класу відносяться ті з них, в яких пошук умовного мінімуму зводиться до безумовної мінімізації функції, що виходить додаванням штрафу за неузгодженість обмежень до цільової функції. У методах другого класу обмеження враховуються безпосередньо, і пошук йде по допустимим точкам з монотонно зменшуваними значеннями цільової функції. До першого класу відносяться методи бар'єрних і штрафних функцій [5]. До другого класу відносяться методи прямого і випадкового пошуку для вирішення завдань з обмеженнями [1, 2, 3, 6-22].

Мета досліджень

Мета дослідження полягала у аналізі багатомірних методів оптимізації пошуку та знаходження оптимального за швидкістю сходження (кількістю ітерацій), точністю пошуку мінімуму функції та швидкістю виконання алгоритму. При

цьому планувалося провести обчислювальний експеримент для трьох функцій за допомогою розробленої комп'ютерної програми.

Метою цього обчислювального експерименту, з одного боку, є важливе самостійне значення для отримання умов застосування методів мінімізації функцій до вирішення практичних інженерних завдань, а з іншого це дасть можливість здійснити пошук глобального екстремуму для реальної (найбільшої) розмірності розв'язуваної задачі.

Результати досліджень

Завдання оптимізації в цілому зводиться до задачі пошуку екстремуму (мінімуму або максимуму) цільової функції з заданими обмеженнями. Її математична постановка виглядає наступним чином: необхідно визначити значення вектору змінних $x = (x_1, x_2 \dots x_n)$, які задовольняють обмеженнями виду:

$$f_i(x_1, x_2 \dots x_n) \le b_i, \quad (1)$$

для всіх $i = 1 \dots k$ при яких досягається максимум чи мінімум цільової функції:

$$f_i(x_1, x_2 \dots x_n) \to (min).$$

Допустимим рішенням задачі буде таке рішення, яке задовольняє її обмеження (1). Сукупність допустимих рішень задачі називають областю допустимих рішень (ОДР) D. Остаточним рішенням завдання є пара (x^{opt}, $f^{opt}(x^{opt})$), що складається з оптимального рішення і оптимального значення цільової функції.

Методи математичного програмування дають велику різноманітність алгоритмів вирішення даного завдання. В цілому алгоритми пошуку реалізують методи спуску до екстремуму, при яких значення цільової функції послідовно поліпшується аж до досягнення екстремуму. Залежно від можливості знаходження алгоритмом локального чи глобального екстремуму, вони діляться на алгоритми локального і глобального пошуку.

Для пошуку локального екстремуму чи одного з локальних екстремумів на безлічі допустимих рішень призначені алгоритми, в яких цільова функція приймає максимальне або мінімальне значення. При їх побудові можуть використовуватися як детермінований спуск в область екстремуму, так і випадковий пошук. Серед детермінованих методів розрізняють методи нульового порядку і градієнтні (1-го і 2-го порядку). Перші обраховують значення функції, яка оптимізується. Другі використовують приватні похідні відповідного порядку. Для пошуку екстремуму в випадках, коли вид оптимізується функції відомий не повністю, або її структура занадто складна, застосовуються методи стохастичного програмування або нейронних мереж. Ефективність процедури пошуку оптимуму – можливість відшукання рішення і збіжність до вирішення по швидкості залежать від виду функції і застосовуваного для неї методу. Розглянемо стратегію кожного методу більш детально, досліджуючи для визначеності мінімізацію цільової функції.

Прямі методи (нульового порядку). З прямих методів найбільш відомі методи: координатного спуску – почергова оптимізація параметрів уздовж осей одним з відомих одновимірних методів; спірального координатного спуску; обертових координат (метод Розенброка); пошуку по симплекс; Хука-Дживса з пошуком за зразком; Розенброка; Пауела, тощо.

Метод координатного спуску полягає в тому, що в якості напрямків траєкторії спуску від попередньої точки пошуку x^{k-1} до подальшої x^k приймаються по черзі напрямки координатних осей x_i $(i = 1 \dots n)$. Після спуску на один крок по координаті x_1 відбувається перехід до спуску на один крок по координаті x_2 і далі, поки не буде знайдена наступна точка пошуку x^k з координатами x_1^k, $x_2^k \dots x_n^k$. Рух по траєкторії спуску триває до тих пір, поки не будуть досягнуті околиці точки мінімуму x^{opt} цільової функції, що визначаються точністю обчислень. Для пошуку координат точки x^k на кожному кроці ітерації можна використовувати будь-який з методів одновимірної мінімізації: метод золотого перетину, метод поділу відрізка навпіл, метод інтерполяції-екстраполяції, тощо.

Метод спірального координатного спуску відрізняється від розглянутого вище лише тим, що крок h змінюється кожного разу при переході від пошуку мінімуму по одній змінній до пошуку мінімуму за іншою змінною. У тривимірному просторі це нагадує спуск до впадини по спіралі. Зазвичай цей метод дає деяке скорочення часу пошуку хоча цей метод менш ефективний при наявності поверхонь з "ярами". Спроба просування в будь-якому напрямку може викликати "погіршення" цільової функції. У той же час просування уздовж "яру" може давати "покращення" цільової функції.

Метод Розенброка спрямований на усунення одного з недоліків методу координатного спуску - високу чутливість до вибору системи координат. В процесі пошуку методом Розенброка проводиться поворот координатних осей так, щоб одна з осей була направлена вздовж напрямку "яру". Розглянемо алгоритм методу в разі одновимірної мінімізації. На кожній ітерації процедура здійснює ітеративний пошук вздовж n лінійно незалежних і ортогональний напрямків. Коли отримана нова точка в кінці ітерації, будується нова множина ортогональних векторів (рис. 1).

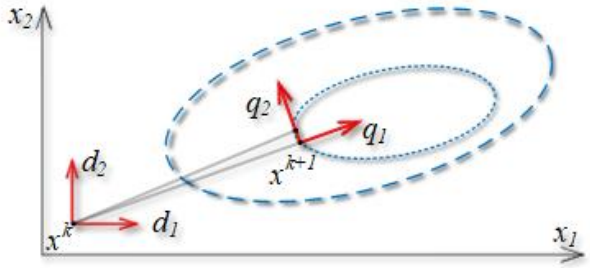


Рис. 1. Схема побудови нової системи координат за методом Розенброка

Fig. 1. The scheme construction a new coordinate system by the method of Rosenbrock.

Побудова напрямків пошуку. Нехай $d_1 \dots d_n$ – лінійно незалежні вектори, за нормою рівні одиниці. Припустимо, що ці вектори взаємно ортогональні, тобто $d_i \cdot d_j = 0$ для $i \neq j$. Починаючи з поточної точки x^k, цільова функція послідовно мінімізується уздовж кожного з напрямків, в результаті чого виходить точка x^{k+1}. Новий набір напрямків $q_1 \dots q_n$ будується за допомогою процедури Грамма-Шмідта:

$$a_j = \begin{cases} d_j, IF\ \lambda_i = 0 \\ \sum_{i=j}^{n} d_i \cdot \lambda_i, IF\ \lambda_i \neq 0' \end{cases} \qquad q_i = \frac{b_j}{|b_j|}. \qquad (2)$$

$$b_j = \begin{cases} a_j, IF\ j = 0 \\ a_j - \sum_{i=1}^{j-1}(a_i \cdot q_i) \cdot q_i, IF\ j \geq 2 \end{cases}$$

Нові напрямки, побудовані описаним чином, є лінійно незалежними і ортогональними. І хоча метод Розенброка усуває проблеми, пов'язані з отриманням рішення заданої точності, проте такий підхід порівняно збільшує час пошуку, що є відносним недоліком описаного методу.

