

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна

Кваліфікаційна наукова
праця на правах рукопису

Гаврилюк Єгор Андрійович

УДК 004.8:519.87

ДИСЕРТАЦІЯ

МОДЕЛІ, МЕТОДИ ТА ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ГІБРИДНОЇ ЕВОЛЮЦІЙНО-РОЙОВОЇ ОПТИМІЗАЦІЇ В ЗАДАЧАХ ПОКРИТТЯ У ТЕРИТОРІАЛЬНО-РОЗПОДІЛЕНИХ СИСТЕМАХ

Спеціальність 122 – Комп'ютерні науки
Галузь знань 12 – Інформаційні технології

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Гаврилюк Є.А.

Науковий керівник: Яковлев Сергій Всеволодович, член-кореспондент НАН
України, доктор фізико-математичних наук, професор

Харків – 2026

АНОТАЦІЯ

Гаврилюк Є. А. Моделі, методи та інформаційна технологія гібридної еволюційно-ройової оптимізації в задачах покриття у територіально-розподілених системах. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття ступеня доктора філософії за спеціальністю 122 – Комп'ютерні науки (Галузь знань 12 – Інформаційні технології). – Харківський національний університет імені В. Н. Каразіна, Міністерства освіти і науки України, Харків, 2026.

Дисертаційна робота присвячена розробці моделей, методів та інформаційної технології гібридної еволюційно-ройової оптимізації для розв'язання задач неперервного максимального геометричного покриття у територіально-розподілених системах. Задача полягає в оптимальному розміщенні скінченної сукупності геометричних об'єктів довільної форми з метою максимізації площі покриття заданої області на евклідовій площині за наявності геометричних обмежень та заборонених зон. Ця задача є NP-важкою, характеризується мультиекстремальним ландшафтом цільової функції та високою обчислювальною вартістю оцінки площі покриття, що обумовлює необхідність розробки спеціалізованих гібридних методів оптимізації. Запропонований підхід базується на поєднанні глобального стохастичного пошуку ройових метаевристик із детермінованою локальною оптимізацією на основі квазіньютонівських методів та нейромережевих механізмів керування оптимізаційним процесом. Результати дисертації мають як наукову новизну, так і практичне значення для побудови ефективних систем просторового планування та оптимізації сервісного покриття.

Перший розділ присвячений комплексному аналізу наукових джерел з питань математичного моделювання задач максимального покриття та методів їх розв'язання. Систематичний огляд літератури виконано за трьома напрямками: математичне моделювання задач покриття – від класичних дискретних

цілочисельних постановок до неперервних узагальнень з об'єктами довільної форми; методи еволюційно-ройової та гібридної оптимізації, включаючи ройові метаевристики та меметичні алгоритми; а також прикладні аспекти задач покриття у територіально-розподілених системах, що охоплюють задачі аварійного реагування, розгортання сенсорних мереж, комерційного розміщення об'єктів обслуговування та задачі в геоінформаційних системах. Розглянуто математичний апарат, що забезпечує формалізацію умов неперетину, належності та покриття для об'єктів складної форми. Показано, що сучасні бібліотеки обчислювальної геометрії створюють практичну основу для точного обчислення площі перетину геометричних фігур довільної конфігурації.

На основі проведеного аналізу виявлено ключові прогалини, що визначають напрям дисертаційного дослідження: переважна більшість існуючих методів зорієнтована на об'єкти простої форми та не забезпечує комплексного розв'язання задачі неперервного максимального покриття із зонами обслуговування довільної форми та заборонними зонами; у літературі відсутнє систематичне порівняння ройових метаевристик у поєднанні з методами локальної оптимізації для задач покриття; питання архітектурної формалізації відповідної інформаційної технології залишається нерозглянутим. Згідно з виявленими прогалинами сформульовано мету дослідження, а також математичну модель задачі нелінійного програмування у просторі неперервних параметрів розміщення з переходом до безумовної оптимізації через механізм зовнішньої штрафної функції.

У другому розділі детально розглянуто та описано методичний апарат для розв'язання задачі максимального покриття. Для обчислення цільової функції проаналізовано та систематизовано п'ять методів оцінки площі конфігурації покриття, що утворюють спектр від найшвидших із низькою точністю, таких як метод мінімальних описаних прямокутників, до найточніших із високою обчислювальною вартістю – точні булеві операції засобами бібліотеки

обчислювальної геометрії. Стохастичний метод Монте-Карло з просторовою індексацією R-Tree визначено як основний метод обчислення цільової функції для багаторазового виклику в циклі оптимізації завдяки оптимальному компромісу між точністю та обчислювальною вартістю. Сформульовано рекомендації щодо вибору методу залежно від етапу оптимізації та кількості покриваючих об'єктів. Проведено обчислювальні експерименти, що підтвердили теоретичні оцінки точності та складності кожного методу.

Для оптимізаційної компоненти чотири ройові метаевристики обчислювального інтелекту (оптимізацію роєм частинок, пошук риб'ячої зграї, алгоритм світлячків та алгоритм штучної бджолоїної колонії) адаптовано до задачі максимального покриття, де кожен агент рою кодує повну конфігурацію розміщення покриваючих об'єктів у високорозмірному просторі параметрів. Для кожного алгоритму обґрунтовано вибір метапараметрів та визначено рекомендовані діапазони їх значень. Порівняльний аналіз алгоритмів на серіях задач зростаючої складності дозволив визначити характерні переваги та обмеження кожного методу. Водночас встановлено, що спільним обмеженням усіх розглянутих ройових алгоритмів є недостатня здатність до точного локального уточнення розв'язків поблизу знайденого оптимуму, що обумовлює необхідність розробки гібридних алгоритмів, які поєднують глобальний ройовий пошук з локальною оптимізацією на основі градієнтних методів.

У третьому розділі досліджено гібридні методи розв'язання задачі неперервного максимального покриття, що поєднують глобальний стохастичний пошук ройових метаевристик із детермінованою локальною оптимізацією та нейромережовим керуванням. Обґрунтовано застосування квазіньютонівського методу BFGS як локальної компоненти гібридних алгоритмів. Розглянуто підхід до чисельного оцінювання градієнта цільової функції на основі матриці площ попарних перетинів покриваючих об'єктів, що забезпечує суттєве прискорення обчислень порівняно з прямим перерахунком площі складного об'єкта.

Квазіньютонівський метод із скінченно-різницевою апроксимацією градієнта продемонстрував суперлінійну збіжність та здатність знаходити високоякісні локальні оптимуми.

Розроблено узагальнену меметичну методологію для задач максимального геометричного покриття, яка поєднує чотири ройові метаевристики з локальною оптимізацією BFGS. Для кожної з чотирьох гібридних комбінацій визначено оптимальну частоту виклику локального пошуку, рекомендовані параметри та критерії зупинки. Показано, що гібридна схема є універсальною і може бути застосована до будь-якого з розглянутих ройових алгоритмів без зміни їхньої базової структури. Встановлено, що меметичний підхід стабільно підвищує площу покриття порівняно з чистими ройовими методами при помірному збільшенні обчислювальних витрат. Проведений порівняльний аналіз підтвердив, що гібридний метод стабільно перевершує як чистий ройовий пошук, так і мультістартову локальну оптимізацію при однаковому обчислювальному бюджеті, причому перевага зростає зі збільшенням кількості покриваючих об'єктів.

Також запропоновано нейронно-керовані гібридні методи максимізації покриття, що інтегрують нейронні мережі у оптимізаційний процес за двома напрямками: адаптивне керування штрафним коефіцієнтом та динамічне налаштування гіперпараметрів ройових алгоритмів. Для кожного напрямку визначено архітектуру нейронної мережі, вхідні ознаки та функцію втрат. На основі порівняльного аналізу трьох парадигм навчання обґрунтовано вибір навчання з підкріпленням як найбільш перспективного підходу, що забезпечує автоматичне формування стратегії керування без потреби у попередньому зборі еталонних даних.

В четвертому розділі представлено цілісну UML-орієнтовану інформаційну технологію розв'язання неперервних задач максимального покриття з об'єктами довільної форми. Технологія формалізує архітектуру,

структури даних, інформаційні потоки та алгоритмічні компоненти системи, забезпечуючи відтворюваність, масштабованість і прозорість процесу проектування програмних рішень для задач покриття. Сформульовано функціональні та нефункціональні вимоги до системи, визначено атрибути якості. Для обґрунтування вибору архітектурної моделі застосовано метод багатокритеріального прийняття рішень TOPSIS з експертними оцінками альтернатив, на основі якого обрано трирівневу архітектуру.

На основі обраної архітектури побудовано комплекс UML-діаграм, що формалізують структурні, поведінкові та інтеграційні аспекти системи та формують стандартизований архітектурний каркас. Введено універсальний метамодуль оцінювання площі покриття, що об'єднує п'ять незалежних стратегій обчислення та підтримує їх адаптивне перемикання. Розроблено оригінальну модель життєвого циклу конфігурації у вигляді діаграми станів, яка формалізує переходи між станами оцінювання, покращення та архівації кандидатних рішень. Сформовано композитну структуру оптимізаційного ядра, що визначає внутрішню організацію глобального пошуку, локального покращення, моніторингу та стратегічного керування. На відміну від існуючих досліджень, що концентруються лише на алгоритмах, запропонована технологія враховує повний життєвий цикл програмної системи.

Сукупність отриманих у дисертації наукових результатів, підтвердження факту їх достовірності, наукової та практичної значущості дають змогу вважати, що сформульована наукова задача розробки моделей, методів та інформаційної технології гібридної еволюційно-ройової оптимізації для підвищення ефективності розв'язання задач максимального геометричного покриття у територіально-розподілених системах, є розв'язаною, а поставлена мета – досягнута.

Ключові слова: задача максимального покриття, оптимізація, математичне моделювання, розміщення, еволюційно-ройові алгоритми, метод

BFGS, гібридна система, безпілотні літальні апарати, інформаційна технологія, нейронні мережі, машинне навчання, глибоке навчання, прийняття рішень, обчислювальна геометрія, агентний підхід.

ABSTRACT

Havryliuk Ye. A. Models, Methods and Information Technology of Hybrid Evolutionary-Swarm Optimization in Coverage Problems in Territorially Distributed Systems. – Qualifying scientific work as a manuscript.

Thesis for the degree of Doctor of Philosophy in Specialty 122 – Computer Science (Field of Knowledge 12 – Information Technologies). – V. N. Karazin Kharkiv National University, Ministry of Education and Science of Ukraine, Kharkiv, 2026.

The dissertation is devoted to the development of models, methods and information technology of hybrid evolutionary-swarm optimization for solving continuous maximum geometric coverage problems in territorially distributed systems. The problem consists in the optimal placement of a finite collection of arbitrarily shaped geometric objects to maximize the coverage area of a given region in the Euclidean plane subject to geometric constraints and restricted zones. This problem is NP-hard, features a multiextremal objective-function landscape and a high computational cost of coverage area evaluation, which necessitates the development of specialized hybrid optimization methods. The proposed approach combines global stochastic search by swarm metaheuristics with deterministic local optimization based on quasi-Newtonian methods and neural-network mechanisms for controlling the optimization process. The results of the dissertation possess both scientific novelty and practical significance for building efficient spatial planning and service coverage optimization systems.

Chapter 1 presents a comprehensive analysis of the scientific literature on mathematical modeling of maximum coverage problems and methods for their solution. A systematic literature review is carried out along three directions: mathematical modeling of coverage problems, ranging from classical discrete integer formulations to continuous generalizations with arbitrarily shaped objects; evolutionary-swarm and hybrid optimization methods, including swarm metaheuristics and memetic algorithms; and applied aspects of coverage problems in territorially distributed systems,

encompassing emergency response, sensor network deployment, commercial facility location, and geographic information systems. The mathematical apparatus that enables the formalization of non-overlap, containment and coverage conditions for complex-shaped objects is examined. It is shown that modern computational geometry libraries provide a practical foundation for the exact computation of the intersection area of geometric figures of arbitrary configuration.

Based on the analysis, the following key gaps that define the direction of the dissertation research have been identified: the vast majority of existing methods are oriented toward simple-shaped objects and do not provide a comprehensive solution to the continuous maximum coverage problem with arbitrarily shaped service zones and restricted zones; the literature lacks a systematic comparison of swarm metaheuristics combined with local optimization methods for coverage problems; and the issue of architectural formalization of the corresponding information technology remains unaddressed. In accordance with the identified gaps, the research objective is formulated along with a mathematical model of a nonlinear programming problem in the space of continuous placement parameters, with a transition to unconstrained optimization via an exterior penalty function mechanism.

Chapter 2 provides a detailed description of the methodological apparatus for solving the maximum coverage problem. For the computation of the objective function, five methods for estimating the coverage configuration area are analyzed and systematized, forming a spectrum from the fastest with low accuracy, such as the minimum bounding rectangle method, to the most accurate with high computational cost -- exact Boolean operations using a computational geometry library. The stochastic Monte Carlo method with R-Tree spatial indexing is identified as the primary method for objective function evaluation within the optimization loop, owing to its optimal trade-off between accuracy and computational cost. Recommendations for method selection depending on the optimization stage and the number of covering objects are

formulated. Computational experiments confirming the theoretical accuracy and complexity estimates for each method are conducted.

For the optimization component, four swarm metaheuristics of computational intelligence (Particle Swarm Optimization, Fish School Search, Firefly Algorithm and Artificial Bee Colony) are adapted to the maximum coverage problem, where each swarm agent encodes a complete placement configuration of covering objects in a high-dimensional parameter space. For each algorithm, the choice of meta-parameters is justified and recommended value ranges are determined. A comparative analysis of the algorithms on problem series of increasing complexity reveals the characteristic advantages and limitations of each method. At the same time, it is established that a common limitation of all considered swarm algorithms is their insufficient ability to accurately refine solutions in the vicinity of the found optimum, which necessitates the development of hybrid algorithms that combine global swarm search with gradient-based local optimization.

Chapter 3 investigates hybrid methods for solving the continuous maximum coverage problem that combine global stochastic search of swarm metaheuristics with deterministic local optimization and neural-network control. The application of the quasi-Newtonian BFGS method as the local component of hybrid algorithms is justified. An approach to numerical gradient estimation of the objective function based on a matrix of pairwise intersection areas of covering objects is considered, which provides a substantial speedup compared to direct recomputation of the complex object's area. The quasi-Newtonian method with finite-difference gradient approximation demonstrates superlinear convergence and the ability to find high-quality local optima.

A generalized memetic methodology for maximum geometric coverage problems is developed, combining four swarm metaheuristics with BFGS local optimization. For each of the four hybrid combinations, the optimal frequency of local search invocation, recommended parameters and stopping criteria are determined. It is

shown that the hybrid scheme is universal and can be applied to any of the considered swarm algorithms without modifying their basic structure. It is established that the memetic approach consistently improves coverage area compared to pure swarm methods with a moderate increase in computational cost. The comparative analysis confirms that the hybrid method consistently outperforms both pure swarm search and multi-start local optimization under an equal computational budget, with the advantage growing as the number of covering objects increases.

Neural-network-driven hybrid methods for coverage maximization are also proposed, integrating neural networks into the optimization process along two directions: adaptive control of the penalty coefficient and dynamic tuning of swarm algorithm hyperparameters. For each direction, the neural network architecture, input features and loss function are specified. Based on a comparative analysis of three learning paradigms, reinforcement learning is selected as the most promising approach, as it enables the automatic formation of a control strategy without the need for prior collection of reference data.

Chapter 4 presents a complete UML-oriented information technology for solving continuous maximum coverage problems with arbitrarily shaped objects. The technology formalizes the architecture, data structures, information flows and algorithmic components of the system, ensuring reproducibility, scalability and transparency of the software design process for coverage problems. Functional and non-functional system requirements are formulated and quality attributes are defined. To justify the selection of the architectural model, the TOPSIS multi-criteria decision-making method with expert assessments of alternatives is applied, on the basis of which a three-tier architecture is chosen.

Based on the selected architecture, a set of UML diagrams is constructed that formalize the structural, behavioral and integration aspects of the system, forming a standardized architectural framework. A universal metamodule for coverage area evaluation is introduced, unifying five independent computation strategies and

supporting their adaptive switching. An original configuration lifecycle model in the form of a state machine diagram is developed, formalizing transitions between evaluation, improvement and archival states of candidate solutions. A composite structure of the optimization engine is formed, defining the internal organization of global search, local improvement, monitoring and strategic control. Unlike existing studies that focus solely on algorithms, the proposed technology accounts for the full lifecycle of the software system.

The totality of scientific results obtained in the dissertation, together with confirmation of their validity, scientific and practical significance, allows one to consider the stated scientific problem of developing models, methods and information technology of hybrid evolutionary-swarm optimization for improving the efficiency of solving maximum geometric coverage problems in territorially distributed systems as solved, and the stated objective as achieved.

Keywords: maximum coverage problem, optimization, mathematical modeling, placement, evolutionary swarm algorithms, BFGS method, hybrid system, unmanned aerial vehicles, information technology, neural networks, machine learning, deep learning, decision making, computational geometry, agent approach.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Статті у наукових фахових виданнях України

1. Гаврилюк Є. А., Коробчинський К. П. UML-орієнтована інформаційна технологія для неперервних задач максимального покриття з об'єктами довільної форми. *Вісник Харківського національного університету імені В. Н. Каразіна. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2025. Вип. 67. С. 18–34. DOI: 10.26565/2304-6201-2025-67.

(Особистий внесок здобувача: розробка UML-орієнтованої інформаційної технології для неперервних задач максимального покриття з об'єктами довільної форми, побудова комплексу UML-діаграм (варіантів використання, класів, діяльності, послідовностей, компонентів, композитної структури, станів та розгортання), формування тривірневої модульної архітектури інформаційної системи, програмна реалізація модулів просторової аналітики та оптимізації покриття, підготовка матеріалів статті. Відповідні результати наведені в теоретичній та практичній частинах роботи.

Особистий внесок Коробчинського К. П.: спільна постановка задач та її аналіз, формування загальної концепції дослідження, валідація результатів дослідження.).

2. Гаврилюк Є. А. Нейронно-керований гібридний метод для оптимального покриття території з обмеженням на розташування. *Системи та технології*. 2025. Т. 70, № 2. С. 112–118. DOI: 10.32782/2521-6643-2025-2-70.12.

3. Гаврилюк Є. А. Гібридні ройові метаевристики для неперервних задач максимального покриття зі складними формами об'єктів. *Проблеми керування та інформатики*. 2026. № 1. С. 14–33. DOI: 10.34229/1028-0979-2026-1-2.

4. Гаврилук Є. А., Струков В. М., Узлов Д. Ю. Інтегрований фреймворк на основі методів штучного інтелекту для задач неперервного покриття з геометричними обмеженнями. *Системні технології*. 2026. Т. 1, № 162. С. 160-176. DOI: 10.34185/1562-9945-5-162-2026-18.

(Особистий внесок здобувача: розробка та формалізація інтегрованого AI-базованого фреймворку для задач неперервного покриття з геометричними обмеженнями, реалізація та порівняння ройових і меметичних методів оптимізації, розробка нейромережових сурогатних моделей для апроксимації цільової функції покриття та адаптивних штрафних механізмів, проведення обчислювальних експериментів на задачах з різними типами геометричних обмежень, підготовка матеріалів статті. Відповідні результати наведені в теоретичній та практичній частинах роботи.

Особистий внесок Струков В. М.: спільна постановка задач та її аналіз, валідація результатів дослідження.

Особистий внесок Узлов Д. Ю.: спільна постановка задач та її аналіз, формування загальної концепції дослідження та інтеграція AI-фреймворку до задач покриття.).

**Наукові праці у наукових фахових виданнях,
що входять до міжнародних наукометричних баз**

5. Uzlov D., Havryliuk Y., Hushchyn I., Strukov V., Yakovlev S. Handling outliers in swarm algorithms: a review. *CEUR Workshop Proceedings*. 2024. Vol. 3777. P. 430-437. *Proceedings of the 4th International Workshop of IT-Professionals on Artificial Intelligence (ProfIT AI 2024)*, Cambridge, MA, USA, September 25–27, 2024). URL: <https://ceur-ws.org/Vol-3777/short13.pdf> (**Scopus**).

(Особистий внесок здобувача: модифікація методу рою часток, дослідження впливу викидів на збіжність ройових алгоритмів у задачах

покриття, презентація доповіді. Відповідні результати наведені в аналітичній частині роботи.

Особистий внесок Uzlov D.: постановка задачі дослідження, загальна концепція дослідження.

Особистий внесок Hushchyn I.: збір та систематизація літературних джерел, аналіз сучасного стану предмету дослідження.

Особистий внесок Strukov V.: перевірка наукової достовірності отриманих результатів.

Особистий внесок Yakovlev S.: наукове керівництво, планування обчислювальних експериментів, перевірка наукової достовірності отриманих результатів.).

6. Pichugina O., Kirichenko L., Koliechkina L., Skob Yu., Havryliuk Ye. Exploring one linear permutation-based stochastic optimization problem with applications. *Proceedings of the 13th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2025)*, Gliwice, Poland, 4-6 September 2025. DOI: 10.1109/IDAACS68557.2025.11322262 (**Scopus**).

(Особистий внесок здобувача: огляд статистичних методів оптимізації, збір та систематизація літературних джерел, аналіз сучасного стану предмету дослідження.

Особистий внесок Pichugina O.: постановка задачі, математична формалізація задачі комбінаторної оптимізації.

Особистий внесок Kirichenko L.: аналіз стохастичних моделей, перевірка наукової достовірності результатів.

Особистий внесок Koliechkina L.: розробка комбінаторних методів розв'язання.

Особистий внесок Skob Yu.: участь у програмній реалізації та візуалізації результатів.).

7. Yakovlev S., Havryliuk Y., Matsyi O., Hulianytskyi A., Kirpich A. A neural network-driven adaptive approach for maximum coverage location with restricted zones. *CEUR Workshop Proceedings*. 2025. Vol. 4164. P. 312-322. *Proceedings of the 5th International Workshop of IT-Professionals on Artificial Intelligence (ProfIT AI 2025)*, Liverpool, UK, October 15–17, 2025). URL: <https://ceur-ws.org/Vol-4164/paper20.pdf> (**Scopus**).

(Особистий внесок здобувача: розробка та реалізація нейромережевого адаптивного підходу для задачі максимального покриття із забороненими зонами, програмна реалізація гібридного алгоритму, що поєднує ройову оптимізацію з нейромережевим керуванням штрафними функціями, проведення обчислювальних експериментів та аналіз результатів. Відповідні результати наведені в практичній частині роботи.

Особистий внесок Yakovlev S.: наукове керівництво, планування обчислювальних експериментів, перевірка наукової достовірності отриманих результатів.

Особистий внесок Matsyi O.: участь в аналізі результатів, аналіз практичного застосування.

Особистий внесок Hulianytskyi A.: участь у математичному обґрунтуванні методу.

Особистий внесок Kirpich A.: участь у програмній реалізації та тестуванні).

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	21
ВСТУП.....	22
РОЗДІЛ 1. АНАЛІЗ СУЧАСНОГО СТАНУ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ	32
1.1 Сучасні підходи до моделювання задач максимального геометричного покриття	32
1.1.1 Моделювання об'єктів довільної форми та заборонених зон.....	35
1.2 Сучасні методи еволюційно-ройової та гібридної оптимізації.....	37
1.2.1 Метаевристичні методи	38
1.2.2 Меметичні методи	40
1.3 Прикладні задачі максимального геометричного покриття.....	41
1.3.1 Розміщення аварійних служб	42
1.3.2 Комерційне розміщення та ділова аналітика	43
1.3.3 Гіс-системи.....	44
1.4 Постановка задачі та математична модель	46
1.4.1 Базові об'єкти та параметризація.....	46
1.4.2 Обмеження на параметри розміщення	50
1.4.3 Оптимізаційна постановка.....	51
Висновки до розділу 1.....	52
РОЗДІЛ 2. МЕТОДИ ОБЧИСЛЕННЯ ЦІЛЬОВОЇ ФУНКЦІЇ ТА РОЙОВІ МЕТАЕВРИСТИКИ В ЗАДАЧАХ МАКСИМАЛЬНОГО ПОКРИТТЯ	54

	18
2.1 Моделі та методи обчислення цільової функції в задачах максимального покриття.....	54
2.1.1 Метод мінімальних описаних прямокутників	55
2.1.2 Метод Монте-Карло	57
2.1.3 Метод Монте-Карло з просторовою індексацією R-Tree	59
2.1.4 Апроксимація формулою включення-виключення другого порядку	61
2.1.5 Точні геометричні бібліотеки.....	63
2.2 Експериментальні дослідження та порівняльний аналіз методів оцінки площі покриття.....	67
2.2.1 Рекомендації щодо вибору методу	72
2.3 Застосування ройових алгоритмів в задачах максимального покриття.....	73
2.3.1 Обґрунтування вибору ройових алгоритмів для задач покриття ..	75
2.3.2 Оптимізація роєм частинок	77
2.3.3 Пошук риб'ячої зграї.....	79
2.3.4 Алгоритм світлячків.....	81
2.3.5 Штучна бджолина колонія	82
2.4 Експериментальні дослідження та порівняльний аналіз ройових алгоритмів в задачах максимального покриття.....	84
2.4.1 Результати обчислювальних експериментів.....	85
2.4.2 Порівняльний аналіз ройових алгоритмів.....	89
Висновки до розділу 2.....	93

РОЗДІЛ 3. ГІБРИДНІ ЕВОЛЮЦІЙНО-РОЙОВІ ТА НЕЙРОКЕРОВАНІ МЕТОДИ ОПТИМІЗАЦІЇ В ЗАДАЧАХ МАКСИМАЛЬНОГО ПОКРИТТЯ 95

3.1 Застосування методів локальної оптимізації в задачах максимального покриття.....	95
3.1.1 Поняття та оцінка градієнту	96
3.1.2 Методи локальної оптимізації з використанням градієнтів	98
3.2 Гібридні алгоритми в задачах максимального покриття.....	99
3.2.1 Переваги меметичного підходу.....	101
3.2.2 Інтеграція з ройовими методами.....	102
3.2.3 Порівняльні властивості ройових методів та їхніх меметичних модифікацій.....	105
3.2.4 Недоліки меметичного підходу.....	106
3.2.5 Рекомендації щодо вибору гібридного ройового алгоритму	107
3.3 Порівняльний аналіз гібридних методів в задачах покриття.....	108
3.3.1 Серія експериментів	110
3.4 Нейронно-керовані гібридні методи максимізації покриття.....	114
3.4.1 Нейронно-керований адаптивний механізм штрафних коефіцієнтів	114
3.4.2 Динамічне налаштування гіперпараметрів ройових алгоритмів .	118
3.4.3 Парадигми навчання нейромережових компонентів	119
Висновки до розділу 3.....	122

РОЗДІЛ 4. РОЗРОБКА UML-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ РОЗВ'ЯЗАННЯ НЕПЕРЕРВНИХ ЗАДАЧ МАКСИМАЛЬНОГО ГЕОМЕТРИЧНОГО ПОКРИТТЯ 125

	20
4.1 Вимоги до інформаційної технології.....	125
4.1.1 Функціональні вимоги	125
4.1.2 Нефункціональні вимоги	126
4.1.3 Атрибути якості.....	127
4.2 Обґрунтування вибору архітектурної моделі	128
4.2.1 Альтернативні моделі системної архітектури	130
4.2.2 Застосування методу TOPSIS.....	133
4.3 UML-специфікація інформаційної технології.....	135
4.3.1 Діаграма варіантів використання.....	135
4.3.2 Діаграма класів	138
4.3.3 Діаграма компонентів	141
4.3.4 Діаграма композитної структури	143
4.4 Поведінкові моделі інформаційної технології.....	145
4.4.1 Діаграма діяльності	145
4.4.2 Діаграма послідовності	148
4.4.3 Діаграма станів	150
Висновки до розділу 4.....	152
ВИСНОВКИ.....	154
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	160
ДОДАТОК А	175
ДОДАТОК Б.....	179

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ABC	– алгоритм штучної бджолоїної колонії
АНР	– метод аналізу ієрархій
BFGS	– метод Бройдена–Флетчера–Голдфарба–Шанно
CMCLP	– задача неперервного максимального покриття
DQN	– глибока Q-мережа
FA	– алгоритм світлячків
FSS	– пошук риб'ячої зграї
IE2	– апроксимація формулою включення-виключення другого порядку
L-BFGS	– метод BFGS з обмеженою пам'яттю
MBR	– мінімальний описаний прямокутник
MCLP	– задача максимального покриття
MDP	– марківський процес прийняття рішень
PPO	– проксимальна оптимізація політики
PSO	– оптимізація роєм частинок
RL	– навчання з підкріпленням
TOPSIS	– метод наближення до ідеального рішення
UML	– уніфікована мова моделювання
VAE	– варіаційний автокодувальник
БПЛА	– безпілотний літальний апарат
ГІС	– геоінформаційна система
ІТ	– інформаційна технологія
СА	– системна архітектура
СКБД	– система керування базами даних

ВСТУП

Задачі оптимального геометричного покриття належать до фундаментальних проблем прикладної математики та операційного дослідження, які виникають у найрізноманітніших практичних контекстах: від проєктування логістичних мереж і розміщення сервісних центрів до побудови сенсорних та телекомунікаційних інфраструктур, планування маршрутів безпілотних апаратів, моніторингу довкілля та просторового аналізу в геоінформаційних системах. Перехід від класичних дискретних моделей покриття до неперервних постановок зумовлений необхідністю врахування складної геометрії реальних областей, наявності заборонених зон, орієнтованих і анізотропних форм зон обслуговування, що призводить до виникнення негладких, багатоекстремальних і обчислювально дорогих цільових функцій, для яких класичні градієнтні та комбінаторні методи є недостатньо ефективними.

Актуальність теми. Сучасний розвиток територіально-розподілених систем, таких як бездротові сенсорні мережі, системи моніторингу, логістичні платформи, мережі безпілотних літальних апаратів та служби екстреного реагування, висуває підвищені вимоги до ефективності просторового покриття. Оптимальне розміщення обмеженої кількості об'єктів із зонами обслуговування довільної форми на складних за геометрією територіях з урахуванням заборонених зон та регіональної неоднорідності попиту є задачею, що за своєю природою є NP-складною, високорозмірною та мультимодальною.

Незважаючи на значний прогрес у галузі геометричних алгоритмів, напівнескінченної оптимізації та сучасних формулювань задач змішаного цілочисельного програмування, ефективне розв'язання неперервних задач максимального геометричного покриття залишається відкритою науковою проблемою. Класичні точні методи стають непрактичними зі зростанням розмірності, а стандартні метаевристики, забезпечуючи глобальний пошук, не

гарантують достатньої точності у локальних околах оптимуму. Це зумовлює актуальність розроблення гібридних підходів, які поєднують глобальний стохастичний пошук ройових алгоритмів із локальними градієнтними методами оптимізації.

Розроблювана інформаційна технологія гібридної еволюційно-ройової оптимізації спрямована на розв'язання широкого класу прикладних задач покриття, зокрема:

- оптимальне розміщення сенсорів у бездротових сенсорних мережах для забезпечення максимального покриття та зв'язності з урахуванням обмежених енергетичних ресурсів;
- планування зон обслуговування в логістичних системах та системах екстреного реагування, де покриття інтерпретується як досяжність клієнтів або постраждалих із обмеженнями за часом;
- оптимізація розміщення базових станцій телекомунікаційних мереж та інфраструктурних об'єктів у територіально-розподілених системах;
- планування траєкторій та зон патрулювання безпілотних літальних апаратів для моніторингу територій, зокрема для виявлення лісових пожеж та екологічного контролю;
- вибір оптимальних локацій для комерційних та сервісних об'єктів з урахуванням просторового розподілу попиту та конкурентного оточення.

Поєднання складної геометрії, багатовимірних просторів параметрів та високої обчислювальної вартості цільової функції природно веде до застосування метаевристичних підходів, насамперед ройових та еволюційних алгоритмів. Зокрема, алгоритми рою частинок (Particle swarm optimization, далі – PSO), пошуку зграї риб (Fish school search, далі – FSS), світлячків (Firefly algorithm, далі – FA) та штучної бджолої колонії (Artificial bee colony, далі – ABC) добре пристосовані до глобального пошуку в нерегулярних просторах. Водночас їхня точність у локальних околах оптимуму часто є недостатньою, що зумовлює

потребу в гібридизації з градієнтними методами локальної оптимізації. Паралельно активно розвиваються нейромеревеві технології, які можуть виступати сурогатними моделями для наближення цільової функції, а також інструментами автоматичного налаштування штрафних функцій у задачах з обмеженнями, забезпечуючи істотне прискорення обчислень.

Для практичної реалізації зазначених підходів необхідна спеціалізована інформаційна технологія, здатна інтегрувати геометричні обчислення, оптимізаційні модулі, нейромеревеві компоненти та інтерфейси до джерел просторових даних. Використання формальних підходів до проєктування архітектури, зокрема UML-орієнтованого (Unified Modeling Language, далі – UML) моделювання, дає змогу побудувати модульну, масштабовану систему, у якій легко поєднувати різні методи оцінювання покриття та алгоритми оптимізації.

Мета дослідження полягає в розробці та обґрунтуванні AI-базованої методології, алгоритмічних методів і UML-орієнтованої інформаційної технології гібридної еволюційно-ройової оптимізації для ефективного розв’язання неперервних задач максимального геометричного покриття в територіально-розподілених системах з урахуванням вимог до швидкодії, масштабованості та точності.

Об’єкт дослідження – процеси функціонування та розробки інтелектуальних інформаційних систем оптимізації просторового покриття в територіально-розподілених системах на основі методів обчислювального інтелекту.

Предмет дослідження – моделі, алгоритмічні методи та інформаційна технологія гібридної еволюційно-ройової оптимізації, включаючи нейросурогатні механізми, багаторівневі методи оцінювання покриття та UML-орієнтовану архітектуру, спрямовані на розв’язання неперервних задач

максимального геометричного покриття шляхом поєднання глобального популяційного пошуку з локальною градієнтною оптимізацією.

Основні завдання дисертації:

1) провести аналіз сучасного стану досліджень у галузі розв'язання неперервних задач максимального геометричного покриття та методів їх алгоритмічного забезпечення з метою виявлення обмежень існуючих підходів і визначення напрямів подальшого розвитку;

2) сформулювати обчислювальну модель неперервної задачі максимального геометричного покриття як теоретичну основу для розроблення алгоритмічних і програмних засобів гібридної оптимізації в територіально-розподілених системах;

3) розробити методологію розв'язання неперервних задач максимального геометричного покриття у територіально-розподілених системах, що ґрунтується на застосуванні методів штучного інтелекту та поєднанні еволюційно-ройових підходів із меметичними принципами;

4) розробити універсальну алгоритмічну схему реалізації гібридної оптимізації, яка забезпечує узгоджене керування глобальним стохастичним пошуком і локальним покращенням у просторі неперервних параметрів;

5) розробити та інтегрувати нейросурогатні й нейронно-адаптивні механізми для прискорення обчислення цільової функції та автоматизованого керування обмеженнями в задачах геометричного покриття;

6) удосконалити методи обчислення критерію покриття шляхом створення багаторівневої адаптивної схеми оцінювання з керованим перемиканням між наближеними та точними геометричними обчисленнями;

7) побудувати UML-орієнтовану архітектуру інформаційної технології гібридної еволюційно-ройової оптимізації, яка формалізує структуру, поведінку та життєвий цикл оптимізаційної системи і забезпечує її масштабованість та відтворюваність;

8) провести експериментальну валідацію розроблених моделей, методів та інформаційної технології й оцінити їх ефективність за показниками якості покриття, швидкодії та обчислювальної стійкості.

Методи дослідження. В процесі розв'язання поставлених задач використовувались такі методи:

– методи обчислювальної геометрії та математичного моделювання: формалізація задач покриття з використанням ω -функцій та конфігураційних просторів, точні та наближені методи обчислення площі покриття, включаючи алгоритми бібліотеки Shapely та методи Монте-Карло з просторовою індексацією;

– метаевристичні та ройові методи оптимізації: алгоритми PSO, FSS, FA, ABC та їх меметичні модифікації з інтеграцією локального оптимізатора Бройдена-Флетчера-Голдфарба-Шанно (Broyden-Fletcher-Goldfarb-Shanno, далі – BFGS) для точного уточнення розв'язків у просторі неперервних параметрів;

– методи диференціального числення та чисельної оптимізації: оцінювання градієнту цільової функції методами скінченних різниць, квазіньютонівські методи локальної оптимізації для уточнення розв'язків у гібридних алгоритмах;

– методи штучного інтелекту та машинного навчання: нейросурогатні моделі для апроксимації цільової функції, нейронно-адаптивні механізми керування штрафними коефіцієнтами, автокодувальники для генерації початкових конфігурацій;

– методи програмної інженерії та архітектурного проєктування: UML-орієнтоване моделювання, принципи модульності, компонентної декомпозиції та побудови масштабованої тривірневої архітектури інформаційної технології;

– статистичний аналіз та валідація результатів: порівняльний аналіз ефективності алгоритмів за критеріями якості покриття, швидкодії та обчислювальної стійкості на тестових задачах різної складності.

Наукова новизна отриманих результатів. Наукова новизна роботи полягає у наступному.

1. Вперше запропоновано методологію розв'язання неперервних задач максимального геометричного покриття, яка базується на інтеграції еволюційно-ройових алгоритмів, меметичних принципів та нейросурогатних моделей і, на відміну від традиційних підходів, забезпечує узгоджене поєднання глобального стохастичного пошуку з локальною градієнтною оптимізацією при зменшенні обчислювальних витрат.

2. Вперше розроблено універсальну алгоритмічну схему гібридизації популяційних методів оптимізації, яка забезпечує узгоджене керування глобальним стохастичним пошуком і локальним покращенням у просторі неперервних параметрів та є інваріантною до вибору базового метаевристичного алгоритму, що підтверджено експериментально.

3. Набула подальшого розвитку інформаційна технологія гібридної еволюційно-ройової оптимізації за рахунок створення UML-орієнтованої архітектурної моделі, яка формалізує структуру, поведінку та життєвий цикл оптимізаційної системи і забезпечує її масштабованість, відтворюваність та інтеграцію в прикладні програмні платформи територіально-розподілених систем.

4. Набули подальшого розвитку підходи до нейросурогатного моделювання та адаптивного керування обмеженнями шляхом інтеграції сурогатних нейронних моделей і нейронно-керованих штрафних механізмів у контур гібридної оптимізації покриття, що дозволило знизити обчислювальні витрати та підвищити стабільність пошуку.

5. Удосконалено методи обчислення критерію покриття за рахунок використання багаторівневої адаптивної схеми оцінювання з керованим перемиканням між наближеними та точними геометричними обчисленнями, що

забезпечує баланс між швидкістю та точністю на різних етапах оптимізаційного процесу.

Практичне значення отриманих результатів. Розроблена інформаційна технологія гібридної еволюційно-ройової оптимізації є інструментом для аналізу, проектування та оптимізації просторового покриття в територіально-розподілених системах і може бути використана як основа для створення програмних систем підтримки прийняття рішень у задачах просторового планування.

Запропонована обчислювальна модель неперервної задачі максимального геометричного покриття та розроблена методологія гібридної оптимізації забезпечують можливість ефективного розв'язання задач великої розмірності зі складною геометрією області, наявністю обмежень і високою обчислювальною вартістю цільової функції. Їх використання дозволяє підвищити площу покриття на 12-25% порівняно з класичними метаевристичними підходами (з 65–77% до 89–90% за даними обчислювальних експериментів). При цьому оптимізація кількості ітерацій глобального пошуку у гібридному підході дозволяє скоротити час обчислень на 30–40% без втрати якості покриття. Застосування методу Монте-Карло з просторовою індексацією R-Tree як основного методу обчислення цільової функції в циклі оптимізації забезпечує похибку в межах 1-3% відносно точних обчислень бібліотекою Shapely при асимптотичній складності $O(N \log n)$ замість $O(nk \log(nk))$, що дозволяє ефективно масштабувати обчислення при зростанні кількості покриваючих об'єктів. Для задач із забороненими зонами гібридний підхід з нейронно-керованим адаптивним механізмом штрафних коефіцієнтів забезпечує площу покриття на рівні 85–88% при дотриманні всіх геометричних обмежень, а скорочення кількості ітерацій глобального пошуку з 500 до 200 зменшує час обчислень утричі (з 550 с до 150 с при $n = 80$) без втрати якості покриття.

Розроблена універсальна алгоритмічна схема гібридизації популяційних методів оптимізації може бути покладена в основу програмних реалізацій, адаптованих до різних класів метаевристик і прикладних задач, без необхідності суттєвої модифікації базових алгоритмів. Це забезпечує гнучкість, повторне використання компонентів та спрощує інтеграцію нових методів оптимізації.

