

УДК 009.4

ЗАСТОСУВАННЯ МЕТОДІВ ОПТИМІЗАЦІЇ В ЗАДАЧІ РОЗМІЩЕННЯ ВЕКТОРНИХ ГРАФІЧНИХ ОБ'ЄКТІВ НА ПЛОЩИНІ

Д. Ю. Паламарчук, О. В. Тимченко, В. О. Демченко

*Українська академія друкарства,
вул. Під Голоском, 19, Львів, 79020, Україна*

Здійснено порівняння результатів застосування алгоритмів оптимізації для розв'язання задачі розміщення векторних графічних об'єктів на площині. Описано алгоритми у вигляді блок-схем, що дає змогу чітко зрозуміти їхню структуру і принципи роботи. Для порівняння ефективності методів було створено програмний застосунок, який працює зі спрощеними вхідними даними у вигляді прямокутних об'єктів, що розміщуються на площинах різного розміру. Результати дослідження наведені у вигляді графіків і таблиць, що ілюструють ефективність кожного з методів. Наведено детальний опис отриманих результатів та зроблено висновки, які свідчать про високий потенціал застосування методів пошуку оптимальних рішень у задачах оптимізації розміщення графічних об'єктів.

Ключові слова: *оптимізація, порівняльний аналіз, векторні графічні об'єкти, генетичний алгоритм, імітаційне моделювання, метод імітації відпалювання.*

Постановка проблеми. Задача розміщення векторних об'єктів на площині є поширеною в багатьох видах діяльності та виробництв — вирізання об'єктів на папері, металі, дереві та інших типах сировини [1]. Якщо є велика кількість деталей, які необхідно вирізати, постає задача їх оптимального розміщення для економії матеріалів або роботи з нестандартними формами площини. Оптимізація розміщення векторних об'єктів на площині є складною задачею через великий простір можливих розташувань і необхідність врахування різних параметрів кожного об'єкта. Наприклад, врахування обертання об'єктів може значно вплинути на загальну ефективність розміщення. Крім того, важливою є допустима відстань між об'єктами для запобігання пошкодженню матеріалу або іншим виробничим дефектам. Зважаючи на це, завдання оптимального розміщення векторних об'єктів належить до класу NP-складних задач, що робить її розв'язання за допомогою точних методів непрактичним для великих обсягів даних. Отже, для розв'язання цієї задачі доцільно використовувати наближені методи оптимізації, які можуть знайти прийнятні рішення за прийнятний час, тому в цій статті запропоновано використання трьох методів оптимального пошуку: генетичний алгоритм, метод імітації відпалювання та імітаційне моделювання. Для порівняння ефективності цих алгоритмів було створено програмний застосунок, який використовує спрощені вхідні дані — прямокутні об'єкти, які необхідно розмістити на заданих площинах різного розміру. Результати цього дослідження можуть сприяти розробці

більш ефективних алгоритмів для задач оптимізації в різних виробничих та інших прикладних галузях, забезпечуючи економію ресурсів та підвищення продуктивності.

Аналіз останніх досліджень та публікацій. Задача пошуку оптимальних рішень активно досліджується в різних наукових та прикладних галузях через її важливість і складність. Для її вирішення було розроблено безліч методів, серед яких найпопулярнішими є генетичні алгоритми, метод імітації відпалу та імітаційне моделювання. Ці методи мають свої унікальні підходи до вирішення задачі пошуку оптимальних рішень.

Метод імітації відпалу — це алгоритм оптимізації, що моделює фізичний процес відпалу металів [2]. У цьому методі система «нагрівається», щоб дослідити різні варіанти, а потім «оохолоджується», дозволяючи уникнути застрягання в локальних мінімумах. Цей алгоритм ефективний в оптимізації складних просторів, де є багато локальних оптимумів. Метод має кілька ключових переваг, зокрема простоту реалізації та здатність знаходити оптимальні рішення для різноманітних задач. Однак він може потребувати значної кількості часу та налаштування параметрів. Завдяки своїй гнучкості та здатності знаходити оптимальні рішення в складних умовах метод імітації відпалу широко застосовується в задачах оптимізації, таких як задачі комівояжера, розкладу, розподілу завдань та інших.