Метод Нелдера-Міда (пошуку по симплексу) використовує геометричну конфігурацію, яка називається симплекс. Симплекс – це опуклий багатогранник з числом вершин, рівним $n + 1$, де n - розмірність простору. Його важливою властивістю є можливість побудови нового симплексу на будь-якій грані вихідного, шляхом перенесення обраної вершини на деяку відстань уздовж прямої, що з'єднує цю вершину з центром ваги інших вершин симплекса.

Алгоритм починається з побудови регулярного симплекса в просторі незалежних змінних задачі і оцінювання значення цільової функції в його вершинах. Потім точка x_i з найбільшим значенням функції віддзеркалюється через центр ваги інших точок:

$$x_c = \frac{1}{N}\sum_{i \neq j, i=0}^{N} x_i. \qquad (3)$$

Нова точка використовується як вершина нового симплекса. Ітерації тривають до тих пір, поки або не буде досягнута точка мінімуму, або не почнеться циклічний рух по двох або більше симплексах.

Серед методів деформації вихідного симплекса (яка відбувається після відкидання найгіршою вершини і подальшому пошуку нової підходящої вершини) зустрічається: відображення вершини (вершина просто відбивається щодо однієї зі сторін симплекса), редукція (симплекс зберігає свою форму, але розміри його зменшуються), розтягнення, стиснення. Однак і цей метод не дивлячись на простоту містить певні недоліки: виникають труднощі пов'язані з масштабуванням завдання (в реальних задачах різні змінні часто не співставні між собою за значеннями); алгоритм працює повільно (не використовується інформація попередніх ітерацій); не існує простого способу зміни розмірів симплекса без перерахунку всіх значень цільової функції.

Метод Хука-Дживса являє собою комбінацію двох типів пошуку: досліджуваний пошук і пошук за зразком. Перший орієнтований на виявлення характеру локальної поведінки цільової функції і визначення напрямків уздовж "яруг". Задається величина кроку, яка може бути різною для різних координатних напрямків і змінюватися в процесі пошуку.

Якщо значення цільової функції в пробній точці не перевищує значення у вихідній, то крок пошуку розглядається як успішний. В іншому випадку, необхідно повернутися в попередню точку і зробити крок в протилежному напрямку. Після перебору всіх N координат досліджуваний пошук закінчується. Отримана точка називається базовою.

Пошук за зразком полягає в реалізації єдиного кроку з отриманої базової точки вздовж прямої, що з'єднує її з попередньою базовою точкою. І нова точка будується за формулою:

$$x_b^{k+1} = x^k + \lambda \cdot (x^k - x^{k-1}), \qquad (4)$$

де x^k – поточна базова точка; x^{k-1} – попередня базова точка; x_b^{k+1} – точка, побудована при русі за зразком; λ – параметр алгоритму.

Якщо рух за зразком не приводить до зменшення цільової функції, то точка x_b^{k+1} фіксується в якості тимчасової базової точки і знову проводиться досліджуваний пошук з цієї точки. Якщо в результаті виходить точка зі значенням функції меншим, ніж у x^k, то вона розглядається як нова базова точка x^{k+1}. Якщо досліджуваний пошук невдалий, то відбувається повернення в x^k і проводиться пошук в протилежному напрямку. Якщо він також не приводить до успіху, то величина кроку зменшується і відновлюється досліджуваний пошук. Пошук завершується, коли величина кроку стає досить малою. Перевагами даного методу є проста стратегія пошуку і невеликий обсяг необхідної пам'яті. Однак алгоритм заснований на циклічному русі по координатам, а це може привести до виродження алгоритму в нескінченну послідовність. Для запобігання ставлять обмежувач ітерацій, після якого алгоритм зупиняється.

Це не всі методи нульового порядку, проте основні ми розглянули.

Градієнтні методи. Група методів, ітераційні процеси яких для вирішення завдань безумовної оптимізації, на кожному кроці збігаються з антиградієнтом функції, називаються градієнтними методами, чи методами спуску. Їх ще називають методами першого порядку, або методами спуску. Ці методи використовують як значення функції, так і значення приватних похідних першого порядку, тому можуть бути застосовані для мінімізації функцій, які диференціюються. Методи першого порядку сходяться швидше методів прямого пошуку, так як в них враховуються похідні, що характеризують напрямок найбільш швидкого спадання функції. Розглянемо деякі з них:

Алгоритм найшвидшого спуску реалізує ітераційну процедуру руху до мінімуму з довільно вибраної початкової точки в напрямку найбільш сильного зменшення функції, визначеному в околиці поточного значення аргументу функції, що мінімізується. Такий напрям протилежний напрямку, що задається вектором градієнта $grad\, f(x) = \nabla f(x)$ функції, що мінімізується $f(x)$. Загальна формула для знаходження значення аргументу x^{k+1} за значенням x^k, знайденому на k–му кроці роботи алгоритму найшвидшого спуску:

$$x^{k+1} = x^k + \lambda^k \cdot s^k,$$

де s^k – вектор одиничної довжини в напрямку, протилежному напрямку градієнта $\nabla f(x^k)$, визначеному в точці x^k; λ^k – крок градієнтної процедури.

$$s^k = \frac{-\nabla f(x^k)}{|\nabla f(x^k)|}, \quad (5)$$

де $|\nabla f(x^k)|$ – норма вектору градієнта.

Алгоритми найшвидшого спуску розрізняються за способом визначення кроку λ^k. Якщо крок λ^k не залежить від k (є постійним), то в околицях екстремуму виникають незгасаючі коливання, амплітуда яких залежить від величини λ і від форми функції, яка мінімізується. Використання постійного кроку: дозволяє побудувати найбільш простий варіант алгоритму; при великих значеннях λ забезпечує швидкий рух до екстремуму, проте призводить до помітних змін на околицях екстремуму; при малих значеннях λ призводить до низької швидкості збіжності до екстремуму; інформація про прийнятну величиною кроку λ отримується у ході налагодження алгоритму. Якщо далеко від екстремуму функція $f(x)$ має малий градієнт, швидкість збіжності може виявитися неприпустимо повільно. Цю проблема вирішують шляхом модифікації алгоритму.

Метод сполучених градієнтів Флетчера і Рівса напрямок спуску у якого відхиляється від напрямку антиградієнта за рахунок додавання до нього вектору напрямку, використовуваного на попередньому кроці, помноженого на деяке позитивне число. Напрями пошуку на кожній ітерації визначаються за допомогою формули:

$$s^k = -\nabla f(x^k) + \sum_{i=0}^{k-1} \alpha^i \cdot s^i. \quad (6)$$

Значення параметрів вибираються таким чином, щоб напрямок був пов'язаним з усіма побудованими раніше напрямками пошуку. Це можливо коли виконується наступна умова:

$$\alpha_{k-1} = \frac{(\nabla f(X_k))^2}{(\nabla f(X_{k-1}))^2}. \quad (7)$$

Практичні дослідження [13] показали, що цей метод сходиться швидше методу найшвидшого спуску, причому його ефективність зростає на завершальних етапах пошуку мінімуму функції. Крім цього, слід зазначити, що даний метод може бути використаний при мінімізації функцій з розривними похідними. Пошук "не зависає на зламі", а йде уздовж лінії, що з'єднує точки зламів ліній рівня, яка, як правило, проходить через точку мінімуму.