Запропоновані нейросурогатні та нейронно-адаптивні механізми дозволяють зменшити обчислювальні витрати за рахунок апроксимації цільової функції та автоматизованого налаштування параметрів штрафних функцій, усуваючи необхідність ручного підбору параметрів для кожної конкретної задачі. Зокрема, нейронно-адаптивне керування штрафним коефіцієнтом та динамічне налаштування гіперпараметрів ройових алгоритмів на основі навчання з підкріпленням дозволяють автоматично формувати стратегію керування оптимізаційним процесом без потреби у попередньому зборі еталонних даних.

Розроблена UML-орієнтована архітектура інформаційної технології створює основу для побудови масштабованих, модульних і відтворюваних програмних систем, придатних до інтеграції з геоінформаційними сервісами, системами моніторингу та іншими прикладними платформами.

Практичне застосування отриманих результатів можливе у таких сферах: оптимізація розміщення сенсорів у бездротових сенсорних мережах; проєктування телекомунікаційної інфраструктури та базових станцій; планування зон обслуговування в логістичних і сервісних системах; управління моніторингом територій із використанням безпілотних літальних апаратів; задачі просторового аналізу в геоінформаційних системах.

Результати роботи можуть бути використані також у навчальному процесі під час викладання дисциплін, пов'язаних із методами оптимізації, штучним інтелектом, інтелектуальними інформаційними системами та програмною інженерією, а також при розробленні курсових і кваліфікаційних робіт.

Особистий внесок здобувача. Основні результати дисертаційного дослідження отримано автором самостійно. У роботах, виконаних у співавторстві, здобувачу належить визначальна роль у проведенні досліджень. Зокрема, автором виконано аналіз сучасного стану досліджень у галузі задач геометричного покриття, розроблено обчислювальні моделі, алгоритмічні методи гібридної еволюційно-ройової оптимізації та універсальну схему гібридизації популяційних алгоритмів. Здобувачем реалізовано програмні модулі інформаційної технології, проведено обчислювальні експерименти, здійснено обробку, узагальнення та інтерпретацію отриманих результатів. Також автором розроблено нейросурогатні та нейронно-адаптивні механізми керування обчислювальним процесом і побудовано UML-орієнтовані моделі архітектури інформаційної системи.

Ідеї дослідження, постановка задач та формування загальної методології здійснювалися спільно з науковим керівником.

Апробація результатів дисертації. Результати дисертації обговорювалися на:

– 4th International Workshop of IT-Professionals on Artificial Intelligence (ProfIT AI 2024), 25–27 September 2024, Cambridge, MA, USA;

– 13th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2025), 4– 6 September 2025, Gliwice, Poland;

– 5th International Workshop of IT-Professionals on Artificial Intelligence (ProfIT AI 2025), 15-17 October 2025, Liverpool, UK.

Публікації. Основні наукові результати дисертаційної роботи викладено у 7 публікаціях, з яких: 4 статті опубліковано у фахових виданнях України категорії Б (Вісник Харківського національного університету імені В. Н. Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»; Системи та технології; Проблеми керування та інформатики;

Системні технології), 3 публікації – у виданнях, що індексуються в наукометричній базі Scopus (CEUR Workshop Proceedings – ProfIT AI 2024 та ProfIT AI 2025; Proceedings of IDAACS 2025).

Структура та обсяг дисертації.

Дисертаційна робота складається зі вступу, чотирьох розділів, загальних висновків, списку використаних літературних джерел та двох додатків. Загальний обсяг дисертації становить 188 сторінок: у тому числі анотації на 11 сторінках, зміст на 4 сторінках, основний текст на 138 сторінках, список використаних джерел із 128 найменувань на 15 сторінках та 2 додатки на 14 сторінках. Робота містить 19 рисунків, 41 таблицю (з них 17 у додатках) та 64 формули.

РОЗДІЛ 1. АНАЛІЗ СУЧАСНОГО СТАНУ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Сучасні підходи до моделювання задач максимального геометричного покриття

Задача максимального покриття (Maximum Coverage Location Problem, далі – MCLP) у широкому класі постановок є предметом вивчення як для теорії операційних досліджень, так і для обчислювальної геометрії. Основні напрями розвитку математичних моделей охоплюють дискретні та частково дискретні постановки, неперервні та плоскі задачі, а також узагальнення на випадок об'єктів довільної форми і заборонених зон. В оглядових роботах [1-7] систематизовано підходи до аналізу розміщення об'єктів на основі моделювання задач покриття: від стратегічного аналізу розміщення об'єктів [3] і теорії задач локалізації [4, 7] до просторового аналізу з використанням геоінформаційних систем (далі – ГІС) [2, 5, 6]. Задачу розміщення певної кількості об'єктів у межах припустимої відстані обслуговування для максимізації попиту в заданому регіоні поставлено і формалізовано у роботі [1]. Для цього розглядалися два дискретні набори точок попиту і можливих розташувань, і задачу MCLP сформульовано у вигляді задачі цілочисельного лінійного програмування. Авторами [8] систематизовано ключові підходи до аналізу розміщення та охарактеризовано їх прикладне значення.

У роботі [9] описано жадібний алгоритм для MCLP, що на кожному кроці обирає розташування з найбільшою кількістю непокритих елементів, і теоретично обґрунтовано, що цей алгоритм є найкращим поліноміальним наближенням. Точне розв'язання планарної задачі є неможливим через NP-важкий характер задачі [9]. В роботі [10] досліджено планарну MCLP з частковим покриттям і прямокутними зонами попиту та обслуговування, де теоретично

обґрунтовано значне скорочення простору пошуку і представлено модифікацію методу гілок і меж.

У роботі [11] метод декомпозиції Бендерса застосовано до дуже великих задач часткового і максимального покриття, що відкрило практичну можливість точного розв'язання екземплярів з тисячами місць розташування. Нечіткі варіанти дискретної MCLP досліджено у [12], де отримано аналітичні результати щодо структури допустимої множини. У роботі [13] розглянуто задачу покриття на мережах з попитом уздовж ребер, де запропоновано алгоритм, що враховує неперервний розподіл попиту між вузлами. Задачу мінімаксного регрету для максимального покриття при невизначених параметрах розміщення проаналізовано у [14]. Задачу розміщення неприємних об'єктів з критерієм мінімаксної відстані і діаграмами Вороного розглянуто у [15]. У роботі [16] досліджено FA для задач зі складними просторами пошуку. Ієрархічні моделі часткового покриття для задач охорони здоров'я розглянуто у [17].

Особливо відзначимо роботу [18], в якій сформульовано модель змішаного цілочисельного нелінійного програмування для розміщення взаємопов'язаних неперервних об'єктів. У суміжному класі задач неперервного просторового розміщення об'єктів нерівної площі [19] розроблено мультістартовий алгоритм відпалу-імітації, що демонструє підходи до глобального пошуку у високовимірних просторах розміщення. В роботі [20] показано, що підходи на основі бінарного покриття, де об'єкт вважається таким, що повністю покриває або не покриває зону покриття, схильні до суттєвих похибок, оскільки часткові покриття ігноруються у фінальному розв'язку.

У роботі [21] запропоновано узагальнення стандартної задачі MCLP, яке дозволяє розміщувати як об'єкти обслуговування, так і точки попиту неперервно на площині – так звана планарна MCLP. В роботі [22] запропоновано геообчислювальну евристику для неперервних задач MCLP, де комбінується оцінка покриття зі стохастичним пошуком у просторі розміщень. Авторами [23]

розглянуто оптимізацію розміщення об'єктів обслуговування у неперервному просторі з інтеграцією геоінформаційних інструментів і показано, що неперервна постановка суттєво змінює структуру допустимої множини порівняно з дискретною. В роботах [24–26] розглянуто різні аспекти оптимізації покриття у неперервному просторі: регіональне моделювання покриття [24], максимізація просторового попиту [25], оптимальне розміщення одного об'єкту у неперервному просторі [26].

Задача визначення місця розташування одного кругового об'єкту у полігональній зоні для максимізації покриття розглянута у [26]. У роботі [27] запропоновано апроксимацію оптимального положення одного кругового об'єкту в незв'язній зоні з обмеженою похибкою; описано ефективний алгоритм точного обчислення площі перетину кола і полігону. Покращені алгоритми зіставлення двох полігональних форм для апроксимації максимального перекриття представлено у [28]. Задачу покриття полігональної зони дисками заданого радіуса з паралельним методом відпалу-імітації описано у [29]. Особливо відзначимо роботу [30], де розглянуто гібридні дискретно-неперервні постановки MCLP, де частина об'єктів розміщується дискретно, а частина – неперервно.

В роботі [31] сформульовано безумовну нелінійну оптимізаційну задачу для неперервного максимального покриття (Continuous Maximum Coverage Location Problem, далі – CMCLP) і запропоновано підхід до чисельного оцінювання градієнта ω -функції на основі різницевої апроксимації, що суттєво скорочує обчислювальні витрати. Задачу визначення оптимального розміщення у неперервному просторі для покриття регіону розглянуто у [26] та теоретично обґрунтовано у [32]. Точні алгоритми для задач планарного максимального покриття еліпсами розроблено у [33]. В роботі [34] запропоновано алгоритм середнього зсуву (mean-shift) для великомасштабних задач планарної MCLP. Задачу неперервного покриття через напівнескінченну

оптимізацію досліджено у [35], де доведено існування та описано структуру оптимального розв'язку.

1.1.1 Моделювання об'єктів довільної форми та заборонених зон

Переважає більшість класичних підходів до MCLP розглядає лише об'єкти простої форми – кола або прямокутники. Для математично строгого опису об'єктів довільної форми та їх взаємного розташування необхідна спеціальна теорія. Рвачьов [36] розробив теорію R-функцій, що дала змогу вирішувати обернену задачу аналітичної геометрії: конструювати функцію, що задає область, із функцій, що задають її частини. Авторами [37] розроблено теорію Phi-функцій для формалізації умов неперетину пари двовимірних геометричних об'єктів, які можуть бути опуклими або неопуклими. Загальні інструменти математичного моделювання задач пакування об'єктів довільної форми систематизовано у [38].

Поняття конфігураційного простору геометричних об'єктів введено Стояном та Яковлевим [39], де конструюється простір, узагальненими змінними якого є параметри розміщення та метричні параметри об'єктів. Класи просторових конфігурацій та їх формалізацію виокремлено у [40], де показано, що задачі пакування, розміщення і покриття є спеціальними випадками загальної задачі оптимізації просторових конфігурацій. У роботі [41] запропоновано спеціальний клас omega-функцій: omega-функція визначає залежність площі параметризованого геометричного об'єкта від узагальнених змінних. Концепцію моделювання задач пакування і покриття з використанням сучасного програмного забезпечення обчислювальної геометрії систематизовано у [42]. Суміжним математичним апаратом є теорія оптимального розбиття множин [43, 44], де задача розміщення генераторів формулюється як нескінченновимірна оптимізаційна задача з розбиттям досліджуваної множини на підмножини. Задачу побудови узагальненої діаграми Вороного з оптимальним розміщенням генераторних точок розглянуто у [45].

Задачі покриття спеціальними класами об'єктів становлять окремий напрям досліджень. Стоян та Пацук [46] розглянули задачу покриття полігональної множини ідентичними колами мінімального радіуса і розробили модифікацію методу здійснення напрямків Зоутендейка. В роботі [47] розв'язано задачу покриття полігональної області прямокутниками з використанням Phi-функцій. Панкратов та ін. [48] запропонували алгоритм оптимального покриття сфероїдів сферами. Задачу орієнтованого покриття смугами змінної довжини досліджено у [49], де розроблено алгоритм гілок і цін для задачі NP-важкого класу.

В роботі [50] задачу максимального покриття узагальнено на випадок довільних форм зон обслуговування. Авторами [50] показано, що задача не зводиться до простих стандартних класів навіть для опуклих форм, і обґрунтовано необхідність формалізації через omega-функцію. У роботі [51] задачу покриття з довільними об'єктами сформульовано і розв'язано з використанням бібліотеки обчислювальної геометрії. Авторами [52] показано застосовність такого підходу для задачі вибору ділового місця з об'єктами складної форми. Задачу покриття з заборонними зонами можна формалізувати через зовнішню штрафну функцію, що суттєво ускладнює ландшафт цільової функції.

Практичне обчислення omega-функції для об'єктів довільної форми стало можливим завдяки бібліотеці Shapely [53], що реалізує стандарт JTS Topology Suite для двовимірних геометрій. В роботах [31, 50, 51, 52] реалізовано і порівняно різні підходи до обчислення площі покриття, в тому числі стохастичні методи Монте-Карло та точні булеві операції над геометріями. Для подолання обчислювальної складності доцільним є триступеневий підхід: стохастична апроксимація Монте-Карло, комбінаторна оцінка IE2 та точні Shapely-операції.

1.2 Сучасні методи еволюційно-ройової та гібридної оптимізації

Оскільки задача неперервного максимального покриття є NP-важкою і характеризується мультиекстремальним ландшафтом цільової функції, для її розв'язання застосовують широкий спектр методів: від детермінованих методів локальної оптимізації до ройових метаевристик, гібридних підходів і нейромережових технологій.

Детерміновані методи локальної оптимізації відіграють ключову роль як у самостійному застосуванні (мультистарт), так і як локальний компонент гібридних алгоритмів. Квазіньютонівський метод BFGS [54] апроксимує матрицю Гессіана та оновлює її з використанням різниць градієнтів, забезпечуючи надлінійну швидкість збіжності в околі локального мінімуму. В роботі [31] запропоновано підхід до застосування методу BFGS з різницевою апроксимацією першого порядку для знаходження локального розв'язку задачі SMCLP, де показано, що вибір стратегії апроксимації градієнта ω -функції суттєво впливає на обчислювальну ефективність.

Метод L-BFGS, запропонований Liu та Nocedal [55], є модифікацією BFGS з обмеженою пам'яттю: замість зберігання повної матриці $n \times n$ метод зберігає лише m останніх пар оновлень ($m = 5 - 20$), що дозволяє масштабуватися на задачі з тисячами змінних при лінійних вимогах до пам'яті. Авторами [56] запропоновано адаптивну стратегію вибору розміру пам'яті для L-BFGS, що підвищує продуктивність при нестационарних ландшафтах. Для задачі SMCLP з $n > 50$ об'єктів L-BFGS є практично кращим за повний BFGS.

На відміну від методів лінійного пошуку, методи довірчої області [57] обмежують крок у кожній ітерації радіусом довірчої області, в межах якої квадратична модель цільової функції є надійною. В роботі [57] систематизовано теорію і практику таких методів для умовних і безумовних задач нелінійної оптимізації; показано, що глобальна збіжність гарантується навіть для сильно

невипуклих функцій. Для задачі SMCLP методи довірчої області є привабливими при різких змінах градієнта ω поблизу меж заборонних зон.

Метод Нельдера-Міда (метод деформованого симплексу), аналіз збіжності якого проведено у [58], не використовує похідних цільової функції. Алгоритм ітеративно трансформує симплекс з $n+1$ вершин (відображення, розтягнення, стиснення, скорочення) у напрямку мінімуму. Авторами [59] запропоновано реалізацію методу Нельдера-Міда з адаптивними параметрами, що залежать від розмірності задачі, і показано, що новий алгоритм перевершує стандартний варіант у вищих вимірах. Для задачі SMCLP метод є привабливим при розривних похідних ω , але рекомендується лише для задач невеликої розмірності.

Порівняльний аналіз методів локальної оптимізації в задачі SMCLP показує, що вибір методу суттєво впливає на якість фінального рішення. Особливо важливою є практична вартість обчислення ω -функції: при точних Shapely-обчисленнях одна оцінка ω займає десятки-сотні мілісекунд, що стимулює мінімізацію кількості звернень до ω на кожній ітерації.

1.2.1 Метаевристичні методи

З огляду на мультиекстремальний характер задачі SMCLP застосування метаевристик (загальних стратегій, що здатні ефективно досліджувати великий простір пошуку) є природним підходом [60, 61]. Ключова перевага ройових методів полягає в тому, що вони оцінюють якість конфігурацій безпосередньо через значення ω -функції, не покладаючись на аналітичний вид градієнта. Метод оптимізації роєм частинок (PSO), запропонований Kennedy та Eberhart [62], є одним з найбільш широко застосовуваних ройових алгоритмів. В роботі [63] наведено огляд теорії та практики PSO; систематичний огляд кумулятивних розвитку PSO з 2018 р. представлено у [64]. Комплексний огляд варіантів і застосувань PSO наведено у [65]. В роботі [66] систематизовано підходи до вибору ваги інерції, фактора стиснення та когнітивно-соціальних коефіцієнтів.

Авторами [67] проведено аналіз збіжності PSO з різними факторами стиснення і теоретично обґрунтовано умови стійкості алгоритму.

Серед варіантів PSO особливе значення мають адаптивні модифікації. Адаптивна PSO (APSO) автоматично регулює параметри на основі оцінки еволюційного стану рою (дослідження, використання, збіжність, вихід з локального оптимуму) [66]. В роботі [68] запропоновано PSO з адаптивним управлінням через затримані швидкості (PSO-AWDV), що підвищує стійкість збіжності при різних типах ландшафтів. Кооперативна PSO (CPSO) використовує кілька підройів для різних груп координат і є ефективною для задач великої розмірності [66]. Адаптивний динамічний PSO з локальним пошуком [69] забезпечує автоматичне перемикання між режимами глобального і локального пошуку. В роботі [70] представлено Standard PSO 2011 (SPSO 2011) – стандартизований варіант PSO з уніфікованими параметрами та сферичною топологією інформування, запропонований спільнотою дослідників PSO і верифікований на наборі тестових функцій CEC-2013.

Серед інших ройових алгоритмів для задачі CMCLP важливу роль відіграють: алгоритм штучної бджолиної колонії (ABC) [71], де механізм розвідника природно запобігає передчасній збіжності; покращений варіант ABC [72] з адаптивним механізмом визначення меж пошуку; алгоритм пошуку рибачої зграї (FSS) [73] з природним саморегулюванням різноманіття; покращений FSS [74] з адаптивними параметрами кроку; алгоритм світлячків (FA) [75], що породжує природну кластеризацію агентів навколо перспективних регіонів; огляд варіантів FA [76]. В роботі [16] досліджено FA для задач зі складними просторами пошуку і показано ефективну диверсифікацію порівняно з базовими алгоритмами. Диференціальна еволюція [77] є конкурентом PSO на гладких мультиекстремальних ландшафтах. Алгоритм мурашиної колонії [78] знаходить застосування у задачах розміщення і маршрутизації, хоча для неперервних CMCLP потребує дискретизації простору параметрів.

Математичний аналіз природонатхненних алгоритмів з урахуванням їх збіжності і порівняльних властивостей представлено у [79]. Оглядова стаття [80] описує метод сірого вовка (GWO) як приклад новітніх ройових алгоритмів, що демонструють конкурентоспроможність порівняно з PSO на ряді задач. Порівняльний аналіз ройових метаевристик стосовно задачі CMCLP підтверджує теорему No Free Lunch [60]: жоден алгоритм не є домінуючим для всіх конфігурацій задач.

1.2.2 Меметичні методи

Меметичні алгоритми, що поєднують глобальний пошук еволюційних або ройових методів із детермінованою локальною оптимізацією, систематизовано у монографії [81]. В роботі [82] наведено огляд меметичних алгоритмів і меметичних обчислень з акцентом на задачах зі складними ландшафтами цільової функції. Авторами [83] систематизовано меметичні алгоритми для задач неперервної оптимізації на основі локальних ланцюжків пошуку і показано, що вибір частоти і типу локального оптимізатора є ключовим проектним рішенням. Нещодавній огляд тенденцій у меметичних алгоритмах [84] показує, що основним напрямом розвитку є адаптивне визначення: коли, як часто і для яких рішень популяції запускати локальний оптимізатор.

Серед гібридних підходів особливо відзначимо роботу [85], де запропоновано ефективний гібридний PSO з локальним пошуком і показано, що точковий запуск локального пошуку суттєво прискорює збіжність та підвищує якість фінального розв'язку. Адаптивний динамічний PSO з локальним пошуком [69] забезпечує автоматичне перемикання між режимами. Поєднання ройових метаевристик (PSO, FSS, FA, ABC) з методом BFGS [54] є перспективним напрямком для задач CMCLP. Такий гібридний підхід забезпечує вищу якість покриття та швидшу збіжність порівняно як з чистими ройовими алгоритмами,

так і з мультістартовою локальною оптимізацією, особливо при числі об'єктів $n > 20$.

Інтеграція нейромережових технологій у задачі CMCLP розвивається за двома основними напрямками: сурогатне апроксимаційне моделювання ω -функції і нейронно-керовані адаптивні механізми. Загальні засади проектування систем за допомогою сурогатного моделювання систематизовано у [86]. В роботі [87] систематично розглянуто дата-орієнтовану еволюційну оптимізацію і показано, що навіть неточні сурогатні моделі суттєво прискорюють збіжність шляхом передфільтрації кандидатних рішень. Фізико-інформовані нейронні мережі (PINNs) [88], що поєднують аналітичні геометричні закономірності з даними, є перспективним підходом для задач покриття. Загальні засади глибокого навчання і нейромережового моделювання викладено у [89].

Архітектури типу Deep Sets [90], що забезпечують інваріантність до перестановок вхідних елементів, природно підходять для апроксимації ω -функції CMCLP, оскільки значення ω не залежить від порядку нумерації покриваючих об'єктів. Перспективним є нейронно-керований гібридний підхід до задачі CMCLP, де нейронна мережа виконує подвійну роль: апроксимаційна модель для обчислення ω -функції та адаптивний механізм управління штрафними коефіцієнтами.

Важливою складовою сучасних інформаційно-аналітичних технологій є системно-інженерний аспект. Питання архітектурної формалізації відповідної інформаційної технології, зокрема розробка UML-орієнтованої архітектури з підтримкою змінних алгоритмічних стратегій, залишається актуальним напрямком досліджень.

1.3 Прикладні задачі максимального геометричного покриття

Задача максимального покриття знаходить широке застосування у різноманітних галузях – від систем аварійного реагування і сенсорних мереж до

комерційного розміщення та систем спостереження. Узагальнення задачі на неперервний простір з об'єктами довільної форми [50] відкриває нові можливості для задач, що раніше не допускали адекватної математичної формалізації.

1.3.1 Розміщення аварійних служб

Традиційним і найбільш дослідженим класом застосувань MCLP є оптимальне розміщення аварійних служб – пожежних станцій, поліцейських підрозділів і центрів медичної допомоги. В роботі [91] досліджено задачу розміщення пожежних станцій для максимального покриття міської забудови на основі постановки Church та ReVelle [1]; показано, що навіть незначне збільшення числа об'єктів суттєво підвищує ефективність покриття. Авторами [92] розроблено спеціалізований програмний інструмент для розв'язання MCLP стосовно задачі розміщення поліцейських підрозділів і наведено результати застосування на реальних даних.

У роботі [93] задачу MCLP застосовано до розміщення центрів екстреного реагування з урахуванням ризику паводку. Авторами [93] інтегровано методи багатосарового персептронну для прогнозування зон ризику і задачу максимального покриття для розміщення рятувальних центрів. Ієрархічну модель часткового покриття для задач охорони здоров'я, що враховує відмінності між первинним і спеціалізованим рівнями обслуговування, розглянуто у [17]. Розміщення мобільних медичних підрозділів та центрів вакцинації в кризових сценаріях досліджено у [50] та [94], де показано, що традиційні моделі з круговими зонами обслуговування не відображають реальних умов роботи виїзних медичних бригад.

Оптимізація покриття є центральним завданням у бездротових сенсорних мережах. Огляд бездротових сенсорних мереж і виклики оптимізації покриття в них систематизовано у [95]. Авторами [51] задачу моніторингу території формалізовано як геометричну задачу максимального покриття та реалізовано з

використанням бібліотеки Shapely [53]. Модельні задачі надійності бездротових сенсорних мереж для моніторингу лісових пожеж розглянуто у [96]; гібридні сенсорні мережі для екологічного і аварійного моніторингу – у роботі [97]. Задачу оптимального позиціонування LoRa-антен для відстеження великої рогатої худоби в гірських умовах Австрії досліджено у [98], де показано практичну ефективність підходу на основі покриття з реальною радіочастотною моделлю.

Задачі покриття для мобільних робототехнічних систем формують окремий важливий клас застосувань. Авторами [99] розроблено децентралізовані алгоритми гарантованого покриття для мобільних сенсорних мереж, засновані на діаграмах Вороного; доведено, що запропонований алгоритм збігається до конфігурацій Centroidal Voronoi. Децентралізоване адаптивне управління покриттям для мережевих роботів, що не потребує центральної координації, розглянуто у [100]. Огляд задач покриття для робототехнічних систем і методів їх вирішення представлено у [101]. Адаптивне широкоплощадне дослідження і картографування за допомогою мережі роботів з урахуванням обмежень на комунікацію – у роботі [102].

Задача розміщення депо для безпілотних літальних апаратів (далі – БПЛА) як різновид MCLP досліджена у [103], де запропоновано надійну багатоперіодну постановку, що враховує стохастичну невизначеність попиту. Задачу розміщення станцій зарядки електромобілів з урахуванням ГІС-критеріїв – у роботі [104].

1.3.2 Комерційне розміщення та ділова аналітика

Задачі вибору ділового місця є одним із найстаріших класів застосувань MCLP. У монографії [5] систематизовано підходи до вибору місця розташування бізнесу з використанням ГІС. Авторами [52] показано застосовність підходу з довільними формами зон обслуговування до задачі вибору ділового місця і наведено обчислювальні експерименти на реальних прикладах бізнес-середовища. В роботі [105] досліджено задачу оптимізації розміщення магазинів

у підземному торговому кварталі на основі даних про рух людей; запропоновано підхід, що поєднує аналіз потоків відвідувачів і задачу максимального покриття.

Прогнозування ділової ефективності в сервісах соціальних мереж на основі параметрів розміщення досліджено у [106]. У роботі [50] виокремлено шість практичних категорій задач, що мотивують постановку SMCLP з довільними формами: використання промислових відходів нерегулярної форми у текстильному та шкіряному виробництві; теплова та акустична ізоляція залишками матеріалів нерегулярної форми; виробництво сонячних панелей із вторинних матеріалів нестандартних форм; виробництво композитних матеріалів; тимчасові укриття та розподіл ресурсів після природних катастроф; розміщення мобільних медичних пунктів із складною конфігурацією зон попиту та обслуговування. Авторами [50] наведено чисельні експерименти, що підтверджують практичну ефективність запропонованих підходів.

1.3.3 Гіс-системи

Розвиток задач покриття нерозривно пов'язаний із геоінформаційними системами. Авторами [5, 6] показано, що постановка і розв'язання задачі MCLP в програмних пакетах ArcGIS та TransCAD є практично доступними; зокрема, функція "maximize coverage" у Network Analyst дозволяє безпосередньо формулювати задачу максимального покриття. В роботі [107] підкреслено роль ГІС як аналітичного інструменту для специфікації і застосування MCLP у широкому колі задач: від управління ресурсами до міського планування.

У роботі [108] застосовано моделі покриття у ГІС-середовищі для вибору природних заповідників, де запропоновано методологію вибору зон збереження природних ресурсів на основі критеріїв максимального покриття видів. Авторами [109] досліджено оптимізацію розміщення станцій велопрокату за ГІС-даними, де задачу покриття використано для максимізації числа охоплених потенційних користувачів. В роботі [6] систематизовано зв'язки між ГІС і задачами аналізу

розміщення та показано, що поява нових просторових даних відкриває нові прикладні можливості для MCLP. Задачу неперервного покриття на мережах з покращеними MIP-формулюваннями розглянуто у [110], де показана ефективність інтеграції із мережевими ГІС-шарами.

Особливо слід відзначити роботу [42], де систематизовано концепцію моделювання задач пакування і покриття з використанням сучасного програмного забезпечення обчислювальної геометрії, зокрема з інтеграцією у ГІС-середовища через відкриті формати просторових даних. Авторами [42] обґрунтовано, що перехід від дискретних ГІС-інструментів до неперервних методів оптимізації відкриває принципово нові можливості для задач покриття в умовах складних просторових обмежень.

Проведений аналіз літературних джерел засвідчує, що задача максимального покриття пройшла шлях від дискретних цілочисельних постановок [1, 9] до неперервних і узагальнених формулювань з об'єктами довільної форми [50], від ізольованих евристик до гібридних ройово-меметичних методів і нейронно-керованих підходів. Практичне застосування охоплює аварійне реагування [91–94], сенсорні та робототехнічні мережі [95–102], комерційне розміщення [105, 106] та ГІС-системи [107–110].

Разом з тим аналіз показує, що ряд аспектів залишається недостатньо дослідженим. По-перше, більшість існуючих методів оптимізації орієнтовані на кола або прямокутники і не підтримують зони обслуговування довільної форми з заборонними зонами у єдиній обчислювальній схемі.

По-друге, незважаючи на наявність окремих гібридних підходів [85], відсутнє систематичне порівняльне дослідження ройових метаевристик у поєднанні з методами локальної оптимізації стосовно задачі CMCLP на об'єктах довільної форми.

По-третє, питання інформаційно-технологічної реалізації – уніфікованої архітектури програмної системи, що підтримує множину алгоритмів і точні геометричні обчислення – практично не розглядалося у наявних дослідженнях.

З урахуванням результатів проведеного огляду, який підтвердив актуальність проблеми та окреслив незаповнені ніші, у наступному розділі формулюється постановка задачі дисертаційного дослідження.

1.4 Постановка задачі та математична модель

Розглянемо задачу оптимального розміщення скінченної сукупності геометричних об'єктів із метою максимізації покриття заданої області на евклідовій площині \mathbb{R}^2 .

1.4.1 Базові об'єкти та параметризація

Нехай $D \subset \mathbb{R}^2$ – компактна область покриття з відомою площею $S(D)$, яка є фіксованою відносно глобальної декартової системи координат Oxy . Задано також сімейство n покриваючих об'єктів (зон обслуговування) $\Omega_1, \Omega_2, \dots, \Omega_n$, кожен з яких є компактною підмножиною площини фіксованої форми та метричних характеристик. Множину $I_n = \{1, 2, \dots, n\}$ називатимемо індексною множиною.

Об'єкти Ω_i можуть мати довільну геометричну форму: опуклі та неопуклі багатокутники, кола, еліпси або їх комбінації. Площі об'єктів $S(\Omega_i)$, $i \in I_n$, є сталими величинами, що визначаються формою та розмірами відповідних фігур.

Для кожного об'єкта Ω_i зафіксуємо характеристичну внутрішню точку o_i , яку називатимемо його полюсом. Вибір полюса зазвичай зумовлений геометричними властивостями об'єкта: для симетричних фігур це центр симетрії, для несиметричних – центроїд, чебишовський центр або інша представницька точка, що однозначно визначає положення об'єкта у просторі.

Положення та орієнтація кожного об'єкта Ω_i у глобальній системі координат Oxy визначаються вектором параметрів розміщення (рисунок 1.1):

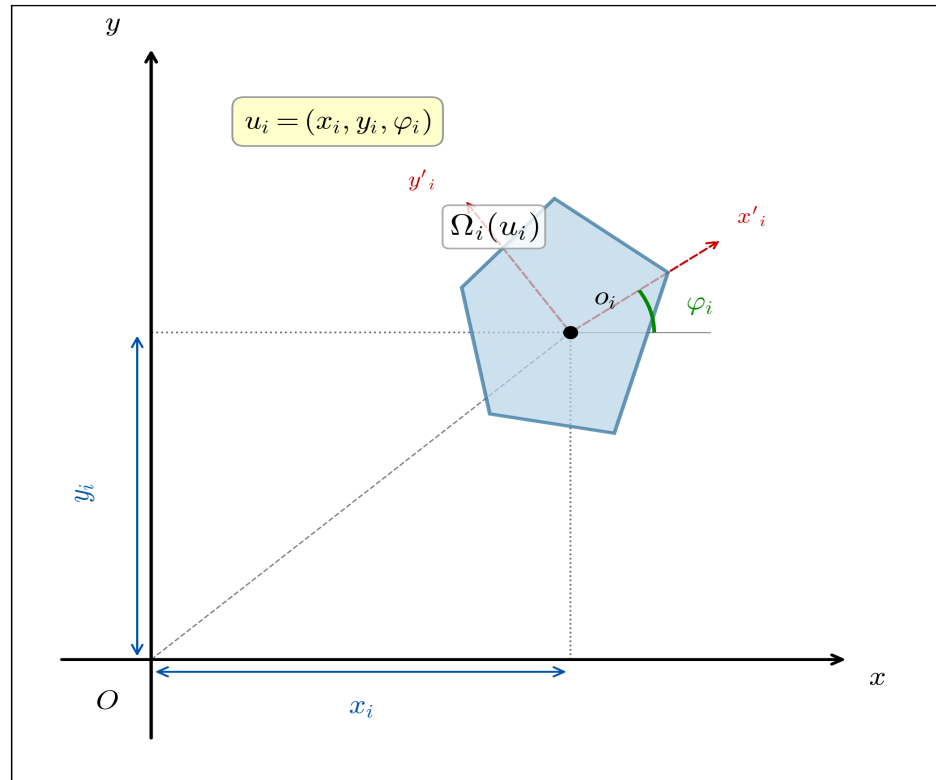


Рисунок 1.1 – Параметри розміщення об'єкта $\Omega_i(u_i)$

Нехай параметрами розміщення кожного об'єкту описуються як:

$$u_i = (x_i, y_i, \varphi_i), \quad i \in I_n, \quad (1.1)$$

де (x_i, y_i) – координати полюса o_i у системі Oxy ;

φ_i – кут повороту об'єкта відносно цієї системи.

Об'єкт Ω_i з параметрами розміщення u_i позначатимемо як $\Omega_i(u_i)$ і називатимемо параметризованим об'єктом. Геометрично, параметризований об'єкт отримується із вихідного шляхом повороту на кут φ_i навколо полюса та паралельного перенесення полюса у точку (x_i, y_i) .

Формально:

$$\Omega_i(u_i) = \left\{ R(\varphi_i) \cdot p + \begin{pmatrix} x_i \\ y_i \end{pmatrix} \mid p \in \Omega_i \right\}, \quad (1.2)$$

де $R(\varphi_i)$ – матриця повороту.

Матриця повороту визначається як:

$$R(\varphi_i) = \begin{pmatrix} \cos\varphi_i & -\sin\varphi_i \\ \sin\varphi_i & \cos\varphi_i \end{pmatrix} \quad (1.3)$$

Позначимо вектор усіх параметрів розміщення як:

$$U = (u_1, u_2, \dots, u_n) \in \mathbb{R}^{3n} \quad (1.4)$$

Об'єднання параметризованих об'єктів утворює складний геометричний об'єкт, який називатимемо конфігурацією покриття:

$$\mathcal{C}(U) = \bigcup_{i=1}^n \Omega_i(u_i) \quad (1.5)$$

Частина області покриття D , що покривається конфігурацією $\mathcal{C}(U)$, визначається перетином:

$$D_{\text{cov}}(U) = D \cap \mathcal{C}(U) = D \cap \left(\bigcup_{i=1}^n \Omega_i(u_i) \right) \quad (1.6)$$

Введемо характеристичну функцію покриття:

$$\chi(x, U) = \begin{cases} 1, & \text{якщо } x \in D_{\text{cov}}(U), \\ 0, & \text{інакше,} \end{cases} \quad (1.7)$$

де $x \in \mathbb{R}^2$.

Тоді площа покритої частини області покриття (рисунок 1.2) визначається як функція параметрів розміщення:

$$F(U) = \iint_D \chi(x, U) dA = S(D \cap C(U)), \quad (1.8)$$

де $S(\cdot)$ позначає площу відповідної множини.

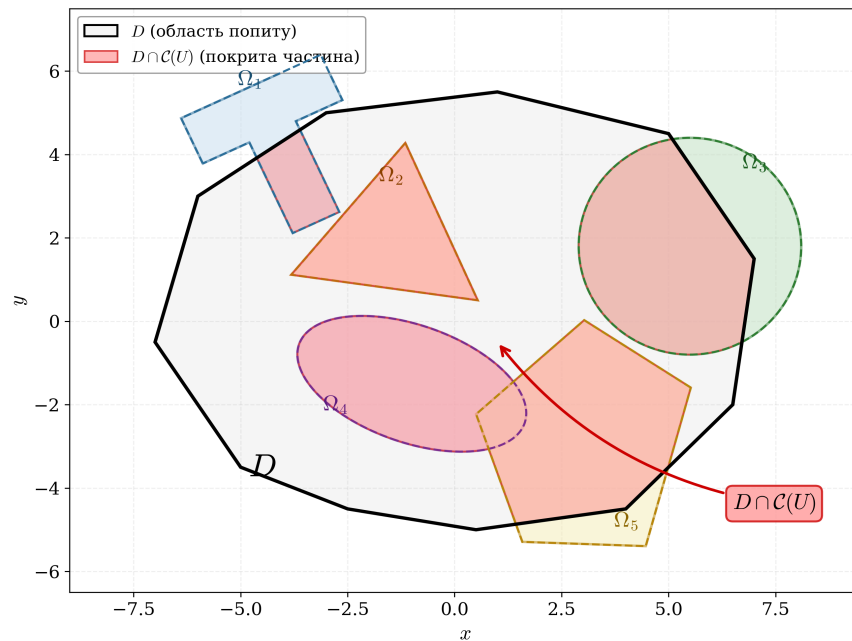


Рисунок 1.2 – Покрита частина області покриття D п'ятьма зонами обслуговування $\Omega_1, \dots, \Omega_5$

Функція $F(U)$ задає залежність площі покриття від параметрів розміщення U і є основним критерієм оптимальності задачі.

1.4.2 Обмеження на параметри розміщення

На відміну від необмежених постановок, у практичних задачах на параметри розміщення покриваючих об'єктів накладаються геометричні обмеження двох типів. Нехай задано m заборонених зон $Z_1, Z_2, \dots, Z_m \subset \mathbb{R}^2$, у яких розміщення полюсів покриваючих об'єктів є неприпустимим. Кожна зона заборони описується системою нерівностей, і обмеження має вигляд:

$$(x_i, y_i) \notin Z_j, \quad i \in I_n, \quad j \in \{1, 2, \dots, m\}, \quad (1.9)$$

що формалізується як:

$$g_j^{(k)}(x_i, y_i) \leq 0, \quad k = 1, \dots, K_j, \quad j = 1, \dots, m \quad (1.10)$$

Заборонені зони виникають внаслідок фізичних перешкод, регуляторних вимог, природних бар'єрів або інших просторових обмежень. Додатково може вимагатися, щоб полюси об'єктів знаходилися всередині або на межі області покриття:

$$(x_i, y_i) \in D, \quad i \in I_n, \quad (1.11)$$

що задається відповідною системою нерівностей:

$$h^{(l)}(x_i, y_i) \leq 0, \quad l = 1, \dots, L. \quad (1.12)$$

Ця умова гарантує, що покриваючі об'єкти розміщуються в безпосередній близькості до області покриття, що є природним для більшості прикладних задач.

Сукупність обмежень (1.9)-(1.12) визначає допустиму область параметрів розміщення $\mathcal{D} \subset \mathbb{R}^{3n}$.

1.4.3 Оптимізаційна постановка

Задача неперервного максимального покриття з обмеженнями формулюється як задача неперервної оптимізації, а саме нелінійного програмування:

$$F(U) = S(D \cap \mathcal{C}(U)) \rightarrow \max_U, \quad (1.13)$$

де $U \in \mathcal{D}$;

\mathcal{D} – допустима множина, що визначається обмеженнями (1.9)-(1.12).

Задача (1.13) є суттєво складною з обчислювальної точки зору з кількох причин. По-перше, розмірність простору параметрів становить $3n$, що за великих n призводить до високорозмірної оптимізації. По-друге, цільова функція $F(U)$ є мультимодальною, негладкою і, як правило, не має зручного аналітичного вигляду. По-третє, обчислення площі $S(D \cap \mathcal{C}(U))$ для об'єктів довільної форми пов'язане зі значними обчислювальними витратами, які визначають загальну ефективність оптимізаційного алгоритму. Нарешті, наявність заборонених зон та умов приналежності створює невиконувальну допустиму множину зі складною топологією. Для чисельного розв'язання задачі (1.13) обмеження вводяться до цільової функції через механізм зовнішньої штрафної функції:

$$P(U) = \sum_{i=1}^n \sum_{j=1}^m \Phi_Z(x_i, y_i, Z_j) + \sum_{i=1}^n \Phi_D(x_i, y_i, D), \quad (1.14)$$

де $\Phi_Z(\cdot)$ та $\Phi_D(\cdot)$ – штрафи за порушення обмежень заборонених зон та умов приналежності відповідно.