Імітаційне моделювання становить потужний інструмент у прикладному системному аналізі, що дає можливість досліджувати складні системи та процеси, особливо в умовах, де управління пов'язане з ухваленням рішень в умовах невизначеності [3]. Порівняно з іншими методами, воно дає змогу враховувати велику кількість альтернатив, покращувати якість управлінських рішень та прогнозувати їх наслідки. Однак використання імітаційного моделювання в практичному управлінні ще не дуже поширене через складність математичного апарату та обробки значних масивів даних. Метод базується на відтворенні процесу функціонування системи із взаємодією з зовнішнім середовищем, а його модель може бути представлена у вигляді програмно або апаратно реалізованих функціональних блоків. Імітаційне моделювання дає змогу враховувати широкий спектр чинників під час ухвалення управлінських рішень і є найпотужнішим інструментом аналізу складних систем і процесів, особливо в умовах суб'єктивності та невизначеності, які потребують надійних аналітичних засобів.

Генетичний алгоритм відрізняється від традиційних методів оптимізації і має істотні переваги [4]. Він працює із закодованою формою значень, здійснюючи пошук у межах популяції, що дає змогу ефективно досліджувати простір рішень. Також цей метод використовує цільову функцію без потреби в її похідних, застосовує ймовірність правила відбору, що допомагає уникати локальних оптимумів, і має внутрішній паралелізм для дослідження різних напрямів пошуку одночасно. Він ефективний в складних областях пошуку, може одночасно оперувати багатьма параметрами, не потребує додаткової інформації про задачу і має високу швидкість знаходження рішень. Основним недоліком є неможливість створити універсальний код для опису функції та критеріїв оптимізації через різні початкові умови. Генетичні алгоритми застосовуються для вирішення різних наукових і технічних

задач, таких як проєктування нейронних мереж, управління виробництвом, економіка, медицина, математика та ін.

Мета статті — аналіз та порівняння ефективності методів імітації відпалювання, імітаційного моделювання та генетичного алгоритму для ефективного розміщення векторних графічних об'єктів на площині.

Виклад основного матеріалу дослідження. Для відображення роботи алгоритмів і порівняння їх ефективності був створений програмний застосунок, який імітує оптимальне розміщення векторних об'єктів на заданій площині. Застосунок реалізований з використанням бібліотеки Java Swing [5] на мові Kotlin. Вхідними даними застосунку є розмір дошки і кількість об'єктів, які ми хочемо розмістити на ній. Для спрощення задачі було використано лише прямокутні фігури розміром від 1 до 40 клітин відносно розміру сітки площини по висоті та ширині, які задані генератором випадкових чисел. Мета роботи застосунку — розмістити фігури на площині, зайнявши якнайменше доступного простору. Фігури розміщуються від лівого верхнього кута площини порядково. Додаток вимірює якість отриманого результату за кількістю вільних клітин, які залишилися від правого нижнього кута.

У програмному застосунку (рис. 1) відображено одразу 3 площини з однаковими вхідними даними, але з використанням різних алгоритмів: імітації відпалювання, імітаційне моделювання та генетичний алгоритм. Також над кожною площиною вказано якість отриманого результату. Очевидно, що вище значення якості, то кращий результат.