Градієнтні методи є досить ефективними, але на початковому етапі мінімізації. На наступних етапах, коли пошукові точки знаходяться поблизу точки мінімуму, необхідно застосовувати методи, що мають більш високу швидкість збіжності. Цими методами є методи другого порядку, до яких відносяться метод Ньютона і пов'язані з ним квазиньютонівські методи.

Метод Ньютона заснований на квадратичної апроксимації функції, що мінімізується в околиці точки x^k. Мінімум квадратичної функції легко знайти, прирівнюючи її градієнт до нуля. Можна відразу ж обчислити положення екстремуму і вибрати його в якості наступного наближення до точки мінімуму. Обчислюючи точку нового наближення за формулою: $x^{k+1} = x^k + \Delta x^k$, а розклавши $f(x^{k+1})$ у ряд Тейлора, отримаємо квадратичну апроксимацію.

$$f_{sq}(x^{k+1}) = f(x^k) + \\ +\left(\nabla f(x^k)\right)^T x^k + \\ +\frac{1}{2!}(x^k)^T \nabla^2 f(x^k) x^k. \quad (8)$$

За умови мінімуму шукають довжину кроку x^k:

$$x^k = -[\![\nabla^2 f(x^k)]\!]^{-1} \cdot \nabla f(x^k). \quad (9)$$

Переваги методу Ньютона:

- якщо функція, що мінімізується є квадратичною, метод дозволить знайти мінімум за один крок;
- якщо функція належить до класу поверхонь обертання (тобто володіє симетрією), то метод також забезпечує збіжність за один крок;
- якщо функція несиметрична, то метод не забезпечує збіжність за кінцеве число кроків. Але для багатьох функцій досягається набагато більш висока швидкість збіжності, ніж при використанні інших модифікацій методу найшвидшого спуску.

Недоліки методу Ньютона пов'язані з необхідністю обчислень і обернених матриць других похідних. При цьому не тільки витрачається машинний час, оскільки можуть з'явитися значні обчислювальні похибки, якщо матриця $\nabla^2 f(x^k)$ виявиться погано обумовленою.

Метод Девідона-Флетчера-Пауелла званий також методом змінної метрики потрапляє в загальний клас квазіньютонівських процедур, в яких напрямки пошуку на k – тій ітерації задаються у вигляді: $s^k = -h^k \nabla f(x^k)$. Напрямок градієнта відхиляється завдяки множенню на h^k, яка є позитивно визначена симетрична матриця порядку $n \times n$, апроксимуюча зворотну матрицю Гессе. На кожному кроці матриця оновлюється, тобто приймає вигляд: $h^{k+1} = h^k + a^k + b^k$.

$$a^k = \frac{\Delta x^k \cdot (\Delta x^k)^T}{(\Delta^k)^T \cdot \Delta g^k}, \qquad \Delta x^k = x^k + x^{k+1}, \\ b^k = -\frac{h^k \cdot (\Delta^k)^T \cdot \Delta g^k \cdot h^k}{\Delta x^k \cdot h^k \cdot (\Delta x^k)^T}, \qquad \Delta g^k = \nabla f(x^{k+1}) - \nabla f(x^k). \quad (10)$$

Алгоритм Девідона-Флетчера-Пауелла іноді може привести до ситуації, коли матриця стає погано обумовлена або ж порушується умова позитивної визначеності. Причиною цього є невдалий вибір початкового наближення, а також наявність помилок округлення. Для подолання цих труднощів слід підвищити точність обчислень і періодично оновлювати ітераційний процес.

Метод Девідона–Флетчера–Пауелла широко використовується для вирішенні найрізноманітніших завдань і відрізняється високою.

Отже, хоча немає жодного універсального методу, який дозволяє успішно вирішувати всі завдання, деякі методи краще пристосовані для вирішення завдань певного типу. Ретельний вибір відповідного алгоритму часто дозволяє заощадити і машинний час, і зусилля, витрачені інженером на вирішення задачі. У розділі дані рекомендації які алгоритми віддати перевагу, для чого розглянуті методи прямого пошуку, градієнтні методи і методи другого порядку.

При створенні і побудові систем прийняття рішень виникає необхідність у вирішенні

найрізноманітніших завдань оптимізації. Розглянуті методи багатовимірної оптимізації як методи вирішення загального завдання пошуку локального екстремуму, та проведені порівняння їх ефективності не дали однозначної відповіді на питання коли і який саме має бути використаний метод. Результат аналізу застосовності розглянутих алгоритмів дослідження спонукав до більш глибокого числового експерименту направленого на виявлення найшвидшого алгоритму з максимальною правдоподібністю.

Числові експерименти (практична реалізація). Для практичного використання була розроблена програма на мові програмування Delphi (рис. 2).

Розроблений інтерфейс програми дозволяє оперативне введення цільової функції і здійснення пошуку рішення одним з 15 методів. Результат такого пошуку містить час затрачений на розрахунок та числові значення проміжних значень на кожному кроці (ітерації) розрахунку (табл. 1).