Тоді задача безумовної оптимізації набуває вигляду:

$$\tilde{F}(U, \lambda) = F(U) - \lambda \cdot P(U) \rightarrow \max_U, \quad (1.15)$$

де $\lambda > 0$ – штрафний коефіцієнт, що поступово збільшується в процесі оптимізації для забезпечення допустимості розв’язку.

Задача (1.15) зберігає всі обчислювальні складності вихідної задачі (1.13) – високу розмірність, мультимодальність цільової функції та високу вартість обчислення площі покриття. Водночас, вона потребує ефективного балансування між оптимізацією покриття та задоволенням геометричних обмежень. Зазначені властивості природно обґрунтовують застосування гібридних (меметичних) підходів, що поєднують глобальний стохастичний пошук популяційних метаевристик із локальною детермінованою оптимізацією.

Висновки до розділу 1

Проведений аналіз літературних джерел засвідчує, що задача максимального покриття пройшла шлях від дискретних цілочисельних постановок [1, 9] до неперервних і узагальнених формулювань з об’єктами довільної форми [50], від ізольованих евристик до гібридних ройово-меметичних методів і нейронно-керованих підходів. Широкий спектр практичних застосувань охоплює задачі аварійного реагування [91–94], розгортання сенсорних та робототехнічних мереж [95–102], комерційного розміщення об’єктів [105, 106], а також задачі в ГІС-системах [107–110]. Наявність розвиненого математичного апарату (теорії R-функцій [36], Phi-функцій [37] і omega-функцій [41]) у поєднанні з сучасними бібліотеками обчислювальної геометрії [53] забезпечує теоретичний фундамент для роботи з об’єктами довільної форми та заборонними зонами.

Разом з тим аналіз виявив три ключові прогалини, що визначають напрям дисертаційного дослідження. По-перше, переважна більшість існуючих методів оптимізації зорієнтована на об'єкти простої форми, такі як кола або прямокутники, і не забезпечує комплексного розв'язання задачі CMCLP із зонами обслуговування довільної форми та заборонними зонами у рамках єдиної обчислювальної схеми. По-друге, незважаючи на наявність окремих гібридних підходів [85], у літературі відсутнє систематичне порівняльне дослідження ройових метаевристик (PSO, FSS, FA, ABC) у поєднанні з методами локальної оптимізації BFGS та L-BFGS стосовно задачі з довільними геометричними обмеженнями. По-третє, питання інформаційно-технологічної реалізації, тобто уніфікованої архітектури програмної системи, що підтримує змінні алгоритмічні стратегії та точні геометричні обчислення через бібліотеку Shapely, залишається практично нерозглянутим у наявних дослідженнях.

Таким чином, проведений огляд підтверджує необхідність розробки нових інформаційних технологій та методів моделювання, які дозволять підвищити ефективність вирішення задач максимального покриття. Розроблена інформаційна технологія демонструє приклад інтеграції метаевристичних алгоритмів для глобальної оптимізації разом з алгоритмами локальної оптимізації. Такий підхід має перспективи для подальшого застосування в багатьох існуючих системах, при розробці нових стратегій та підходів, або оптимізації існуючих.

РОЗДІЛ 2. МЕТОДИ ОБЧИСЛЕННЯ ЦІЛЬОВОЇ ФУНКЦІЇ ТА РОЙОВІ МЕТАЕВРИСТИКИ В ЗАДАЧАХ МАКСИМАЛЬНОГО ПОКРИТТЯ

2.1 Моделі та методи обчислення цільової функції в задачах максимального покриття

Оцінка значення площі конфігурації покриття при заданих параметрах розміщення покриваючих об'єктів є найбільш трудомісткою операцією в задачі максимального покриття геометричних об'єктів довільної форми. Від швидкості та точності її обчислення суттєво залежать як загальний час роботи оптимізаційного алгоритму, так і якість отриманих рішень. У зв'язку з цим вибір ефективного методу обчислення площі покриття є критично важливим етапом при розробці алгоритмів розв'язання задачі максимального покриття.

У цьому розділі використовуються позначення, введені у підрозділі 1.4: область покриття D , покриваючі об'єкти $\Omega_1(u_1), \dots, \Omega_n(u_n)$ з параметрами розміщення $u_i = (x_i, y_i, \varphi_i)$, вектор усіх параметрів $U \in \mathbb{R}^{3n}$, та цільова функція $F(U) = S(D \cap C(U))$, де $S(\cdot)$ позначає площу відповідної множини (формули (1.1)-(1.8)).

Обчислення площі об'єднання довільних геометричних фігур є нетривіальною задачею обчислювальної геометрії. Складність виникає через необхідність коректної обробки перетинів між об'єктами, які можуть мати довільну форму та орієнтацію. Для криволінійних об'єктів, таких як еліпси, аналітичне обчислення площі перетину є особливо складним, оскільки перетин двох еліпсів може утворювати фігуру, обмежену кривими четвертого порядку.

У процесі оптимізації цільова функція обчислюється багаторазово: для популяційних алгоритмів з розміром популяції P та кількістю ітерацій T загальна кількість обчислень становить $P \times T$, що може сягати десятків тисяч викликів.

Таким чином, навіть незначне зменшення часу одного обчислення суттєво впливає на загальний час роботи алгоритму. У зв'язку з цим було розглянуто низку підходів до обчислення площі покриття, що забезпечують різний компроміс між точністю обчислень та обчислювальною складністю [31, 50].

2.1.1 Метод мінімальних описаних прямокутників

Метод мінімальних описаних прямокутників є найпростішим підходом до оцінки площі покриття. Основна ідея полягає в заміні кожного геометричного об'єкта його мінімальним описаним прямокутником, орієнтованим по осях координат. Такий прямокутник визначається чотирма значеннями: мінімальними та максимальними координатами об'єкта по осях x та y .

Алгоритм методу складається з наступних кроків. Спочатку для кожного об'єкта $\Omega_i (u_i)$ обчислюється мінімальний описаний прямокутник B_i , що повністю містить цей об'єкт. Для еліпса з центром (x_c, y_c) , півосями a, b та кутом повороту φ координати прямокутника обчислюються за формулами:

$$x_{min} = x_c - \sqrt{(a^2 \cos^2 \varphi + b^2 \sin^2 \varphi)}, \quad (2.1)$$

$$x_{max} = x_c + \sqrt{(a^2 \cos^2 \varphi + b^2 \sin^2 \varphi)}, \quad (2.2)$$

$$y_{min} = y_c - \sqrt{(a^2 \sin^2 \varphi + b^2 \cos^2 \varphi)}, \quad (2.3)$$

$$y_{max} = y_c + \sqrt{(a^2 \sin^2 \varphi + b^2 \cos^2 \varphi)} \quad (2.4)$$

Аналогічно, область покриття D замінюється своїм мінімальним описаним прямокутником B_D .

Площа покриття оцінюється як площа об'єднання прямокутників, перетнутого з прямокутником області:

$$F_{BB}(U) \approx S(B_D \cap (B_1 \cup B_2 \cup \dots \cup B_n)) \quad (2.5)$$

Обчислювальна складність методу становить $O(n)$, де n – кількість покриваючих об'єктів, оскільки операції з прямокутниками, орієнтованими по осях, виконуються за константний час. Об'єднання прямокутників може бути ефективно обчислене за допомогою алгоритму замітання площини зі складністю $O(n \log(n))$, проте для грубої оцінки достатньо простішого підходу.

Основною перевагою методу є надзвичайно низька обчислювальна вартість та простота реалізації, що робить його придатним для швидкого відсіювання явно неоптимальних конфігурацій на початкових етапах оптимізації. Водночас метод має суттєві недоліки: він створює значну похибку при роботі з об'єктами довільної форми, особливо з еліпсами та повернутими фігурами, оскільки прямокутник завжди має більшу площу за вписану криволінійну фігуру. Для еліпса співвідношення площі описаного прямокутника до площі еліпса становить $4/(\pi) \approx 1.27$, тобто похибка складає щонайменше 27% для одного об'єкта, а для конфігурації з багатьох об'єктів похибка може сягати 40% і більше.

Таким чином, метод bounding boxes доцільно застосовувати лише як грубу верхню оцінку площі покриття або для найперших кроків глобального пошуку, коли необхідно швидко відкинути явно неперспективні конфігурації.

Для точної оцінки якості рішення цей метод є неприйнятним, що обумовлює необхідність застосування більш точних, хоча й обчислювально витратніших підходів.

2.1.2 Метод Монте-Карло

Метод Монте-Карло є стохастичним підходом до оцінки площі покриття, що базується на статистичному принципі оцінки інтегралів шляхом випадкової вибірки. Цей метод належить до класу методів статистичного моделювання та широко застосовується в обчислювальній геометрії для оцінки площ та об'ємів складних фігур. У контексті задачі CMCLP метод Монте-Карло застосовано для обчислення цільової функції у роботах [31, 51]. Теоретичним підґрунтям методу є закон великих чисел, згідно з яким середнє арифметичне випадкових величин збігається до їх математичного сподівання при збільшенні кількості спостережень. У контексті обчислення площі це означає, що відношення кількості точок, які потрапили в область, до загальної кількості точок збігається до відношення площі області до площі всієї вибіркової зони.

Алгоритм методу реалізується наступним чином. В області D рівномірно генерується N випадкових точок p_1, p_2, \dots, p_n з використанням генератора псевдовипадкових чисел. Для забезпечення рівномірності розподілу координати кожної точки генеруються незалежно в межах мінімального описаного прямокутника області D . Точки, що не належать області D (для випадку області складної форми), відкидаються або враховуються окремо.

Для кожної точки p_j перевіряється умова входження хоча б в один покриваючий об'єкт. Ця перевірка виконується послідовно для всіх об'єктів до першого позитивного результату:

$$hit(p_j) = 1, \text{ якщо } \exists i: p_j \in \Omega_i(u_i); 0 - \text{ інакше} \quad (2.6)$$

Для еліпса з центром (x_c, y_c) , півосями a, b та кутом повороту φ умова входження точки (x, y) перевіряється за формулою:

$$\begin{aligned} & ((x - x_c)\cos\varphi + (y - y_c)\sin\varphi)^2/a^2 + ((y - y_c)\cos\varphi - \\ & - (x - x_c)\sin\varphi)^2/b^2 \leq 1 \end{aligned} \quad (2.7)$$

Площа покриття оцінюється як добуток площі області на частку точок, що потрапили в покриття:

$$F_{MC}(U) \approx S(D) \cdot (1/N) \cdot \sum_j hit(p_j) \quad (2.8)$$

Базова обчислювальна складність методу становить:

$$O(N \cdot n), \quad (2.9)$$

де N – кількість випадкових точок;

n – кількість покриваючих об'єктів.

Це пояснюється тим, що для кожної з N точок потенційно необхідно перевірити входження в кожен з n об'єктів. На практиці середня кількість перевірок менша, оскільки перевірка припиняється після першого позитивного результату.

Точність методу визначається центральною граничною теоремою. Стандартне відхилення оцінки площі зменшується пропорційно \sqrt{N} , що означає повільну збіжність: для зменшення похибки вдвічі необхідно збільшити кількість точок вчетверо. Для досягнення відносної похибки близько 2% рекомендується використовувати $N = 1000 - 2000$ точок; для похибки 0.5–1% необхідно $N \geq 3000$ точок.

Суттєвою перевагою методу Монте-Карло є його універсальність: алгоритм працює з об'єктами будь-якої форми без модифікації, оскільки потребує лише функції перевірки входження точки в об'єкт. Метод легко паралелізується на багатоядерних процесорах та графічних прискорювачах, оскільки перевірки для різних точок є незалежними. Крім того, точність методу контролюється через регулювання кількості точок, що дозволяє адаптувати обчислювальні витрати до вимог конкретного етапу оптимізації.

Основним недоліком методу є його стохастична природа: результат може варіюватися між запусками, що може ускладнювати порівняння близьких за якістю конфігурацій. Крім того, при великій кількості об'єктів квадратичний характер зростання часу $O(N \cdot n)$ стає критичним обмеженням, що обумовлює необхідність оптимізації алгоритму.

2.1.3 Метод Монте-Карло з просторовою індексацією R-Tree

Метод Монте-Карло з просторовою індексацією є оптимізованою версією базового методу Монте-Карло, що використовує структуру даних R-Tree для прискорення перевірки входження точок у покриваючі об'єкти. Цей підхід дозволяє значно зменшити обчислювальну складність при великій кількості об'єктів. Основна проблема базового методу Монте-Карло полягає у необхідності перевірки кожної випадкової точки на входження в кожен з n об'єктів, що призводить до складності $O(N \cdot n)$. При великій кількості об'єктів ($n > 50 - 100$) це стає вузьким місцем алгоритму, особливо враховуючи багаторазове обчислення цільової функції в процесі оптимізації.

R-Tree – це збалансована деревоподібна структура даних, розроблена для індексації просторових об'єктів. Структура була запропонована А. Гуттманом у 1984 році [111] та набула широкого застосування в геоінформаційних системах і базах даних з просторовими запитами. Кожен вузол дерева містить мінімальний описаний прямокутник (Minimum Bounding Rectangle, далі – MBR), що охоплює

всі об'єкти у відповідному піддереві. Листові вузли містять посилання на фактичні геометричні об'єкти, а внутрішні вузли – посилання на дочірні вузли.

Ключова властивість R-Tree полягає в тому, що якщо точка запиту не потрапляє в MBR вузла, то вона гарантовано не належить жодному об'єкту в піддереві цього вузла. Це дозволяє ефективно відсікати великі групи об'єктів, які не потребують детальної перевірки.

Алгоритм методу складається з двох фаз: побудови індексу та обробки запитів. На етапі побудови для кожного покриваючого об'єкта $\Omega_i(u_i)$ обчислюється мінімальний описаний прямокутник, і всі прямокутники організуються в ієрархічну структуру R-Tree. Побудова виконується один раз перед початком обчислень і має складність $O(n \log(n))$. На етапі обробки для кожної випадкової точки p_j виконується запит до R-Tree, який повертає множину об'єктів-кандидатів C_j , чий MBR містять точку p_j . Запит реалізується рекурсивним обходом дерева: якщо точка потрапляє в MBR вузла, виконується рекурсивний спуск у дочірні вузли; інакше піддерево відкидається. Середня складність одного запиту становить $O(\log(n))$ при рівномірному розподілі об'єктів. Після отримання множини кандидатів для кожного об'єкта з C_j виконується точна перевірка входження точки. Оскільки кількість кандидатів зазвичай значно менша за загальну кількість об'єктів, загальний час обробки суттєво скорочується. Оцінка площі обчислюється за тією ж формулою (2.8), що й у базовому методі Монте-Карло.

Загальна обчислювальна складність методу становить $O(n \log(n))$ для побудови індексу та $O(N \log(n))$ для обробки всіх точок, що дає асимптотику $O((N + n) \log(n))$. При типових значеннях $N \gg n$ це суттєво краще за $O(N \cdot n)$ базового методу. Наприклад, при $N = 10000$ та $n = 100$ базовий метод виконує до 1000000 перевірок, тоді як метод з R-Tree – близько 70000 (при середній глибині дерева 7).

Перевагою методу є значне прискорення при великій кількості об'єктів, при цьому зберігаються всі переваги базового методу Монте-Карло: універсальність, контрольована точність та можливість паралелізації. Структура R-Tree також забезпечує ефективне використання кешу процесора завдяки локальності даних.

Недоліками методу є накладні витрати на побудову та зберігання індексу, що робить його менш ефективним при малій кількості об'єктів ($n < 20$). Крім того, при сильному перекритті об'єктів MBR багатьох об'єктів містять одні й ті самі точки, що зменшує ефективність відсікання. Проте для типових задач покриття з помірним перекриттям об'єктів метод забезпечує прискорення в 5–10 разів порівняно з базовим методом Монте-Карло. Стохастична природа методу Монте-Карло, навіть з оптимізацією R-Tree, залишає проблему варіативності результатів між запусками, що ускладнює порівняння близьких за якістю конфігурацій на етапі локального пошуку. Це обумовлює доцільність розгляду детермінованих підходів до оцінки площі покриття.

2.1.4 Апроксимація формулою включення-виключення другого порядку

Метод апроксимації формулою включення-виключення другого порядку (Inclusion-Exclusion of second order, далі – IE2) базується на комбінаторній формулі обчислення міри об'єднання множин. На відміну від стохастичних методів, цей підхід є детермінованим і забезпечує вищу точність при помірних обчислювальних витратах. Підхід на основі матриці площ попарних перетинів для ефективного обчислення приросту цільової функції запропоновано у [31]. Точна формула включення-виключення для площі об'єднання n множин виражає цю площу через знакозмінну суму площ перетинів усіх можливих підмножин:

$$S(\Omega_1 \cup \dots \cup \Omega_n) = \sum_i S(\Omega_i) - \sum_{i < j} S(\Omega_i \cap \Omega_j) + \sum_{i < j < k} S(\Omega_i \cap \Omega_j \cap \Omega_k) - \dots, \quad (2.10)$$

Ця формула містить $2^n - 1$ доданків, що робить її пряме застосування непрактичним для великих n .

Апроксимація другого порядку обмежується лише першими двома сумами, ігноруючи члени з потрійними та вищими перетинами:

$$F_{IE2}(U) \approx \sum_i S(\Omega_i) - \sum_{i < j} S(\Omega_i \cap \Omega_j) \quad (2.11)$$

Перший доданок, сума площ усіх покриваючих об'єктів, обчислюється тривіально за відомими формулами (для еліпса $S(\Omega_i) = \pi a_i b_i$). Другий доданок потребує обчислення площ попарних перетинів, що є більш складною задачею.

Для кіл та еліпсів існують аналітичні формули площі перетину, хоча для еліпсів загального положення вони є досить громіздкими. Для багатокутників та апроксимованих полігонами криволінійних об'єктів площа перетину обчислюється за допомогою алгоритмів булевих операцій, реалізованих у бібліотеках Shapely [53], Clipper або CGAL.

Теоретична обчислювальна складність методу становить $O(n^2)$, оскільки необхідно обчислити площі всіх $C(n, 2) = n(n - 1)/2$ попарних перетинів. Однак на практиці граф непорожніх перетинів є розрідженим, оскільки більшість пар об'єктів не перетинаються. Для оптимізованих конфігурацій покриття середня кількість сусідів (об'єктів, що перетинаються з даним) становить 3–6, для випадкових конфігурацій — $O(\sqrt{n})$. Таким чином, реальна кількість непорожніх попарних перетинів масштабується як $O(n)$, і обчислювальна складність на практиці має квазілінійний характер $O(n \log n)$.

Похибка методу IE2 виникає через ігнорування членів вищих порядків у формулі включення-виключення. За відсутності потрійних перетинів метод дає точний результат. При наявності потрійних перетинів похибка пропорційна сумарній площі областей, покритих трьома та більше об'єктами одночасно. Для

типових конфігурацій покриття з помірним перекриттям метод забезпечує точність 0.5–1%.

Перевагами методу ІЕ2 є висока точність при помірному перекритті об'єктів, детермінований характер результату та ефективність при наявності аналітичних формул перетину. Метод особливо добре підходить для локального пошуку, де необхідно точно порівнювати близькі за якістю конфігурації.

Недоліками методу є втрата точності при значних потрійних перетинах, необхідність реалізації алгоритмів перетину для кожного типу об'єктів та квадратична складність у найгіршому випадку (при щільному розташуванні об'єктів). Для задач з великою кількістю об'єктів ($n > 100$) або при щільному покритті рекомендується використовувати стохастичні методи.

Окрім стохастичних та аналітичних наближених підходів, існує можливість точного обчислення площі покриття за допомогою спеціалізованих бібліотек обчислювальної геометрії, що реалізують булеві операції над полігональними представленнями геометричних об'єктів.

2.1.5 Точні геометричні бібліотеки

Метод точних геометричних обчислень базується на використанні спеціалізованих бібліотек обчислювальної геометрії, що реалізують точні булеві операції над полігональними об'єктами. Серед найбільш поширених бібліотек слід відзначити Shapely для мови Python [53], CGAL (Computational Geometry Algorithms Library) для C++, Boost.Geometry та JTS (Java Topology Suite). У даному дослідженні використовується бібліотека Shapely як найбільш зручна для інтеграції з науковими обчисленнями на Python.

Бібліотека Shapely є Python-обгорткою над бібліотекою GEOS (Geometry Engine – Open Source), яка, у свою чергу, є портом Java Topology Suite. Обидві бібліотеки реалізують специфікацію Simple Features Access консорціуму Open

Geospatial Consortium, що забезпечує стандартизований інтерфейс для роботи з геометричними об'єктами та їх взаємовідносинами.

Принцип представлення геометричних об'єктів у Shapely базується на полігональній апроксимації. Кожен геометричний об'єкт представляється як полігон – впорядкована послідовність вершин, що утворюють замкнутий контур. Прості полігони задаються зовнішнім кільцем (exterior ring), складні полігони з отворами – додатково внутрішніми кільцями (interior rings). Криволінійні об'єкти, такі як еліпси, апроксимуються полігонами з достатньою кількістю вершин.

Для апроксимації еліпса полігоном використовується рівномірна дискретизація параметричного рівняння еліпса. Точки на контурі еліпса з центром (x_c, y_c) , півсями a, b та кутом повороту φ обчислюються за формулами:

$$x(t) = x_c + a \cdot \cos(t) \cdot \cos(\varphi) - b \cdot \sin(t) \cdot \sin(\varphi), \quad (2.12)$$

$$y(t) = y_c + a \cdot \cos(t) \cdot \sin(\varphi) + b \cdot \sin(t) \cdot \cos(\varphi), \quad (2.13)$$

де t приймає значення від 0 до 2π з кроком $2\pi/k$;

k – кількість точок апроксимації.

Типове значення k становить 64–128 точок, що забезпечує похибку апроксимації площі менше 0.01%. При $k = 64$ відносна похибка площі еліпса складає приблизно 0.003%.

Алгоритм обчислення площі покриття за допомогою Shapely складається з чотирьох основних етапів. На першому етапі виконується створення геометрій: кожен покриваючий об'єкт Ω_i перетворюється на об'єкт типу Polygon бібліотеки Shapely. Для еліпсів це реалізується через генерацію точок апроксимації та створення полігону з цих точок.

На другому етапі виконується операція об'єднання (union) всіх покриваючих об'єктів. Бібліотека Shapely надає функцію `unary_union`, яка ефективно обчислює об'єднання колекції геометричних об'єктів. Внутрішньо ця операція реалізована як послідовність попарних об'єднань з використанням каскадного алгоритму (cascaded union), що зменшує загальну кількість операцій.

Операція union є найбільш трудомісткою частиною алгоритму і займає приблизно 90–95% загального часу обчислення площі. Ця операція включає пошук усіх точок перетину контурів об'єктів, побудову топології об'єднаної фігури (визначення, які частини контурів належать межі об'єднання) та обробку вироджених випадків (дотики, співпадіння ребер, самоперетини). Складність операції union для двох полігонів з k_1 та k_2 вершинами становить $O((k_1 + k_2) \log(k_1 + k_2))$ при використанні алгоритму Бентлі-Отмана для пошуку перетинів [112].

На третьому етапі результат об'єднання перетинається з областю покриття D за допомогою операції `intersection`. Ця операція також є булевою операцією над полігонами і має аналогічну складність. Результатом може бути простий полігон, мультиполігон (колекція непов'язаних полігонів) або порожня множина.

На четвертому етапі для отриманого полігону (або мультиполігону) обчислюється площа за допомогою атрибута `.area`. Внутрішньо площа обчислюється за формулою шолейс (shoelace formula, також відома як формула Гауса або формула землемірів):

$$S(P) = \frac{1}{2} |\sum_{i=1}^n (x_i y_{i+1} - x_{i+1} y_i)| \quad (2.14)$$

де $(x_1, y_1), \dots, (x_n, y_n)$ – вершини полігону в порядку обходу;

$$(x_{n+1}, y_{n+1}) = (x_1, y_1).$$

Для полігону з отворами площа обчислюється як площа зовнішнього кільця мінус суми площ внутрішніх кілець. Складність обчислення площі становить $O(k)$, де k – загальна кількість вершин результуючого полігону.

Загальна обчислювальна складність методу Shapely залежить від кількості об'єктів n та складності їх контурів k . Для n об'єктів з k вершинами кожен час роботи операції `unary_union` масштабується приблизно як $O(n \cdot k \cdot \log(n \cdot k))$ при помірному перекритті об'єктів. На практиці спостерігається лінійне зростання часу зі збільшенням числа об'єктів при фіксованій складності контурів. Експериментальні дослідження показують наступні характерні значення часу обчислення на типовому настільному комп'ютері. Для конфігурації з 50 еліпсів ($k = 64$ точки кожен) час обчислення площі становить приблизно 0.02 секунди. Для 100 еліпсів час зростає до 0.04 секунди, для 500 еліпсів – до 0.24 секунди, для 1000 еліпсів – до 0.64 секунди. Ці значення є прийнятними для одноразового обчислення, проте стають критичними при багаторазовому виклику в циклі оптимізації.

Розглянемо типовий сценарій використання: алгоритм PSO з популяцією 50 частинок та 100 ітераціями потребує 5000 обчислень цільової функції. При часі одного обчислення 0.64 секунди (для 1000 об'єктів) загальний час тільки на обчислення площі складе близько 53 хвилин, що є неприйнятним для практичного застосування. Саме тому метод Shapely рекомендується використовувати лише для верифікації фінального рішення, а не для всіх ітерацій оптимізації.

Основними перевагами методу точних геометричних бібліотек є машинна точність обчислень (похибка обмежена лише точністю представлення чисел з плаваючою комою та похибкою полігональної апроксимації криволінійних об'єктів), коректна обробка всіх геометричних випадків включаючи складні топології з отворами та островами, а також широкий набір підтримуваних операцій для подальшого аналізу результатів.

Недоліками методу є найвища обчислювальна вартість серед розглянутих методів, суттєва залежність часу роботи від складності геометрії та непрактичність для багаторазового виклику в циклі оптимізації. У зв'язку з цим метод рекомендується використовувати у двох випадках: для калібрування та перевірки точності наближених методів на етапі розробки алгоритму; для остаточної верифікації найкращого знайденого рішення після завершення оптимізації.

2.2 Експериментальні дослідження та порівняльний аналіз методів оцінки площі покриття

Проведений аналіз п'яти методів обчислення площі конфігурації покриття дозволяє сформулювати їх порівняльну характеристику за основними критеріями: обчислювальною складністю, точністю та рекомендованою областю застосування. Узагальнені результати порівняння наведено в таблиці 2.1. З таблиці видно, що методи утворюють спектр від найшвидших з низькою точністю до найточніших з високою обчислювальною вартістю.

Таблиця 2.1 – Порівняння методів обчислення площі покриття

Метод	Складність	Похибка	Рекомендоване застосування
Bounding Boxes	$O(n)$	Висока (20-40%)	Груба верхня оцінка, перші ітерації глобального пошуку
Монте-Карло (базовий)	$O(N \cdot n)$ (N – кількість точок методу)	1–3%	Глобальний пошук, $n < 50$
Монте-Карло + R-Tree	$O(N \log n)$ (N – кількість точок методу)	1–3%	Глобальний пошук, $n > 50 - 100$
IE2	$O(n \log n)$	~8% для щільного розміщення	Локальний пошук, аналітичні форми об'єктів
Shapely	$O(n \cdot k \cdot \log(nk))$ (k – кількість вершин об'єкту)	Машинна точність	Верифікація, калібрування

Для обґрунтування вибору методу обчислення площі покриття та визначення меж ефективного застосування кожного підходу було проведено серію обчислювальних експериментів.

Експерименти проводились на комп'ютері з такою конфігурацією: Macbook Pro, M4 processor, 16Gb RAM. Як тестові дані використовувались випадково згенеровані конфігурації еліпсів з рівномірно розподіленими параметрами. Результати експериментів наведено на рисунках 2.1-2.3 (Для методів Монте-Карло кількість точок = 1000). Результати часу обчислення наведено у таблиці 2.2:

Таблиця 2.2 – Час обчислення площі різними методами

Кількість об'єктів	Shapely (мс)	Bounding boxes (мс)	Monte-Carlo (мс)	Monte-Carlo + R-Tree (мс)	IE2 (мс)
10	0.21	0.24	0.27	0.37	0.33
20	0.44	0.37	0.46	0.53	0.66
30	0.62	0.46	0.60	0.62	0.96
40	0.87	0.57	0.77	0.74	1.27
50	1.15	0.69	0.97	0.88	1.60
100	2.33	1.20	1.75	1.31	2.95

Результати точності обчислення (% похибки відносно Shapely) зведено у таблиці 2.3:

Таблиця 2.3 – Точність обчислення площі різними методами

Кількість об'єктів	Bounding boxes (%)	Monte-Carlo (%)	Monte-Carlo + R-Tree (%)	IE2 (%)
10	+20.5	+0.2	-0.6	0.0
20	+33.9	+3.6	+2.0	-3.1
30	+28.8	-0.8	+1.9	-7.4
40	+18.7	-0.8	-2.1	-5.2
50	+28.3	-1.2	+1.3	-6.7
100	+29.0	+4.6	-1.6	-7.0

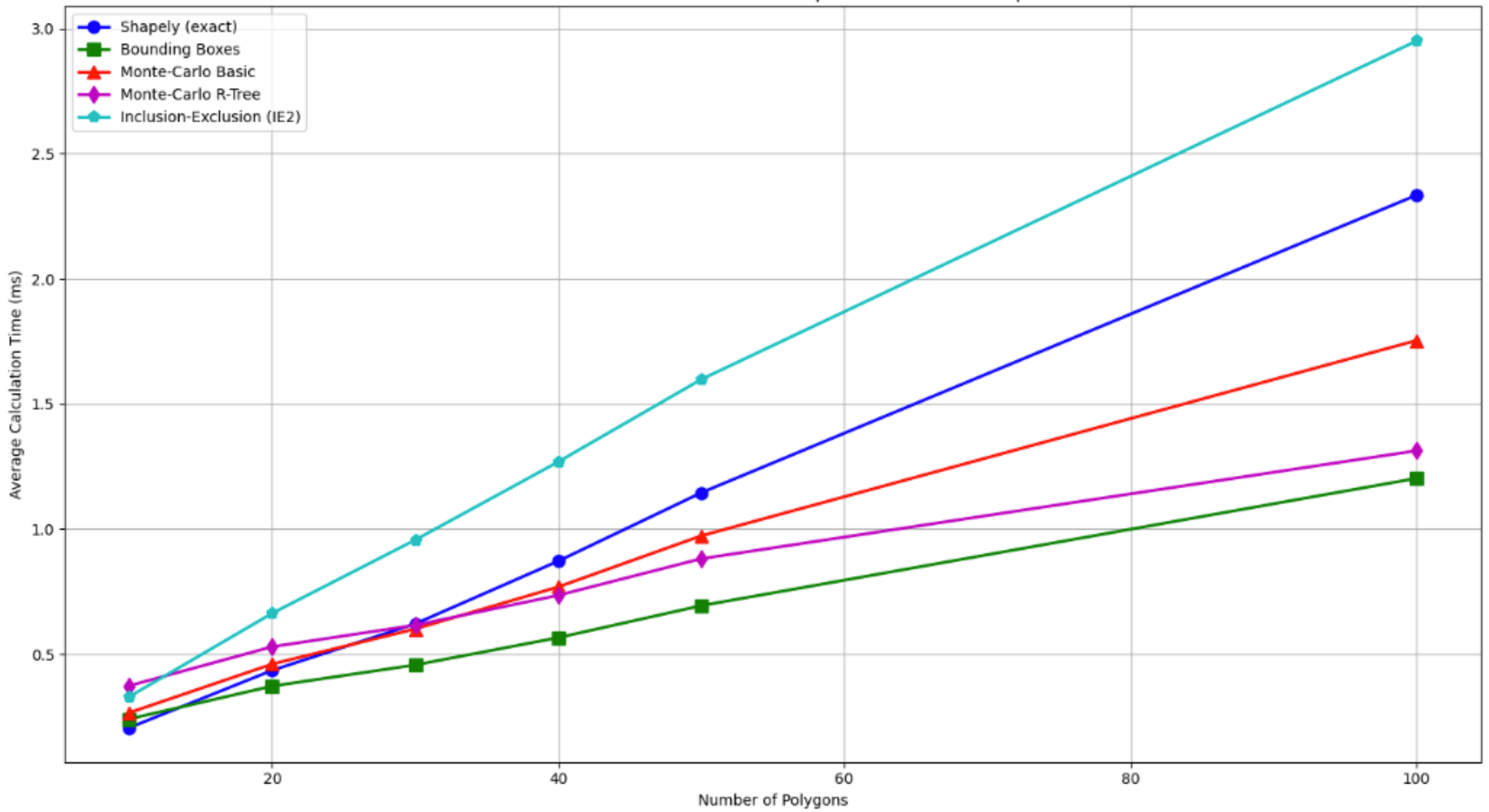


Рисунок 2.1 – Порівняння часу обчислення площі різними методами

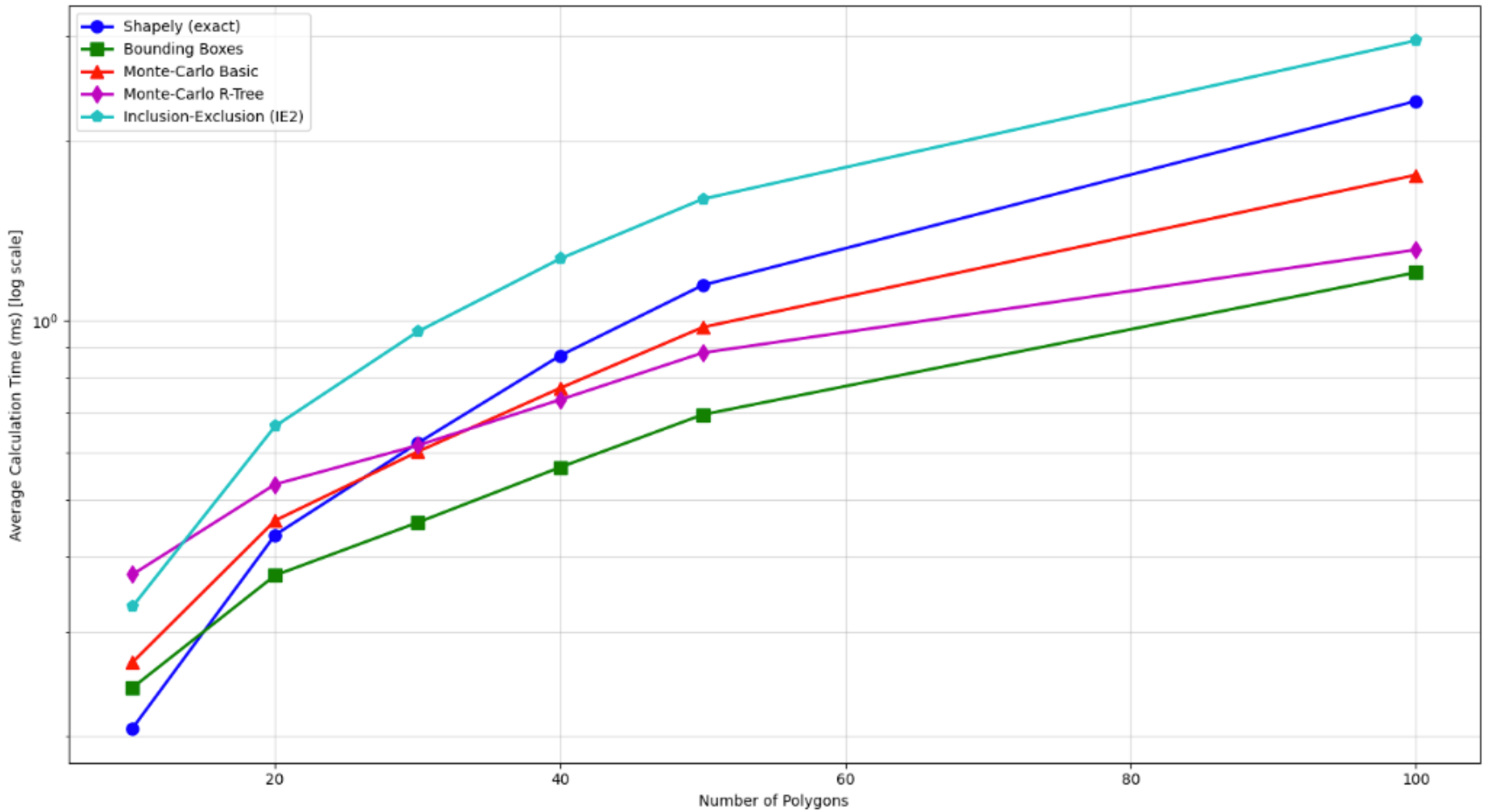


Рисунок 2.2 – Порівняння часу обчислення площі різними методами (логарифмічна шкала)

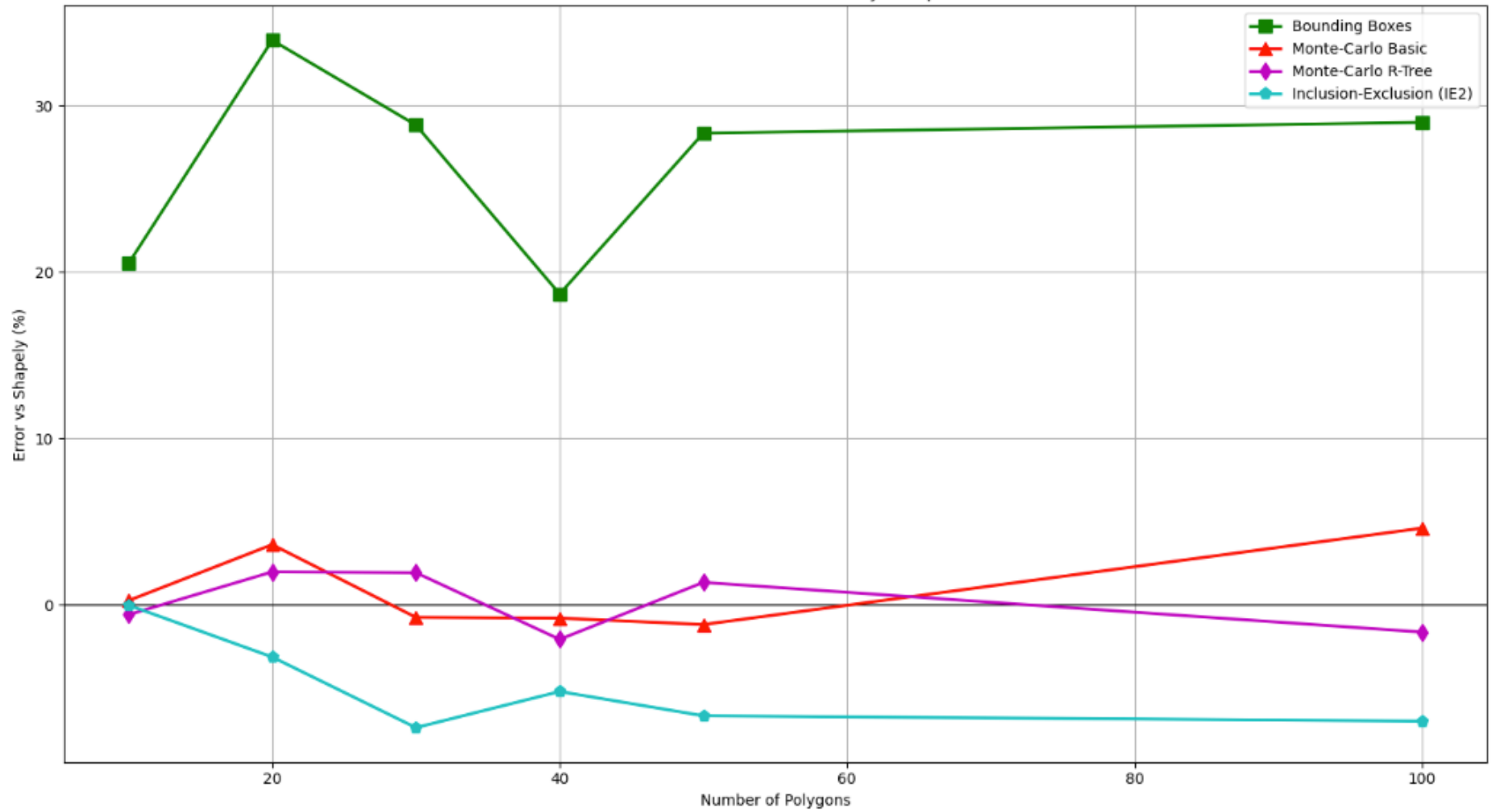


Рисунок 2.3 – Порівняння точності обчислення площі різними методами відносно Shapely

Результати експериментів дозволяють сформулювати кількісні рекомендації щодо вибору методу обчислення площі покриття залежно від параметрів задачі.