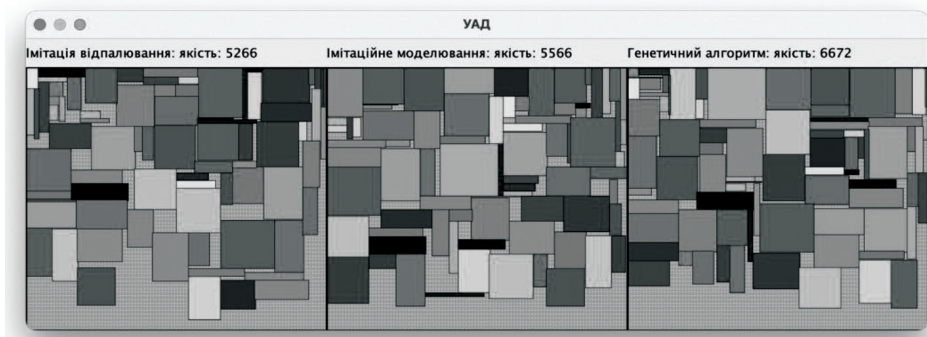


Рис. 1. Програмний застосунок

Метод імітації відпалювання. Метод імітаційного відпалювання має декілька модифікацій, які відрізняються різними законами зміни температур [6]. В кожного є свої плюси і мінуси, такі як швидкість, гарантія знаходження глобального мінімуму та складність реалізації. Для цієї задачі вибрана модифікація алгоритму під назвою надшвидке відпалювання («Very Fast Annealing»). Для кращого розуміння роботи алгоритму в задачі оптимального розміщення фігур на площині його можна представити послідовністю таких кроків:

1. Задаються дві величини: початкова T_0 та кінцева T_{end} температури.
2. Генеруємо початковий коректний розв'язок x_0 , оцінюємо його якість і отримуємо оцінку зберігаємо як r_0 .

3. Присвоюємо поточній температурі початкове значення $T = T_0$.

4. Створюємо нову ітерацію. В кожній ітерації буде відбуватися обмін двома фігурами, визначеними генератором випадкових чисел. Якщо визначені фігури неможливо розмістити на нових позиціях, розрахунок номерів фігур відбувається заново. Після визначення нових номерів робимо обмін і отримуємо новий розв'язок x .

5. Оцінюємо новий отриманий розв'язок за допомогою функції $r=f(x)$. Якщо $r < r_0$, то розв'язок покращився, зберігаємо нове рішення як поточне $x_0=x$, $r_0=r$. Інакше, якщо $r > r_0$, нове рішення зберігається як поточне лише з ймовірністю:

$$h(\Delta E, T) = \frac{1}{1 + \exp(\Delta E / T)},$$

де $\Delta E = r_0 - r$. Отже, ймовірність прийняття гіршого рішення як поточне менша, що менша температура і що більша різниця між поточною енергією r і оптимальною r_0 .

6. Обчислюємо нову температуру так:

$$T_i(k) = T_{(i,0)} \exp(c_i k^{1/D}), c_i > 0,$$

де D — загальна кількість занять; k — поточний номер ітерації; c — змінна, яка використовується в загальній задачі пошуку шляху в системі координат для визначення конкретної температури в кожній точці, тому для задачі пошуку оптимального рішення спростимо і збережемо її як константу $c=1$. Температура буде зменшуватися за експоненціальним законом.

7. Якщо $T > T_{end}$, переходимо на крок 4 і продовжуємо пошук, інакше закінчуємо роботу алгоритму.

Блок-схема алгоритму подана на рис. 2.

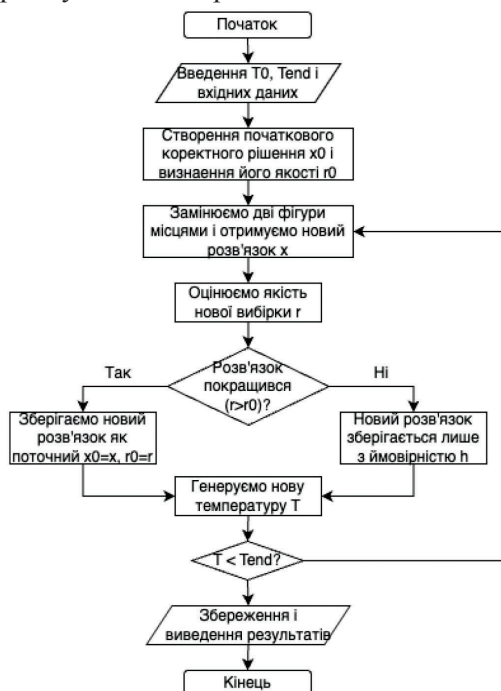


Рис. 2. Блок-схема роботи методу імітації відпалювання

Імітаційне моделювання — це підхід до дослідження, за якого реальна система замінюється моделлю, що достатньо точно описує її функціонування [7]. Більш детально для задачі знаходження оптимального розміщення фігур на площині цей алгоритм можна описати так:

1. Вибираємо ще не розміщену фігуру.
2. Для обраної фігури знаходимо всі можливі варіанти розміщення на площині, які задовольняють його жорсткі обмеження, і заносимо їх в список *list*.
3. Оцінюємо якість кожної з позицій, в результаті фігура розміщується в найбільш вигідному з варіантів.

Принци роботи алгоритму наведено на блок-схемі (рис. 3).

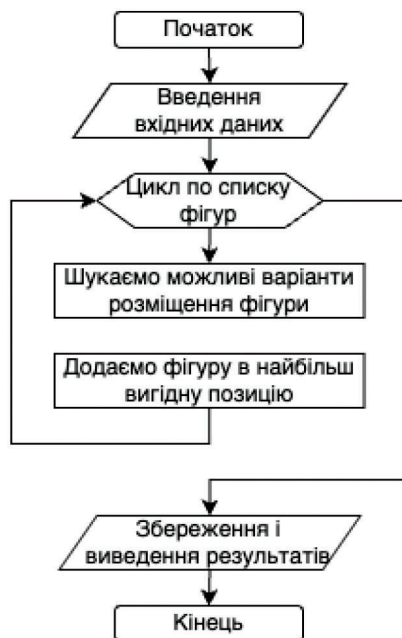


Рис. 3. Блок-схема роботи методу імітаційного моделювання

Генетичний алгоритм — алгоритм, в основі якого лежить природний відбір, тобто коли більш пристосовані види мають більше шансів на виживання [8]. Використання генетичного алгоритму в знаходженні оптимального розміщення фігур на площині можна навести так:

1. Вводимо необхідну кількість ітерацій N , визначену попередньо експериментальним шляхом. Як лічильник використаємо змінну $count=0$.
2. Отримуємо список ще не розміщених фігур. На його основі генеруємо початковий коректний розв'язок x_0 . Оцінюємо розв'язок за допомогою методу $f(x)$ і зберігаємо оцінку як r_0 .
3. Випадковим чином змінюємо порядок фігур в списку і на його основі генеруємо новий коректний результат x , після чого оцінюємо його $r=f(x)$.
4. Якщо $r < r_0$, зберігаємо новий розв'язок як поточний $x_0=x$, $r_0=r$.

5. Збільшуємо значення лічильника $count++$. Якщо $count$ рівний N , закінчуємо пошук.

Послідовність кроків виконання алгоритму наведено на блок-схемі (рис. 4).



Рис. 4. Блок-схема роботи генетичного алгоритму

Проведення досліджень. Для визначення і порівняння ефективності кожного з алгоритмів було проведено по 20 експериментів на 3 різних типах вхідних даних: 1) площиною 200 на 200 клітинок і 80 фігурами; 2) площиною 300 на 300 клітинок і 170 фігурами; 3) площиною 400 на 400 клітинок і 300 фігурами. Розмір площини, кількість фігур і кількість ітерацій було обрано експериментальним шляхом так, щоб дозволити відображення роботи алгоритмів з різними площинами при кількості фігур, яка б не перевищувала максимальну місткість, а час пошуку рішення був схожим незалежно від вибраного способу. Результати експериментів подано в табл. 1. У наведеній таблиці подано значення якості отриманих результатів для кожного з алгоритмів.

Для порівняння роботи алгоритмів ці результати зображено за допомогою графіків на рис. 5–7. У лівій шкалі вказано оцінки якості результатів, а в нижній номери ітерацій проведених дослідів.