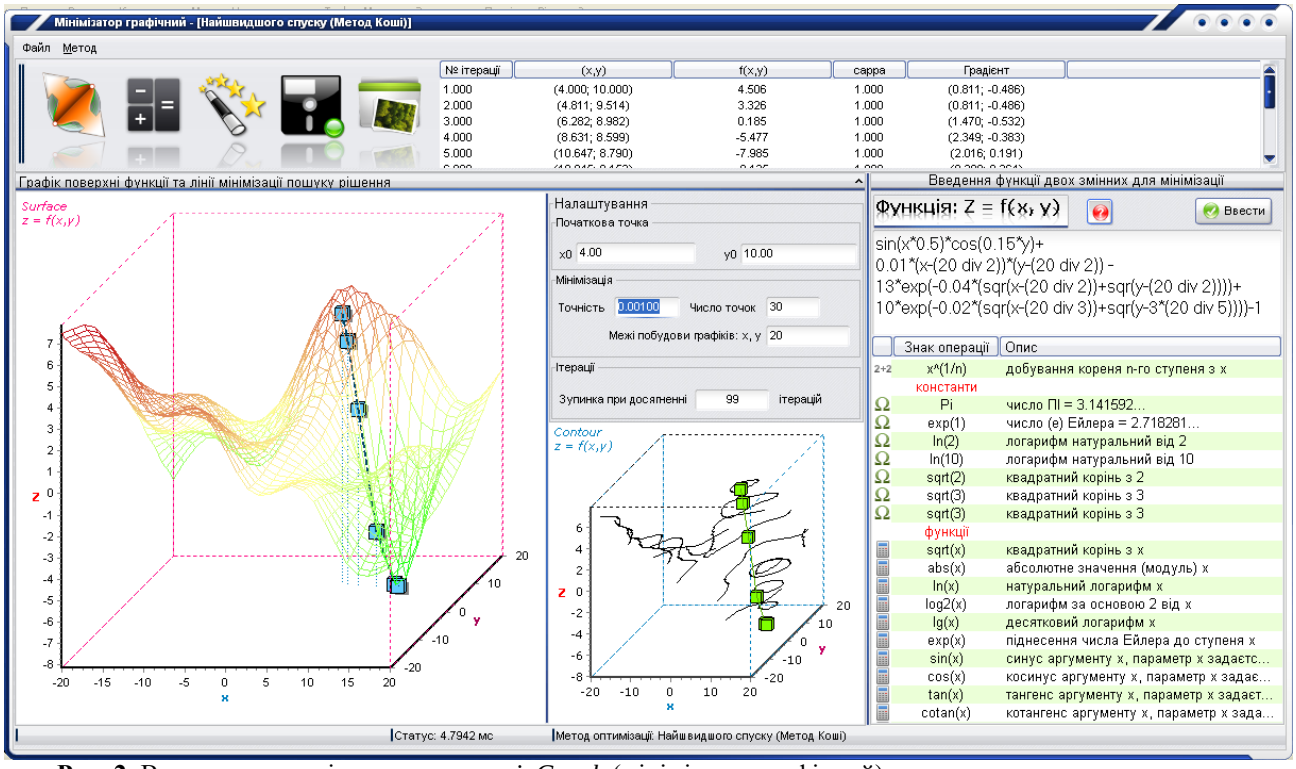


Рис. 2. Вид головного вікна програми *minGraph* (мінімізатор графічний).
Fig. 2. View of the main window of the *minGraph* program (graphic minimizer).

Таблиця 1. Результат генерації проміжних значень розрахунку програмою *minGraph*.
Table 1. The result of generating intermediate values of the calculation program *minGraph*.

$$Z(x) = 8 \cdot x^2 - 4 \cdot x \cdot y + 5 \cdot y^2 + 8 \cdot \sqrt{5} \cdot (x + 2 \cdot y) + 64$$

Метод: По координатний спуск (золотий перетин) - метод Гаусса - Зайделя
Час розрахунку: *12.2292 мс*

Ітер.	(x, y)	Z = *f(x, y)*	скалярний аргумент*	градієнт
1	(-15.000; 10.000)	3053.443	0.000	(-15.000; 10.000)
2	(-4.868; -5.525)	13.869	1.000	(-4.868; -5.525)
3	(-2.499; -4.578)	-35.501	1.000	(-2.499; -4.578)
4	(-2.263; -4.482)	-35.995	1.000	(-2.263; -4.482)
5	(-2.239; -4.473)	-36.000	1.000	(-2.239; -4.473)
6	(-2.237; -4.473)	-36.000	1.000	(-2.237; -4.473)
7	(-2.236; -4.472)	-36.000	1.000	(-2.236; -4.472)
8	(-2.236; -4.472)	-36.000	1.000	(-2.236; -4.472)

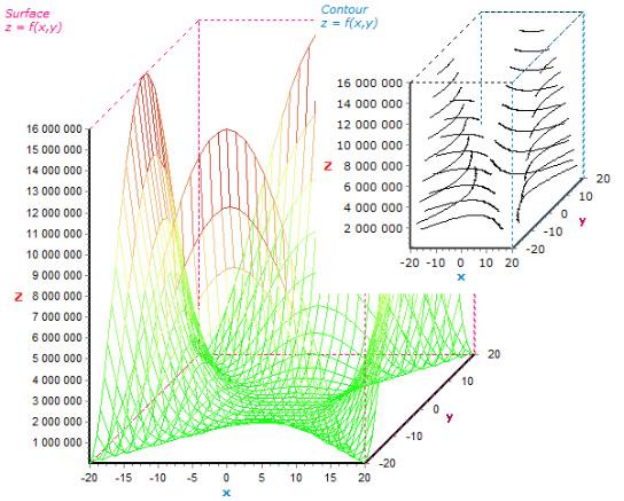
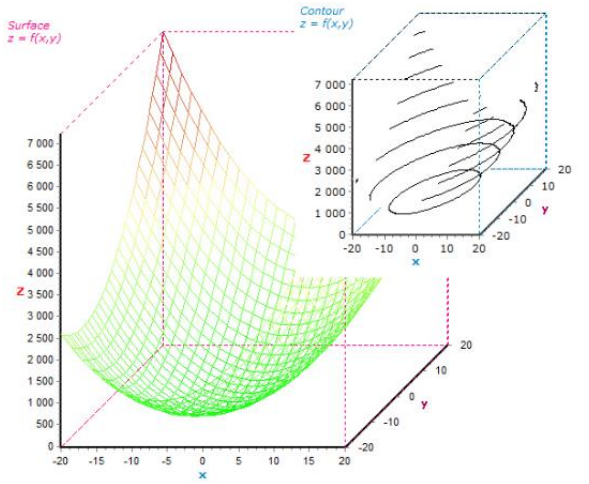
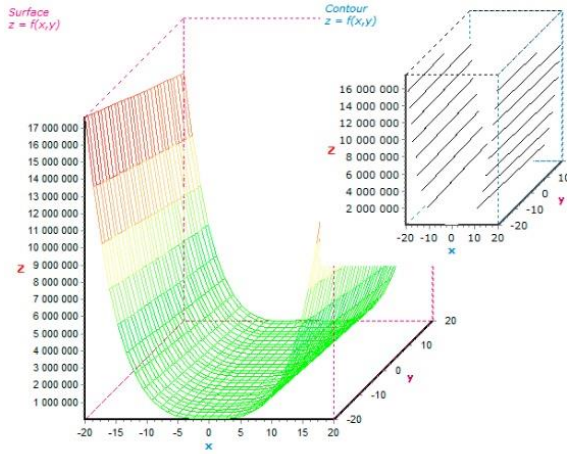
Примітка * – скалярний аргумент *k* скалярної функції виду: $\Psi(k) = f\left(\overrightarrow{x^k} + k \cdot \overrightarrow{y^k}\right)$.

Методика числового експерименту полягала в наступному: для трьох цільових функцій при певних початкових умовах (табл. 2) досліджувалися швидкість та точність сходження при використанні 15 методів пошуку мінімуму: *1* – по координатний спуск (золотий перетин) – метод Гаусса-Зайделя; *2* – випадкового пошуку; *3* – градієнтного спуску з дробленням кроку; *4* – найшвидшого спуску (метод Коші); *5* – пов'язаних напрямків; *6* – Флетчера-Рівса; *7* – ДФП (Давідона-Флетчера-Пауелла); *8* – циклічного по координатного спуску; *9* – нульового порядку – Хука-Дживса; *10* – нульового порядку –Розенброка; *11* – нульового порядку –Пауелла; *12* – нерегулярний симплекс – Нелдера-Міда; *13* – метод Ньютона; *14* – умовного

градієнта; *15* – проекції антиградієнта. При цьому фіксувалися наступні параметри: кількість ітерацій, час затрачений на пошук рішення, знайдене значення функції і наступне порівняння з точним значенням шляхом пошуку абсолютної похибки (табл. 3).