2.2.1 Рекомендації щодо вибору методу

На основі теоретичного аналізу та експериментальних досліджень сформульовано практичні рекомендації щодо вибору методу обчислення площі покриття залежно від етапу оптимізації та кількості покриваючих об'єктів. Узагальнені рекомендації наведено в таблиці 2.4.

Таблиця 2.4 – Рекомендований вибір методу обчислення площі

Кількість об'єктів	Глобальний пошук	Локальний пошук	Верифікація
$n < 30$	Монте-Карло (N=1000–2000)	IE2 або Монте-Карло (N=2000)	Shapely
$30 \leq n < 100$	Монте-Карло + R-Tree	IE2	Shapely
$n \geq 100$	Монте-Карло + R-Tree	Монте-Карло + R-Tree (N=2000-3000)	Shapely (одноразово)

Для задач з малою кількістю об'єктів ($n < 30$) на етапі глобального пошуку достатньо базового методу Монте-Карло з 1000–2000 точками, який забезпечує прийнятну точність при мінімальних накладних витратах. На етапі локального пошуку рекомендується використовувати метод IE2 для об'єктів з аналітичними формулами перетину або метод Монте-Карло зі збільшеною кількістю точок ($N \geq 2000$) для об'єктів довільної форми.

Для задач із середньою кількістю об'єктів ($30 \leq n < 100$) на етапі глобального пошуку рекомендується застосовувати метод Монте-Карло з просторовою індексацією R-Tree, який забезпечує логарифмічну складність пошуку. На етапі локального пошуку ефективним залишається метод IE2 завдяки розрідженості графу перетинів.

Для задач з великою кількістю об'єктів ($n \geq 100$) метод Монте-Карло з R-Tree є основним методом як для глобального, так і для локального пошуку. Метод IE2 стає менш ефективним через квадратичне зростання кількості потенційних перетинів. Метод Shapely рекомендується використовувати лише одноразово для верифікації фінального рішення. Незалежно від кількості об'єктів верифікація найкращого знайденого рішення виконується методом Shapely, який гарантує машинну точність обчислення площі покриття.

Таким чином, розглянуті методи обчислення площі покриття забезпечують необхідний інструментарій для ефективного оцінювання цільової функції задачі максимального покриття. Наступним кроком є вибір та адаптація методів оптимізації, здатних знаходити конфігурації покриваючих об'єктів, що максимізують значення цієї цільової функції. З огляду на властивості задачі, такі як висока розмірність простору параметрів, мультимодальність та негладкість цільової функції, найбільш перспективними є метаевристичні методи, зокрема ройові алгоритми обчислювального інтелекту.

2.3 Застосування ройових алгоритмів в задачах максимального покриття

Задачі оптимізації, що виникають у прикладній математиці, інженерії та операційному дослідженні, часто характеризуються високою розмірністю простору параметрів, мультимодальністю цільової функції, наявністю складних обмежень та відсутністю аналітичних виразів для градієнтів. Класичні детерміновані методи оптимізації, такі як метод Ньютонів, градієнтний спуск або метод спряжених градієнтів, у таких умовах виявляються недостатньо ефективними, оскільки вони, як правило, збігаються до найближчого локального екстремуму й не здатні забезпечити пошук глобального оптимуму у складних ландшафтах цільової функції.

Для подолання цих обмежень у другій половині ХХ століття було розроблено клас методів, що отримали назву метаевристичних. Метаевристики являють собою загальні стратегії пошуку, які керують процесом дослідження простору розв'язків за допомогою механізмів, що не залежать від конкретної постановки задачі. Ройові метаевристики належать до класу методів агентного підходу, де кожна частинка, риба чи світлячок виступає автономним агентом, що взаємодіє з іншими агентами за локальними правилами. На відміну від точних методів, метаевристики не гарантують знаходження глобального оптимуму за скінченне число кроків, проте забезпечують знаходження прийнятно якісних розв'язків за обмежений час навіть для задач надвисокої складності.

Характерними рисами метаевристичних підходів є: стохастична природа пошуку, що дозволяє уникати детермінованих пасток локальних екстремумів; балансування між дослідженням нових областей простору параметрів (exploration) та поглибленою експлуатацією перспективних зон (exploitation); відсутність потреби в аналітичних виразах для градієнтів цільової функції; та універсальність застосування до широкого класу задач. Серед суміжних стохастичних підходів до задач розміщення слід відзначити оптимізацію на основі перестановок [113].

Ройовий інтелект об'єднує методи, що імітують колективну поведінку біологічних організмів: оптимізацію роєм частинок, пошук риб'ячої зграї, алгоритм світлячків, алгоритм штучної бджолоїної колонії, мурашиний алгоритм та інші. Ці методи базуються на принципі емерджентності, коли складна колективна поведінка виникає з простих правил взаємодії між окремими агентами.

Важливою властивістю будь-якого метаевристичного алгоритму є баланс між дослідженням та експлуатацією. Дослідження передбачає переміщення пошукових агентів до раніше не відвідуваних областей простору параметрів з метою виявлення нових перспективних зон. Експлуатація, навпаки, зосереджена

на поглибленому аналізі вже виявлених перспективних областей для знаходження локального оптимуму в їхніх околицях. Надмірний акцент на дослідженні призводить до повільної збіжності, тоді як надмірна експлуатація спричиняє передчасну збіжність до локального екстремуму. Жоден метаевристичний алгоритм не є оптимальним для всіх класів задач, що зумовлює необхідність ретельного аналізу відповідності характеристик методу структурним властивостям конкретної задачі оптимізації.

2.3.1 Обґрунтування вибору ройових алгоритмів для задач покриття

Для розв'язання сформульованої задачі безумовної оптимізації у просторі $3n$ параметрів, де n позначає кількість покривальних об'єктів, обрано ройові метаевристики обчислювального інтелекту. Цей вибір зумовлений кількома вагомими аргументами, що безпосередньо впливають зі структури та властивостей задачі максимального геометричного покриття.

По-перше, ройові алгоритми працюють безпосередньо з неперервними параметрами, що відповідає природній параметризації задачі покриття, де кожен об'єкт описується координатами полюса та кутом обертання. На відміну від генетичних алгоритмів, які потребують кодування та декодування параметрів, ройові методи оперують у тому самому геометричному просторі конфігурацій, що й сама задача.

По-друге, висока розмірність простору параметрів ($3n$ змінних, де для практичних задач n може досягати 50–100 об'єктів) створює серйозні обчислювальні виклики для класичних методів оптимізації. Ройові алгоритми демонструють кращу масштабованість у задачах високої розмірності порівняно з методами, що базуються на обчисленні матриці Гессе або повному переборі. Питання обробки викидів у ройових алгоритмах та їх вплив на збіжність при розв'язанні задач покриття досліджено у [114].

По-третє, цільова функція задачі покриття є мультимодальною та негладкою. Площа покриття змінюється нелінійно при зсуві або обертанні будь-якого з об'єктів, причому невеликі зміни параметрів можуть призводити до значних стрибків значення цільової функції. Ройові алгоритми, завдяки стохастичній природі пошуку та колективному обміну інформацією, ефективно долають такі негладкості та уникають передчасної збіжності до локальних екстремумів.

По-четверте, наявність заборонених зон та невивуклість цільової області створюють фрагментовану допустиму область у просторі параметрів. Ройові алгоритми, на відміну від градієнтних методів, не потребують зв'язності допустимої множини й здатні паралельно досліджувати відокремлені фрагменти простору конфігурацій.

Нарешті, ройові алгоритми природним чином інтегруються з зовнішньою штрафною функцією, що використовується для формулювання задачі безумовної оптимізації. Штрафні доданки за порушення геометричних обмежень коректно обробляються ройовими методами без модифікації їхньої базової структури.

У всіх розглянутих нижче алгоритмах кожна особина рою (частинка, риба, світлячок чи бджола) кодує повний вектор параметрів розміщення $U \in \mathbb{R}^{3n}$ (формула (1.4)), що містить координати полюса та кут обертання кожного з n покриваючих об'єктів. Якість кожної особини визначається значенням цільової функції $\tilde{F}(U, \lambda)$ (формула (1.15)). Розмірність простору параметрів становить $3n$. Для стислості запису у формулах ройових алгоритмів значення цільової функції $\tilde{F}(U, \lambda)$ позначатимемо $f(U)$. У подальших підрозділах детально розглянуто чотири ройові метаевристики: оптимізацію роєм частинок, пошук риб'ячої зграї, алгоритм світлячків та алгоритм штучної бджолоїної колонії.

2.3.2 Оптимізація роєм частинок

Алгоритм оптимізації роєм частинок, запропонований Кеннеді та Еберхартом у 1995 році [62], є одним із найбільш поширених та добре вивчених ройових методів оптимізації. Метод імітує колективну поведінку зграї птахів або косяку риби, де кожна особина (частинка) рухається у просторі пошуку під впливом власного досвіду та досвіду інших членів популяції.

У контексті задачі максимального покриття кожна частинка рою кодує повну конфігурацію з n покривальних об'єктів. Позиція частинки являє собою вектор параметрів розміщення $U \in \mathbb{R}^{3n}$. Кожна частинка асоціюється з вектором швидкості $v_i \in \mathbb{R}^{3n}$, який визначає, яким чином конфігурація змінюється між ітераціями алгоритму.

На кожній ітерації t алгоритм оновлює швидкість та позицію кожної частинки i за формулами:

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot r_1 \cdot (pbest_i - U_i(t)) + c_2 \cdot r_2 \cdot (gbest - U_i(t)), \quad (2.15)$$

$$U_i(t+1) = U_i(t) + v_i(t+1), \quad (2.16)$$

де w – інерційний коефіцієнт;

c_1, c_2 – когнітивний та соціальний коефіцієнти прискорення відповідно;

r_1, r_2 – випадкові числа з рівномірного розподілу $Uniform(0,1)$, що генеруються незалежно для кожної компоненти вектора на кожній ітерації;

$pbest_i$ – найкраща позиція, знайдена i -ю частинкою за весь час роботи алгоритму;

$gbest$ – глобально найкраща позиція, виявлена будь-яким членом рою.

Формула (2.15) складається з трьох фундаментальних компонент:

1) інерційний член $w \cdot v_i(t)$ зберігає частину попереднього руху частинки й тим самим сприяє продовженню поточного напрямку зсувів та обертань у просторі конфігурацій;

2) когнітивний член $c_1 \cdot r_1 \cdot (pbest_i - U_i(t))$ притягує частинку до її власної найкращої конфігурації, тобто до тієї розстановки об'єктів, яка забезпечила найбільшу площу покриття саме для цієї частинки;

3) соціальний член $c_2 \cdot r_2 \cdot (gbest - U_i(t))$ притягує частинку до глобально найкращої конфігурації, виявленої будь-яким членом рою.

Для забезпечення стійкості алгоритму застосовується обмеження максимальної швидкості:

$$v_{ij}(t + 1) = clamp(v_{ij}(t + 1), -v_max_j, v_max_j). \quad (2.17)$$

У задачах покриття рекомендовано обмежувати максимальну швидкість величиною порядку 10–20% від характерного лінійного розміру цільової області для позиційних компонент та 0.1–0.3 радіан для кутових компонент.

Інерційний коефіцієнт w регулює ступінь збереження попереднього напрямку руху. Висока інерція ($w \approx 0.9$) відповідає довгим, плавним траєкторіям, що сприяє дослідженню віддалених областей. Низька інерція ($w \approx 0.4$) призводить до більш локалізованих рухів. Для задач покриття рекомендовано $w \in [0.72, 0.74]$. Когнітивний коефіцієнт $c_1 = 1.496$ та соціальний коефіцієнт $c_2 = 1.496$ забезпечують збіжність до високоякісних конфігурацій з теоретичними гарантіями стабільності рою. Розмір рою складає від $15n$ до $20n$, де загальна кількість агентів не перевищує 1500.

Перевагами методу PSO у контексті задач максимального покриття є висока швидкість формування щільних покриттів завдяки ефективному механізму обміну інформацією між частинками, простота реалізації та налаштування, добра масштабованість для задач помірної розмірності. Основним

недоліком PSO є схильність до передчасного застрягання у локальних екстремумах, що особливо проявляється у задачах із складною геометрією цільової області та наявністю заборонених зон.

2.3.3 Пошук риб'ячої зграї

Алгоритм пошуку риб'ячої зграї, запропонований Бастос-Фільо та де Ліма-Нето у 2008 році [73], моделює колективну поведінку зграї риб, що рухається у безперервному середовищі. У задачі покриття кожна «риба» відповідає конфігурації покривальних об'єктів. Алгоритм поєднує три основні типи руху.

Кожна риба виконує невелике випадкове зміщення у просторі параметрів:

$$U_i'(t) = U_i(t) + step_ind \cdot Uniform(-1, 1) \quad (2.18)$$

де $step_ind$ – амплітуда індивідуального кроку.

Якщо нова конфігурація забезпечує більшу площу покриття ($f(U_i') > f(U_i)$), риба переміщується до неї; інакше залишається на попередній позиції. Геометрично це відповідає локальним випадковим збуренням позицій та орієнтацій покривальних об'єктів.

Після індивідуального руху оновлюється вага кожної риби:

$$W_i(t + 1) = W_i(t) + \Delta f_i(t) / \max(|\Delta f_j(t)|), \forall j, \quad (2.19)$$

де $\Delta f_i(t) = f(U_i(t + 1)) - f(U_i(t))$ – зміна значення цільової функції.

Конфігурації, що забезпечують покращення площі покриття, набирають вагу та стають більш впливовими при обчисленні колективних напрямків руху.

Риби, що покращили своє покриття, формують зважений вектор напрямку:

$$I(t) = \Sigma(\Delta U_i(t) \cdot \Delta f_i(t)) / \Sigma(\Delta f_i(t)), \forall i: \Delta f_i > 0, \quad (2.20)$$

$$U_i(t + 1) = U_i(t) + I(t). \quad (2.21)$$

Зграя стискається або розширюється навколо центру мас залежно від динаміки загального покриття:

$$B(t) = \Sigma(U_i(t) \cdot W_i(t)) / \Sigma(W_i(t)), \quad (2.22)$$

$$U_i(t + 1) = U_i(t) - step_{vol} \cdot (U_i(t) - B(t)) / \|U_i(t) - B(t)\|, \\ \text{якщо } \Sigma W(t + 1) > \Sigma W(t), \quad (2.23)$$

$$U_i(t + 1) = U_i(t) + step_{vol} \cdot (U_i(t) - B(t)) / \|U_i(t) - B(t)\|, \\ \text{якщо } \Sigma W(t + 1) \leq \Sigma W(t), \quad (2.24)$$

де $B(t)$ – зважений центр мас зграї;

$step_{vol}$ – амплітуда волітивного кроку.

Якщо загальне покриття зросло, зграя стискається навколо центру мас, якщо покриття погіршилося, зграя розширюється для дослідження нових областей. Розмір зграї обирається пропорційно кількості змінних: від $15n$ до $25n$. Амплітуда індивідуального кроку:

$$step_{ind} \in [0.05, 0.15] \times diam, \quad (2.25)$$

де $diam$ – діаметр цільової області.

Амплітуда волітивного кроку $step_{vol} \in [0.01, 0.05] \times diam$. Перевагами FSS є добре збереження різноманітності конфігурацій та висока стійкість при складній геометрії з забороненими зонами. Основним недоліком є дещо більша обчислювальна вартість кожної ітерації порівняно з PSO.

2.3.4 Алгоритм світлячків

Алгоритм світлячків, запропонований Янгом у 2009 році [75], базується на моделюванні світлових сигналів між світлячками. Кожен світлячок представляє конфігурацію покривальних об'єктів, а його «яскравість» є пропорційною площі покриття. Основний принцип: кожен світлячок рухається до інших, яскравіших світлячків, причому сила притягання зменшується зі зростанням відстані.

Привабливість між двома світлячками визначається як функція відстані:

$$\beta(r_{ij}) = \beta_0 \cdot \exp(-\gamma \cdot r_{ij}^2), \quad (2.26)$$

де β_0 – привабливість при нульовій відстані;

γ – коефіцієнт поглинання світла;

$r_{ij} = ||U_i - U_j||$ – евклідова відстань між світлячками i та j у просторі параметрів.

Рух світлячка i до яскравішого світлячка j описується формулою:

$$U_i(t+1) = U_i(t) + \beta_0 \cdot \exp(-\gamma \cdot r_{ij}^2) \cdot (U_j(t) - U_i(t)) + \alpha \cdot \varepsilon_i, \quad (2.27)$$

де α – параметр стохастичності;

ε_i – вектор випадкових чисел з нормального або рівномірного розподілу.

У задачах покриття другий доданок відповідає за притягання конфігурації до кращих патернів покриття, а третій – за випадкові збурення, що дозволяють

уникати вироджених ситуацій. Для найяскравішого світлячка (з найбільшою площею покриття) рух визначається лише випадковим збуренням:

$$U_best(t + 1) = U_best(t) + \alpha \cdot \varepsilon \quad (2.28)$$

Параметр привабливості $\beta_0 \in [0.8, 1.2]$ визначає максимальну силу притягання. Параметр поглинання $\gamma \in [0.5, 2.0]$ контролює дальність взаємодії: малі значення забезпечують далекий вплив, великі – локальний пошук. Стохастичний крок:

$$\alpha \in [0.15, 0.25] \times L, \quad (2.29)$$

де L – розмір об'єкта.

Суттєвою відмінністю FA від інших розглянутих алгоритмів є квадратична обчислювальна складність кожної ітерації $O(N^2)$, оскільки кожен світлячок порівнюється з кожним іншим.

2.3.5 Штучна бджолина колонія

Алгоритм штучної бджолиної колонії, запропонований Карабогою та Бастурком у 2007 році [71], імітує поведінку бджолиної колонії, що шукає джерела нектару. У контексті задачі покриття кожне джерело нектару відповідає конфігурації покривальних об'єктів, а «кількість нектару» пропорційна площі покриття. Алгоритм складається з трьох фаз. Кожна фаза реалізує окремий механізм дослідження простору конфігурацій: бджоли-робітниці здійснюють локальну модифікацію поточних конфігурацій, бджоли-спостерігачі обирають найперспективніші рішення для подальшого покращення, а бджоли-розвідники замінюють виснажені конфігурації випадковими новими, забезпечуючи вихід з локальних екстремумів.

Фаза бджіл-робітниць. Кожна бджола-робітниця асоціюється з конфігурацією та виконує локальну модифікацію:

$$v_{ij} = U_{ij} + \psi_{ij} \cdot (U_{ij} - U_{kj}), \quad (2.30)$$

де v_{ij} – нова кандидатна позиція;

ψ_{ij} – випадкове число з $Uniform(-1, 1)$;

j – випадково обрана координата;

k – випадково обрана інша конфігурація ($k \neq i$).

Якщо $f(v_i) > f(U_i)$, нова конфігурація замінює стару; інакше зберігається попередня.

Фаза бджіл-спостерігачів. Спостерігачі обирають конфігурації з імовірностями, пропорційними якості покриття:

$$p_i = fit_i / \sum fit_k, \quad (2.31)$$

де $fit_i = 1/(1 + f(U_i))$ для мінімізації;

$fit_i = f(U_i)$ для максимізації.

Обрані конфігурації зазнають такої самої локальної модифікації за формулою (2.30). Цей механізм підсилює експлуатацію перспективних патернів покриття, оскільки на уточнення конфігурацій з більшим покриттям спрямовується більше пошукових ресурсів.

Фаза бджіл-розвідниць. Якщо конфігурація U_i не покращувалася протягом $limit$ ітерацій, відповідна бджола стає розвідницею і генерує нову випадкову конфігурацію:

$$U_{ij} = U_{min_j} + rand(0,1) \cdot (U_{max_j} - U_{min_j}) \quad (2.32)$$

Геометрично це відповідає стрибку з насиченого локального регіону простору параметрів до віддаленої, раніше не дослідженої області.

Розмір колонії складає від $15n$ до $25n$. Амплітуда модифікації $A \in [0.1, 0.3] \times diam$. Поріг виснаження $limit \in [5, 10]$ циклів. Перевагами ABC є добре збереження різноманітності конфігурацій, мінімальна кількість параметрів та ефективний вихід з локальних екстремумів. Основні недоліки: повільніша локальна збіжність та потреба у більшій кількості ітерацій.

Описані чотири ройові алгоритми реалізують різні стратегії балансування між дослідженням та експлуатацією простору конфігурацій, що обумовлює необхідність їх порівняльного експериментального дослідження на задачах покриття різної складності.

2.4 Експериментальні дослідження та порівняльний аналіз ройових алгоритмів в задачах максимального покриття

Для порівняльного аналізу ефективності розглянутих ройових алгоритмів у задачах максимального покриття проведено серію обчислювальних експериментів. У цьому підрозділі описано умови експерименту, параметри цільової області та характеристики покривальних об'єктів.

Як цільову область D обрано три полігональні області (для різних серій експериментів) на площині, які використовуються у всіх експериментах для всіх чотирьох алгоритмів, що забезпечує коректність порівняння. Для кожної серії експериментів, область D є багатокутником із визначеними координатами вершин. Площа цільової області становить $S(D)$.

У якості покривальних об'єктів використано геометричні фігури довільної форми (кола, еліпси, прямокутники, трикутники та інші полігони), кожна з яких характеризується метричними параметрами (розміри, форма) та параметрами розміщення (координати полюса x_i, y_i та кут обертання φ_i). Набір об'єктів є

однаковим для всіх чотирьох алгоритмів у кожній серії експериментів, що забезпечує коректність порівняння.

Для оцінки масштабованості алгоритмів проведено три серії експериментів з різною кількістю покривальних об'єктів, але з однаковою цільовою областю D :

- 1) серія 1: $n = 20$ покривальних об'єктів довільної форми;
- 2) серія 2: $n = 40$ покривальних об'єктів довільної форми;
- 3) серія 3: $n = 80$ покривальних об'єктів довільної форми.

У кожній серії набір з 20 (40, 80) об'єктів є однаковим для всіх чотирьох алгоритмів (PSO, FSS, FA, ABC), тобто 20 об'єктів для PSO – це ті самі 20 об'єктів, що використовуються для FSS, FA та ABC. Це гарантує, що різниця у результатах зумовлена виключно відмінностями у механізмах пошуку алгоритмів, а не варіаціями вхідних даних. У кожній серії експериментів загальна сума площ покриваючих об'єктів складає приблизно 95% площі області покриття.

Для кожного алгоритму використано рекомендовані значення метапараметрів, наведені у підрозділах 2.3.2–2.3.5. Кількість ітерацій, початкова ініціалізація та критерії зупинки є однаковими для всіх алгоритмів у кожній серії. Для оцінки площі покриття використано бібліотеку Shapely, яка забезпечує точне обчислення булевих операцій над геометричними об'єктами.

Параметри зон покриття, покриваючих об'єктів, налаштування метапараметрів алгоритмів для кожної серії подано у Додатку Б.

2.4.1 Результати обчислювальних експериментів

У цьому підрозділі подано результати обчислювальних експериментів для кожного з чотирьох ройових алгоритмів у трьох серіях з різною кількістю покривальних об'єктів.

Серія 1: $n = 20$ покривальних об'єктів (рисунок 2.4).

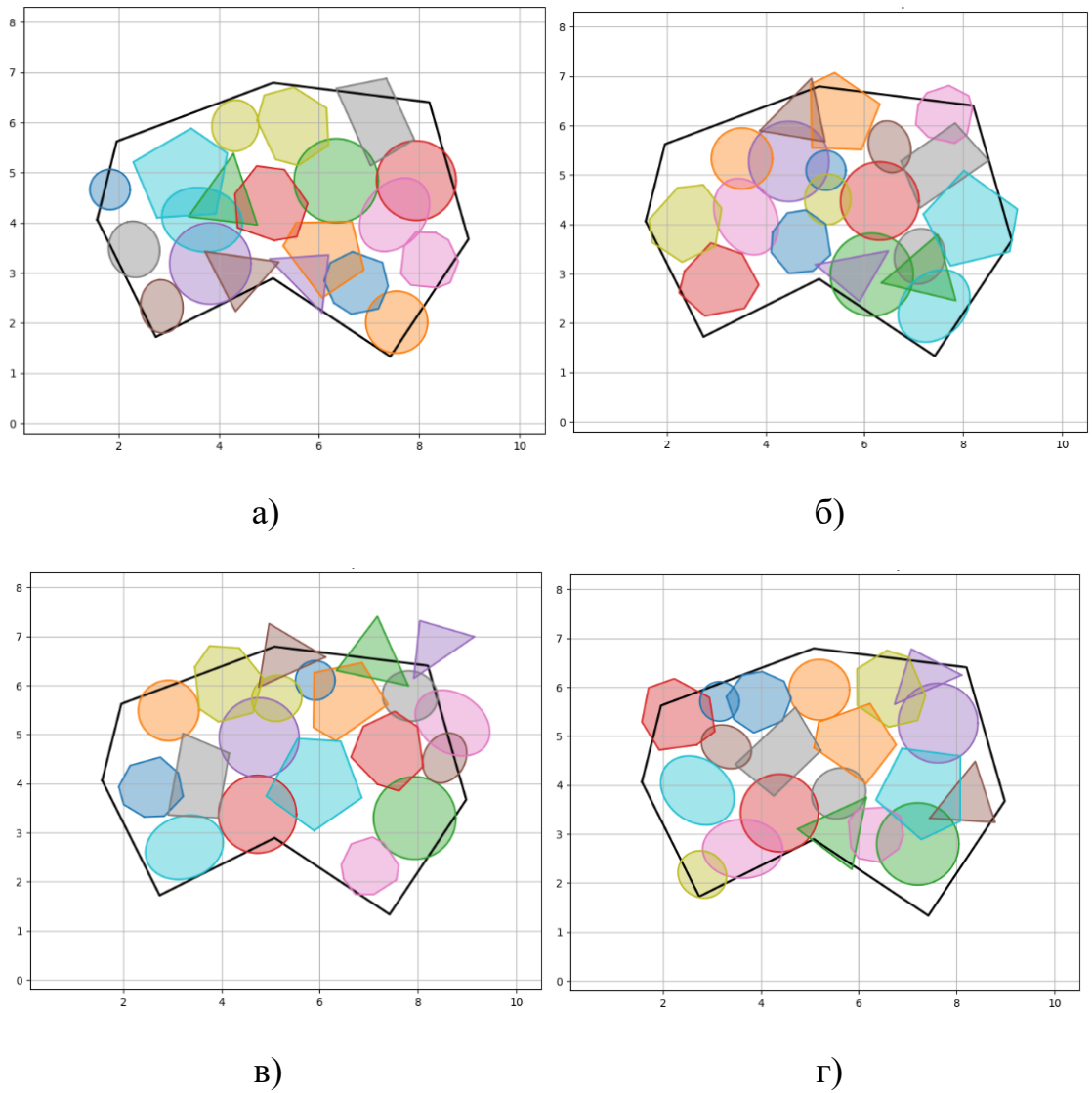


Рисунок 2.4 – Результат покриття серії 1: а) PSO; б) FSS; в) FA; г) ABC

Зведена таблиця результатів за серією 1 (таблиця 2.5):

Таблиця 2.5 – Результати експериментів за серією 1

Алгоритм	Покриття (%)	Час виконання (с)
PSO	77.4	11.3
FSS	78.7	15.8
FA	75.6	10.2
ABC	79.9	12.5

Серія 2: $n = 40$ покривальних об'єктів. Результати подано на рисунку 2.5.

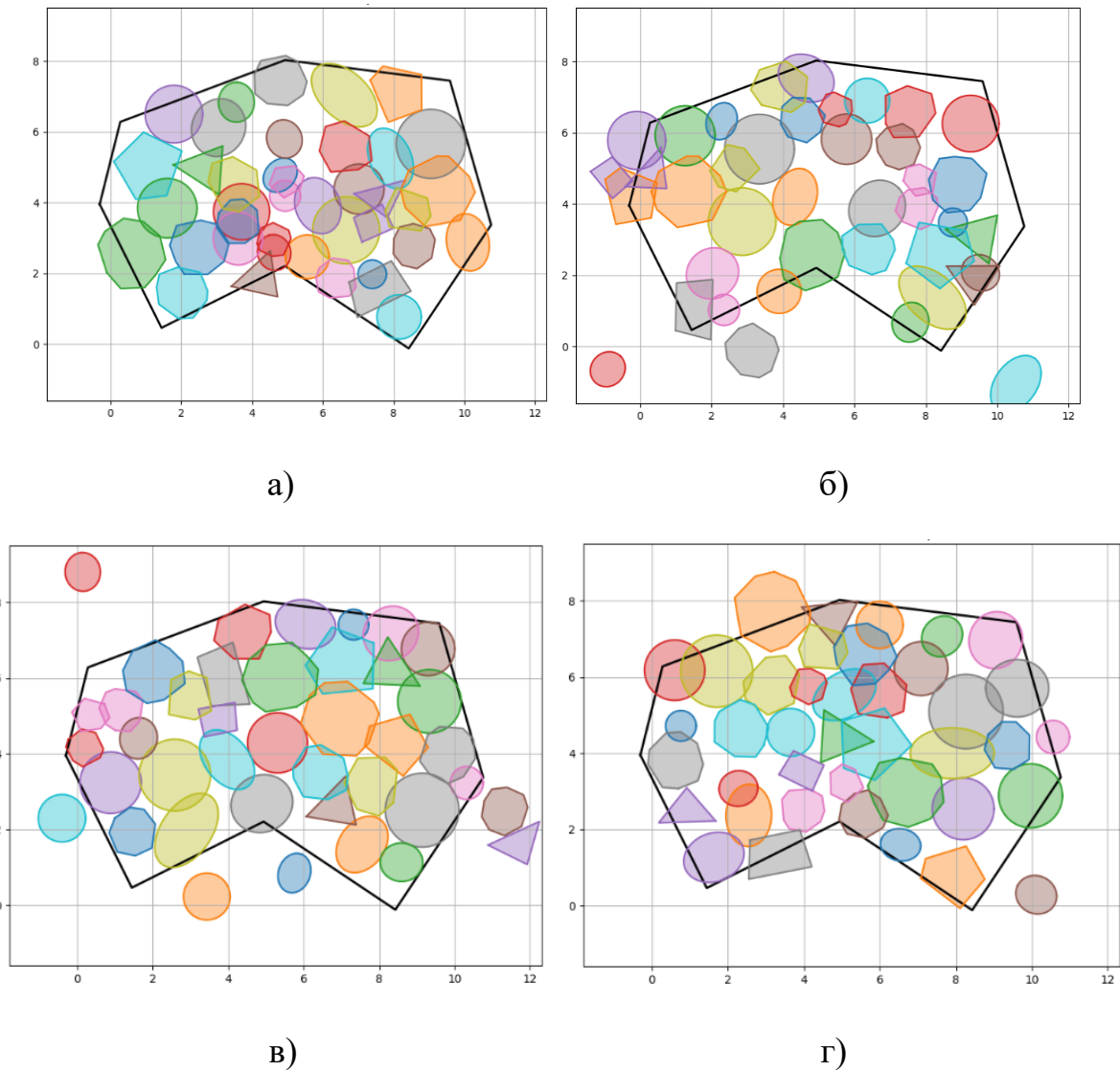


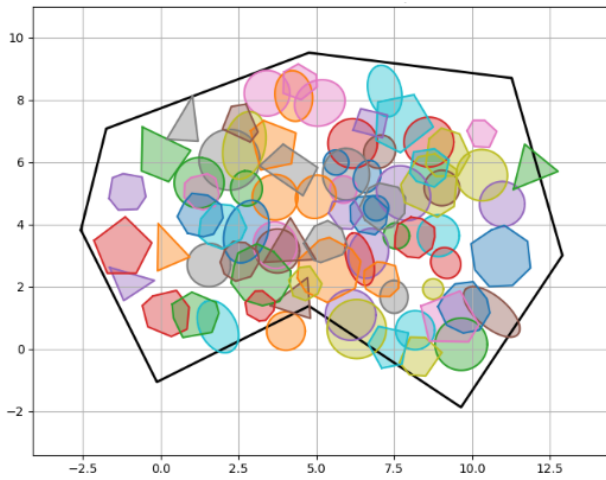
Рисунок 2.5 – Результат покриття серії 2: а) PSO; б) FSS; в) FA; г) ABC

Результати серії 2 також відображено у зведеній таблиці (таблиця 2.6):

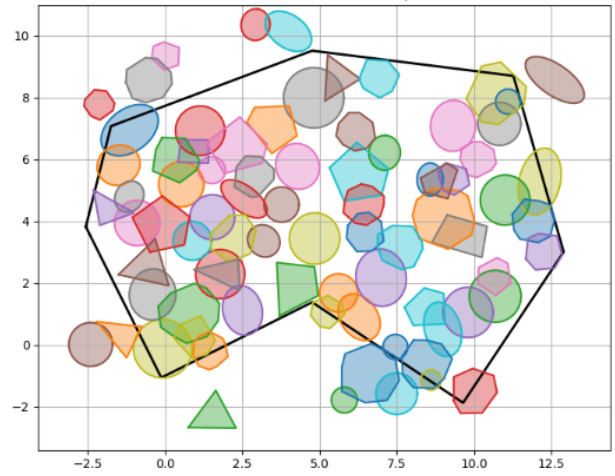
Таблиця 2.6 – Результати експериментів за серією 2

Алгоритм	Покриття (%)	Час виконання (с)
PSO	75.8	47.1
FSS	70.3	54.8
FA	79.3	29.1
ABC	75.4	57.4

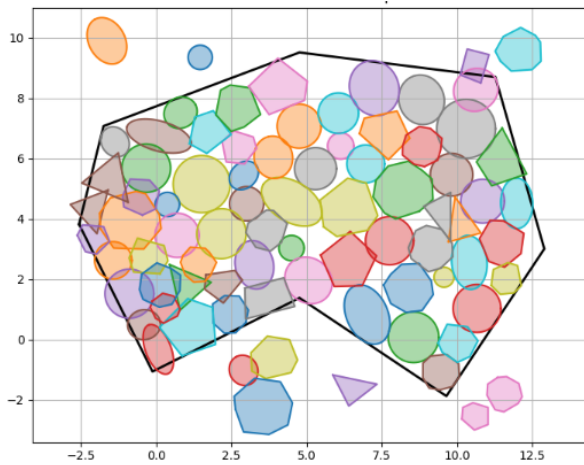
Серія 3: $n = 80$ покривальних об'єктів. Результати подано на рисунку 2.6.



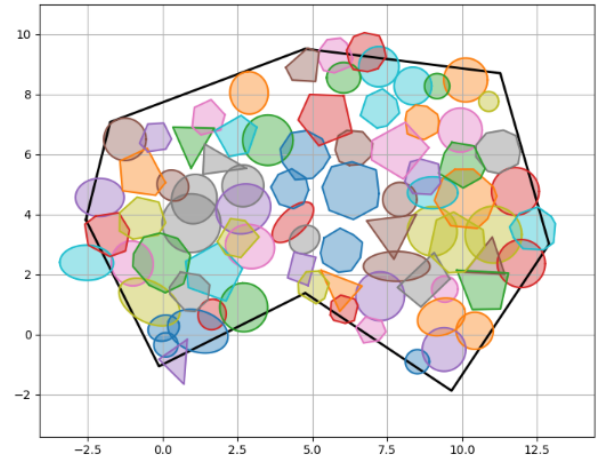
а)



б)



в)



г)

Рисунок 2.6 – Результат покриття серії 3: а) PSO; б) FSS; в) FA; г) ABC

Результати серії 3 також відображено у зведеній таблиці (таблиця 2.7):

Таблиця 2.7 – Результати експериментів за серією 3

Алгоритм	Покриття (%)	Час виконання (с)
PSO	65.1	410.4
FSS	64.8	514.6
FA	80.1	130.7
ABC	77.2	345.4

Як видно з графіків та результатів, чим більше розмірність пошуку – тим гірше справляються алгоритми. Варто окремо зазначити роботу алгоритму FA на великих розмірностях. Хоча у таблиці 2.6 результат роботи алгоритму набагато кращий за конкурентів, його стабільність є сумнівною. Розглянувши детальніше серії 1-3, а також після проведення великої кількості експериментів, було з'ясовано, що середнє відхилення відсотку покриття між експериментами для PSO, FSS та ABC приблизно дорівнює 5-10%. В той же час середнє відхилення для FA може сягати 20-25%, тобто в одному експерименті покриття може дорівнювати 60%, в іншому – 80%. Це обумовлено високою чутливістю до параметру γ , що робить алгоритм складнішим у підлаштуванні до кожної окремої задачі та ненадійним для задач покриття різного характеру. Далі буде наведено більш детальний порівняльний аналіз алгоритмів.

2.4.2 Порівняльний аналіз ройових алгоритмів

Для систематизації властивостей розглянутих ройових алгоритмів та обґрунтування вибору конкретного методу залежно від структурних характеристик задачі покриття проведено порівняльний аналіз за ключовими критеріями ефективності, стійкості та придатності. Рекомендовані значення метапараметрів кожного алгоритму з урахуванням специфіки задач покриття наведено в таблиці 2.8.

Таблиця 2.8 – Метапараметри ройових алгоритмів для задач покриття

Алгоритм	Метапараметр	Рекомендоване значення	Геометричний зміст у задачах покриття
PSO	Розмір рою N_p	$15n - 20n, N_p \leq 1500$	Різноманітність траєкторій у 50–200 змінних
	Інерція w	0.72-0.74	Баланс дослідження/експлуатації конфігурацій
	Когнітивний c_1	1.496	Тяжіння до індивідуально найкращих конфігурацій
	Соціальний c_2	1.496	Групування навколо глобально найкращого покриття

Продовження таблиці 2.8

Алгоритм	Метапараметр	Рекомендоване значення	Геометричний зміст у задачах покриття
FSS	Розмір зграї	$15n - 25n$	Стійке колективне переміщення у складній геометрії
	$step_ind$	$0.05 - 0.15 \times diam$	Радіус локального зондування конфігурацій
	$step_vol$	$0.01 - 0.05 \times diam$	Швидкість стиснення/розширення хмари конфігурацій
FA	Привабливість β_0	0.8-1.2	Максимальна сила притягання кращих конфігурацій
	Поглинання γ	0.005-1.0	Дальність взаємодії між конфігураціями
	Стохастичний α	$0.15 - 0.25 \times L$	Амплітуда збурень для виходу з вироджень
ABC	Розмір колонії	$15n - 25n$	Кількість паралельно досліджуваних конфігурацій
	Амплітуда модифікації	$0.1 - 0.3 \times diam$	Масштаб збурень при локальному пошуку
	Поріг limit	$limit \approx 0.6 * max_iters$	Швидкість відмови від непродуктивних конфігурацій

Наведені в таблиці 2.8 метапараметри визначають поведінку кожного алгоритму в просторі конфігурацій покриття. Для більш детального порівняння сильних і слабких сторін алгоритмів та їхньої придатності до різних типів задач покриття систематизовано якісні характеристики (таблиця 2.9).

Таблиця 2.9 – Порівняльні властивості ройових алгоритмів у задачах покриття

Алгоритм	Сильні сторони	Слабкі сторони	Придатність у задачах покриття
PSO	Висока швидкість збіжності; ефективний обмін інформацією; простота; добра масштабованість	Передчасне застрягання у локальних екстремумах; втрата різноманітності популяції	Оптимальний для задач помірної складності; найшвидше формування щільних покриттів
FSS	Добре збереження різноманітності; стійкість при складній геометрії; адаптивне стиснення–розширення	Вища обчислювальна вартість ітерації; потребує ретельного налаштування кроків	Найкращий для невіпуклих областей з забороненими зонами та вузькими проходами

Продовження таблиці 2.9

Алгоритм	Сильні сторони	Слабкі сторони	Придатність у задачах покриття
FA	Природне фокусування; плавний перехід між кластерами; проста структура; немає глобального лідера	Висока чутливість до γ	Ефективний для мультимодальних задач з аналітичною оцінкою площі
ABC	Добре збереження різноманітності; вихід з локальних мінімумів; мінімум параметрів	Повільніша локальна збіжність; більше ітерацій; чутливість до limit	Підходить для задач з великою кількістю обмежень; ефективний з гібридизацією BFGS

Окрім якісних характеристик, важливим критерієм вибору алгоритму є його обчислювальна складність та здатність масштабуватися зі зростанням кількості покриваючих об'єктів. Відповідні кількісні оцінки наведено в таблиці 2.10.