Таблиця 1

Оцінки якості результатів роботи алгоритмів

Площина: 200 x 200 Кількість фігур: 80			Площина: 300 x 300 Кількість фігур: 170			Площина: 400 x 400 Кількість фігур: 300		
ІВ	ІМ	ГА	ІВ	ІМ	ГА	ІВ	ІМ	ГА
5509	5433	6376	16168	16191	17440	21593	21208	23196
5652	5804	6426	16223	17286	17419	21593	21405	23239
5649	6272	6225	16255	17061	18054	21593	22523	23274
5652	5797	6785	16168	16892	17364	21593	22464	23067
5645	5727	6242	16168	16285	17327	21593	23253	23439
5652	5710	6877	16168	17391	17443	21601	22335	23297
5652	5389	6357	16168	17242	17566	21593	21525	23135
5652	5623	6429	16168	15603	17434	21593	21632	23558
5643	4843	6295	16168	16665	17657	21593	21067	23212
5652	5927	6275	16108	16503	17787	21564	21607	23470
5652	6335	6373	16255	16250	17603	21641	22228	23381
5652	5068	6529	16168	16310	17496	21593	21811	23587
5652	5166	6521	16168	16952	17856	21676	21667	23255
5652	5642	6357	16168	16852	17820	21593	20690	23612
5649	5432	6405	16177	17237	17449	21593	21648	22903
5640	5991	6509	16168	17158	17354	21593	22392	23058
5640	5978	6391	16168	16881	17766	21593	21702	23143
5652	5957	6611	16168	16856	17582	21593	22635	23521
5645	5762	6314	16223	16155	17734	21673	21343	23329
5640	5828	6313	16129	16457	17593	21593	21811	23380