При пошуку мінімуму функцій ітераційним методами одним з вирішальних чинників слід вважати стабільність результатів. Це означає, що невеликі відхилення від початкових значень шуканих функцій не повинні призводити до істотної зміни кінцевого результату. Для наочності вид поверхонь і обчислювальний процес мінімізації для двох змінних показаний на рис. 3.

Таблиця 2. Вихідні дані моделювання пошуку кращого алгоритму для трьох функцій.
Table 2. The output of the simulation search for the best algorithm for the three functions.

	*Функція 1 – f_1:* $f(x,y) = 100 \cdot (y^2 - x^2)^2 + (1 - x)^2$
	Початкові дані: Початкова точка – $(x0 = -19.00; y0 = 0.00)$ Точність – $e = 0.001$ Число точок – 30 Межі побудови графіка – x, y = 20
	***Функція 2 – f_2:*** $f(x,y) = 8x^2 - 4xy + 5y^2 + 8\sqrt{5} \cdot (x + 2y) + 64$
	Початкові дані: Початкова точка – $(x0 = -15.00; y0 = 10.00)$ Точність – $e = 0.001$ Число точок – 30 Межі побудови графіка – x, y = 20
	*Функція 2 –f_3:* $f(x,y) = 100 \cdot (y - x^2)^2 + (y - 1)^2$
	Початкові дані: Початкова точка – $(x0 = -19.00; y0 = 0.00)$ Точність – $e = 0.001$ Число точок – 30 Межі побудови графіка – x, y = 20

У цьому обчислювальному експерименті було використано навмисне складні умови роботи. Для цього початкові точки були вибрані якнайдалі від ймовірного мінімуму.

Після знаходження оптимуму функцій (рис. 3) було проведено порівняння результатів обрахунку за методами для трьох функцій на швидкість сходження, точність сходження.

Таблиця 3. Розрахункові мінімальні значення функцій для різних алгоритмів.

Table 3. Estimated minimum values of functions for different algorithms.

Ме-тод	Знайдене значення	Ітера-ції	Час, мс	Δ
$\boldsymbol{функція - f_1\ (min\ =\ 0)}$				
1.	323,997	3	6,4131	-323,997
2.	0,059	74	116,97	-0,059
3.	3592310	2	4,2522	-3592310
4.	0,771	5	6,4882	-0,771
5.	0,771	4	5,5555	-0,771
6.	0,771	8	21,8611	-0,771
7.	0,557	98	159,5394	-0,557
8.	99,994	3	202,1436	-99,994
9.	110,249	4	7,2663	-110,249
10.	0	5	6,6056	0
11.	0	5	6,6534	0
12.	0,001	2	4,4209	-0,001
13.	0,771	16	29,3393	-0,771
14.	0,049	143	223,6698	-0,049
15.	4	308	480,2124	-4
$\boldsymbol{функція - f_2\ (min\ =\ -36)}$				
1.	-36	8	15,32	0
2.	-35,745	79	142,667	-0,255
3.	-36	29	58,9249	0
4.	-36	9	22,9714	0
5.	-36	3	8,9138	0
6.	-36	3	5,9635	0
7.	-36	3	4,7089	0
8.	-36	8	214,511	0
9.	-36	12	20,3669	0
10.	-36	7	24,5134	0
11.	-36	3	11,0239	0
12.	-36	2	4,3565	0
13.	-36	3	4,8218	0
14.	-20,22	3	4,9149	-15,78
15.	-34,997	7	25,2023	-1,003
$\boldsymbol{функція - f_3\ (min\ =\ 0)}$				
1.	0,15	99	180,254	-0,15
2.	4,977	74	134,857	-4,977
3.	3586954	2	14,4432	-3586954
4.	0,361	98	160,222	-0,361
5.	0,478	49	104,287	-0,478
6.	43,643	99	162,982	-43,643
7.	0	92	156,589	0
8.	0	987	7604	0
9.	0,044	480	1867,2	-0,044
10.	0	135	248,126	0
11.	0	6	30,9801	0
12.	0	2	8,6935	0
13.	0	22	43,0731	0
14.	0	17	39,3458	0
15.	0	371	570,204	0

Обчислювальний експеримент показав, що перша функція виявилась досить складною майже для всіх алгоритмів (*мінімум при:* $x = 1; y = 1$). Найгіршу швидкість сходження показали методи 7, 8 та 14 і 15. Ці методи мають до того ж незадовільну точність. Оптимальне рішення не було знайдено, а значення наближення до мінімуму склали від 0,049…99,994. Найбільш ефективно себе показали методи 10, 11 та 11. Точність даних методів склала 100%. Недопустимо використовувати для пошуку мінімуму цієї функції 3 методом (градієнтного спуску з дробленням кроку). Звісно якщо врахувати значення функції у початковій точці (13032500), то знайдене значення 0,049..0,771 (це майже мінімум, з деяким припущенням можна вважати достатнім рішенням методами 2,4,5,6,7, 13 та 14.

Друга функція (*мінімум = -36*) виявилася простішою для пошуку рішення. Це пояснюється більш вираженим піком мінімуму (табл.2). Із задачею не справилися лише три методи 2, 14 і 15. Всі інші показали високу точність. Щодо швидкості сходження то 2 і 3 методи виконали роботу за 79 і 29 ітерацій відповідно. Час на роботу алгоритму при цьому залежав не так від кількості ітерацій, як від складності алгоритму.

Залежності швидкості сходження трьох функцій для 15 алгоритмів (рис. 4) показав найкращі показники для 11, 12 та частково 3 методів. Однак, враховуючи сучасну швидкість роботи комп'ютерів кількість ітерацій наближену до 100 не є такою вже важливою, хоча різниця в часі (рис. 5) є більш разючою (8 і 9 методи – 2000, 8000 мс). Більш визначальною для методів є точність пошуку мінімуму.

Третя експериментальна функції при пошуку мінімуму показала відмінну роботу методів 7–15. І це зважаючи на складність функції Розенброка. Ця функція – класична проблема оптимізації, також відома як бананова функція. Вона має велике повільно спадаюче плато. Знайти плато – тривіальна задача, однак конвергенція до глобального оптимуму важка. Ця функція використовується для оцінки роботи алгоритмів оптимізації, а тому і була нами обрана для дослідження. Всі інші методи спіткнулися на цій функції і показали сумнівні результати, особливо 3 метод. Особливу зацікавленість виявили в роботі 8 і 9 методи (циклічного по координатного спуску і нульового порядку – Хука-Дживса). Відмічається досить погане сходження та великий час на роботу алгоритму. Проте результат виявився відмінним для 8 та достатнім для 9 методу.

Проведені обчислювальні експерименти по мінімізації трьох функції (табл. 2) для 15 методів оптимізації показали, що для всіх досліджуваних функцій можна рекомендувати два методи: *10* – нульового порядку –Розенброка та *11* – нульового порядку –Пауелла (рис. 3).

Це пояснюється тим, що дані алгоритми відразу рухаються на локальний мінімум і мають найбільш широку область тяжіння, в силу чого з більшою ймовірністю хоча б одна з ітерацій потрапляє в зазначену область і чинить максимальний вплив на подальшу поведінку алгоритму пошуку.