Таблиця 2.10 – Обчислювальна складність та масштабованість

Алгоритм	Складність ітерації	Масштабованість	Стійкість	Потреба в гібридизації
PSO	$O(N)$	До 150–200 змінних	Середня	Висока – для подолання локальних мінімумів
FSS	$O(N)$	До 200–250 змінних	Висока	Помірна – для точного локального уточнення
FA	$O(N^2)$	До 300–400 змінних	Низька	Помірна – компенсація повільної збіжності
ABC	$O(N)$	До 200–250 змінних	Висока	Висока – прискорення локальної збіжності

Тут N – кількість внутрішніх ітерацій циклу алгоритму. Але варто зазначити, що обчислювальна складність ітерації для, наприклад, PSO, сильно відрізняється від такої для FA. Справжнім показником обчислювальної складності для порівняння алгоритмів має бути кількість викликів підрахунку цільової функції (таблиця 2.11):

Таблиця 2.11 – Кількість викликів цільової функції за ітерацію

Алгоритм	Виклик цільової ф-ції
PSO	N
FSS	2N
FA	N
ABC	2N + scout

В такій оцінці однозначно ліduють PSO та FA, але завдяки дуже ефективному використанню кожного виклику фітнесу FA показує себе набагато краще за інші алгоритми при високій розмірності пошуку.

Теоретичні оцінки складності та масштабованості (таблиця 2.10) підтверджено результатами обчислювальних експериментів на задачах зростаючої розмірності.

Як видно з порівняльного аналізу, кожен з розглянутих ройових алгоритмів має характерні переваги та обмеження, що визначають його придатність для різних типів задач покриття. PSO забезпечує найшвидшу збіжність для задач із відносно гладкою геометрією цільової функції, проте втрачає ефективність при наявності заборонених зон та складної топології області. FSS демонструє найвищу стійкість при навігації у фрагментованих допустимих областях. FA характеризується природним механізмом фокусування на перспективних зонах без необхідності явного глобального лідера, дуже ефективно використовує кожен виклик фітнесу завдяки одночасному тяжінню до всіх яскравіших. ABC вирізняється мінімальною кількістю параметрів та ефективним механізмом глобального дослідження.

Спільною рисою всіх чотирьох алгоритмів є обмежена здатність до точного локального уточнення розв'язків. Стохастична природа ройових методів забезпечує ефективний глобальний пошук, проте не може конкурувати з детермінованими градієнтними методами у задачах тонкого налаштування параметрів поблизу виявленого оптимуму.

Висновки до розділу 2

У другому розділі розроблено та досліджено методичний апарат для розв'язання задачі максимального геометричного покриття: методи обчислення цільової функції (площі покриття) та ройові алгоритми оптимізації розміщення покриваючих об'єктів довільної форми. Проведено теоретичний аналіз обчислювальної складності та точності методів, а також обчислювальні експерименти для підтвердження теоретичних оцінок та порівняння ефективності алгоритмів на задачах різної розмірності. Основні результати розділу полягають у наступному.

По-перше, проаналізовано та систематизовано п'ять методів обчислення площі конфігурації покриття, що утворюють спектр від найшвидших з низькою точністю (метод мінімальних описаних прямокутників, $O(n)$) до найточніших з високою обчислювальною вартістю (бібліотека Shapely, $O(nk \log(nk))$). Метод Монте-Карло з просторовою індексацією R-Tree, що забезпечує складність $O(N \log n)$, визначено як основний метод обчислення цільової функції для багаторазового виклику в циклі оптимізації завдяки оптимальному компромісу між точністю та обчислювальною вартістю [115, 116]. Експериментально підтверджено, що метод Монте-Карло з просторовою індексацією R-Tree забезпечує похибку в межах 3% відносно точних обчислень бібліотекою Shapely при асимптотичній складності $O(N \log n)$ замість $O(N \cdot n)$. Сформульовано кількісні рекомендації щодо вибору методу залежно від етапу оптимізації (глобальний пошук, локальне уточнення, верифікація) та кількості покриваючих об'єктів.

По-друге, чотири ройові метаевристики обчислювального інтелекту, що реалізують агентний підхід, а саме PSO, FSS, FA та ABC, адаптовано до задачі максимального геометричного покриття, де кожен агент рою кодує повну конфігурацію розміщення n покриваючих об'єктів у просторі \mathbb{R}^{3n} [115, 50]. Для

кожного алгоритму обґрунтовано вибір метапараметрів з урахуванням специфіки задачі покриття та визначено рекомендовані діапазони їх значень.

По-третє, проведено обчислювальні експерименти на трьох серіях задач зростаючої складності ($n = 20, 40, 80$ покриваючих об'єктів, що відповідає розмірності простору параметрів 60, 120 та 240 змінних відповідно). Порівняльний аналіз показав, що ройові алгоритми досягають площі покриття від 65% до 80% залежно від складності задачі: при $n = 20$ найкращий результат демонструє ABC (79.9%), при $n = 40$ – FA (79.3%), при $n = 80$ – FA (80.1%). PSO забезпечує найшвидшу збіжність для задач з відносно гладкою геометрією, FSS демонструє найвищу стійкість при наявності заборонених зон, FA характеризується природним механізмом фокусування без глобального лідера, а ABC вирізняється ефективним механізмом виходу з локальних екстремумів. Водночас встановлено, що спільним обмеженням усіх розглянутих ройових алгоритмів є недостатня здатність до точного локального уточнення розв'язків поблизу знайденого оптимуму. Стохастична природа ройових методів забезпечує ефективний глобальний пошук, проте не може конкурувати з детермінованими методами оптимізації у задачах тонкого налаштування параметрів. Ця обставина обумовлює необхідність розробки гібридних алгоритмів, що поєднують глобальний ройовий пошук з локальною оптимізацією на основі градієнтних методів, чому присвячено наступний розділ дисертації.

РОЗДІЛ 3. ГІБРИДНІ ЕВОЛЮЦІЙНО-РОЙОВІ ТА НЕЙРОКЕРОВАНІ МЕТОДИ ОПТИМІЗАЦІЇ В ЗАДАЧАХ МАКСИМАЛЬНОГО ПОКРИТТЯ

3.1 Застосування методів локальної оптимізації в задачах максимального покриття

Перехід до гібридних алгоритмів, які є предметом дослідження цього розділу, потребує першочергового розгляду методів локальної оптимізації, що складають детерміновану компоненту гібридних схем. Методи глобальної оптимізації вже детально розглянуто у розділі 2. Тому в цьому підрозділі зосереджено увагу на градієнтних та квазіградієнтних методах локального пошуку, їхній адаптації до специфіки задачі неперервного максимального покриття та підходах до ефективного оцінювання градієнта цільової функції.

Необхідність застосування локальної оптимізації у задачах максимального покриття зумовлена кількома факторами. По-перше, цільова функція, що визначає площу покриття області сімейством геометричних об'єктів, є неперервною відносно параметрів розміщення, що створює передумови для використання градієнтних методів [31]. По-друге, навіть незначні зміни позицій та орієнтацій об'єктів можуть суттєво впливати на площу покриття, особливо в околі меж перекриття об'єктів. По-третє, як буде показано далі у цьому розділі, поєднання глобальних ройових методів із локальним градієнтним пошуком забезпечує суттєве підвищення якості рішень.

Загальна схема розв'язання задачі неперервного максимального покриття включає такі етапи [31]: формування вхідних даних щодо області покриття та сімейства покривальних об'єктів; визначення узагальнених змінних конфігурації покриття; обчислення міри покритої частини області; вибір методу локальної оптимізації; оцінювання градієнта цільової функції з урахуванням геометричних властивостей задачі; вибір методу глобальної оптимізації. Перші два етапи було

висвітлено у розділі 2, а останній – реалізується через ройові алгоритми. Решта етапів є предметом розгляду цього підрозділу.

3.1.1 Поняття та оцінка градієнту

Ключовою передумовою застосування градієнтних методів є можливість оцінювання градієнта цільової функції $F(U)$, яка визначає площу покритої частини області D сімейством покривальних об'єктів $\Omega_1(u_1), \dots, \Omega_n(u_n)$. Аналітичне обчислення градієнта є надзвичайно складною задачею, оскільки вимагає аналітичного опису межі складного геометричного об'єкта [31]. Межа об'єднання довільних геометричних фігур може складатися з великої кількості криволінійних та прямолінійних сегментів, що робить отримання замкнутого аналітичного виразу для похідної площі за параметрами розміщення практично неможливим. Тому на практиці застосовують чисельне оцінювання за допомогою скінченних різниць першого порядку.

Нехай вектор параметрів розміщення об'єкта Ω_i має вигляд $u_i = (x_i, y_i, \varphi_i)$. Для оцінки часткової похідної по k -й компоненті вектора u_i надамо їй приріст δ та сформуємо збурений вектор:

$$\tilde{U} = (\hat{u}_1, \dots, \hat{u}_{i-1}, u_{i1}, \dots, u_{ik} + \delta, \dots, u_{i3}, \hat{u}_{i+1}, \dots, \hat{u}_n) \quad (3.1)$$

Найбільш прямий підхід до оцінювання приросту цільової функції полягає у повному перерахунку площі покриття:

$$\Delta F(\hat{U}) = F(\tilde{U}) - F(\hat{U}) \quad (3.2)$$

Цей підхід є універсальним, проте повне оцінювання градієнта потребує $3n$ обчислень площі покриття, що при великих n є обчислювально витратним.

Для зменшення витрат у [31] запропоновано підхід на основі симетричної матриці площ попарних перетинів:

$$M = [\mu_{ij}](n + 1) \times (n + 1), \mu_{ij} = S(\Omega_i(\hat{u}_i) \cap \Omega_j(\hat{u}_j)), \quad (3.3)$$

де $i, j \in \{0, 1, \dots, n\}$;

індекс 0 відповідає області покриття D ;

$\Omega_0 \equiv D$;

матриця M є сильно розрідженою.

Використовуючи її, приріст цільової функції обчислюється ефективніше:

$$\Delta^i_k F(\hat{U}) = \sum_{j=0}^n [S(\Omega_i^k(\tilde{U}) \cap \Omega_j(\hat{u}_j)) - \mu_{ij}], \quad (3.4)$$

де підсумовування виконується лише по тих j , для яких $\mu_{ij} \neq 0$;

$\Omega_i^k(\tilde{U})$ – об'єкт Ω_i зі збуреною k -ю компонентою вектора параметрів.

Замість повного перерахунку $F(U)$ обчислюються лише площі попарних перетинів збуреного об'єкта з сусідами. Експерименти [31] підтверджують ефективність: при $n = 100$ час за формулою (3.2) становить ~ 0.08 с, за (3.4) – ~ 0.004 с; при $n = 500$ – 0.36 с проти 0.011 с, тобто прискорення на порядок.

Часткова похідна апроксимується скінченною різницею:

$$\partial F / \partial u_{ik} \approx \Delta^i_k F(\hat{U}) / \delta \quad (3.5)$$

де $\delta > 0$ – крок скінченної різниці.

Вибір параметра δ є важливим: занадто мале значення призводить до числових похибок округлення, а занадто велике – до неточної апроксимації похідної. На практиці δ обирається у діапазоні $10^{-3} - 10^{-5}$ відносно характерного розміру об'єктів покриття.

3.1.2 Методи локальної оптимізації з використанням градієнтів

На основі розглянутих підходів до оцінювання градієнта застосовуються квазіградієнтні методи локальної оптимізації. Основним методом локального пошуку у гібридних алгоритмах обрано BFGS (Broyden–Fletcher–Goldfarb–Shanno) – класичний метод неперервної оптимізації, що будує апроксимацію оберненої матриці Гессе ітеративно, використовуючи лише градієнт [54]. На кожній ітерації t визначається напрям пошуку та оновлюються параметри:

$$d_t = H_t \nabla F(U_t), \quad (3.6)$$

$$U_{t+1} = U_t + \alpha_t d_t, \quad (3.7)$$

де H_t – апроксимація оберненої матриці Гессе;

α_t – крок лінійного пошуку.

Вибір BFGS зумовлений суперлінійною збіжністю, потребою лише у першому порядку похідних та стійкістю до неточностей скінченно-різницевих апроксимацій [31, 54].

Практична реалізація здійснюється засобами пакета SciPy (модуль `scipy.optimize`), а обчислення площі покриття – бібліотекою Shapely [31, 53]. Початкові параметри розміщення генеруються рівномірно: координати центрів – у межах області, кути повороту – на інтервалі $(0, \pi)$. Найпростішою стратегією глобального пошуку є схема мультистарту: багаторазова генерація початкових конфігурацій з подальшою локальною оптимізацією та вибором найкращого результату. Однак більш ефективним є використання BFGS у складі гібридних алгоритмів, де початковою точкою слугує поточна найкраща конфігурація ройового алгоритму, що суттєво підвищує якість стартової точки.

Результати експериментів [31] для сімейства еліпсів, що покривають прямокутну область, свідчать про таке: для $n = 30$ час локальної оптимізації становить 22 секунди, для $n = 50 - 51$ секунд, для $n = 75 - 79$ секунд, для $n = 100 - 103$ секунди. Залежність має квадратичний характер, що пояснюється зростанням розмірності простору ($3n$) та кількості ненульових елементів матриці попарних перетинів.

Показовими є граничні випадки. При $n = 50$ BFGS знаходить конфігурацію пакування [39, 40] (еліпси не перетинаються, повністю всередині області), а при $n = 75$ – конфігурацію покриття (кожна точка області належить хоча б одному еліпсу). Обидва випадки є глобальними розв’язками, що підтверджує здатність BFGS знаходити високоякісні оптимуми.

Разом з тим, при зростанні n ландшафт цільової функції стає мультимодальним, і ймовірність потрапляння у локальний оптимум зростає. Це обґрунтовує необхідність інтеграції BFGS із глобальними ройовими методами.

Підсумовуючи, метод BFGS є ефективним інструментом для тонкого налаштування конфігурацій покриття в околі перспективних точок простору параметрів. Ефективність його застосування суттєво підвищується завдяки запропонованому у [31] підходу до оцінювання градієнта через матрицю попарних перетинів, що зменшує обчислювальні витрати на порядок. Інтеграція BFGS як локальної компоненти у складі гібридних ройових алгоритмів, яка детально розглядається у підрозділі 3.2, дозволяє поєднати глобальні пошукові можливості ройових метаевристик із високою точністю градієнтного локального пошуку.

3.2 Гібридні алгоритми в задачах максимального покриття

Як було встановлено у підрозділі 3.1 та у розділі 2, ройові алгоритми забезпечують ефективний глобальний пошук, проте спільною рисою всіх чотирьох алгоритмів (PSO, FSS, FA, ABC) є обмежена здатність до точного

локального уточнення розв'язків. Для подолання цього обмеження запропоновано гібридну систему оптимізації, що поєднує глобальний ройовий пошук із детермінованою локальною оптимізацією BFGS. Стохастична природа оновлення позицій обмежує здатність ройових методів до тонкого градієнтного налаштування конфігурацій, що призводить до втрати 5-10% потенційної площі покриття порівняно з теоретичним оптимумом. Ця фундаментальна обмеженість зумовлює необхідність розвитку гібридних підходів.

Меметичні алгоритми, вперше запропоновані Москато у 1989 році, являють собою клас гібридних метаевристик, які поєднують глобальний стохастичний пошук популяційних методів із локальною детермінованою оптимізацією. Назва «меметичний» походить від концепції «мему» Докінза, що означає одиницю культурної інформації, яка передається та вдосконалюється в процесі передачі. Загальні принципи меметичних алгоритмів докладно описано в літературі [81, 82].

У меметичній схемі ройовий алгоритм виступає глобальним движком, що досліджує простір конфігурацій, виявляє перспективні області та підтримує різноманітність популяції. На окремих ітераціях до найкращих конфігурацій застосовується оператор локального пошуку. Цей оператор виконує кілька кроків локальної оптимізації, коригуючи позиції та орієнтації об'єктів у градієнтному стилі для збільшення покриття з урахуванням геометричних обмежень через штрафний доданок.

З геометричної точки зору глобальні оновлення рою відповідають грубим переміщенням усієї конфігурації у просторі параметрів, зсуваючи та обертаючи об'єкти таким чином, щоб виявляти нові кандидатні патерни покриття. Локальний пошук, навпаки, виконує тонке налаштування в малих околицях перспективних конфігурацій. Це можна інтерпретувати як двомасштабний процес: ройові алгоритми сканують «ландшафт» можливих покриттів із грубою

роздільною здатністю, тоді як локальний пошук «полірує» найкращі виявлені піки поверхні покриття.

Частота та інтенсивність активації локального пошуку визначають баланс між дослідженням і експлуатацією в меметичному алгоритмі. Якщо локальний пошук застосовується занадто часто, популяція може передчасно сійтися до малої області простору конфігурацій. Якщо його застосовувати надто рідко, алгоритм поводить себе подібно до чистого ройового методу й може не досягти покриттів найвищої якості. Завдяки ретельній інтеграції локального уточнення в динаміку рою меметичні алгоритми використовують взаємодоповнюючі сильні сторони обох компонентів.

Як локальний детермінований оптимізатор у меметичних схемах використовується метод BFGS, детально описаний у підрозділі 3.1.2. Градієнт цільової функції оцінюється чисельно за допомогою ефективного підходу на основі матриці попарних перетинів (формула (3.4)), що забезпечує прийнятну обчислювальну вартість навіть для задач великої розмірності.

Загальна обчислювальна вартість меметичного підходу визначається сумою витрат на ройовий пошук та локальну оптимізацію. Для типових задач покриття з $n = 10 - 100$ об'єктами витрати на ройовий пошук становлять 60–80 відсотків загального часу, а витрати на BFGS – 20-40 відсотків. При цьому гібридна схема забезпечує результати, які перевершують як чистий ройовий пошук, так і мультістартову локальну оптимізацію при однаковому обчислювальному бюджеті.

3.2.1 Переваги меметичного підходу

Комбінація глобального та локального пошуку забезпечує суттєве підвищення якості розв'язків. За даними чисельних досліджень, інтеграція BFGS додає від 1 до 3 відсотків до площі покриття у складних конфігураціях. Хоча ця величина може здаватися незначною, у задачах із десятками покривальних

об'єктів та невипуклою геометрією області навіть такий приріст означає суттєве практичне покращення якості сервісного охоплення.

Гібридна схема є універсальною та може бути застосована до будь-якого з розглянутих ройових алгоритмів: PSO→PSO-BFGS, FSS→FSS-BFGS, FA→FA-BFGS, ABC→ABC-BFGS – без зміни їхньої базової структури. Це забезпечує модульність підходу: дослідник може обрати найбільш відповідний ройовий алгоритм для конкретної задачі та доповнити його локальним оптимізатором без необхідності перепроєктування алгоритмічної архітектури.

Періодичний (а не постійний) запуск локального пошуку забезпечує оптимальний баланс між обчислювальними витратами та якістю розв'язків. Запускати локальний пошук на кожній ітерації є обчислювально дорогим, проте періодичне застосування BFGS до найкращих конфігурацій забезпечує значне покращення при помірних додаткових витратах.

Локальний пошук ефективно розкриває дрібні локальні пастки, які чисті ройові методи не здатні подолати через свою стохастичну природу. У задачах покриття це проявляється у тонкому підлаштуванні позицій та орієнтацій об'єктів, що усуває невеликі непокриті ділянки та зменшує надмірне перекриття між сусідніми об'єктами.

Висока масштабованість меметичного підходу забезпечує його придатність для задач різної складності. Навіть при кількості об'єктів $n \approx 100$ (що відповідає 300 змінним оптимізації) час розв'язання залишається прийнятним. Стійкість до заборонених зон та невипуклої геометрії забезпечується гнучким глобальним пошуком ройового компонента, а точність фінальних конфігурацій – локальним градієнтним уточненням.

3.2.2 Інтеграція з ройовими методами

Розглянемо конкретні рекомендації щодо інтеграції локального оптимізатора BFGS з кожним із чотирьох ройових алгоритмів.

PSO+BFGS. Глобальний PSO швидко формує щільні кластери розміщення, проте може «застрягати» у локальних екстремумах через сильне притягання до глобально найкращої позиції. Додавання BFGS через кожні кілька поколінь суттєво підвищує площу покриття та покращує стабільність результатів. Рекомендована частота виклику BFGS становить кожні 3-5 поколінь для задач невеликої розмірності ($n < 30$) та кожні 5-8 поколінь для задач більшої розмірності. При кожному виклику BFGS застосовується до найкращої глобальної конфігурації та до 2-3 найкращих індивідуальних конфігурацій.

FSS+BFGS. FSS добре зберігає різноманіття популяції та є ефективним при складній геометрії завдяки механізму адаптивного стиснення-розширення зграї. Меметичний FSS забезпечує найвищу надійність серед усіх гібридних комбінацій: навіть у невивулених областях та при наявності багатьох заборонених зон покращення площі покриття становить стабільно 1-2 відсотки. Рекомендована частота виклику BFGS становить кожні 3-5 поколінь для $n \leq 50$ та кожні 8-12 поколінь для $n > 50$, оскільки FSS самостійно забезпечує кращу локальну експлуатацію.

FA+BFGS. FA добре працює у задачах, де оцінка площі покриття може бути виконана аналітично або напіваналітично, що суттєво зменшує обчислювальну вартість оцінки градієнтів BFGS. Меметична модифікація FA компенсує повільнішу збіжність, характерну для алгоритму світлячків, та дозволяє формувати щільніші кластери покриття. Відсутність глобального лідера у FA забезпечує природне фокусування на кількох перспективних зонах одночасно, що добре узгоджується з локальним уточненням BFGS. Рекомендована частота виклику BFGS становить кожні 4-7 поколінь.

ABC+BFGS. У ABC фази зайнятих бджіл та спостерігачів забезпечують експлуатацію перспективних джерел, проте без градієнтної інформації важко точно покращити положення об'єктів. Інтеграція BFGS перетворює ABC на ефективний комбінований метод, що поєднує три компоненти: розвідку з

пошуком нових конфігурацій, спостереження з посиленням уточненням, та BFGS з високоточною локальною оптимізацією. Узагальнені рекомендації щодо налаштування меметичних схем систематизовано у таблиці 3.1.

Таблиця 3.1 – Параметри налаштування меметичних схем

Комбінація	Параметр пошуку	Рекомендоване значення	Обґрунтування застосування
PSO+BFGS	Частота виклику BFGS	Кожні 3–5 поколінь для $n < 30$; Кожні 5–8 поколінь для $n \geq 50$	У задачах покриття BFGS ефективно покращує локальні густі структури; часті виклики дозволяють компенсувати недоліки PSO у негладких областях
	Поріг зупинки BFGS	10^{-5} – 10^{-6} відносної зміни площі покриття	Гарантує локальну точність без надмірних обчислень
	Максимальна кількість ітерацій	80–120	Оптимальний компроміс між точністю та витратами
FSS+BFGS	Частота локального пошуку	3–5 для $n \leq 50$; 8–12 для $n > 50$	FSS добре досліджує складні області; BFGS потрібен для точного усунення геометричних конфліктів.
	Метод оцінки градієнтів	Кінцеві різниці або стохастичні оцінки (Monte Carlo $N \approx 105$)	Використовується при відсутності аналітичних похідних
FA+BFGS	Частота локальних оновлень	4–7	У FA надто частий локальний пошук може зруйнувати динаміку тяжіння; рекомендовано помірне використання
	Параметр γ (адаптивний)	Зменшення при наближенні до локальних максимумів	Дозволяє FA здійснювати глобальну розвідку з великим радіусом взаємодії на початкових етапах та переходити до точного локального уточнення на завершальних ітераціях
ABC+BFGS	Частота викликів локального пошуку	4–8	ABC виграє від інтенсивної локальної експлуатації, оскільки сам по собі має слабкий градієнтний компонент

З даних таблиці 3.1 видно, що алгоритми мають різні сфери застосування, а також свої сильні та слабкі сторони.

3.2.3 Порівняльні властивості ройових методів та їхніх меметичних модифікацій

Порівняльні властивості ройових методів та їхніх меметичних модифікацій узагальнено у таблиці 3.2.

Таблиця 3.2 – Порівняльні властивості меметичних модифікацій

Алгоритм	Сильні сторони	Слабкі сторони	Типові випадки застосування у задачах покриття
PSO+BFGS	Найшвидша збіжність серед гібридів; ефективний обмін інформацією; добра масштабованість; BFGS компенсує передчасне застрягання	Втрата різноманітності популяції при складній топології; чутливість до початкової ініціалізації рою	Оптимальний для задач помірної складності з гладкою геометрією; найшвидше формування щільних покриттів; задачі з $n < 50$ об'єктами
FSS+BFGS	Найвища надійність; добре збереження різноманітності; стійкість при складній геометрії; адаптивне стиснення–розширення	Вища обчислювальна вартість ітерації; потребує ретельного налаштування кроків	Найкращий для невипуклих областей із забороненими зонами та вузькими проходами; складна топологія допустимої множини
FA+BFGS	Природне фокусування на кількох зонах; плавний перехід між кластерами; відсутність глобального лідера запобігає згортанню	Квадратична складність ітерації; висока чутливість до параметра γ ; повільніша збіжність	Ефективний для мультимодальних задач з аналітичною або напіваналітичною оцінкою площі покриття
ABC+BFGS	Тришарова структура; добре збереження різноманітності (розвідниці); мінімум параметрів; BFGS перетворює на точний комбінований метод	Потреба у більшій кількості ітерацій; чутливість до порогу виснаження	Підходить для задач з великою кількістю обмежень розташування; ефективне дослідження обмеженого простору

Як видно з порівняльного аналізу, меметичні модифікації суттєво покращують характеристики базових ройових алгоритмів. Найбільш помітний ефект спостерігається для PSO+BFGS, де локальне уточнення компенсує основний недолік PSO – схильність до передчасної збіжності. Для FSS+BFGS покращення менш виражене кількісно, проте гібрид забезпечує найвищу надійність серед усіх комбінацій. ABC+BFGS демонструє найбільший відносний приріст якості завдяки тому, що локальний оптимізатор ефективно компенсує повільну локальну збіжність базового ABC.

3.2.4 Недоліки меметичного підходу

Основні недоліки меметичного підходу пов'язані із вищими обчислювальними витратами на оцінку градієнтів. Кожен виклик BFGS потребує багаторазового обчислення цільової функції для апроксимації часткових похідних методом скінченних різниць, що збільшує загальний час оптимізації на 20–40 відсотків порівняно з чистим ройовим пошуком.

Додаткова складність полягає у необхідності налаштування параметрів інтеграції: частоти виклику локального пошуку, глибини BFGS-оптимізації, вибору методу оцінки градієнтів та критеріїв зупинки. Неоптимальне налаштування цих параметрів може призвести або до надмірних обчислювальних витрат (при занадто частому виклику BFGS), або до недостатнього покращення якості (при занадто рідкому виклику).

У задачах із сильно негладкою цільовою функцією точність скінченно-різницевої апроксимації градієнта може бути недостатньою, що знижує ефективність BFGS. У таких випадках доцільно використовувати стохастичні оцінки градієнтів на основі Монте-Карло, що додатково збільшує обчислювальну вартість.

Однак у задачах максимального покриття гібридні схеми перевершують будь-який чистий алгоритм на 12-25% при тому ж обчислювальному бюджеті, що виправдовує додаткову складність реалізації.

3.2.5 Рекомендації щодо вибору гібридного ройового алгоритму

Узагальнена систематизація рекомендацій щодо вибору гібридного ройового алгоритму наведена у таблиці 3.3.

Таблиця 3.3 – Рекомендації щодо вибору гібридного ройового алгоритму

Структурні властивості задачі	Типові труднощі оптимізації	Гібридний метод	Обґрунтування застосування
Геометрично проста випукла область, невелика кількість об'єктів ($n < 30$)	Цілком гладка та помірно мультимодальна поверхня; швидка збіжність	PSO +BFGS	PSO забезпечує найшвидше формування щільних конфігурацій; BFGS додає точність локального уточнення
Невипукла область з забороненими зонами, складна топологія допустимої множини	Фрагментована допустима область; вузькі проходи між зонами; PSO може застрягати	FSS +BFGS	FSS стійкий до невивуклості; адаптивне стиснення–розширення дозволяє обходити заборонені зони; найвища надійність
Мультимодальна поверхня з багатьма локальними оптимумами приблизно рівної якості	Висока мультимодальність; ризик передчасної збіжності до субоптимуму	FA +BFGS	FA природно фокусується на кількох перспективних зонах одночасно; відсутність глобального лідера запобігає згортанню до одного піку
Задача з великою кількістю обмежень на розташування та орієнтацію об'єктів	Сильно обмежена допустима область; висока частка недопустимих конфігурацій	ABC +BFGS	Механізм розвідниць забезпечує ефективне дослідження обмеженого простору; мінімум параметрів спрощує адаптацію
Великомасштабна задача ($n > 50$ об'єктів, розмірність > 150)	Висока розмірність; велика обчислювальна вартість оцінки цільової функції	PSO +BFGS або FSS +BFGS	PSO масштабується добре при гладкій геометрії; FSS при складній; обидва потребують зрідженого виклику BFGS

Наведені рекомендації базуються на систематичному аналізі характерних властивостей кожного ройового алгоритму та його поведінки у різних геометричних постановках задачі покриття. Слід зазначити, що ці рекомендації мають орієнтовний характер, і остаточний вибір алгоритму доцільно підтверджувати чисельними експериментами для конкретної задачі. Тим не менш, наведена систематизація суттєво звужує простір пошуку ефективного методу та скорочує час налаштування оптимізаційного процесу.

Окремо зазначимо, що для всіх розглянутих алгоритмів кількість поколінь (ітерацій) не є внутрішнім параметром і слугує лише зовнішнім критерієм зупинки. У практичних задачах покриття доцільнішим є використання спільного часового бюджету для забезпечення коректної порівнянності різних методів при однакових обчислювальних витратах.

3.3 Порівняльний аналіз гібридних методів в задачах покриття

Наведемо серію чисельних експериментів, яка ілюструє застосовність запропонованого меметичного підходу до задач максимального покриття з покривальними об'єктами довільної форми. Головною метою експериментів є не отримання абсолютно найкращого розв'язку, а демонстрація ефективності та практичної придатності гібридного підходу порівняно з чистим ройовим методом при різних масштабах задачі.

Конфігурації експериментів повністю відповідає тій, що була описана у розділі 2. Покривальні об'єкти мають довільну форму: полігональні, кругові та еліптичні. Положення кожного об'єкта описується параметрами:

$$(x_i, y_i, \varphi_i), \quad (3.8)$$

де (x_i, y_i) – координати полюса;

φ_i – кут обертання.

Для кругових об'єктів кут обертання фіксується: $\varphi_i = 0$. Для кожної конфігурації параметрів формується об'єднання всіх покривальних об'єктів, після чого обчислюється площа перетину з областю покриття з використанням бібліотеки Shapely.

Параметри ройового алгоритму та часовий бюджет оптимізації є ідентичними тим, що використовувалися у розділі 2, що забезпечує коректну порівняльність результатів. Спільний часовий бюджет для всіх методів становить 10 хвилин. Кожен метод може завершитися раніше, якщо досягає власного критерію збіжності.

За результатами порівняльного аналізу, проведеного у розділі 2, серед чотирьох досліджених ройових алгоритмів (PSO, FSS, FA, ABC) найкращими кандидатами для побудови меметичних модифікацій є алгоритми:

- 1) PSO для серії 1 (20 покриваючих об'єктів);
- 2) FA для серії 2 (40 покриваючих об'єктів);
- 3) FSS для серії 3 (80 покриваючих об'єктів).

Для проведення експериментів з гібридними варіаціями було обрано один з них – PSO. Цей вибір обумовлений найшвидшою збіжністю PSO на ранніх стадіях оптимізації та найкращим балансом між глобальним дослідженням простору параметрів і локальним уточненням конфігурацій, що було продемонстровано у розділі 2. Крім цього, PSO характеризується більшою стійкістю до зміни гіперпараметрів за FA та більш простою реалізацією за FSS, а також його поширеними варіаціями у сучасній літературі.

Таким чином, у серії експериментів порівнюються дві стратегії: чистий PSO та гібридний PSO+BFGS. Параметри PSO відповідають рекомендаціям, наведеним у розділі 2: розмір рою $N_p = 20n$ частинок; інерційний коефіцієнт $w = 0.72$; коефіцієнти $c_1 = c_2 = 1.496$. Для гібриду PSO+BFGS частота виклику BFGS становить кожні 5 поколінь; критерій зупинки за приростом площі – менше 0.001%; максимальна кількість ітерацій BFGS – 80.

Для повноти порівняння також наводяться результати стратегії мультистарт + BFGS, де з множини випадкових початкових конфігурацій запускається лише локальна оптимізація BFGS з тими самими параметрами. Середній час одного запуску BFGS становить приблизно 10 с.

3.3.1 Серія експериментів

Для систематичного дослідження масштабованості гібридного підходу проведено серію з трьох експериментів, конфігурації яких відповідають серіям експериментів у розділі 2, деталі конфігурацій – у Додатку Б.

Результати для першої серії експериментів (рисунок 3.1):

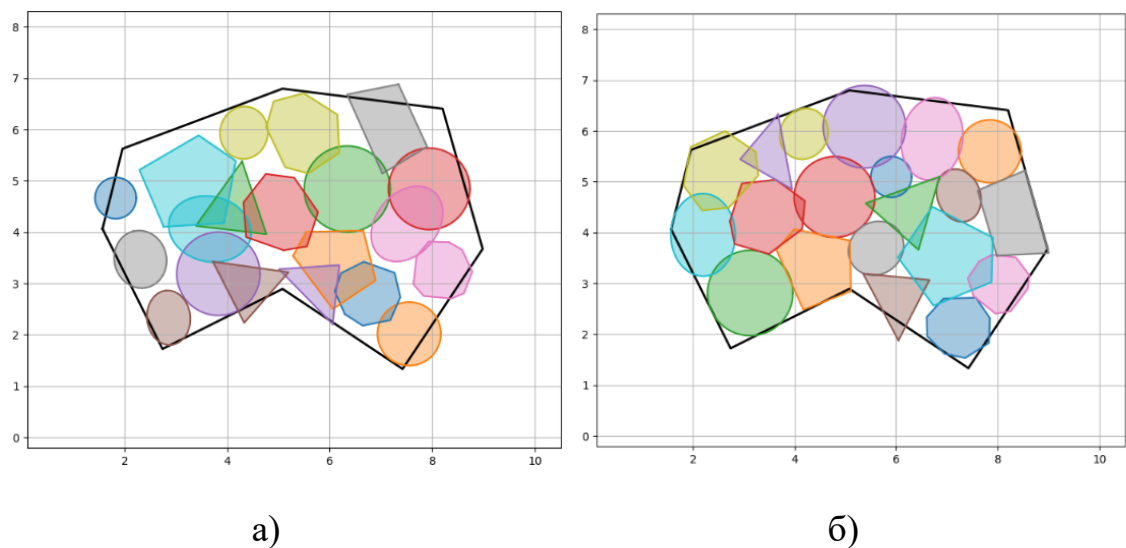


Рис. 3.1 – Результати покриття серії 1: а) PSO; б) PSO+BFGS

Результати експерименту демонструють, що навіть при відносно невеликій кількості об'єктів гібридний PSO+BFGS забезпечує помітне покращення якості покриття порівняно з PSO. Відсоток покриття при цьому для PSO+BFGS дорівнює 89.6%, при 77.4% для звичайного PSO, при різниці в часі $\sim +100\%$. Це кращий результат за будь-який з ройових алгоритмів, що були розглянуті у розділі 2. Локальний оптимізатор BFGS ефективно уточнює положення та

орієнтацію об'єктів у перспективних зонах, знайдених глобальним пошуком, що призводить до формування щільніших конфігурацій.

Результати для другої серії експериментів (рисунок 3.2):

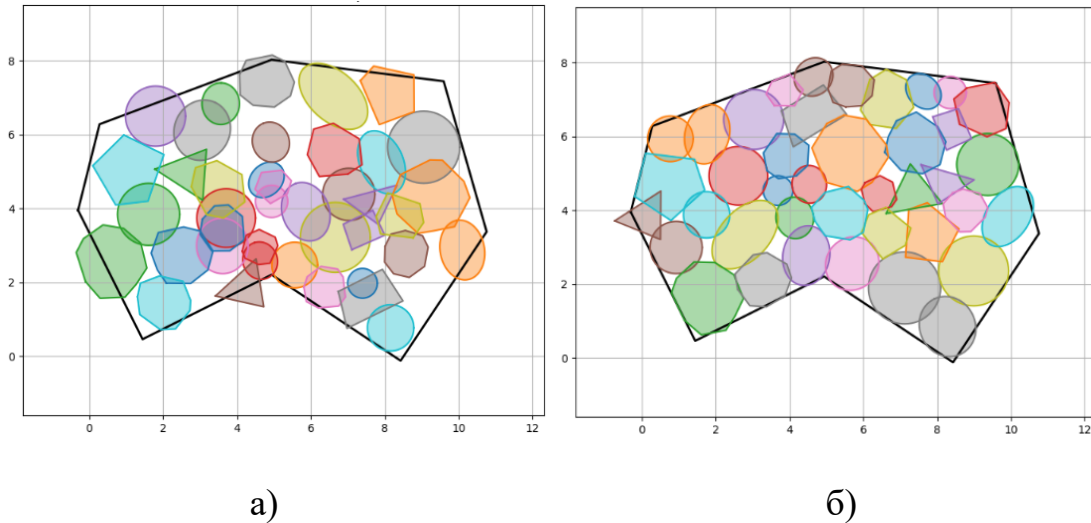


Рис. 3.2 – Результати покриття серії 2: а) PSO; б) PSO+BFGS

При $n = 40$ перевага гібридного підходу стає більш виразною. Звичайний PSO при подвоєній розмірності потребує значно більше поколінь для досягнення порівнянної якості покриття, тоді як PSO+BFGS компенсує цю складність за рахунок періодичного локального вдосконалення конфігурацій. PSO+BFGS забезпечує 89.9% покриття, тоді як PSO – всього лише 75.8%. При цьому слід також зазначити, що час виконання збільшився трохи менше, ніж вдвічі – 85.3 секунди порівняно з 47.1 секундами.

Гібридний метод формує помітно щільніші та геометрично узгодженіші конфігурації, автоматично «заповнюючи» вузькі проходи та виступи в невиконаній області. Це досягається завдяки тому, що BFGS здійснює точне градієнтне налаштування кожного об'єкта з урахуванням його геометричного оточення, що є принципово недосяжним для стохастичних ройових методів.

Також гібридний підхід забезпечує краще покриття за всі розглянуті ройові алгоритми у розділі 2.

Результати для третьої серії експериментів (рисунок 3.3):

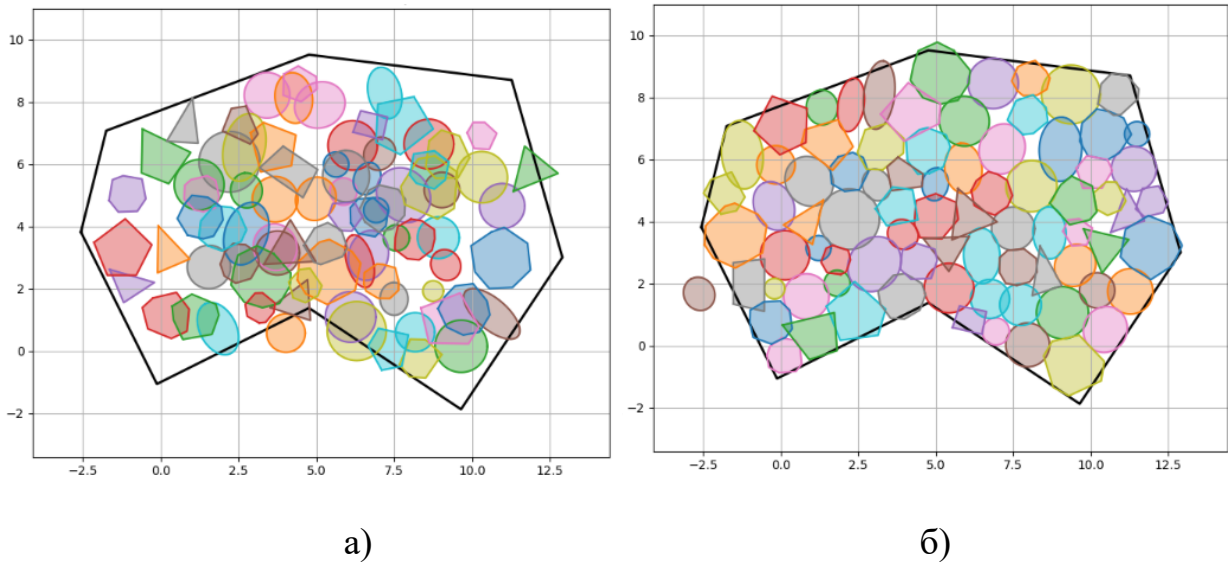


Рис. 3.3 – Результати покриття серії 3: а) PSO; б) PSO+BFGS

При великій кількості об'єктів покриття перевага гібридного підходу є найбільш суттєвою. Звичайний PSO за відведену кількість ітерацій отримує лише 65.1% покриття. Натомість PSO+BFGS ефективно використовує кожне знайдене перспективне рішення, локально вдосконалюючи його за допомогою BFGS, результат – 89.8% покриття. Саме при великих n найяскравіше виявляється двомасштабна природа меметичного підходу: глобальний ройовий пошук швидко знаходить перспективні регіони, а локальний оптимізатор забезпечує їх ефективну експлуатацію. При цьому час виконання збільшився пропорційно: 410.4 секунд для PSO та 610.1 секунди для гібридного підходу.