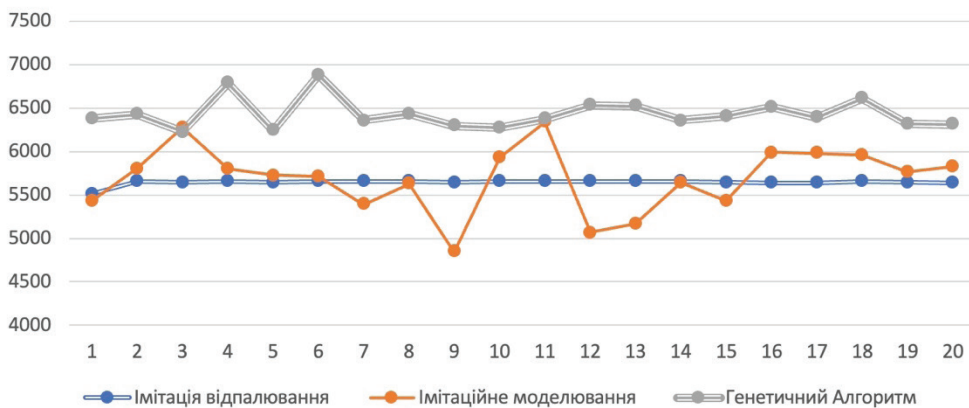


Рис. 5. Порівняння результатів на площині 200 на 200 клітинок

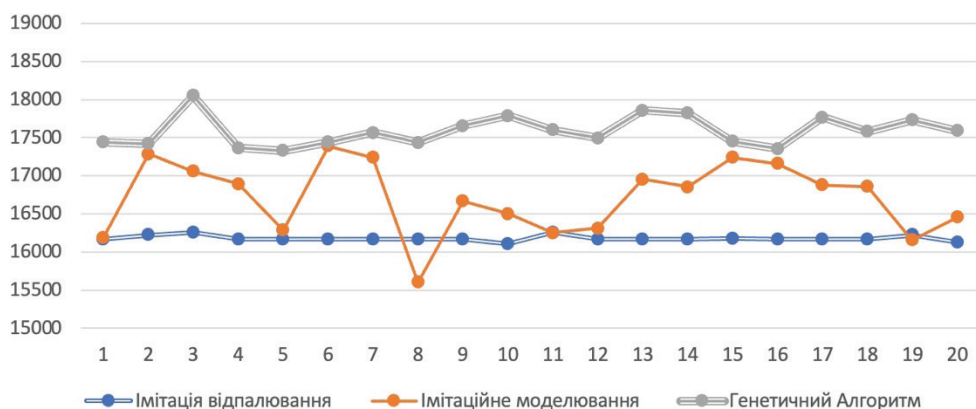


Рис. 6. Порівняння результатів на площині 300 на 300 клітинок

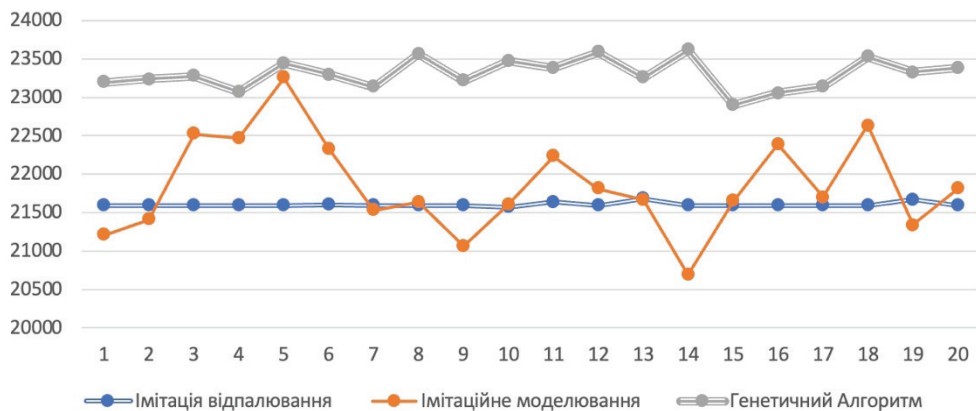


Рис. 7. Порівняння результатів на площині 400 на 400 клітинок

З наведених графіків можна зробити висновок, що в кожному з варіантів розмірів площин найкращі результати показує генетичний алгоритм. Метод імітації відпалювання показує низьку ефективність, оскільки його складніше застосувати в задачах такого типу. Для його застосування потрібно знаходити 2 фігури, які можна змінити місцями, що не завжди є можливим. Метод імітаційного моделювання інколи може показувати результати близькі до генетичного алгоритму, але все ж не є достатньо стабільним.

Висновки. Отже, розглянуто застосування методів оптимізації на прикладі задачі розміщення векторних графічних об'єктів на площині. Для вирішення цієї задачі в статті запропоновано використання наближених методів оптимізації: генетичного алгоритму, методу імітації відпалу та імітаційного моделювання. Було створено програмний застосунок для порівняння ефективності алгоритмів, який використовує спрощені вхідні дані — прямокутні об'єкти, що розміщуються на

площинах різного розміру. Результати цього дослідження можуть сприяти розробці більш ефективних алгоритмів для задач оптимізації у різних виробничих та інших прикладних галузях, забезпечуючи економію ресурсів та підвищення продуктивності. Отримані показники свідчать про високий потенціал застосування генетичних алгоритмів у задачах оптимізації розміщення графічних об'єктів, а також вказують на необхідність подальших досліджень для вдосконалення методів і підвищення їх ефективності в різних умовах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Jayananden Meenakchisundaram. Review on Building A Cost Efficient Pen Plotter. *International Research Journal of Engineering and Technology (IRJET)*. 2020. Pp. 352–355.
2. Боднар О. С. Інформаційна система для вивчення та підбору алгоритмів дослідження операцій. Аналіз та порівняння алгоритмів. 2022.
3. Братушка С. М. Імітаційне моделювання як інструмент дослідження складних економічних систем. Вісник Української академії банківської справи. 2009. № 2 (27). С. 113–118.
4. Мартинова О. В., Степанова К. В. Генетичний алгоритм для розв'язання оптимізаційних задач. Сучасні проблеми управління підприємствами: теорія та практика. 2019.
5. John Zukowski. *The Definitive Guide to Java Swing*. New York. 2005.
6. Гавриленко В. В., Івохіна К. Є., Рудоман Н. В. Про застосування методу імітації відпалу для розв'язання нечіткої задачі комівояжера. *Control, Navigation and Communication Systems*. 2022. № 3. С. 60–63.
7. Zhanghan Li, Shuming Gan. Optimal arrangement of heliostat field based on simulated annealing algorithm. *Highlights in Science Engineering and Technology*. 2024. 82. Pp. 172–180.
8. Colin Richard Reeves. *Genetic Algorithms. Handbook of Metaheuristics*. 2010. Pp. 109–139.