Лише ці методи показали для всіх функцій найкращу сходження та точність обчислення. Інші методи частково показали відмінні результати для певних функцій, тоді як для інших функцій не зовсім.

Слід зазначити, що доцільно використання цих методів на більш складних функціях, які окрім глобального мінімуму містять декілька локальних мінімумів, що може становити інтерес при вирішенні

прикладних інженерних задач. Використання ж розробленого інструменту пришвидшить аналіз інженерних функцій, і загалом підвищить стабільність отриманих результатів.

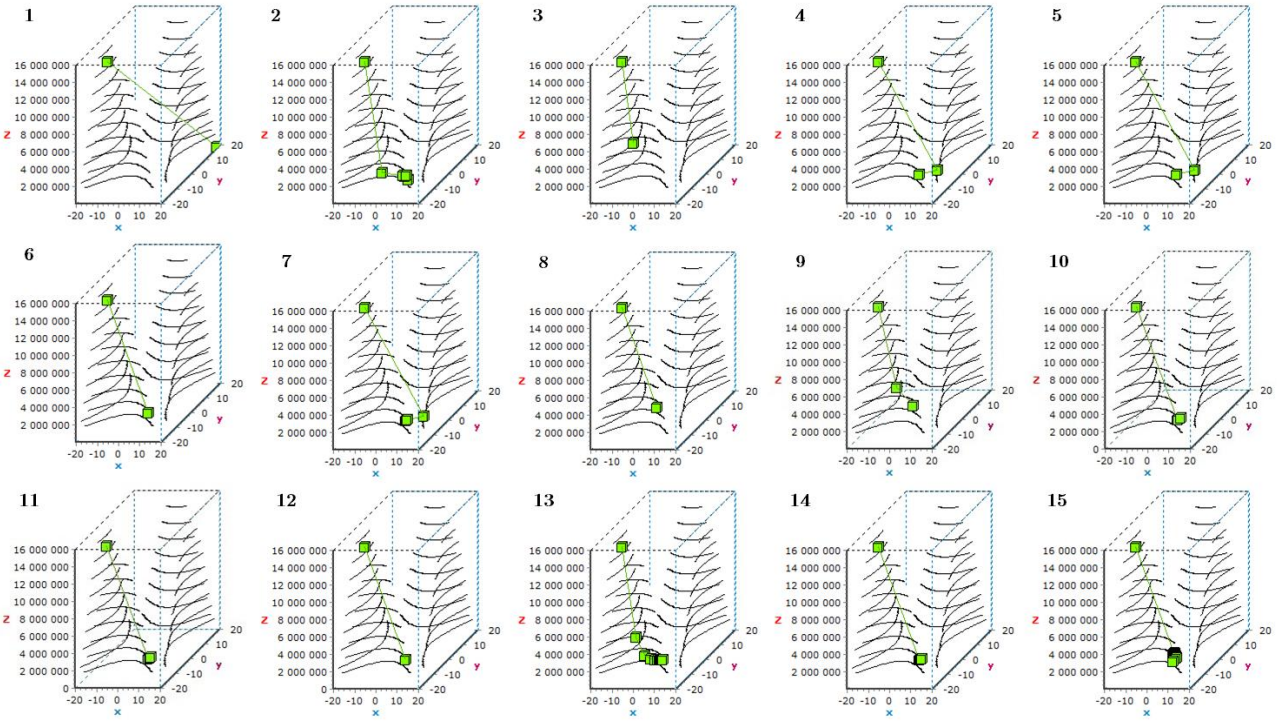


Рис. 3. Множина зображень пошуку мінімуму для функції f_1 різними методами.
Fig. 3. A set of minimum search images for a function f_1 by different methods.

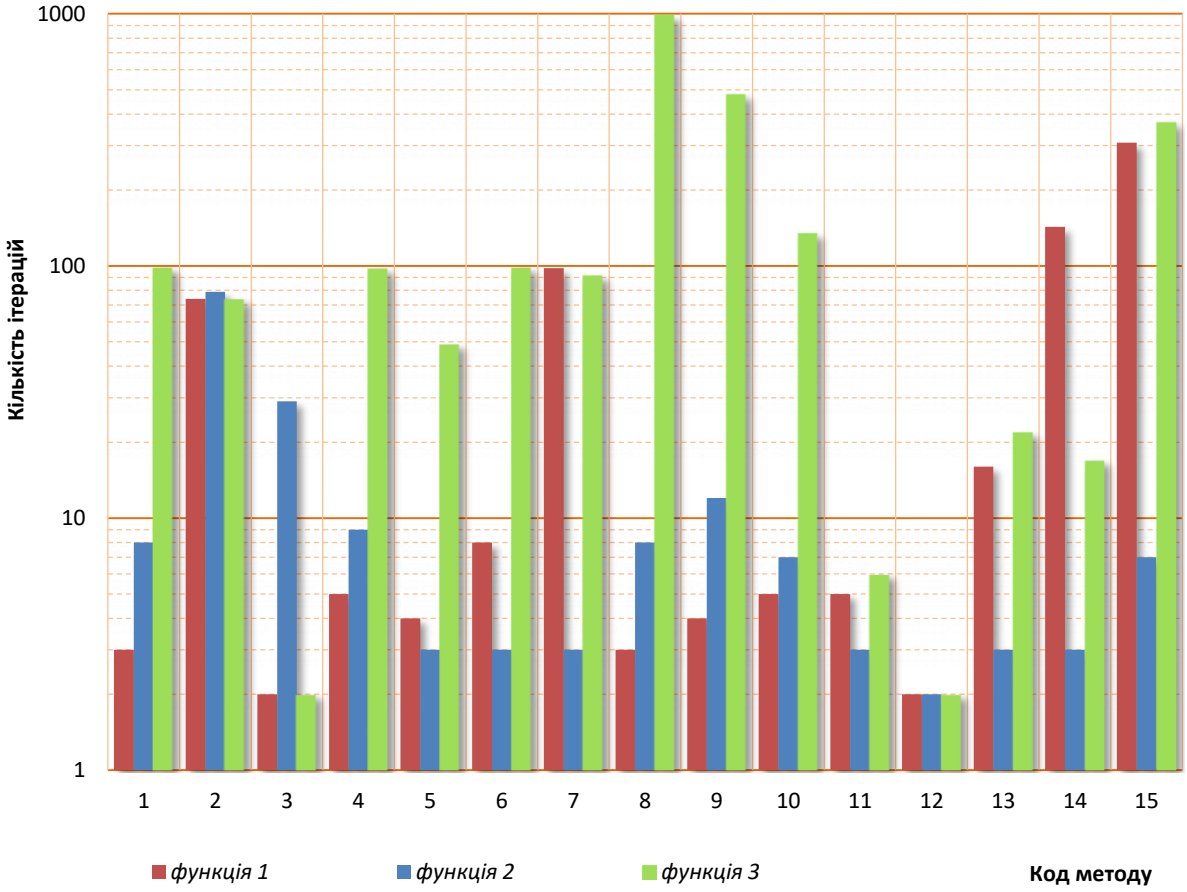


Рис. 4. Гістограми сходження експериментальних функцій для 15 методів.
Fig. 4. Histograms of convergence of experimental functions for 15 methods.