Результати порівняльних чисельних експериментів для трьох серій систематизовано у таблиці 3.4. Отримані результати демонструють, що гібридний метод PSO+BFGS стабільно перевершує загальний PSO та інші ройові алгоритми за площею покриття при всіх трьох масштабах задачі.

Таблиця 3.4 – Зведені результати порівняння PSO та PSO+BFGS

Алгоритм	Покриття (%)	Час виконання (с)	Серія
PSO	77.4	11.3	1
PSO + BFGS	89.6	20.1	1
PSO	75.8	47.1	2
PSO + BFGS	89.9	85.3	2
PSO	65.1	410.4	3
PSO + BFGS	89.8	610.1	3

Отримані результати демонструють, що гібридний метод PSO+BFGS стабільно перевершує загальний PSO та інші ройові алгоритми за площею покриття при всіх трьох масштабах задачі.

Більш того, за результатами багаточисельних експериментів, у гібридному підході кількість ітерацій PSO, а також популяція часток, можуть бути зменшені майже в 2 рази без втрати % покриття, що дозволить отримати високі показники покриття набагато швидше (приблизно -30-40% часу обчислення). Це зумовлено тим, що в гібридному підході PSO вирішує задачу грубої оцінки, що потребує меншу кількість обчислень. Крім цього, масштабування ройових алгоритмів (збільшення популяцій, кількості ітерацій) має сенс лише до порогу у 82-85% покриття. Для отримання кращого покриття ройовим алгоритмам в середньому потрібно значно більше часу та ітерацій, ніж гібридному підходу, або, як у випадку з PSO, алгоритм може застрягти у локальному оптимумі і ніколи не вийти з нього (залишаючи покриття у межах 80-83%).

Найшвидшу збіжність гібрид демонструє при малих n , де вартість обчислення площі є відносно невеликою і BFGS може виконати значну кількість локальних оптимізацій. При великих n перевага гібриду за якістю покриття зростає, оскільки чистий PSO не встигає повноцінно дослідити простір параметрів за відведену кількість ітерацій та скоріш за все застрягне у локальному оптимумі.

3.4 Нейронно-керовані гібридні методи максимізації покриття

У попередніх підрозділах розглянуто гібридні алгоритми, що поєднують глобальний ройовий пошук із локальною оптимізацією BFGS. Порівняльний аналіз (підрозділ 3.3) засвідчив, що меметичний підхід стабільно підвищує площу покриття $F(U)$ на 12–25% порівняно з чистими ройовими методами. Разом з тим, ефективність гібридних алгоритмів обмежена двома факторами: фіксованими гіперпараметрами ройових методів та статичним розкладом зростання штрафного коефіцієнта λ . Обидва фактори потребують ручного налаштування під кожну конкретну задачу, що ускладнює практичне застосування.

Інтеграція методів машинного навчання, зокрема нейронних мереж, у оптимізаційний процес відкриває принципово нові можливості подолання цих обмежень. У цьому підрозділі розглянуто два напрямки нейромережевої гібридизації: адаптивне керування штрафним коефіцієнтом λ та динамічне налаштування гіперпараметрів ройових алгоритмів. Для кожного напрямку визначено архітектуру нейронної мережі, функцію втрат та проаналізовано парадигми навчання, з обґрунтуванням вибору навчання з підкріпленням як найбільш перспективного підходу.

3.4.1 Нейронно-керований адаптивний механізм штрафних коефіцієнтів

У розділі 1 було сформульовано задачу безумовної оптимізації (формула (1.15)), яку для зручності наведено повторно:

$$\tilde{F}(U, \lambda) = F(U) - \lambda \cdot P(U) \rightarrow \max_U, \quad (3.9)$$

де $F(U) = S(D \cap C(U))$ – площа покриття;

$P(U)$ – штрафна функція за порушення обмежень (формула (1.14));

$\lambda > 0$ – штрафний коефіцієнт.

Ефективність цього підходу критично залежить від динаміки зміни λ у процесі оптимізації. Традиційний підхід полягає у фіксованому розкладі зростання λ :

$$\lambda_{t+1} = \alpha \cdot \lambda_t, \alpha \in [1.5; 3.0] \quad (3.10)$$

Проблема такого підходу – фундаментальна дилема: занадто повільне зростання λ призводить до тривалого перебування у недопустимій області, тоді як занадто швидке – до передчасного звуження пошуку. Оптимальна стратегія є різною для задач з різною кількістю, розміром та розташуванням заборонених зон Z_j .

Для забезпечення гладкості штрафної функції визначимо згладжену функцію порушення обмежень для кожного об'єкта $\Omega_i(u_i)$:

$$v_i = \sum_{j=1}^m \max(0, 1 - \|(x_i, y_i) - c_j\| / r_j), \quad (3.11)$$

де c_j – центр j -ї забороненої зони;

r_j – її характерний радіус.

Загальний штраф визначається як $P(U) = \sum_i v_i$. Така параметризація є згладженою альтернативою індикаторної штрафної функції (1.14). Пропонується замінити фіксований розклад (3.10) нейронною мережею NN_θ , яка визначає значення λ на кожній ітерації:

$$\lambda(t+1) = NN_m(\lambda(t), v(t), \Delta F(t), d(t), t/T) \quad (3.12)$$

де $\lambda(t)$ – поточне значення штрафного коефіцієнта;

$v(t)$ – середній рівень порушення обмежень $P(U)$ у популяції;

$\Delta F(t)$ – зміна найкращого значення площі покриття за останні k ітерацій;

$d(t)$ – міра різноманітності популяції;

t/T – нормалізований час оптимізації.

Запропонований нейронно-керований гібридний метод детально описано у [117]. Архітектура нейронної мережі: вхідний вектор розмірності 5–10, 2–3 повнозв’язних шарів з 32–64 нейронами, активація ReLU, вихідний нейрон із функцією *softplus* для гарантії невід’ємності λ . Навчання мережі мінімізує композитну функцію втрат:

$$\mathcal{L}_{penalty}(\theta) = -\sum_{k=1}^K F_k(T) + \mu \cdot \max(0, P_k(T)) \quad (3.13)$$

де $F_k(T)$ – фінальна площа покриття в k -му оптимізаційному епізоді;

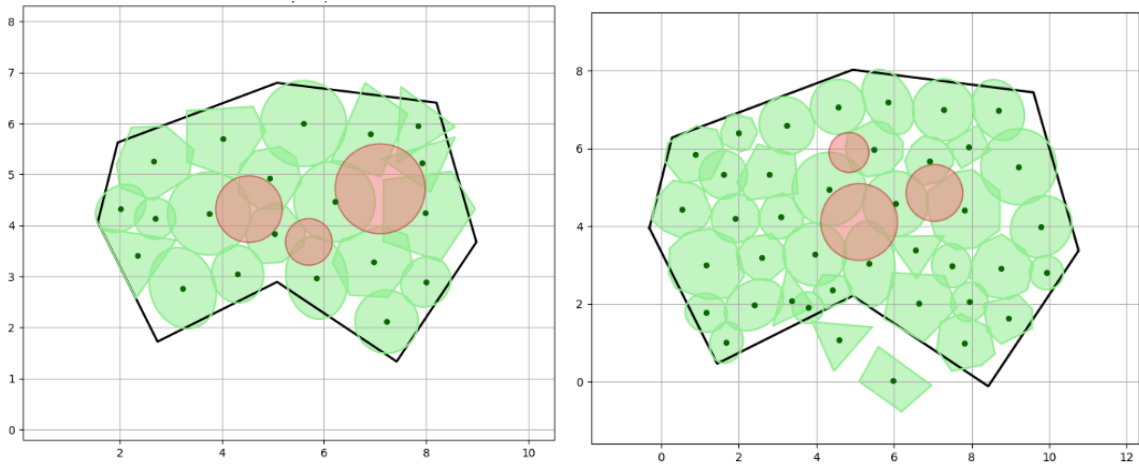
$P_k(T)$ – фінальне порушення обмежень;

μ – ваговий множник важливості дотримання обмежень.

Таким чином можна розглянути задачу покриття із забороненими зонами, та застосування описаного підходу до неї. Нейромережевий адаптивний підхід до задачі максимального покриття із забороненими зонами детальніше розглянуто у [118]

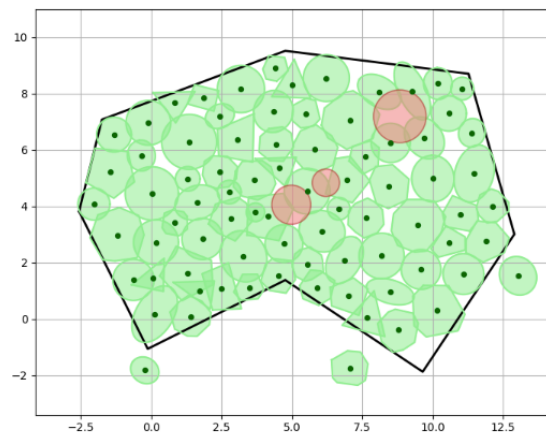
Було проведено три експерименти (конфігурації відповідають тим, що були наведені у розділі 2) із забороненими зонами, де було поєднано даний підхід із гібридним алгоритмом PSO+BFGS. У кожному експерименті заборонені зони розміщено у різних частинах області покриття з метою моделювання реалістичних сценаріїв, де окремі ділянки території є недоступними для розміщення обладнання. Заборонені зони являють собою червоні кола, та центри об’єктів покриття (темно-зелені) не можуть бути розміщені у цих зонах (рисунок 3.1). Нейронно-керований адаптивний механізм автоматично регулює штрафний коефіцієнт λ протягом оптимізації, забезпечуючи поступовий перехід від вільного дослідження простору параметрів до суворого дотримання геометричних обмежень. Конфігурації заборонених зон для кожної серії експериментів наведено у Додатку Б.

Результати експериментів наведено на рисунку 3.4, у всіх 3 випадках заборонені зони не були порушені:



а)

б)



в)

Рисунок 3.4 – Результати покриття: а) $n = 20$; б) $n = 40$; в) $n = 80$

Для експерименту 1, покриття склало 87.6% зони покриття; для експерименту 2 – 85.7%; для експерименту 3 – 86.1%. З рисунку 3.4 видно, що запропонована модифікація демонструє гарні результати. Відсоток покриття менший, ніж при виконанні алгоритму без заборонених зон, але все ще на високому рівні (більше ніж 85%). До того ж, під час проведення ряду експериментів, було з'ясовано, що

ефективну кількість ітерацій для PSO у випадку, наприклад, з кількістю покриваючих об'єктів, що дорівнює 80, можна зменшити з 500 до 200 без втрати ефективності покриття при використанні гібридного підходу. Це суттєво зменшує час, що потрібен для вирішення задачі. Так, наприклад, у наведених на рисунку 3.4 даних для експерименту 3 знадобилось всього лише 150 секунд та 200 ітерацій PSO для отримання результату у 85-86% покриття, що й при 500 ітераціях PSO та понад 550 секундах виконання.

3.4.2 Динамічне налаштування гіперпараметрів ройових алгоритмів

Другий можливий напрямок нейромережевої гібридизації спрямований на автоматичне налаштування внутрішніх гіперпараметрів ройових алгоритмів під час їхньої роботи. Традиційно ці параметри встановлюються перед запуском та залишаються фіксованими або змінюються за простим детерміністичним розкладом. Об'єктами динамічного налаштування є: для PSO – інерційний коефіцієнт w та коефіцієнти когнітивного і соціального навчання c_1, c_2 ; для FA – параметр поглинання γ та початкова привабливість β_0 ; для FSS – амплітуди індивідуального та колективного кроків; для ABC – поріг виснаження та амплітуда локальної модифікації.

Вхідні ознаки нейронної мережі поділяються на статичні (кількість об'єктів n , розмірність $3n$, площа та периметр області D , кількість і площа заборонених зон Z_j , відношення сумарної площі об'єктів до площі області) та динамічні (нормалізований час t/T , поточна найкраща площа покриття $F^*(t)$, темп покращення $\Delta F/\Delta t$, міра різноманітності популяції, частка недопустимих конфігурацій).

Архітектура мережі включає 3-4 повнозв'язних шари з 64–128 нейронами та активацією ReLU. Така глибина мережі є достатньою для апроксимації нелінійних залежностей між станом оптимізаційного процесу та оптимальними значеннями гіперпараметрів, водночас не створюючи надмірних

обчислювальних витрат на етапі інференсу. Вихідний шар використовує sigmoid для нормалізації виходів до допустимих діапазонів:

$$\hat{w} = w_{min} + (w_{max} - w_{min}) \cdot \sigma(z_w) \quad (3.14)$$

де \hat{w} – прогнозоване значення параметра;

w_{min}, w_{max} – межі допустимого діапазону;

$\sigma(z) = 1/(1 + e^{-z})$ – сигмоїдна функція.

Для PSO мережа прогнозує трійку $(w, c1, c2)$, для FA – пару (γ, β_0) .

3.4.3 Парадигми навчання нейромережових компонентів

Ефективність нейромережових компонентів суттєво залежить від обраної парадигми машинного навчання. Запропоновані нейромережові компоненти (підрозділи 3.4.1–3.4.2) потребують ефективних стратегій навчання. Розглянемо три фундаментальні парадигми та обґрунтуємо вибір найбільш перспективної з них.

Навчання з учителем передбачає попередній збір бази пар «вхід–вихід» шляхом систематичного перебору конфігурацій. Для кожного екземпляра задачі покриття виконується серія запусків оптимізації з різними значеннями гіперпараметрів, після чого фіксуються найкращі комбінації, що забезпечили максимальну площу $F(U)$. Перевагою є швидкість прогнозування після офлайн-навчання (частки мілісекунди на інференс). Основне обмеження – якість моделі визначається повнотою навчального набору: при зустрічі з нетиповою конфігурацією заборонених зон узагальнююча здатність різко знижується.

Навчання без учителя спрямоване на виявлення прихованих структур у просторі конфігурацій. Автокодувальники стискають вектор параметрів $U \in \mathbb{R}^{3n}$ до компактного латентного представлення $z = E(U)$ розмірності $d \ll 3n$, мінімізуючи втрату реконструкції $\|U - D(E(U))\|^2$. Кластеризація популяції (k -

means, DBSCAN) дозволяє оцінити різноманітність та виявити передчасну конвергенцію. Варіаційні автокодувальники (Variational Autoencoder, далі – VAE) [119] генерують якісніші початкові конфігурації порівняно з рівномірною ініціалізацією. Однак ці методи не оптимізують безпосередньо якість покриття і слугують допоміжними компонентами.

Навчання з підкріпленням (Reinforcement learning, далі – RL) формалізує процес оптимізації як марківський процес прийняття рішень (Markov decision process, далі – MDP) [120]. Середовищем є ройовий алгоритм, станом – поточні характеристики оптимізації, а діями – зміни керованих параметрів. Для задачі штрафного коефіцієнта (підрозділ 3.4.1) стан доповнюється рівнем порушення обмежень, а дії полягають у зміні λ . Функція винагороди поєднує приріст площі покриття зі штрафом:

$$r(t) = (F * (t + 1) - F * (t)) - \mu \cdot \max(0, P(U *_{t+1})), \quad (3.15)$$

де $F * (t)$ – найкраща площа покриття на кроці t ;

μ – ваговий множник штрафу.

Агент максимізує сукупну дисконтовану винагороду:

$$G = \sum_{t=0}^T \gamma^t \cdot r(t), \gamma \in [0.95; 0.99] \quad (3.16)$$

Для реалізації рекомендується алгоритм оптимізації проксимальних політик (Proximal Policy Optimization, далі – PPO) [121] для неперервних просторів дій або глибоких Q-мереж (Deep Q-Network, далі – DQN) для дискретизованих. Архітектура Actor-Critic складається з двох гілок: Actor $\pi(a|s)$ генерує розподіл дій, Critic $V(s)$ оцінює очікувану винагороду. Мережа містить 2–3 повнозв’язних шари з 64–128 нейронами.

У таблиці 3.5 наведено порівняння парадигм навчання для задач нейромережевого керування.

Таблиця 3.5 – Порівняння парадигм навчання для нейромережевого керування

Критерій	З учителем	Без учителя	З підкріпленням
Потреба у розмічених даних	Висока	Відсутня	Відсутня
Швидкість прогнозування	Найвища	Висока	Висока
Здатність до узагальнення	Обмежена	Середня	Висока
Стратегічний горизонт	Миттєвий	Не застос.	Довгостроковий
Відкриття нових стратегій	Ні	Ні	Так
Трансферне навчання	Обмежене	Так	Так

Навчання з підкріпленням обрано як основну парадигму з наступних міркувань. По-перше, агент навчання з підкріпленням не потребує попереднього збору еталонних даних, що усуває проблему обмеженості навчального набору. По-друге, RL природно відображає послідовну природу задач керування: на кожній ітерації ройового алгоритму агент спостерігає стан, приймає рішення та отримує зворотний зв'язок. По-третє, завдяки дисконтованій винагороді RL-агент враховує довгострокові наслідки – наприклад, свідомо допускає тимчасове зниження λ на ранніх ітераціях для дослідження перспективних регіонів з подальшим посиленням штрафу. Такий стратегічний горизонт є принципово недосяжним для контрольованого навчання.

Додатковою перевагою є трансферне навчання: RL-агент, попередньо навчений на спрощених задачах покриття з невеликою кількістю об'єктів та заборонених зон, дотренується на складніших конфігураціях зі значно меншими витратами порівняно з навчанням з нуля, що забезпечує масштабованість при переході до задач зростаючої складності.

Висновки до розділу 3

У цьому розділі розроблено, обґрунтовано та досліджено гібридні (меметичні) методи розв'язання задачі неперервного максимального покриття, що поєднують глобальний стохастичний пошук ройових метаевристик із детермінованою локальною оптимізацією та нейромережовим керуванням. Основні результати розділу полягають у такому.

По-перше, обґрунтовано застосування методу BFGS як локальної компоненти гібридних алгоритмів для задач максимального покриття. Розглянуто підхід до чисельного оцінювання градієнта ω -функції на основі матриці площ попарних перетинів об'єктів, що забезпечує прискорення на порядок порівняно з прямим перерахунком площі складного об'єкта (0.004 с проти 0.08 с при $n = 100$). Квазіньютонівський метод BFGS із скінченно-різницевою апроксимацією градієнта продемонстрував суперлінійну збіжність та здатність знаходити високоякісні локальні оптимуми, що у граничних випадках ($n = 50, n = 75$) збігаються з глобальними розв'язками. Використання підходу до оцінювання градієнта через матрицю попарних перетинів [31] забезпечує прискорення обчислень на порядок порівняно з прямим перерахунком площі складного об'єкта.

По-друге, розроблено узагальнену меметичну методологію для задач максимального геометричного покриття, яка поєднує чотири ройові метаевристики (PSO, FSS, FA, ABC) з локальною оптимізацією BFGS. Для кожної з чотирьох гібридних комбінацій (PSO+BFGS, FSS+BFGS, FA+BFGS, ABC+BFGS) визначено оптимальну частоту виклику локального пошуку, рекомендовані параметри та критерії зупинки. Показано, що гібридна схема є універсальною і може бути застосована до будь-якого з розглянутих ройових алгоритмів без зміни їхньої базової структури. Встановлено, що меметичний підхід стабільно підвищує площу покриття на 12–25% порівняно з чистими ройовими методами (від 12.2% при $n = 20$ до 24.7% при $n = 80$, таблиця 3.4).

При цьому кількість ітерацій глобального пошуку у гібридному підході може бути скорочена вдвічі без втрати якості покриття, що зменшує загальний час обчислень на 30–40%.

По-третє, проведено порівняльний аналіз гібридних методів на серії чисельних експериментів зі зростаючою кількістю покривальних об'єктів ($n = 20, 40, 80$). На прикладі комбінації PSO+BFGS підтверджено, що гібридний метод стабільно перевершує чистий ройовий пошук: приріст площі покриття становить 12.2% при $n = 20$ (з 77.4% до 89.6%), 14.1% при $n = 40$ (з 75.8% до 89.9%) та 24.7% при $n = 80$ (з 65.1% до 89.8%). Встановлено, що чисті ройові алгоритми мають поріг ефективності на рівні 82-85% покриття, після якого подальше масштабування кількості ітерацій та популяції не дає суттєвого покращення. Перевага гібридного підходу зростає зі збільшенням кількості об'єктів, що підтверджує його масштабованість та практичну цінність для задач великої розмірності.

По-четверте, запропоновано нейронно-керовані гібридні методи максимізації покриття, що інтегрують нейронні мережі у оптимізаційний процес за двома напрямками: адаптивне керування штрафним коефіцієнтом λ та динамічне налаштування гіперпараметрів ройових алгоритмів. Для задач із забороненими зонами нейронно-керований адаптивний механізм штрафних коефіцієнтів забезпечує площу покриття на рівні 85.7–87.6% при повному дотриманні геометричних обмежень. Скорочення кількості ітерацій PSO з 500 до 200 при $n = 80$ зменшує час обчислень з 550 с до 150 с без втрати якості покриття. На основі порівняльного аналізу трьох парадигм навчання (з учителем, без учителя, з підкріпленням) обґрунтовано вибір навчання з підкріпленням як найбільш перспективного підходу, що забезпечує автоматичне формування стратегії керування без потреби у попередньому зборі еталонних даних.

Сформульовані рекомендації щодо вибору та налаштування гібридних методів залежно від геометричної структури задачі, її розмірності та наявності

заборонених зон створюють методичну основу для практичного застосування вдосконалених метаевристик у задачах просторового планування та оптимізації сервісного покриття. Подальші дослідження спрямовано на програмну реалізацію запропонованих методів у межах уніфікованої інформаційної технології, що є предметом розгляду розділу 4.

РОЗДІЛ 4. РОЗРОБКА UML-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ РОЗВ'ЯЗАННЯ НЕПЕРЕРВНИХ ЗАДАЧ МАКСИМАЛЬНОГО ГЕОМЕТРИЧНОГО ПОКРИТТЯ

4.1 Вимоги до інформаційної технології

У попередніх розділах досліджено математичні моделі, методи обчислення площі покриття, локальні, метаевристичні та гібридні алгоритми оптимізації. Цей розділ присвячено розробленню інформаційної технології, що реалізує гібридну систему оптимізації та інтегрує зазначені методи у єдину масштабовану програмну платформу.

4.1.1 Функціональні вимоги

Функціональні вимоги визначають можливості системи з точки зору основного користувача. Система повинна забезпечувати:

- 1) імпорт просторових даних з файлів або зовнішніх ГІС-сервісів;
- 2) налаштування геометричної моделі з вибором типів покриваючих об'єктів та їх параметрів;
- 3) вибір методу обчислення покриття з п'яти доступних стратегій;
- 4) вибір алгоритмів оптимізації: ройових, генетичних, меметичних та локального пошуку;
- 5) запуск адаптивної оптимізації з автоматичним перемиканням між грубими та точними методами оцінювання;
- 6) моніторинг прогресу та візуалізацію конфігурації покриття;
- 7) експорт рішень у ГІС-форматах;
- 8) керування шаблонами сценаріїв; зупинку та відновлення процесу оптимізації.

Програмна система повинна складатися з чотирьох функціональних підсистем: геометричне ядро (Geometry Core), підсистема оцінювання покриття (Coverage Evaluation), оптимізаційна підсистема (Optimization Engine) та інфраструктурна підсистема. Детальну специфікацію класів, інтерфейсів та зв'язків між підсистемами наведено у підрозділі 4.3.

4.1.2 Нефункціональні вимоги

Нефункціональні вимоги визначають технологічні обмеження та зовнішні інтерфейси системи [122]. Основною мовою реалізації обчислювального ядра є Python (бібліотеки Shapely, scipy, numpy, matplotlib). Для зберігання сценаріїв, логів та результатів використовується система управління базами даних (далі – СКБД) PostgreSQL з розширенням PostGIS. Система повинна працювати на операційних системах Linux та Windows з підтримкою контейнеризації Docker.

Зовнішні інтерфейси системи включають:

- 1) графічний інтерфейс для взаємодії з аналітиком;
- 2) REST API (JSON) для комунікації між рівнем представлення та прикладної логіки;
- 3) інтерфейс роботи з файловою системою (GeoJSON, Shapefile, HDF5, CSV);
- 4) інтеграцію з ГІС-сервісами за протоколами WMS/WFS;
- 5) SQL/ORM-інтерфейс до СКБД; опціонально – gRPC-інтерфейс для з'єднання з обчислювальним кластером.

Серед додаткових обмежень:

- 1) кількість покриваючих об'єктів у конфігурації – від 1 до 80 та більше;
- 2) типи підтримуваних форм – багатокутники, кола, еліпси та їх композиції; кількість методів обчислення покриття – не менше п'яти;
- 3) алгоритм оптимізації за замовчуванням – PSO;

4) метод обчислення покриття за замовчуванням – BoundingBoxEstimator.

Наведені обмеження визначають технологічний стек та межі реалізації програмної системи.

4.1.3 Атрибути якості

Атрибути якості описують вимірювання характеристик продукту, що є важливими для користувача або розробника [122]. Нижче наведено основні атрибути якості, яким повинна відповідати розроблена інформаційна технологія.

Продуктивність. Система повинна мати можливість обробляти задачі з 20 та більше покриваючими об'єктами одночасно. При типовій задачі (20–40 об'єктів) один цикл оптимізації (1000 ітерацій) повинен виконуватися за прийнятний час.

Ключові показники подано у таблиці 4.1:

Таблиця 4.1 – Ключові показники продуктивності системи

Показник	Цільове значення (мс)
Час одного обчислення покриття (BoundingBox, 20 об'єктів)	< 0,1
Час одного обчислення покриття (Монте-Карло, 20 об'єктів, 10000 точок)	< 5
Час одного обчислення покриття (Shapely, 20 об'єктів)	< 50
Максимальна кількість об'єктів у конфігурації	≥ 80
Накладні витрати на комунікацію між модулями прикладного рівня	≈ 0 (in-process)

Продуктивність є ключовим критерієм для даного класу систем, оскільки обчислювальне ядро виконує тисячі оцінок покриття за ітерацію оптимізації.

Модульність та розширюваність. Архітектура повинна дозволяти додавання нових алгоритмів оптимізації, методів оцінювання площі та типів геометричних об'єктів без суттєвої модифікації існуючого коду. Це досягається завдяки шаблонам проєктування Strategy, Factory та Facade: кожна нова

реалізація реєструється через відповідний інтерфейс без впливу на решту компонентів (детальну специфікацію інтерфейсів наведено у підрозділі 4.3).

Переносимість. Система повинна мати можливість розгортання у різних середовищах:

- локальна машина дослідника (Windows/Linux) для індивідуальних експериментів;
- серверне розгортання (Linux, Docker) для обчислювально-інтенсивних задач;
- хмарне розгортання (AWS, GCP) з можливістю підключення обчислювального кластера.

Переносимість забезпечується використанням Python та контейнеризацією Docker.

Серед інших атрибутів якості виділяються надійність, масштабованість та зручність експлуатації. Надійність передбачає автоматичне перемикавання на альтернативний метод обчислення при збої основного (через CoverageStrategyManager), виключення збійних конфігурацій з популяції без зупинки процесу та збереження checkpoint кожні N ітерацій. Масштабованість забезпечується через вертикальне нарощування ресурсів, опціональне винесення обчислень на зовнішній кластер (gRPC) та паралельне обчислення покриття в межах одного сервера (multiprocessing). Зручність експлуатації досягається завдяки інтуїтивному інтерфейсу, візуалізації проміжних результатів у реальному часі та повторному використанню збережених сценаріїв. Зазначені атрибути в сукупності визначають вимоги до архітектурного рішення, обґрунтування вибору якого наведено у підрозділі 4.2.

4.2 Обґрунтування вибору архітектурної моделі

У попередньому підрозділі визначено структурований перелік функціональних та нефункціональних вимог до інформаційної технології (далі –

IT) розв'язання неперервних задач максимального покриття з об'єктами довільної форми. Зокрема, серед атрибутів якості виділено продуктивність (обробка ≥ 80 покриваючих об'єктів, < 50 мс на оцінку покриття методом Shapely), модульність (додавання алгоритмів через інтерфейси CoverageEstimator та OptimizationAlgorithm за шаблонами Strategy/Factory), переносимість (Docker, REST API), масштабованість (вертикальне та горизонтальне через gRPC-кластер) і мінімальну складність інфраструктури для дослідницької системи.

Такий функціональний обсяг вимагає обґрунтованого вибору моделі системної архітектури (далі – СА). Для прийняття рішення застосовано метод багатокритеріального прийняття рішень (Technique for Order of Preference by Similarity to Ideal Solution, далі – TOPSIS) [123, 124] у поєднанні з методом аналізу ієрархій (Analytic Hierarchy Process, далі – АНР) [125, 126] для визначення вагових коефіцієнтів критеріїв.

На основі атрибутів якості (підрозділ 4.1.3) визначено п'ять критеріїв: продуктивність (мінімізація накладних витрат на комунікацію між модулями), модульність (додавання алгоритмів через інтерфейси без модифікації ядра), переносимість (Windows/Linux/Docker, стандартизовані інтерфейси), масштабованість (від 20 до 80+ об'єктів, вертикальне та горизонтальне через gRPC) та складність інфраструктури як негативний критерій.

Для визначення відносної важливості критеріїв побудовано матрицю попарних порівнянь за методом аналізу ієрархій (таблиця 4.2) [125, 126].

Таблиця 4.2 – Матриця попарного порівняння критеріїв

Критерій	Продукт.	Модульн.	Перенос.	Масштаб.	Складн.
Продуктивність	1,000	2,000	3,000	4,000	7,000
Модульність	0,500	1,000	2,000	3,000	5,000
Переносимість	0,333	0,500	1,000	2,000	4,000
Масштабованість	0,250	0,333	0,500	1,000	3,000
Складність інфр.	0,143	0,200	0,250	0,333	1,000
Сума стовпців	2,226	4,033	6,750	10,333	20,000

Після нормалізації матриці та обчислення середнього арифметичного по кожному рядку отримано ваги критеріїв (таблиця 4.3).

Таблиця 4.3 – Нормалізована матриця та вагові коефіцієнти критеріїв

Критерій	Прод.	Мод.	Пер.	Масш.	Скл.	Ваги
Продуктивн.	0,449	0,496	0,444	0,387	0,350	0,425
Модульність	0,225	0,248	0,296	0,290	0,250	0,262
Переносим.	0,150	0,124	0,148	0,194	0,200	0,163
Масштабов.	0,112	0,083	0,074	0,097	0,150	0,103
Складність	0,064	0,050	0,037	0,032	0,050	0,047

Перевірка консистентності: $\lambda_{max} = 5,078$; $CI = 0,020$; $CR = 0,017 < 0,1$ — ваги є консистентними [125, 126].

Отже, продуктивність (0,425) та модульність (0,262) складають понад 68% загальної ваги. Це закономірно: для обчислювально-інтенсивної системи, що виконує тисячі оцінок покриття за ітерацію з необхідністю динамічного перемикання між п'ятьма стратегіями обчислення, саме ці два фактори є визначальними.

4.2.1 Альтернативні моделі системної архітектури

Для аналізу обрано п'ять розповсюджених моделей системної архітектури [127, 128]:

1. Монолітна архітектура – єдиний застосунок, усі модулі у спільному адресному просторі. Мінімальні накладні витрати на комунікацію, але відсутність структурованого розподілу відповідальності. Додавання нового алгоритму оптимізації потребує перекомпіляції всього додатку.

2. Клієнт-сервер 2-рівнева – клієнт (інтерфейс та частина логіки) та сервер (дані). Не дозволяє відокремити обчислювальне ядро від інтерфейсу, що

ускладнює розвиток підсистем Coverage Evaluation та Optimization Engine незалежно від UI.

3. Клієнт-сервер 3-рівнева (модульна) – три чітко розділені рівні: представлення, прикладна логіка, дані. Модулі прикладного рівня (Geometry Core, Coverage Evaluation, Optimization Engine, Strategy Manager) реалізують шаблони Strategy/Factory/Facade з чіткими інтерфейсами. Поєднує продуктивність in-process викликів з модульною організацією.

4. Мікросервісна архітектура (MCA) – кожен компонент є автономним сервісом з власною БД. Найвища масштабованість та незалежність розгортання. Проте кожен виклик CoverageEstimator.estimateCoverage() потребує серіалізації, мережевого передавання та десеріалізації, що при тисячах викликів за ітерацію додає суттєві затримки.

5. Конвеєрна архітектура (Pipe-and-Filter) – дані передаються послідовно через ланцюг фільтрів-обробників. Природна для потокової обробки, але погано підходить для ітеративної оптимізації зі зворотним зв'язком: оптимізаційний цикл потребує повторних викликів Coverage Evaluation з модифікованими конфігураціями, що не вкладається у лінійний конвеєр.

На основі аналізу відповідності кожної моделі СА сформульованим вимогам надано експертні оцінки (таблиця 4.4) за 5-бальною шкалою (1 – найнижче, 5 – найвище). Критерій «Складність інфраструктури» є критерієм витрат (Cost): вище значення означає більшу складність, що є негативним.

Таблиця 4.4 – Експертні оцінки моделей СА за критеріями

Модель СА / Критерій	Продукт.	Модульн.	Перенос.	Масштаб.	Складн. (Cost)
Монолітна	4	2	4	1	1
КС 2-рівнева	3	2	3	2	2
КС 3-рівнева (мод.)	5	5	4	4	3
Мікросервісна	3	5	3	5	5
Конвеєрна	4	3	3	2	2

Продуктивність. Згідно з вимогами, час одного обчислення покриття (Shapely, 20 об'єктів) не має перевищувати 50 мс, а накладні витрати на комунікацію між модулями прикладного рівня мають бути ≈ 0 (in-process). КС 3-рівнева (5): Geometry Core, Coverage Evaluation та Optimization Engine працюють у спільному адресному просторі, виклик estimateCoverage() – це звичайний виклик функції Python. Монолітна (4): також in-process, але відсутність структурованого розподілу призводить до неоптимальних внутрішніх залежностей. МСА (3): кожен виклик estimateCoverage() проходить через серіалізацію JSON \rightarrow HTTP \rightarrow десеріалізацію, що додає 1–5 мс на виклик; при 10 000 оцінок за ітерацію \times 500 ітерацій – це години чистих мережевих затримок.

Модульність. Згідно з вимогами, додавання нового методу обчислення покриття має зводитися до реалізації інтерфейсу CoverageEstimator та реєстрації у CoverageStrategyManager. КС 3-рівнева (5) та МСА (5) однаково забезпечують цю можливість через чіткі інтерфейси (ICoverageEstimate, IOptimizationService). Різниця: у 3-рівневій модульність досягається на рівні компонентів (in-process), у МСА – на рівні сервісів. Монолітна (2): формальні інтерфейси відсутні, додавання алгоритму потребує змін у ядрі.

Переносимість. Згідно з вимогами: підтримка Windows/Linux/Docker, REST API, SQL/ORM, WMS/WFS. Монолітна (4): один артефакт, легко контейнеризується, але обмежена у варіантах розгортання. КС 3-рівнева (4): рівні розділені, кожен може розгортатися у власному контейнері. МСА (3): хоча кожен сервіс переноситься легко, сукупність сервісів потребує оркестрації, що ускладнює перенесення всієї системи.

Масштабованість. Згідно з вимогами: вертикальне масштабування та опціональне горизонтальне через gRPC-кластер. МСА (5): лідер завдяки незалежному масштабуванню кожного сервісу. КС 3-рівнева (4): масштабується вертикально та через винесення Optimization Engine на зовнішній кластер (gRPC). Монолітна (1): масштабується лише вертикально.

Складність інфраструктури. Згідно з обмеженнями: мінімізація кількості сервісів, БД – PostgreSQL. Монолітна (1): найпростіша – один процес + БД. КС 3-рівнева (3): помірна – сервер додатків + БД + опціональний кластер. МСА (5): потребує оркестрації контейнерів (Kubernetes), service mesh, розподілених БД, API gateway – надмірно для дослідницької системи.

4.2.2 Застосування методу TOPSIS

Метод TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) [123, 124] визначає альтернативу, найближчу до ідеального рішення та найдальшу від анти-ідеального. Ідеальне рішення формується з найкращих значень кожного критерію серед усіх альтернатив, а анти-ідеальне – з найгірших. Кожна альтернатива оцінюється за відносною близькістю до ідеального рішення, що дозволяє побудувати однозначне ранжування навіть за наявності суперечливих критеріїв. Для забезпечення порівнянності критеріїв з різними одиницями виміру та діапазонами значень нормалізацію виконано за формулою:

$$r_{ij} = x_{ij} / \text{sqrt}(\sum x_{ij}^2) \quad (4.1)$$

Результати подано у таблиці 4.5:

Таблиця 4.5 – Нормалізована матриця рішень

Модель СА	Продукт.	Модульн.	Перенос.	Масштаб.	Складн.
Монолітна	0,462	0,244	0,521	0,141	0,152
КС 2-рівнева	0,346	0,244	0,391	0,283	0,305
КС 3-рівнева (мод.)	0,577	0,611	0,521	0,566	0,457
Мікросервісна	0,346	0,611	0,391	0,707	0,762
Конвеєрна	0,462	0,367	0,391	0,283	0,305

Зважену матрицю отримано множенням нормалізованих значень на відповідні ваги критеріїв. Для критеріїв типу Benefit ідеальне найкраще – максимум, для Cost – мінімум (таблиця 4.6).

Таблиця 4.6 – Зважена нормалізована матриця рішень

Модель СА	Продукт.	Модульн.	Перенос.	Масштаб.	Складн.
Монолітна	0,196	0,064	0,085	0,015	0,007
КС 2-рівнева	0,147	0,064	0,064	0,029	0,014
КС 3-рівнева (мод.)	0,246	0,160	0,085	0,058	0,021
Мікросервісна	0,147	0,160	0,064	0,073	0,036
Конвеєрна	0,196	0,096	0,064	0,029	0,014
Тип критерію	BENEFIT	BENEFIT	BENEFIT	BENEFIT	COST
Ідеальне найкраще	0,246	0,160	0,085	0,073	0,007
Ідеальне найгірше	0,147	0,064	0,064	0,015	0,036

Обчислено евклідові відстані до ідеального ($D +$) та анти-ідеального ($D -$) рішень, а також відносну близькість $C = D - / (D + + D -)$ (таблиця 4.7).

Таблиця 4.7 – Відстані до ідеалів та відносна близькість до ідеального рішення

Модель СА	D+	D-	C	Ранг
Монолітна	0,123	0,061	0,331	4
КС 2-рівнева	0,146	0,026	0,150	5
КС 3-рівнева (мод.)	0,020	0,146	0,878	1
Мікросервісна	0,104	0,112	0,518	2
Конвеєрна	0,094	0,064	0,404	3

Найбільше значення відносної близькості до ідеального рішення має клієнт-серверна 3-рівнева модульна архітектура ($C = 0,878$). Вона суттєво випереджає МСА (0,518) та інші альтернативи. Розрив між першим та другим місцем (0,360) є значним і свідчить про явну перевагу обраної моделі для даного класу задач.

Таким чином, обґрунтовано вибір трирівневої модульної архітектури. У наступному підрозділі цю архітектуру формалізовано засобами UML-моделювання.

4.3 UML-специфікація інформаційної технології

На основі обраної трирівневої архітектури (підрозділ 4.2) побудовано комплекс UML-діаграм, що формалізують структуру, поведінку та розгортання інформаційної технології.

Архітектура реалізується у вигляді класичної трирівневої моделі:

Рівень представлення (Presentation Layer) – графічний або веб-інтерфейс аналітика, модулі візуалізації покриття, засоби налаштування задачі.

Рівень прикладної логіки (Application / Logic Layer) – підсистеми геометричного моделювання, оцінювання покриття, оптимізації, керування сценаріями, адаптивного вибору стратегій.