REFERENCES

1. Jayananden Meenakchisundaram. Review on Building A Cost Efficient Pen Plotter: *International Research Journal of Engineering and Technology (IRJET)*. 2020, 352–355 (in English).
2. Bodnar, O. S. *Informatsiina systema dlia vyvchennia ta pidboru alhorytmiv doslidzhennia operatsii. Analiz ta porivniannia alhorytmiv*. 2022 (in Ukrainian).
3. Bratushka, S. M. (2009). *Imitatsiine modeliuвання yak instrument doslidzhennia skladnykh ekonomichnykh system: Visnyk Ukrainskoi akademii bankivskoi spravy*, 2 (27), 113–118 (in Ukrainian).
4. Martynova, O. V., & Stiepanova, K. V. (2019). *Henetychnyi alhorytm dlia rozv'iazannia optyimizatsiinykh zadach: Suchasni problemy upravlinnia pidpriemstvamy: teoriia ta praktyka* (in Ukrainian).
5. John, Zukowski. (2005). *The Definitive Guide to Java Swing*. New York (in English).
6. Havrylenko, V. V., Ivokhina, K. Ye., & Rudoman, N. V. (2022). *Pro zastosuvannia metodu imitatsii vidpalu dlia rozv'iazannia nechitkoi zadachi komivoiazhera: Control, Navigation and Communication Systems*, 3, 60–63 (in Ukrainian).
7. Zhanghan, Li, & Shuming Gan. (2024). *Optimal arrangement of heliostat field based on simulated annealing algorithm: Highlights in Science Engineering and Technology*, 82, 172–180 (in English).

8. Colin Richard Reeves. Genetic Algorithms. Handbook of Metaheuristics. 2010, 109–139 (in English).

doi: 10.32403/1998-6912-2024-1-68-61-70

APPLICATION OF OPTIMIZATION METHODS IN THE PROBLEM OF PLACEMENT OF VECTOR GRAPHIC OBJECTS ON THE PLANE

D. Yu. Palamarchuk, O. V. Tymchenko, V. O. Demchenko

*Ukrainian Academy of Printing,
19, Pid Holoskom St., Lviv, 79020, Ukraine
o_tymch@ukr.net*

The article considers a comprehensive comparison of optimization algorithms for solving the problem of placing vector graphic objects on a plane. Algorithms are described in the form of block diagrams, which allow one to clearly understand their structure and principles of operation. These diagrams provide a step-by-step visual representation of the processes involved, making it easier to follow the logical process and identify the key components of each algorithm. Software is developed to evaluate the effectiveness of these methods. This program uses simplified input data in the form of rectangular objects that are placed on planes of different sizes. A set of rectangular objects and different dimensions of the plane makes it possible to perform a variety of analysis in different scenarios, guaranteeing the reliability of the results. Research results are carefully presented in the form of graphs and tables that clearly illustrate the effectiveness of each method. Furthermore, the article provides a thorough discussion of the obtained results and conclusions, indicating the high potential of optimization search methods in solving the problem of placing graphic objects. It highlights the practical implications of these findings, suggesting that these methods can significantly improve efficiency and resource utilization in various applications, such as manufacturing, printing, and layout design. The results of this study can guide future development and innovation in optimization methods, contributing to advances in fields that require accurate and efficient feature placement strategies.

Keywords: *optimization, comparative analysis, vector graphic objects, genetic algorithm, simulation modeling, annealing simulation method.*

Стаття надійшла до редакції 06.05.2024.

Received 06.05.2024.