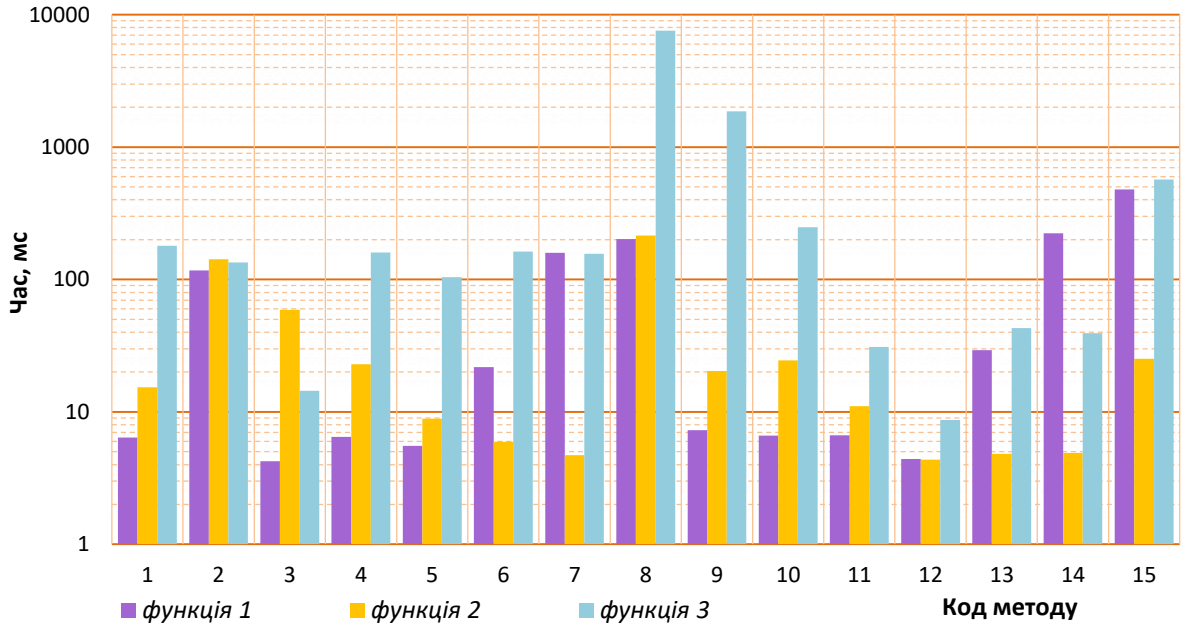


Рис. 5. Гістограми швидкості роботи 15 методів пошуку для трьох експериментальних функцій.
Fig. 5. Speed histograms of 15 search methods for three experimental functions.

Висновки

1. В ході даної роботи були розглянуті основні методи оптимізації, що застосовуються в сучасних інженерних задачах та методах навчання нейронних мереж. У процесі дослідження було проведено аналіз властивостей і особливостей розглянутих методів, а також сформульовані умови та обґрунтування їх найбільш оптимального застосування в різних практичних завданнях з точки зору їх сходження і точності.
2. Дані аналізу супроводжуються практичними результатами, які підтверджують сформульовані рекомендації по використанню розглянутих методів класичного і стохастичного градієнтного спуску, імпульсних, адаптивних, а також квазіньютонівських алгоритмів оптимізації в інженерних задачах.
3. Напрямки подальших досліджень. Розглянуті багатомірні методи оптимізації можуть бути в майбутньому використані для навчання нейронних мереж..

Список літератури

1. *Антонов А. И.* Переходный процесс в двумерной экстремальной системе при наличии запрещенных областей и случайном методе поиска. Задачи статистической оптимизации. Рига: Зинтане, 2011. С. 69-81.
2. *Медведев Г. А., Рыжаков А. П.* О применении алгоритмов случайного поиска в системах автоматической оптимизации. Задачи статистической оптимизации. Рига: Зинтане, 2011. С. 81-92.
3. *Растригин Л. А., Рипа К. К.* Автоматная теория случайного поиска. Рига: Зинтане, 2013. 342 с.
4. *Таев И. С., Егоров Е. Г., Горшков Ю. Е., Попова Е. П.* Оптимизация параметров дугогасительной камеры электрического аппарата. Москва. Аппараты низкого напряжения. 2011. Вып. 1(92). С. 24-32.
5. *Петров А. А.* Численные методы условной оптимизации. Москва. Мир. 2017. 290 с.
6. *Box M. J.* A new method of constrained optimization and a compare son with other methods. Computer. 2015. №8. P. 42-52.
7. *Burkhard G.* Uber das Lichtbogenverha1ten in Loschkammern und derem Bemessung. E1ektrie. 2014. Vol. E05. P. 96-105.
8. *Hooke R., Jeeves T.* Direct seach solution of numerical and ststist1cal problems. JACM. 2011. Vol. 8. P. 212-229.
9. *Klingman W. R., Himmelblau D. M.* Nonlinear programming with the aid of a multiple gradient summation technique. JACM. 2014. Vol. 11. P. 400-415.
10. *Luws R., Jaakola T.* Optimization by direct search and systematic reduction of the size of seach region. Ch.E. Journal. 2013. Vol. 19. P. 760-766.
11. *Rosenbrock H. H.* An automatic method for finding the greatest or least value of a function. Computer. 2020. Vol. 3. P. 175-184.
12. *Rumelhart D. E.* Learning representations of back-propagation errors. Nature. 2016. Vol. 323. P. 533-536.
13. *Реклейтис Г.* Оптимизация в технике. Москва. Мир. 2016. 220 с.
14. *Goldberg D. E.* Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, 2019. 308 p.
15. *Колков Д. А.* Анализ интервальных методов поиска глобального экстремума. Фундаментальные исследования. 2006. Вып. 2. С. 22-23.
16. *Rogovskii I. L., Titova L. L., Trokhaniak V. I., Rosamaha Yu. O., Blesnyuk O. V., Ohiienko M. M. Ohiienko A. V.* Engineering management of two-phase coulter systems of seeding machines for implementing

precision farming technologies. INMATEH. Agricultural Engineering. 2019. Bucharest. Vol. 58. No 2. P. 137-146. doi: 10.35633/INMATEH-58-15.

17. *Romaniuk W., Polishchuk V., Marczuk A., Titova L., Rogovskii I., Borek K.* Impact of sediment formed in biogas production on productivity of crops and ecologic character of production of onion for chives**.** Agricultural Engineering (wir.ptir.org). Krakow. Poland. 2018. Vol. 22. №1. P. 105-125. https://doi.org/10.1515/agriceng-2018-0010**.**

18. *Rogovskii I., Titova L., Trokhaniak V., Trokhaniak O., Stepanenko S.* Experimental study in a pneumatic microbiocature separator with apparatus camera. Bulletin of the Transilvania University of Brasov, Series II: Forestry, Wood Industry, Agricultural Food Engineering, 2019. Vol. 12 (1). P. 117-128.

19. *Тітова Л. Л., Роговський І. Л.* Методичні положення потреби в мобільних засобах техобслуговування лісових МЕЗ. Науковий вісник Національного університету біоресурсів і природокористування України. Серія: техніка та енергетика АПК. Київ. 2014. Вип. 196, ч. 3. С. 146-152.