Рівень даних (Data Layer) – сховище геоданих, параметрів сценаріїв, історії конфігурацій, логів.

UML-діаграми, що описуються далі, послідовно розкривають ці рівні з різних точок зору: функціональної (Use Case), структурної (Class, Component, Composite Structure), поведінкової (Activity, Sequence, State Machine).

4.3.1 Діаграма варіантів використання

UML-діаграма варіантів використання формалізує зовнішню функціональну специфікацію інформаційної технології з точки зору користувачів та зовнішніх систем. Вона визначає, які задачі може виконувати аналітик, які сервіси надає система та як відбувається інтеграція з ГІС і обчислювальною інфраструктурою. На діаграмі виокремлено три категорії акторів: основний користувач (аналітик), який ініціює процеси оптимізації та аналізує результати; зовнішні джерела просторових даних (ГІС-сервіси), що забезпечують імпорт

геометрії області покриття та заборонених зон; та обчислювальна інфраструктура, що виконує ресурсоміткі оптимізаційні розрахунки. Кожен варіант використання відповідає окремій функціональній вимозі, сформульованій у підрозділі 4.1.1, що забезпечує простежуваність між вимогами та проектними рішеннями. Зв'язки типу «include» та «extend» між варіантами використання відображають обов'язкові та опціональні залежності між функціями системи.

Основні актори:

- аналітик – формулює задачу, обирає типи геометричних об'єктів, методи обчислення покриття та оптимізації, запускає обчислення, аналізує результати;
- адміністратор системи – керує конфігураціями, правами доступу, обчислювальними ресурсами;
- зовнішній GIS/DB-сервіс – постачає та приймає просторові дані (геометрія області, зон заборони, довідкова інфраструктура);
- обчислювальний кластер покриття – виконує ресурсоміткі операції з оцінювання покриття та оптимізації (може бути абстрагований як зовнішній сервіс).

Ключові варіанти використання (рисунок 4.1) відповідають функціональним вимогам, визначеним у підрозділі 4.1.1, і охоплюють повний цикл роботи з системою: від імпорту просторових даних (UC1) та налаштування геометричної моделі (UC2) до експорту результатів (UC8) та керування сценаріями (UC9). Така організація варіантів використання забезпечує послідовний та контрольований процес розв'язання задачі покриття, де кожен крок логічно зумовлює наступний, а користувач має змогу повернутися до попередніх етапів для корекції параметрів без необхідності перезапуску всього процесу.

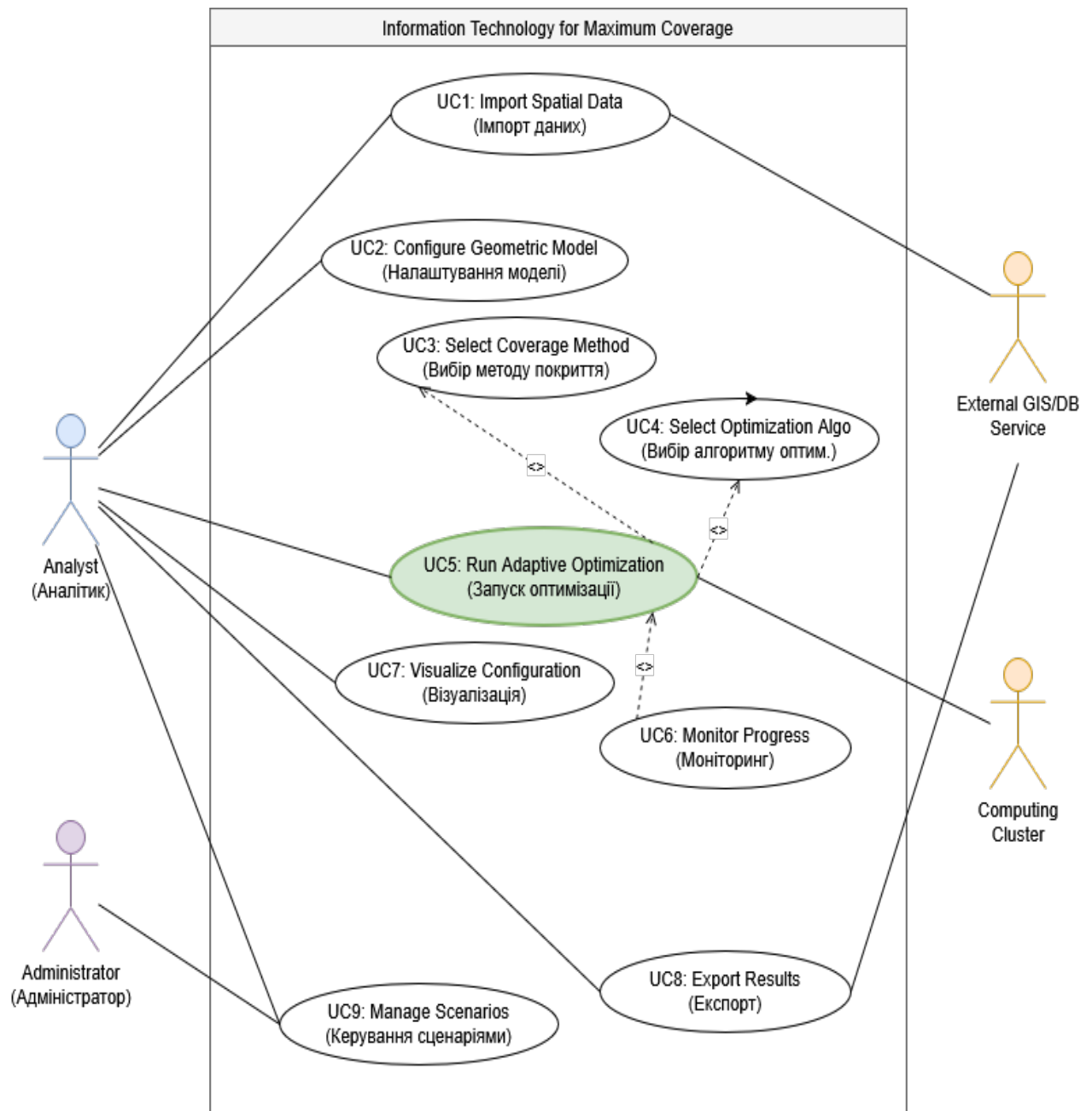


Рисунок 4.1 – Діаграма варіантів використання UML-орієнтованої інформаційної технології для неперервних задач максимального покриття з об'єктами довільної форми

На діаграмі наведено основних акторів (аналітик, адміністратор, зовнішні сервіси) та ключові варіанти використання, включаючи вибір геометричної моделі, методів обчислення покриття та алгоритмів оптимізації.

4.3.2 Діаграма класів

UML-діаграма класів задає статичну структуру інформаційної технології: сутності предметної області, їх атрибути, методи та зв'язки. Для задачі максимального покриття виділено чотири взаємопов'язані підсистеми: геометричну, підсистему оцінювання покриття, оптимізаційну та інфраструктурну.

Геометрична підсистема відповідає за представлення просторових об'єктів. Базовий абстрактний клас `AbstractGeometry` визначає загальний інтерфейс для всіх геометричних сутностей (методи `area()`, `contains()`, `intersect()`, `transform()`) і має три предметно-орієнтовані нащадки: `Region` (область покриття, що може містити отвори), `ForbiddenRegion` (заборонена зона з відповідним рівнем штрафу) та `CoverageObject` (покриваючий об'єкт з параметрами повороту та трансляції). Конкретні форми об'єктів інкапсульовані у класах `PolygonShape`, `CircleShape` та `EllipseShape`, кожен з яких реалізує спеціалізовані методи обчислення площі та перетинів.

Підсистема оцінювання покриття організована за шаблоном `Strategy`. Інтерфейс `CoverageEstimator` визначає єдиний метод `estimateCoverage()`, що повертає об'єкт `CoverageResult` (з атрибутами `coveredArea`, `coverageRatio`, `uncoveredRatio`, `penaltyValue`). П'ять конкретних реалізацій відповідають методам обчислення площі, описаним у розділі 2: `MonteCarloEstimator` (швидкий наближений метод для ранніх етапів оптимізації), `PairwiseInclusionExclusionEstimator` (метод подвійних перетинів для кругових та еліптичних об'єктів), `GeometryLibraryEstimator` (точне обчислення через бібліотеку `Shapely` для фінальних етапів), `TriangulationEstimator` (триангуляція області для детальної локальної оцінки) та `BoundingBoxEstimator` (швидка груба оцінка для первинного відбору конфігурацій). Адаптивне перемикання між цими стратегіями координує клас `CoverageStrategyManager`, який обирає метод оцінювання залежно від номеру ітерації, рівня стагнації та розмірності задачі.

Оптимізаційна підсистема керує пошуком конфігурації, що максимізує покриття. Клас Configuration зберігає набір покриваючих об'єктів, вектор рішень та результат оцінювання, а ProblemDefinition інкапсулює постановку задачі з усіма обмеженнями. Абстрактний клас OptimizationAlgorithm (з методами initialize(), iterate(), getBestSolution()) має конкретні реалізації для кожної групи алгоритмів: PSOAlgorithm, GAAlgorithm, MemeticAlgorithm, LocalSearchAlgorithm та HybridAlgorithm. Клас OptimizationEngine виступає фасадом (шаблон Facade), що координує взаємодію алгоритму оптимізації, менеджера стратегій оцінювання та PenaltyManager, який реалізує штрафні функції для врахування порушень геометричних обмежень.

Інфраструктурна підсистема включає модулі DataImporter/GISAdapter (імпорт даних), ScenarioManager (керування сценаріями), ResultExporter (експорт результатів), VisualizationService (візуалізація) та SystemLogger/MonitoringService (логування та моніторинг). Основні класи та їхні методи узагальнено у таблиці 4.8.

Таблиця 4.8 – Приклад основних класів та методів

Клас	Основні атрибути	Основні методи
Region	id, boundary, holes	area(), clip(), toPolygonalApprox()
CoverageObject	id, shapeType, baseShape, orientation, params	placeAt(), getFootprint()
CoverageEstimator	estimatorType	estimateCoverage()
MonteCarloEstimator	numSamples, samplingScheme	estimateCoverage()
GeometryLibraryEstimator	backendType, tolerance	estimateCoverage()
Configuration	objects, decisionVector, coverageResult	evaluate(), isFeasible()
OptimizationEngine	algorithm, strategyManager, penaltyManager	run(), step(), refineBestSolution()
CoverageStrategyManager	currentEstimator, switchCriteria	selectEstimator()

Архітектура побудована на шаблонах Strategy, Factory та Facade. Діаграму класів представлено на рисунку 4.2:

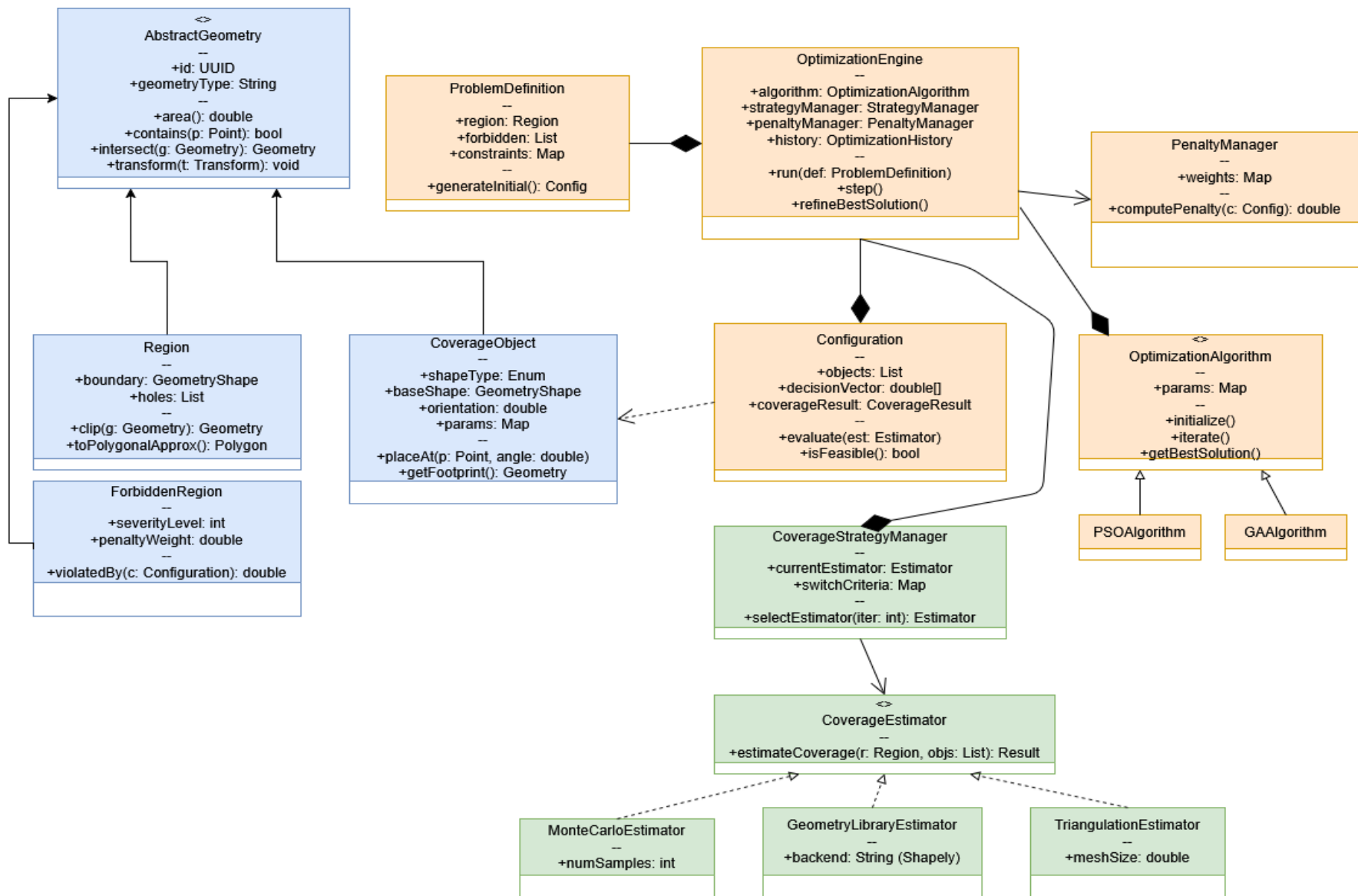


Рисунок 4.2 – Діаграма класів UML-орієнтованої інформаційної технології для неперервних задач максимального покриття з об'єктами довільної форми

На діаграмі відображено основні класи геометричної підсистеми, підсистеми оцінювання покриття, оптимізаційного ядра та інфраструктури, а також їхні зв'язки та узагальнення.

4.3.3 Діаграма компонентів

Діаграма компонентів відображає високорівневу модульну архітектуру системи та інтерфейси взаємодії між компонентами. Вона узгоджує логіку, задану діаграмою класів, із тривірневою архітектурою.

На рівні прикладної логіки виділено чотири ключові компоненти. Geometry Core (інтерфейси IGeometryModel, IGeometryOperations) відповідає за створення, зберігання та перетворення геометричних об'єктів. Coverage Evaluation (інтерфейс ICoverageEstimate) інкапсулює реалізації всіх CoverageEstimator та надає єдину точку доступу для обчислення покриття. Optimization Engine (інтерфейс IOptimizationService) реалізує запуск глобальних і локальних алгоритмів, взаємодіючи з Coverage Evaluation через адаптивний менеджер стратегій. Strategy Management (інтерфейси IStrategyConfig, IEstimatorSelector) керує вибором алгоритмів оптимізації та методів обчислення площі на різних етапах оптимізаційного процесу.

На рівні представлення розташовано компонент Visualization & Reporting (інтерфейси IVisualization, IReporting), що відповідає за графічне представлення конфігурацій та формування звітів, а також частину компонента Scenario & Configuration (інтерфейс IScenarioManagement), що забезпечує зберігання та повторне використання постановок задачі.

На рівні даних функціонує компонент Data Access / GIS Integration (інтерфейси IDataImport, IDataExport, IGISAdapter), який забезпечує роботу з зовнішніми ГІС, БД та файлами, а також Logging & Monitoring (інтерфейси ILogging, IMonitoring) для збору інформації про роботу системи. Діаграму компонентів представлено на рисунку 4.3:

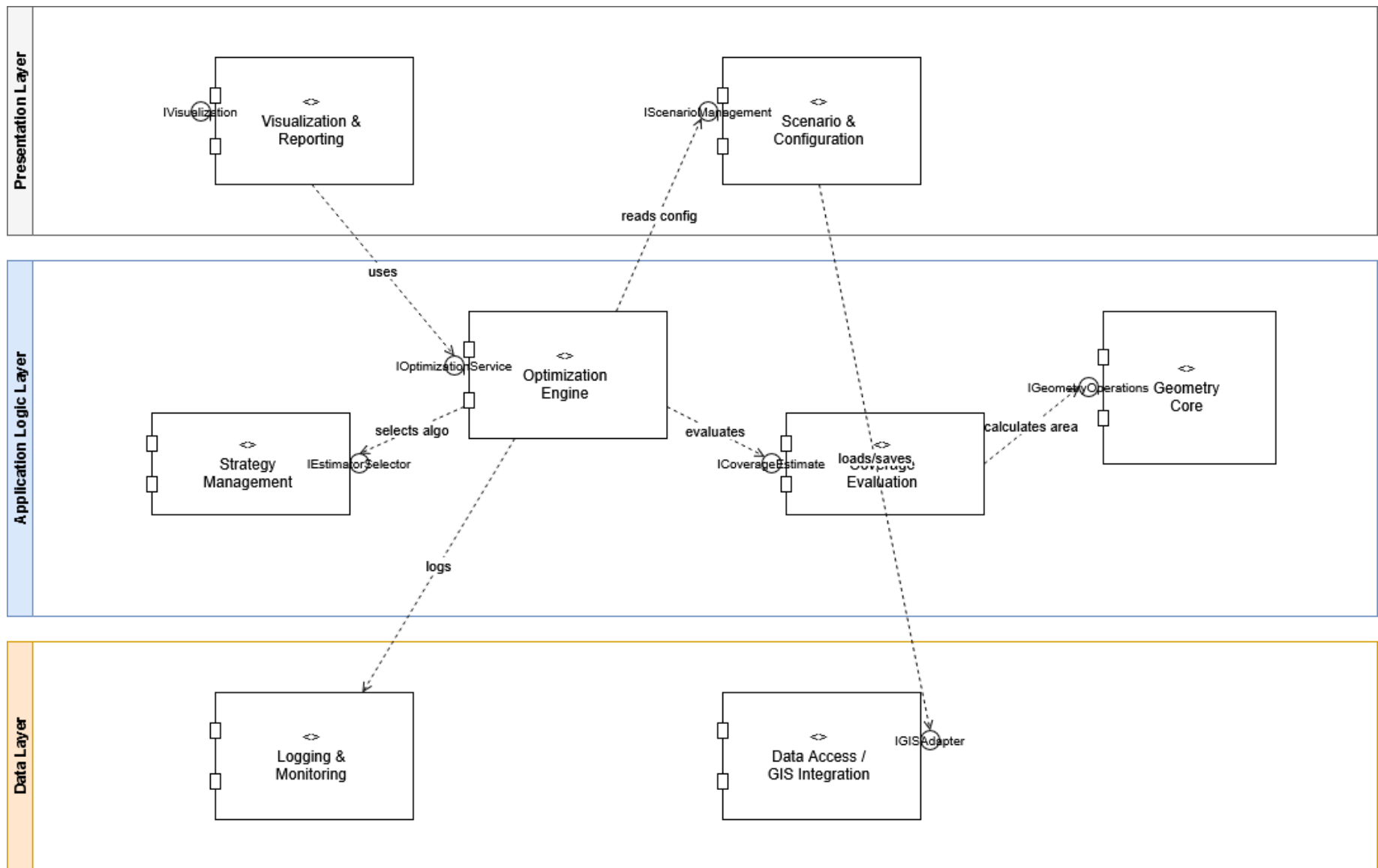


Рисунок 4.3 – Діаграма компонентів UML-орієнтованої інформаційної технології для неперервних задач максимального покриття

На діаграмі показано модулі геометричного ядра, оцінювання покриття, оптимізаційного рушія, керування стратегіями, даними, візуалізацією та моніторингом, а також їхні інтерфейси.

4.3.4 Діаграма композитної структури

Діаграма композитної структури деталізує внутрішню організацію оптимізаційного ядра (OptimizationEngine) як складного компонента, що складається з кількох підмодулів.

Генерація та управління конфігураціями забезпечується двома підмодулями: CandidateGenerator створює початкові та проміжні конфігурації, а PopulationManager зберігає популяцію кандидатів та реалізує операції відбору. Обидва підмодулі безпосередньо взаємодіють із GlobalSearchModule (реалізує PSO, GA, меметичні алгоритми) та LocalSearchModule (локальний пошук та fine-tuning кращих конфігурацій).

Координацію оцінювання конфігурацій здійснює EvaluationCoordinator, який через порт CoverageStrategyPort отримує поточний метод оцінювання від зовнішнього компонента Coverage Evaluation, а також взаємодіє з PenaltyManager для розрахунку штрафів. Рішення про перемикання між грубими та точними методами оцінювання приймає StrategySelector на основі аналізу історії значень покриття, що накопичується у HistoryManager. Саме HistoryManager забезпечує детекцію стагнації, яка є тригером для переходу до більш точних методів обчислення площі. Логування роботи всіх підмодулів здійснюється через виділений порт LogPort. Діаграму композитної структури показано на рисунку 4.4:

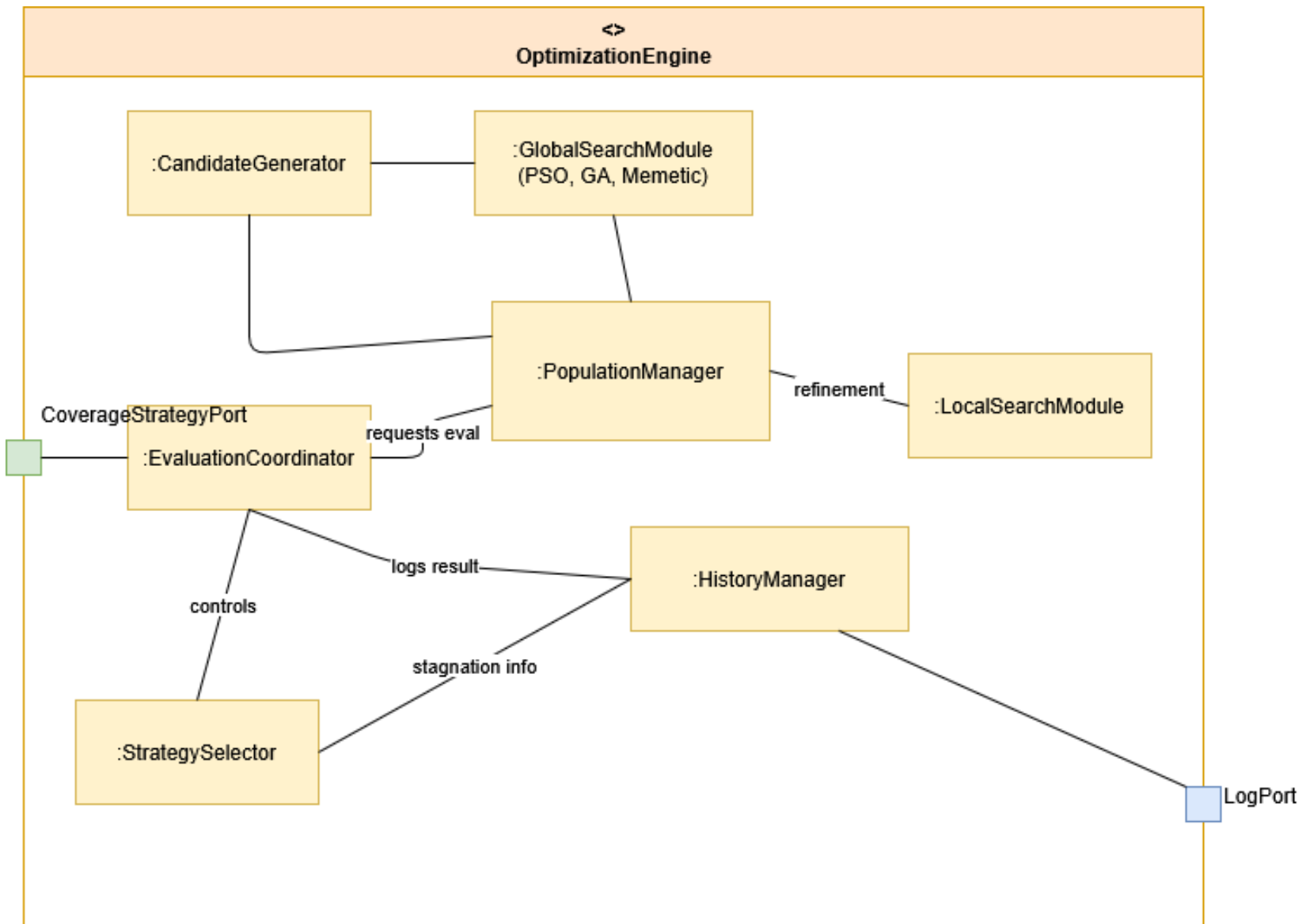


Рисунок 4.4 – Діаграма композитної структури компонента OptimizationEngine в UML-орієнтованій інформаційній технології максимального покриття

На діаграмі показано внутрішні частини рушія оптимізації, порти та з'єднання між модулями глобального й локального пошуку, координації оцінювання, вибору стратегій та керування історією. Описані структурні моделі визначають статичну організацію системи. У наступному підрозділі розглянуто поведінкові моделі, що описують динаміку роботи інформаційної технології.

4.4 Поведінкові моделі інформаційної технології

Поведінку системи можна також описати за допомогою UML-діаграм. У цьому підрозділі наведено такі діаграми, як діаграма діяльності, діаграма послідовності та діаграма станів. Разом з діаграмою варіантів використання, діаграмою класів, діаграмою компонентів та діаграмою композитної структури вони описують систему в повній мірі таким чином, щоб її можна було побудувати незалежно від технологій.

4.4.1 Діаграма діяльності

Діаграма діяльності описує робочий процес інформаційної технології від моменту постановки задачі до отримання остаточного рішення. Процес складається з трьох послідовних фаз: підготовки, грубої оптимізації та уточнення.

На фазі підготовки аналітик визначає сценарій (область покриття, типи об'єктів, обмеження та вимоги до точності), після чого відбувається імпорт та попередня обробка геоданих (діяльності LoadRegion, LoadForbiddenRegions, PreprocessGeometry). Далі налаштовується геометрична модель, обираються початковий метод обчислення покриття (типово BoundingBoxEstimator або MonteCarloEstimator) та алгоритм оптимізації (PSO, GA, меметичний або локальний пошук).

Фаза грубої оптимізації реалізується як ітераційний цикл: генерація початкової популяції (InitPopulation), оцінка покриття для всієї популяції

обраним грубим методом (`EvaluateCandidates`), врахування штрафів за порушення обмежень (`ApplyPenalties`), відбір кращих рішень та побудова нових конфігурацій (`SelectAndUpdate`) і перевірка критеріїв зупинки (`CheckCoarseStoppingCriteria`). Цикл повторюється до досягнення стабілізації покриття, після чого `CoverageStrategyManager` автоматично перемикає метод оцінювання на більш точний (`GeometryLibraryEstimator` або `TriangulationEstimator`).

На фазі уточнення до відібраних конфігурацій застосовується локальний пошук (`RefineBestConfigurations`), кандидати переоцінюються точним методом (`AccurateEvaluate`), контролюються фінальні критерії точності та обмежень (`CheckFinalCriteria`). Після проходження валідації (`ValidateFinalSolution`) формується остаточне рішення, виконується візуалізація покриття та експорт результатів у ГІС-формати. Обчислення покриття для різних конфігурацій можуть виконуватися паралельно на всіх фазах процесу. Діаграму діяльності процесу адаптивної оптимізації максимального покриття показано на рисунку 4.5:

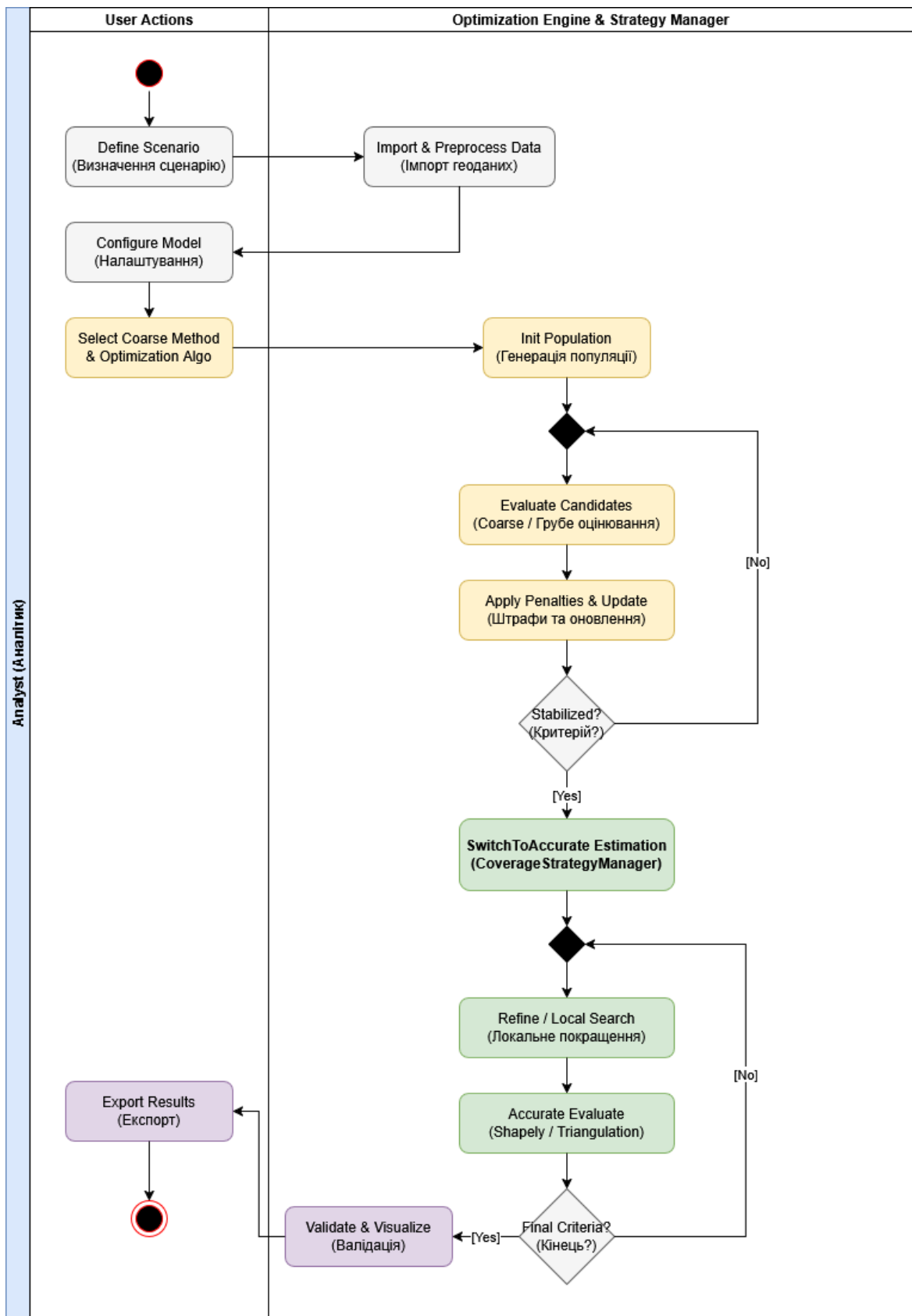


Рисунок 4.5 – Діаграма діяльності процесу адаптивної оптимізації
максимального покриття

Діаграма відображає послідовність кроків від завдання сценарію до валідації та експорту рішення, включно з перемиканням між грубими та точними методами оцінювання.

4.4.2 Діаграма послідовності

Діаграма послідовності деталізує динаміку взаємодії між основними об'єктами при запуску адаптивної оптимізації.

У взаємодії беруть участь User (аналітик), ScenarioController, OptimizationEngine, CoverageStrategyManager, один із CoverageEstimator (BoundingBoxEstimator, MonteCarloEstimator або GeometryLibraryEstimator), GeometryCore, PenaltyManager, ResultRepository та VisualizationService.

Типовий сценарій взаємодії: User ініціює оптимізацію через ScenarioController, який конфігурує OptimizationEngine. На початку CoverageStrategyManager обирає BoundingBoxEstimator. OptimizationEngine генерує початкові конфігурації через GeometryCore, для кожної з них викликає estimateCoverage() через обраний CoverageEstimator та computePenalty() через PenaltyManager. Найкращі конфігурації зберігаються у ResultRepository. Після досягнення грубих критеріїв зупинки CoverageStrategyManager перемикається на точніший метод (наприклад, GeometryLibraryEstimator), і цикл повторюється для відібраних конфігурацій. Після завершення OptimizationEngine повертає фінальне рішення, яке візуалізується через VisualizationService. Діаграму послідовності подано на рисунку 4.6:

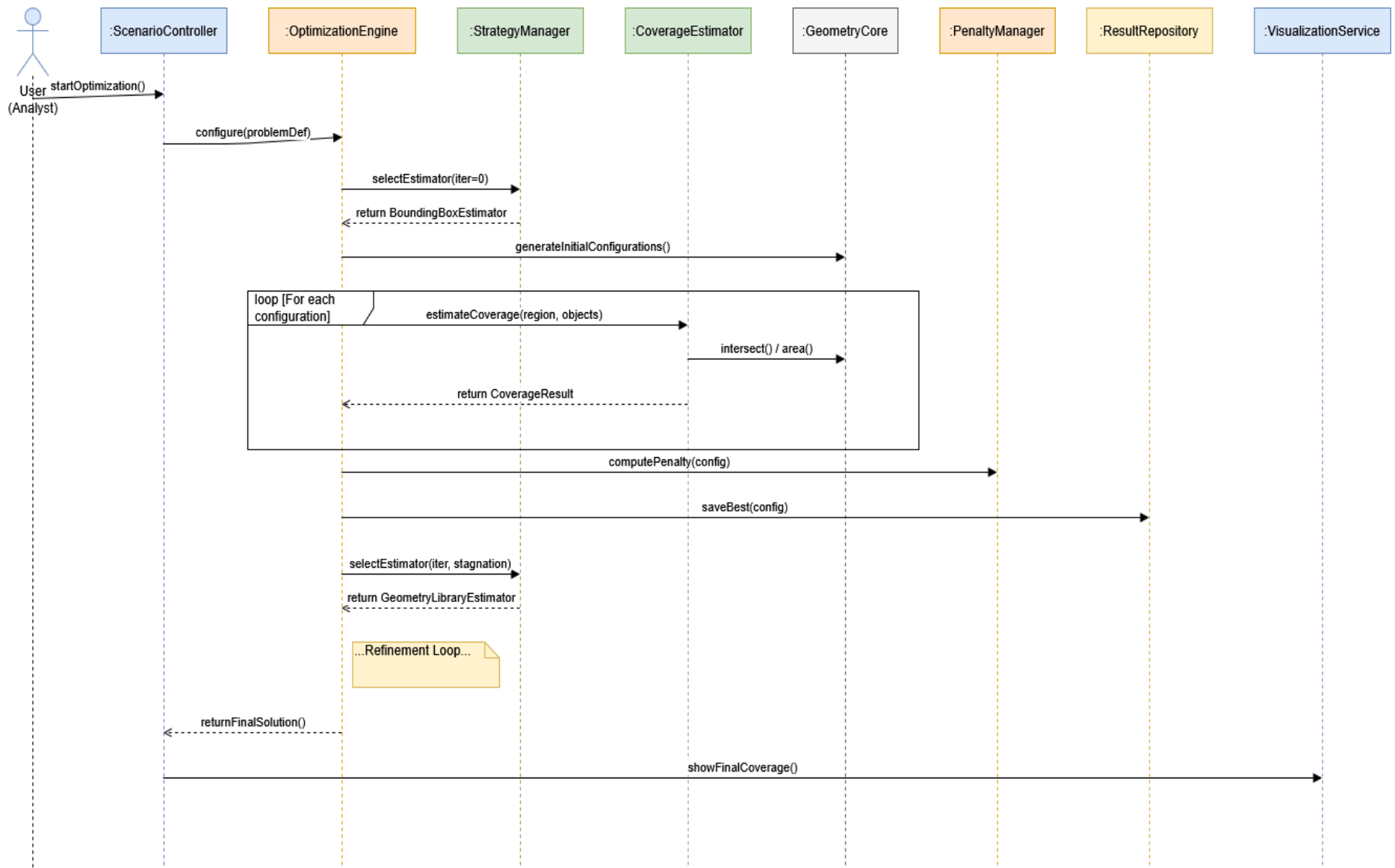


Рисунок 4.6 – Діаграма послідовності взаємодії компонентів

Діаграма демонструє динамічний обмін повідомленнями між користувачем, керуючим сценарієм, оптимізаційним ядром, менеджером стратегій покриття, геометричним ядром та сервісом візуалізації.

4.4.3 Діаграма станів

Діаграма станів описує життєвий цикл конфігурації покриваючих об'єктів у процесі адаптивної оптимізації. Модель визначає одинадцять станів, що відображають шлях конфігурації від генерації до фінального рішення.

Після створення конфігурації (стан `Generated`) вона проходить грубе оцінювання (`CoarseEvaluating` → `CoarseEvaluated`), за результатами якого або відкидається через низьку якість чи значні порушення обмежень (`RejectedByCoarse`), або відбирається для подальшого аналізу (`SelectedForRefinement`). Відібрані конфігурації оцінюються точним методом (`AccurateEvaluating` → `AccurateEvaluated`), після чого до них може бути застосований локальний пошук (`LocalImprovement`). Конфігурації, що демонструють найкращі результати, потрапляють до множини кандидатів (`BestCandidate`) і зберігаються в історії (`Archived`). На завершення оптимізаційного процесу одна конфігурація обирається як остаточне рішення (`FinalSolution`).

Переходи між станами ініціюються подіями оцінювання (`coarseEvaluate`, `accurateEvaluate`), перевірки перспективності (`isPromising`), локального покращення (`applyLocalSearch`) та фінального відбору (`selectAsFinal`). Така модель забезпечує контрольований перехід конфігурації через усі етапи оптимізації з можливістю раннього відсіювання неперспективних рішень. Діаграму станів подано на рисунку 4.7:

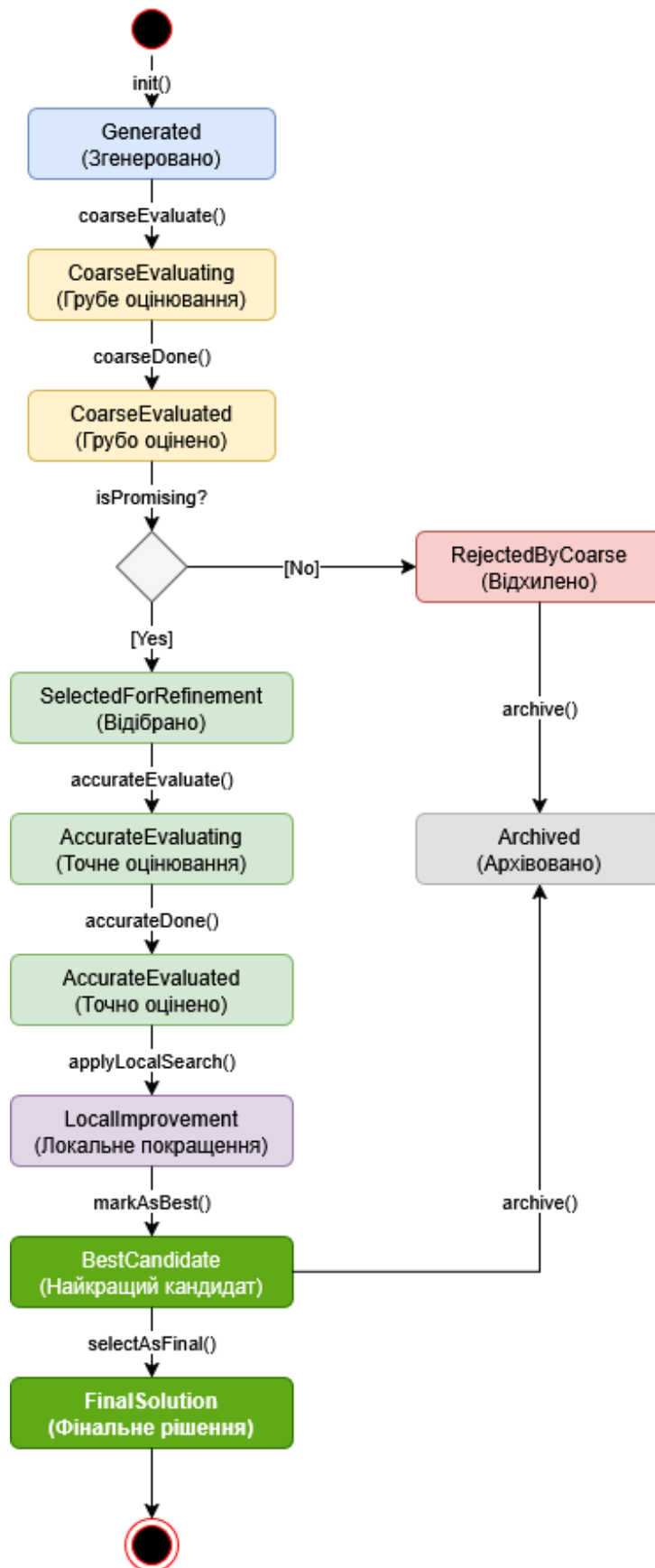


Рисунок 4.7 – Діаграма станів конфігурації покриваючих

Діаграма відображає переходи між станами генерації, грубого та точного оцінювання, локального покращення, відбору найкращих рішень та формування фінального результату.