20. *Тітова Л. Л., Роговський І. Л.* Методи технічного обслуговування лісових машин. Вісник Харківського національного технічного університету сільського господарства імені Петра Василенка. Харків. 2014. Вип. 155. С. 132-137.

21. *Тітова Л. Л., Роговський І. Л.* Обґрунтування технічних заходів підтримання працездатності машин для лісотехнічних робіт. Вісник Харківського національного технічного університету сільського господарства імені Петра Василенка. Харків. 2015. Вип. 160. С. 189-195.

22. *Тітова Л. Л., Роговський І. Л.* Аналіз періодичності техобслуговування машин для лісотехнічних робіт. Науковий вісник Національного університету біоресурсів і природокористування України. Серія: техніка та енергетика АПК. Київ. 2015. Вип. 212, ч. 1. С. 322-328.

References

1. *Antonov A. I.* (2011). Transient process in a two-dimensional extremal system in the presence of forbidden regions and a random search method. Statistical optimization problems. Riga: Zintane. 69-81.

2. *Medvedev G. A., Ryzhakov A. P.* (2011). On the application of random search algorithms in automatic optimization systems. Statistical optimization problems. Riga. Zintane. 81-92.

3. *Rastrigin L. A., Ripa K. K.* (2013). Automatic theory of random search. Riga. Zintane. 342.

4. *Taev I. S., Egorov E. G., Gorshkov Yu. E., Popova E. P.* (2011). Optimization of parameters of the electric apparatus arc-extinguishing chamber. Moscow. Low voltage devices. 1(92). 24-32.

5. *Petrov A. A.* (2017). Numerical methods of conditional optimization. Moscow. Myr. 290.

6. *Box M. J.* (2015). A new method of constrained optimization and a compare son with other methods. Computer. 8. 42-52.

7. *Burkhard G.* (2014). Uber das Lichtbogenverha1ten in Loschkammern und derem Bemessung. E1ektrie. E05. 96-105.

8. *Hooke R., Jeeves T.* (2011). Direct seach solution of numerical and ststist1cal problems. JACM. 8. 212-229.

9. *Klingman W. R., Himmelblau D. M.* (2014). Nonlinear programming with the aid of a multiple gradient summation technique. JACM. 11. 400-415.

10. *Luws R., Jaakola T.* (2013). Optimization by direct search and systematic reduction of the size of seach region. Ch.E. Journal. 19. 760-766.

11. *Rosenbrock H. H.* (2020). An automatic method for finding the greatest or least value of a function. Computer. 3. 175-184.

12. *Rumelhart D. E.* (2016). Learning representations of back-propagation errors. Nature. 323. 533-536.

13. *Rekleytis G.* (2016). Optimization in technology. Moscow. Myr. 220.

14. *Goldberg D. E.* (2019). Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, 308.

15. *Kolkov D. A.* (2006). Analysis of interval methods of searching for a global extremum. Fundamental research. 2. 22-23.

16. *Rogovskii I. L., Titova L. L., Trokhaniak V. I., Rosamaha Yu. O., Blesnyuk O. V., Ohiienko M. M. Ohiienko A. V.* (2019). Engineering management of two-phase coulter systems of seeding machines for implementing precision farming technologies. INMATEH. Agricultural Engineering. 2019. 58(2). 137-146. doi: 10.35633/INMATEH-58-15.

17. *Romaniuk W., Polishchuk V., Marczuk A., Titova L., Rogovskii I., Borek K.* (2018). Impact of sediment formed in biogas production on productivity of crops and ecologic character of production of onion for chives. Agricultural Engineering (wir.ptir.org). Krakow. Poland. 22(1). 105-125. https://doi.org/10.1515/agriceng-2018-0010**.**

18. *Rogovskii I., Titova L., Trokhaniak V., Trokhaniak O., Stepanenko S.* (2019). Experimental study in a pneumatic microbiocature separator with apparatus camera. Bulletin of the Transilvania University of Brasov, Series II: Forestry, Wood Industry, Agricultural Food Engineering. 12(1). 117-128.

19. *Titova L. L., Rogovsky I. L.* (2014). Methodical provisions of the need for mobile means of maintenance of forest MEZ. Scientific Herald of National University of Life and Environmental Science of Ukraine. Series: Technique and energy of APK. Kyiv. 196(3). 146-152.

20. *Titova L. L., Rogovsky I. L.* (2014). Methods of maintenance of forest machines. Bulletin of the Petro Vasylenko Kharkiv National Technical University of Agriculture. Kharkiv. 155. 132-137.

21. *Titova L. L., Rogovskii I. L.* (2015). Substantiation of technical measures to maintain the efficiency of machines for forestry work. Bulletin of the Petro Vasylenko Kharkiv National Technical University of Agriculture. Kharkiv. 160. 189-195.

22. *Titova L. L., Rogovsky I. L.* (2015). Analysis of periodicity of maintenance of machines for forestry works. Scientific Herald of National University of Life and Environmental Science of Ukraine. Series: Technique and energy of APK. Kyiv. 2015. 212(1). 322-328.

МЕТОД ПОИСКА ЭКСТРЕМУМА МНОГОМЕРНЫХ ФУНКЦИЙ ПРИ РЕШЕНИИ ИНЖЕНЕРНЫХ ЗАДАЧ МАШИН ДЛЯ ЛЕСОТЕХНИЧЕСКИХ РАБОТ

Л. Л. Титова, А. В. Надточий

Аннотация. Данная статья посвящена анализу наиболее распространенных методов оптимизации, используемых в практических инженерных задачах поиска экстремума многомерных функций и формирование на основе выявленных свойств рекомендаций по выбору лучшего на различных наборах данных. В процессе анализа были рассмотрены различные реализации методов градиентного спуска, импульсные методы, адаптивные методы и квазиньютоновские методы, обобщенны преимущества и проблемы каждого из методов при их использовании. Разработанная компьютерная программа, реализующая использование всех рассмотренных методов. Проведенный вычислительный эксперимент для трех функций показал, что наиболее эффективными оказались методы нулевого порядка -Розенброка и нулевого порядка -Пауэлла.

Ключевые слова: методы оптимизации, метод градиентного спуска, стохастический градиент, квазиньютоновские методы, целевая функция.

METHOD FOR SEARCHING THE EXTREMUM OF MULTIDIMENSIONAL FUNCTIONS IN SOLVING ENGINEERING PROBLEMS OF MACHINES FOR FORESTRY WORKS

L. L. Titova, O. V. Nadtochiy

Abstract. This article is devoted to the analysis of the most common optimization methods used in practical engineering problems of finding the extremum of multidimensional functions and the formation on the basis of the identified properties of recommendations for choosing the best on different data sets. In the process of analysis, various implementations of gradient descent methods, pulse methods, adaptive methods and quasi-Newtonian methods were considered, and the advantages and problems of each of the methods in their use were summarized. Developed computer program that implements the use of all considered methods. The computational experiment performed for the three functions showed that the zero -Rosenbrock and zero -Powell methods proved to be the most effective.

Keywords: optimization methods, gradient descent method, stochastic gradient, quasi-Newtonian methods, objective function.

Л. Л. Тітова ORCID 0000-0001-7313-1253.
О. В. Надточій ORCID 0000-0002-2876-3021.