Висновки до розділу 4

У розділі представлено цілісну UML-орієнтовану інформаційну технологію розв'язання неперервних задач максимального покриття з об'єктами довільної форми. Технологія формалізує архітектуру, структури даних, інформаційні потоки та алгоритмічні компоненти системи, забезпечуючи відтворюваність, масштабованість і прозорість процесу проєктування програмних рішень для задач покриття. Розроблений комплекс UML-діаграм охоплює структурні, поведінкові та інтеграційні аспекти системи, що дає змогу однозначно описати її логіку та підтримувати розширюваність.

Запропоновано повну архітектурну формалізацію задач максимального покриття засобами UML. Побудовано комплекс діаграм (Use Case, Class, Component, Activity, Sequence, State Machine, Composite Structure), який формує стандартизований архітектурний каркас системи. На відміну від існуючих досліджень, що концентруються лише на алгоритмах, технологія враховує повний життєвий цикл програмної системи. Введено універсальний метамодуль оцінювання площі покриття, що об'єднує п'ять незалежних стратегій обчислення та підтримує їх адаптивне перемикання. Така інтеграція вперше формалізована UML-моделями та дозволяє ефективно працювати зі складними геометріями та великими конфігураціями. Розроблено новий підхід до адаптивної оптимізації, який поєднує грубі й точні методи оцінювання, глобальні й локальні алгоритми, а також механізм динамічного вибору стратегії. Це створює гібридну оптимізаційну платформу, яка є науково унікальною. Запропоновано оригінальну модель життєвого циклу конфігурації у вигляді діаграми станів, яка формалізує переходи між станами оцінювання, покращення та архівації кандидатних рішень.

Сформовано композитну структуру оптимізаційного ядра, що визначає внутрішню організацію глобального пошуку, локального покращення, моніторингу та стратегічного керування. Це забезпечує масштабованість і відтворюваність архітектурних рішень.

Таким чином, розділ формує клас інформаційних технологій, у яких UML виступає не просто засобом документування, а фундаментом архітектурного проектування систем оптимізації покриття.

ВИСНОВКИ

1. На етапі дослідження проведено аналіз задач максимального геометричного покриття у територіально-розподілених системах, в результаті якого виявлено, що такі задачі характеризуються NP-важкістю, мультиекстремальним ландшафтом цільової функції та високою обчислювальною вартістю оцінки площі покриття для об'єктів довільної форми. Крім того, встановлено, що існуючі методи оптимізації переважно зорієнтовані на об'єкти простої форми, такі як кола або прямокутники, і не забезпечують комплексного розв'язання задачі неперервного максимального покриття із зонами обслуговування довільної конфігурації та заборонними зонами у рамках єдиної обчислювальної схеми. Також виявлено відсутність систематичного порівняння ройових метаевристик у поєднанні з методами локальної оптимізації для задач покриття та нерозглянутість питання архітектурної формалізації відповідної інформаційної технології. Показано, що для ефективного розв'язання поставленої задачі необхідно вирішити завдання на трьох рівнях: обчислення цільової функції з контрольованою точністю, глобальної та локальної оптимізації розміщення покриваючих об'єктів, а також побудови інтегрованої програмної платформи, що забезпечує уніфіковану взаємодію зазначених компонентів.

2. Основними науковими результатами, що були отримані в роботі, є наступні.

Проведено аналіз сучасного стану досліджень у галузі розв'язання неперервних задач максимального геометричного покриття та методів їх алгоритмічного забезпечення. Систематичний огляд літератури виконано за трьома напрямками: математичне моделювання задач покриття, методи еволюційно-ройової та гібридної оптимізації, а також прикладні аспекти задач покриття у територіально-розподілених системах. Розглянуто математичний апарат для роботи з об'єктами довільної форми. На основі проведеного аналізу виявлено обмеження існуючих підходів та визначено напрями подальшого

розвитку, що обґрунтовують доцільність розробки гібридних еволюційно-ройових методів оптимізації.

Сформовано обчислювальну модель неперервної задачі максимального геометричного покриття як задачі нелінійного програмування у просторі неперервних параметрів розміщення з переходом до безумовної оптимізації через механізм зовнішньої штрафної функції. Модель враховує об'єкти довільної форми, заборонені зони та умови належності полюсів області покриття, що створює теоретичну основу для розроблення алгоритмічних і програмних засобів гібридної оптимізації в територіально-розподілених системах.

Удосконалено методи обчислення критерію покриття шляхом створення багаторівневої адаптивної схеми оцінювання площі конфігурації покриття. Систематизовано п'ять методів, що утворюють впорядкований спектр від найшвидших із низькою точністю до найточніших із високою обчислювальною вартістю: метод мінімальних описаних прямокутників, стохастичний метод Монте-Карло, метод Монте-Карло з просторовою індексацією R-Tree, комбінаторну оцінку методом включення-виключення другого порядку та точні булеві операції засобами бібліотеки обчислювальної геометрії. Сформульовано рекомендації щодо керованого перемикання між наближеними та точними геометричними обчисленнями залежно від етапу оптимізації та кількості покриваючих об'єктів. Експериментально підтверджено, що метод Монте-Карло з R-Tree забезпечує похибку в межах 3% відносно Shapely при обчислювальній складності $O(N \log n)$, що створює методичну основу для побудови адаптивних обчислювальних схем.

Розроблено методологію розв'язання неперервних задач максимального геометричного покриття у територіально-розподілених системах, що ґрунтується на поєднанні еволюційно-ройових підходів із меметичними принципами. Чотири ройові метаевристики обчислювального інтелекту – оптимізацію роєм частинок, пошук риб'ячої зграї, алгоритм світлячків та алгоритм штучної бджолоїної колонії

– адаптовано до задачі максимального покриття з об'єктами довільної форми та заборонними зонами. Ройові алгоритми досягають площі покриття від 65% до 80% залежно від кількості покриваючих об'єктів та складності геометрії задачі. Обґрунтовано застосування квазіньютонівського методу BFGS як локальної компоненти гібридних алгоритмів. Розглянуто підхід до чисельного оцінювання градієнта цільової функції на основі матриці площ попарних перетинів покриваючих об'єктів, що забезпечує суттєве прискорення обчислень порівняно з прямим перерахунком площі складного об'єкта.

Створено універсальну алгоритмічну схему гібридизації популяційних методів оптимізації, яка забезпечує узгоджене керування глобальним стохастичним пошуком і локальним покращенням у просторі неперервних параметрів. Для кожної з чотирьох гібридних комбінацій (PSO+BFGS, FSS+BFGS, FA+BFGS, ABC+BFGS) визначено оптимальну частоту виклику локального пошуку, рекомендовані параметри та критерії зупинки. Показано, що гібридна схема є універсальною і може бути застосована до будь-якого з розглянутих ройових алгоритмів без зміни їхньої базової структури. Встановлено, що меметичний підхід стабільно підвищує площу покриття на 12–25% порівняно з чистими ройовими методами (від 12.2% при $n = 20$ до 24.7% при $n = 80$). При оптимізації гібридного підходу кількість ітерацій глобального пошуку може бути скорочена вдвічі, що зменшує час обчислень на 30–40% без втрати якості покриття.

Запропоновано нейросурогатні й нейронно-адаптивні механізми для прискорення обчислення цільової функції та автоматизованого керування обмеженнями в задачах геометричного покриття. Запропоновано нейронно-керовані гібридні методи за двома напрямками: адаптивне керування штрафним коефіцієнтом, що дозволяє автоматично балансувати між оптимізацією покриття та дотриманням геометричних обмежень залежно від поточного стану пошуку, та динамічне налаштування гіперпараметрів ройових алгоритмів під час їхньої

роботи, що усуває необхідність ручного підбору параметрів для кожної конкретної задачі. Для задач із забороненими зонами нейронно-керований адаптивний механізм забезпечує площу покриття 85.7–87.6% при повному дотриманні обмежень, а скорочення кількості ітерацій глобального пошуку з 500 до 200 при $n = 80$ зменшує час обчислень з 550 с до 150 с. На основі порівняльного аналізу трьох парадигм навчання – з учителем, без учителя та з підкріпленням – обґрунтовано вибір навчання з підкріпленням як найбільш перспективного підходу, що забезпечує автоматичне формування стратегії керування без потреби у попередньому зборі еталонних даних.

Побудовано UML-орієнтовану архітектуру інформаційної технології гібридної еволюційно-ройової оптимізації, яка формалізує структуру, поведінку та життєвий цикл оптимізаційної системи. Сформульовано функціональні та нефункціональні вимоги до системи, визначено атрибути якості. Обґрунтовано вибір трирівневої архітектурної моделі на основі методу багатокритеріального прийняття рішень TOPSIS з експертними оцінками альтернатив. Побудовано комплекс UML-діаграм, що формалізують структурні, поведінкові та інтеграційні аспекти системи. Введено універсальний метамодуль оцінювання площі покриття, що об'єднує п'ять незалежних стратегій обчислення та підтримує їх адаптивне перемикання. Розроблено оригінальну модель життєвого циклу конфігурації, що формалізує переходи між станами оцінювання, покращення та архівації кандидатних рішень. Запропонована технологія забезпечує масштабованість, відтворюваність та розширюваність програмної системи.

Проведено експериментальну валідацію розроблених моделей, методів та інформаційної технології на серіях задач зростаючої складності. Порівняльний аналіз показав, що PSO забезпечує найшвидшу збіжність для задач із відносно гладкою геометрією, FSS демонструє найвищу стійкість при наявності заборонених зон, FA характеризується природним механізмом фокусування на перспективних зонах, а ABC вирізняється ефективним механізмом виходу з

локальних екстремумів. Підтверджено, що гібридний метод стабільно перевершує як чистий ройовий пошук, так і мультістартову локальну оптимізацію при однаковому обчислювальному бюджеті. Гібридний PSO+BFGS досягає площі покриття 89.6–89.9% при $n = 20 - 80$, тоді як чистий PSO – лише 65.1–77.4%, при цьому чисті ройові алгоритми мають поріг ефективності на рівні 82–85%. Ефективність розроблених методів за показниками якості покриття, швидкодії та обчислювальної стійкості підтверджує їх практичну придатність для задач різної розмірності та геометричної складності.

Таким чином, усі завдання дисертаційної роботи виконано в повному обсязі. Для розроблених методів та моделей побудовано UML-орієнтовану архітектурну специфікацію, що визначає структуру, поведінку та інтерфейси програмних компонентів системи. Запропонована архітектура з модульною організацією забезпечує можливість незалежного розширення та заміни окремих алгоритмічних стратегій, а також створює формалізовану основу для подальшої програмної реалізації інформаційної технології.

3. Отримані результати підтверджують можливість практичного використання запропонованих методів та інформаційної технології для розв'язання задач максимального геометричного покриття у територіально-розподілених системах. Обґрунтовано, що систематизовані методи обчислення площі покриття дозволяють ефективно оцінювати якість конфігурацій для об'єктів довільної форми при різних вимогах до точності та швидкодії. Підтверджено, що меметичний підхід забезпечує підвищення площі покриття на 12-25% порівняно з чистими ройовими методами, а оптимізація кількості ітерацій глобального пошуку дозволяє скоротити час обчислень на 30-40%. Показано, що нейронно-керовані механізми адаптації дозволяють автоматизувати процес налаштування оптимізаційних алгоритмів та забезпечують площу покриття 85-88% для задач із забороненими зонами при скороченні часу обчислень утричі. Розроблена UML-орієнтована інформаційна технологія забезпечує уніфіковану

платформу для інтеграції зазначених методів у єдину масштабовану програмну систему.

4. Достовірність отриманих наукових положень, висновків та рекомендацій забезпечується коректністю математичних моделей, аргументованими результатами обчислювальних експериментів та підтверджується співставленням з результатами існуючих досліджень. Теоретичну базу дослідження складають методи нелінійної оптимізації, обчислювальної геометрії, теорії ω -функцій, ройового інтелекту, меметичних алгоритмів, теорії нейронних мереж та навчання з підкріпленням, а також методи програмної інженерії та UML-моделювання. Обчислювальні експерименти проведено на серіях задач зростаючої складності.

5. Результати, що були отримані в роботі, можуть бути рекомендовані до застосування в науково-дослідних та проектних організаціях для розв'язання задач просторового планування та оптимізації сервісного покриття у територіально-розподілених системах, зокрема при розгортанні сенсорних мереж, плануванні розміщення об'єктів обслуговування, оптимізації аварійного реагування та в геоінформаційних системах. Сформульовані рекомендації щодо вибору та налаштування гібридних методів залежно від геометричної структури задачі, її розмірності та наявності заборонених зон створюють методичну основу для практичного застосування вдосконалених метаевристик.

6. Виходячи з наведених у дисертації наукових і практичних результатів, а також підтвердження факту їх достовірності, наукової та практичної значущості, дають змогу вважати, що сформульована наукова задача розробки моделей, методів та інформаційної технології гібридної еволюційно-ройової оптимізації для підвищення ефективності розв'язання задач максимального геометричного покриття у територіально-розподілених системах – розв'язаною, а поставлену мету – досягнутою.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Church R. L., ReVelle C. The maximal covering location problem. *Papers of the Regional Science Association*. 1974. Vol. 32. P. 101–118. DOI: 10.1111/j.1435-5597.1974.tb00902.x.
2. Murray A. T. Maximal coverage location problem: impacts, significance, and evolution. *International Regional Science Review*. 2016. Vol. 39. P. 5–27. DOI: 10.1177/0160017615600222.
3. Owen S. H., Daskin M. S. Strategic facility location: a review. *European Journal of Operational Research*. 1998. Vol. 111. P. 423–447. DOI: 10.1016/S0377-2217(98)00186-6.
4. Facility Location: Applications and Theory / Eds. Z. Drezner, H. W. Hamacher. Berlin : Springer, 2002. 330 p.
5. Church R. L., Murray A. T. Business Site Selection, Location Analysis, and GIS. Hoboken, NJ : John Wiley & Sons, 2009. 328 p. DOI: 10.1002/9780470432761.
6. Murray A. T. Advances in location modeling: GIS linkages and contributions. *Journal of Geographical Systems*. 2010. Vol. 12. P. 335–354. DOI: 10.1007/s10109-009-0105-9.
7. Foundations of Location Analysis / Eds. H. A. Eiselt, V. Marianov. New York : Springer, 2011. 522 p. DOI: 10.1007/978-1-4419-7572-0.
8. Daskin M. S. What you should know about location modeling. *Naval Research Logistics*. 2008. Vol. 55, No. 4. P. 283–294. DOI: 10.1002/nav.20284.
9. Hochbaum D. S. Approximating covering and packing problems. *Approximation Algorithms for NP-Hard Problems*. Boston : PWS Publishing Co., 1996. P. 94–143.
10. Bansal M., Kianfar K. Planar maximum coverage location problem with partial coverage and rectangular demand and service zones. *INFORMS Journal on Computing*. 2017. Vol. 29. P. 152–169. DOI: 10.1287/ijoc.2016.0722.

11. Cordeau J. F., Furini F., Ljubic I. Benders decomposition for very large scale partial set covering and maximal covering location problems. *European Journal of Operational Research*. 2019. Vol. 275, No. 3. P. 882–896. DOI: 10.1016/j.ejor.2018.12.021.
12. Arana-Jimenez M., Blanco V., Fernandez E. On the fuzzy maximal covering location problem. *European Journal of Operational Research*. 2020. Vol. 283, No. 2. P. 692–705. DOI: 10.1016/j.ejor.2019.11.036.
13. Berman O., Kalcsics J., Krass D. On covering location problems on networks with edge demand. *Computers and Operations Research*. 2016. Vol. 74. P. 214–227. DOI: 10.1016/j.cor.2015.04.005.
14. Baldomero-Naranjo M., Kalcsics J., Rodriguez-Chia A. M. Minmax regret maximal covering location problems with edge demands. *Computers and Operations Research*. 2021. Vol. 130. P. 105181. DOI: 10.1016/j.cor.2020.105181.
15. Drezner Z., Kalczyński P., Salhi S. The planar multiple obnoxious facilities location problem: a Voronoi based heuristic. *Omega*. 2019. Vol. 87. P. 105–116. DOI: 10.1016/j.omega.2018.08.013.
16. Yang X.-S., He X. Firefly algorithm: recent advances and applications. *International Journal of Swarm Intelligence*. 2013. Vol. 1, No. 1. P. 36–50. DOI: 10.1504/IJSI.2013.055801.
17. Ozturk O. T., Karasakal E. K., Karasakal O. A partial coverage hierarchical location allocation model for health services. *European Journal of Industrial Engineering*. 2023. Vol. 17, No. 1. DOI: 10.1504/EJIE.2023.127742.
18. Blanco V., Gazquez R. Continuous maximal covering location problems with interconnected facilities. *Computers & Operations Research*. 2021. Vol. 132. P. 105310. DOI: 10.1016/j.cor.2021.105310.
19. Allahyari M. Z., Azab A. Mathematical modeling and multi-start search simulated annealing for unequal-area facility layout problem. *Expert Systems with Applications*. 2018. Vol. 91. P. 46–62. DOI: 10.1016/j.eswa.2017.07.049.

20. Davari S., Zarandi M. H. F., Hemmati A. Maximal covering location problem (MCLP) with fuzzy travel times. *Expert Systems with Applications*. 2011. Vol. 38. P. 14535–14541. DOI: 10.1016/j.eswa.2011.05.031.
21. Church R. L. The planar maximal covering location problem. *Journal of Regional Science*. 1984. Vol. 24. P. 185–201. DOI: 10.1111/j.1467-9787.1984.tb01031.x.
22. Murray A. T., Matisziw T. C., Wei H., Tong D. A geocomputational heuristic for coverage maximization. *Transactions in GIS*. 2008. Vol. 12. P. 757–773. DOI: 10.1111/j.1467-9671.2008.01125.x.
23. Murray A. T., Tong D. Coverage optimization in continuous space facility siting. *International Journal of Geographical Information Science*. 2007. Vol. 21. P. 757–776. DOI: 10.1080/13658810601169857.
24. Murray A. T., O'Kelly M. E., Church R. L. Regional service coverage modeling. *Computers & Operations Research*. 2008. Vol. 35. P. 339–355. DOI: 10.1016/j.cor.2006.03.004.
25. Tong D., Murray A. T. Maximising coverage of spatial demand for service. *Papers in Regional Science*. 2009. Vol. 88. P. 85–97. DOI: 10.1111/j.1435-5957.2008.00168.x.
26. Matisziw T. C., Murray A. T. Siting a facility in continuous space to maximize coverage of a region. *Socio-Economic Planning Sciences*. 2009. Vol. 43. P. 131–139. DOI: 10.1016/j.seps.2008.02.009.
27. Coll N., Fort M., Sellares J. A. On the overlap area of a disk and a piecewise circular domain. *Computers & Operations Research*. 2019. Vol. 104. P. 59–73. DOI: 10.1016/j.cor.2018.11.007.
28. Mount D. M., Silverman R., Wu A. Y. On the area of overlap of translated polygons. *Computer Vision and Image Understanding*. 1996. Vol. 64. P. 53–61. DOI: 10.1006/cviu.1996.0045.

29. Coll N., Fort M., Saus M. Coverage area maximization with parallel simulated annealing. *Expert Systems with Applications*. 2022. Vol. 202. P. 117185. DOI: 10.1016/j.eswa.2022.117185.
30. Blanco V., Gazquez R., Saldanha-da-Gama F. Multi-type maximal covering location problems: hybridizing discrete and continuous problems. *European Journal of Operational Research*. 2023. Vol. 307, No. 3. P. 1040–1054. DOI: 10.1016/j.ejor.2022.10.037.
31. Yakovlev S., Kartashov O., Podzeha D. Mathematical models and nonlinear optimization in continuous maximum coverage location problem. *Computation*. 2022. Vol. 10. P. 119. DOI: 10.3390/computation10070119.
32. Wei R., Murray A. T. Continuous space maximal coverage: insights, advances and challenges. *Computers & Operations Research*. 2015. Vol. 62. P. 325–336. DOI: 10.1016/j.cor.2014.04.010.
33. Tedeschi D., Andretta M. New exact algorithms for planar maximum covering location by ellipses problems. *European Journal of Operational Research*. 2021. Vol. 291, No. 1. P. 114–127. DOI: 10.1016/j.ejor.2020.09.029.
34. He Z., Fan B., Cheng T., Wang S. Y., Tan C. H. A mean-shift algorithm for large-scale planar maximal covering location problems. *European Journal of Operational Research*. 2016. Vol. 250, No. 1. P. 65–76. DOI: 10.1016/j.ejor.2015.09.006.
35. Krieg H., Seidel T., Schwientek J., Kufer K. H. Solving continuous set covering problems by means of semi-infinite optimization. *Mathematical Methods of Operations Research*. 2022. Vol. 96, No. 1. P. 39–82. DOI: 10.1007/s00186-022-00776-y.
36. Rvachev V. L. Theory of the R-Function and Some of Its Applications. Kyiv : Naukova Dumka, 1982. 552 p.
37. Stoyan Y., Scheithauer G., Gil N., Romanova T. Phi-functions for complex 2D-objects. *4OR*. 2004. Vol. 2. P. 69–84.

38. Bennell J., Scheithauer G., Stoyan Y., Romanova T. Tools of mathematical modeling of arbitrary object packing problems. *Annals of Operations Research*. 2010. Vol. 179. P. 343–368. DOI: 10.1007/s10479-008-0456-5.
39. Stoyan Y. G., Yakovlev S. V. Configuration space of geometric objects. *Cybernetics and Systems Analysis*. 2018. Vol. 54. P. 716–726. DOI: 10.1007/s10559-018-0073-5.
40. Yakovlev S. V. On some classes of spatial configurations of geometric objects and their formalization. *Journal of Automation and Information Sciences*. 2018. Vol. 50. P. 38–50. DOI: 10.1615/JAutomatInfScien.v50.i9.30.
41. Yakovlev S. V. Formalizing spatial configuration optimization problems with the use of a special function class. *Cybernetics and Systems Analysis*. 2019. Vol. 55. P. 581–589. DOI: 10.1007/s10559-019-00167-y.
42. Yakovlev S. V. The concept of modeling packing and covering problems using computational geometry software. *Cybernetics and Systems Analysis*. 2023. Vol. 59, No. 1. P. 108–119. DOI: 10.1007/s10559-023-00547-5.
43. Kiseleva E. M., Kadochnikova Y. E. Solving a continuous single-product problem of optimal partitioning with additional conditions. *Journal of Automation and Information Sciences*. 2009. Vol. 41. P. 48–63. DOI: 10.1615/JAutomatInfScien.v41.i7.30.
44. Kiseleva E. M. The emergence and formation of the theory of optimal set partitioning. *Journal of Automation and Information Sciences*. 2018. Vol. 50. P. 1–24. DOI: 10.1615/JAutomatInfScien.v50.i9.10.
45. Kiseleva E. M., Hart L. L., Prytomanova O. M., Zhuravel S. V. Construction of a generalized Voronoi diagram with optimal placement of generator points. *Matematychni Studii*. 2020. Vol. 53. P. 109–112. DOI: 10.30970/ms.53.1.109-112.

46. Stoyan Y. G., Patsuk V. Covering a compact polygonal set by identical circles. *Computational Optimization and Applications*. 2010. Vol. 46. P. 75–92. DOI: 10.1007/s10589-008-9191-8.
47. Stoyan Y. G., Romanova T., Scheithauer G., Krivulya A. Covering a polygonal region by rectangles. *Computational Optimization and Applications*. 2011. Vol. 48. P. 675–695. DOI: 10.1007/s10589-009-9258-1.
48. Pankratov A., Romanova T., Litvinchev I., Marmolejo-Saucedo J. A. An optimized covering spheroids by spheres. *Applied Sciences*. 2020. Vol. 10. P. 1846. DOI: 10.3390/app10051846.
49. Hu X., Zhu W., Ma H., An B., Zhi Y., Wu Y. Orientational variable-length strip covering problem: a branch-and-price-based algorithm. *European Journal of Operational Research*. 2021. Vol. 289, No. 1. P. 254–269. DOI: 10.1016/j.ejor.2020.07.003.
50. Yakovlev S., Shekhovtsov S., Kirichenko L., Matsyi O., Podzaha D., Chumachenko D. Continuous maximum coverage location problem with arbitrary service shapes. *Symmetry*. 2025. Vol. 17, No. 5. P. 676. DOI: 10.3390/sym17050676.
51. Yakovlev S. V., Kartashov O. V., Mumrienko O. A. Formalization and solution of the maximum area coverage problem using library Shapely for territory monitoring. *Radioelectronic and Computer Systems*. 2022. No. 2(102). P. 35–48. DOI: 10.32620/reks.2022.2.03.
52. Yakovlev S., Kiseleva O., Chumachenko D., Podzaha D. Maximum service coverage in business site selection using computer geometry software. *Electronics*. 2023. Vol. 12, No. 10. P. 2329. DOI: 10.3390/electronics12102329.
53. Gillies S. The Shapely User Manual. 2023. URL: <https://shapely.readthedocs.io/en/stable/manual.html> (дата звернення: 15.01.2026).

54. Fletcher R. *Practical Methods of Optimization*. 2nd ed. New York : John Wiley & Sons, 2013. 436 p.
55. Liu D. C., Nocedal J. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*. 1989. Vol. 45, No. 1–3. P. 503–528. DOI: 10.1007/BF01589116.
56. Berahas A. S., Curtis F. E., Zhou B. Limited-memory BFGS with displacement aggregation. *Mathematical Programming*. 2022. Vol. 194. P. 121–157. DOI: 10.1007/s10107-021-01621-6.
57. Conn A. R., Gould N. I. M., Toint P. L. *Trust-Region Methods*. Philadelphia : SIAM, 2000. 959 p. DOI: 10.1137/1.9780898719857.
58. Lagarias J. C., Reeds J. A., Wright M. H., Wright P. E. Convergence properties of the Nelder–Mead simplex method in low dimensions. *SIAM Journal on Optimization*. 1998. Vol. 9, No. 1. P. 112–147. DOI: 10.1137/S1052623496303470.
59. Gao F., Han L. Implementing the Nelder–Mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications*. 2012. Vol. 51. P. 259–277. DOI: 10.1007/s10589-010-9329-3.
60. Glover F., Sorensen K. Metaheuristics. *Scholarpedia*. 2015. Vol. 10. P. 6532. DOI: 10.4249/scholarpedia.6532.
61. Salhi S., Thompson J. An overview of heuristics and metaheuristics. *The Palgrave Handbook of Operations Research*. London : Palgrave Macmillan, 2022. DOI: 10.1007/978-3-030-96935-6_11.
62. Kennedy J., Eberhart R. Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks (ICNN'95)*. Perth, 1995. P. 1942–1948. DOI: 10.1109/ICNN.1995.488968.
63. Poli R., Kennedy J., Blackwell T. Particle swarm optimization: an overview. *Swarm Intelligence*. 2007. Vol. 1. P. 33–57. DOI: 10.1007/s11721-007-0002-0.

64. Rosso M., Cucuzza R., Aloisio A., Marano G. C. Cumulative major advances in particle swarm optimization from 2018 to the present. *Archives of Computational Methods in Engineering*. 2025. DOI: 10.1007/s11831-024-10185-5.
65. Shami T. M., El-Saleh A. A., Alswaitti M., Al-Tashi Q., Summakieh M. A., Mirjalili S. Particle swarm optimization: a comprehensive survey. *IEEE Access*. 2022. Vol. 10. P. 10031–10061. DOI: 10.1109/ACCESS.2022.3142859.
66. Sengupta S., Basak S., Peters R. A. Particle swarm optimization: a survey of historical and recent developments with hybridization perspectives. *Machine Learning and Knowledge Extraction*. 2019. Vol. 1. P. 157–191. DOI: 10.3390/make1010010.
67. Tarekegn Nigatu A., Ferede T. Convergence analysis of particle swarm optimization algorithms for different constriction factors. *Frontiers in Applied Mathematics and Statistics*. 2024. Vol. 9. P. 1304268. DOI: 10.3389/fams.2024.1304268.
68. Xu L., Li Z., Li S., Xiao N. PSO with adaptive weighted delay velocity (PSO-AWDV) for global optimization. *Systems Science & Control Engineering*. 2021. Vol. 9, No. 1. P. 164–178. DOI: 10.1080/21642583.2021.1891153.
69. Li X., Li B. An adaptive dynamic particle swarm optimizer with local search. *Expert Systems with Applications*. 2022. Vol. 203. P. 117115.
70. Zambrano-Bigiarini M., Clerc M., Rojas R. Standard particle swarm optimisation 2011 at CEC-2013. *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2013)*. Cancun, 2013. P. 2337–2344. DOI: 10.1109/CEC.2013.6557848.
71. Karaboga D., Basturk B. Artificial bee colony (ABC) algorithm. *Journal of Global Optimization*. 2007. Vol. 39. P. 459–471. DOI: 10.1007/s10898-007-9149-x.
72. Akay B., Karaboga D. A modified artificial bee colony algorithm for real-parameter optimization. *Applied Soft Computing*. 2012. Vol. 12, No. 1. P. 342–359. DOI: 10.1016/j.asoc.2011.08.038.

73. Bastos-Filho C. J. A., de Lima-Neto F. B., Lins A. J. C. C., Nascimento A. I. S., Lima M. P. Fish school search. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC 2008)*. Singapore, 2008. P. 3069–3074.
74. Jamal A., Mohamed A.-W. An improved fish school search algorithm for global optimization. *Expert Systems with Applications*. 2021. Vol. 168. P. 114356.
75. Yang X.-S. Firefly algorithms for multimodal optimization. *Lecture Notes in Computer Science*. 2009. Vol. 5792. P. 169–178. DOI: 10.1007/978-3-642-04944-6_14.
76. Fister I., Yang X.-S., Brest J., Fister D. A comprehensive review of firefly algorithm variants. *Swarm and Evolutionary Computation*. 2013. Vol. 13. P. 34–46. DOI: 10.1016/j.swevo.2013.06.001.
77. Storn R., Price K. Differential evolution. *Journal of Global Optimization*. 1997. Vol. 11, No. 4. P. 341–359. DOI: 10.1023/A:1008202821328.
78. Dorigo M., Stutzle T. Ant Colony Optimization. Cambridge, MA : MIT Press, 2004. 305 p. DOI: 10.7551/mitpress/1290.001.0001.
79. Yang X.-S. Mathematical analysis of nature-inspired algorithms. Nature-Inspired Algorithms and Applied Optimization / Ed. X.-S. Yang. Cham : Springer, 2018. P. 1–25. DOI: 10.1007/978-3-319-67669-2_1.
80. Mirjalili S., Mirjalili S. M., Lewis A. Grey wolf optimizer. *Advances in Engineering Software*. 2014. Vol. 69. P. 46–61. DOI: 10.1016/j.advengsoft.2013.12.007.
81. Handbook of Memetic Algorithms / Eds. F. Neri, C. Cotta, P. Moscato. Berlin : Springer, 2012. 370 p. DOI: 10.1007/978-3-642-23247-3.
82. Neri F., Cotta C. Memetic algorithms and memetic computing optimization: a literature review. *Swarm and Evolutionary Computation*. 2012. Vol. 2. P. 1–14. DOI: 10.1016/j.swevo.2011.11.003.

83. Molina D., Lozano M., Herrera F. Memetic algorithms for continuous optimisation based on local search chains. *Evolutionary Computation*. 2015. Vol. 23, No. 1. P. 1–28.
84. Cotta C. Harnessing memetic algorithms. *TOP*. 2025. DOI: 10.1007/s11750-024-00694-8.
85. Liu B., Wang L., Jin Y. An effective hybrid PSO with local search for global optimization. *Applied Soft Computing*. 2016. Vol. 49. P. 563–580.
86. Forrester A., Sobester A., Keane A. Engineering Design via Surrogate Modelling. Chichester : John Wiley & Sons, 2008. 228 p. DOI: 10.1002/9780470770801.
87. Jin Y., Wang H., Chugh T., Guo D., Miettinen K. Data-driven evolutionary optimization. *IEEE Transactions on Evolutionary Computation*. 2019. Vol. 23, No. 3. P. 442–458. DOI: 10.1109/TEVC.2018.2869001.
88. Raissi M., Perdikaris P., Karniadakis G. E. Physics-informed neural networks. *Journal of Computational Physics*. 2019. Vol. 378. P. 686–707. DOI: 10.1016/j.jcp.2018.10.045.
89. Goodfellow I., Bengio Y., Courville A. Deep Learning. Cambridge, MA : MIT Press, 2016. 800 p.
90. Zaheer M., Kottur S., Ravanbakhsh S., Póczos B., Salakhutdinov R., Smola A. J. Deep sets. *Advances in Neural Information Processing Systems*. 2017. Vol. 30. P. 3391–3401.
91. Murray A. T. Optimising the spatial location of urban fire stations. *Fire Safety Journal*. 2013. Vol. 62. P. 64–71. DOI: 10.1016/j.firesaf.2013.03.002.
92. Fajardo Calderín J. F., Nodarse C. P., Yera L. S., Rodríguez D. E. E. Software tool for model and solve the maximum coverage location problem, case study: locations police officers. *Investigación Operacional*. 2017. Vol. 38. P. 141–149.
93. Rizeei H. M., Pradhan B., Saharkhiz M. A. Allocation of emergency response centres in response to pluvial flooding-prone demand points. *International*

Journal of Disaster Risk Reduction. 2019. Vol. 38. P. 101205. DOI: 10.1016/j.ijdrr.2019.101205.

94. Yakovlev S., Kirichenko L., Matsyi O., Kirpich A., Chumachenko D. Optimization of mobile medical service locations. *Proceedings of the IADIS International Conference on Information Systems (IS 2025)*. Porto, 2025. P. 538–541.

95. Akyildiz I. F., Su W., Sankarasubramaniam Y., Cayirci E. Wireless sensor networks: a survey. *Computer Networks*. 2002. Vol. 38, No. 4. P. 393–422. DOI: 10.1016/S1389-1286(01)00302-4.

96. Leichenko K., Skorobohatko S., Fesenko H., Kharchenko V., Yakovlev S. Assessment of the reliability of wireless sensor networks for forest fire monitoring. *Cybernetics and Systems Analysis*. 2025. Vol. 61, No. 1. P. 137–147. DOI: 10.1007/s10559-025-00753-3.

97. Skorobohatko S., Fesenko H., Kharchenko V., Yakovlev S. Architecture and reliability models of hybrid sensor networks for environmental monitoring. *Cybernetics and Systems Analysis*. 2024. Vol. 60, No. 2. P. 293–304. DOI: 10.1007/s10559-024-00670-x.

98. Welscher F., Bulbul R., Scholz J., Lederer P. Optimising antenna positioning for maximum coverage: LoRa cattle tracking. *Lecture Notes in Computer Science*. 2022. Vol. 12238. P. 61–70. DOI: 10.1007/978-3-031-06245-2_6.

99. Cortes J., Martinez S., Karatas T., Bullo F. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*. 2004. Vol. 20, No. 2. P. 243–255. DOI: 10.1109/TRA.2004.824698.

100. Schwager M., Rus D., Slotine J.-J. Decentralized, adaptive coverage control for networked robots. *The International Journal of Robotics Research*. 2009. Vol. 28, No. 7. P. 735–752. DOI: 10.1177/0278364908100177.

101. Choset H. Coverage for robotics: a survey of recent results. *Annals of Mathematics and Artificial Intelligence*. 2001. Vol. 31, No. 1–4. P. 113–126. DOI: 10.1023/A:1016639210559.

102. Low K. H., Dolan J. M., Khosla P. Adaptive multi-robot wide-area exploration and mapping. *Autonomous Robots*. 2009. Vol. 27, No. 2. P. 129–148.
103. Rajesh Chauhan D., Unnikrishnan A., Figliozzi M. A., Boyles S. D. Robust multi-period maximum coverage drone facility location problem. *Transportation Research Record*. 2022. Vol. 2677. P. 98–114. DOI: 10.1177/03611981221087240.
104. Zafar U., Bayram I. S., Bayhan S. A GIS-based optimal facility location framework for fast EV charging stations. *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE)*. Kyoto, 2021. DOI: 10.1109/ISIE45552.2021.9576448.
105. Fujii N., Watanabe R., Kokuryo D., Kaihara T., Son M., Doi M., Yoshida Y., Suzuki Y. A study on store location optimization in underground shopping street. *Procedia CIRP*. 2022. Vol. 112. P. 17–21. DOI: 10.1016/j.procir.2022.09.017.
106. Chang X., Li J. Business performance prediction in location-based social commerce. *Expert Systems with Applications*. 2019. Vol. 126. P. 112–123. DOI: 10.1016/j.eswa.2019.01.086.
107. Longley P. A., Goodchild M. F., Maguire D. J., Rhind D. W. *Geographic Information Systems and Science*. 4th ed. Chichester : John Wiley & Sons, 2015. 496 p.
108. Gerrard R. A., Church R. L., Stoms D. M., Davis F. W. Selecting conservation reserves using species-covering models. *Transactions in GIS*. 1997. Vol. 2. P. 45–60. DOI: 10.1111/j.1467-9671.1997.tb00004.x.
109. Garcia-Palomares J. C., Gutierrez J., Latorre M. Optimizing the location of stations in bike-sharing programs: a GIS approach. *Applied Geography*. 2012. Vol. 35. P. 235–246. DOI: 10.1016/j.apgeog.2012.07.002.
110. Pelegrin M., Puerto J., Rufian-Lizana A. Continuous covering on networks: improved MIP formulations. *Omega*. 2023. Vol. 117. P. 102835. DOI: 10.1016/j.omega.2023.102835.

111. Guttman A. R-Trees: a dynamic index structure for spatial searching. *Proceedings of the ACM SIGMOD International Conference on Management of Data*. Boston, 1984. P. 47–57. DOI: 10.1145/971697.602266.
112. Bentley J. L., Ottmann T. A. Algorithms for reporting and counting geometric intersections. *IEEE Transactions on Computers*. 1979. Vol. C-28, No. 9. P. 643–647. DOI: 10.1109/TC.1979.1675432.
113. Pichugina O., Kirichenko L., Koliechkina L., Skob Yu., Havryliuk Ye. Exploring one linear permutation-based stochastic optimization problem with applications. *Proceedings of the 13th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2025)*. Gliwice, Poland, 4–6 September 2025. DOI: 10.1109/IDAACS68557.2025.11322262.
114. Uzlov D., Havryliuk Y., Hushchyn I., Strukov V., Yakovlev S. Handling outliers in swarm algorithms: a review. *CEUR Workshop Proceedings*. 2024. Vol. 3777. P. 430–437. URL: <https://ceur-ws.org/Vol-3777/short13.pdf>.
115. Гаврилук Є. А. Гібридні ройові метаевристики для неперервних задач максимального покриття зі складними формами об'єктів. *Проблеми керування та інформатики*. 2026. № 1. С. 14–33. DOI: 10.34229/1028-0979-2026-1-2.
116. Гаврилук Є. А., Струков В. М., Узлов Д. Ю. Інтегрований фреймворк на основі методів штучного інтелекту для задач неперервного покриття з геометричними обмеженнями. *Системні технології*. 2026. Т. 1, № 162. С. 160–176. DOI: 10.34185/1562-9945-5-162-2026-18.
117. Гаврилук Є. А. Нейронно-керований гібридний метод для оптимального покриття території з обмеженням на розташування. *Системи та технології*. 2025. Т. 70, № 2. С. 112–118. DOI: 10.32782/2521-6643-2025-2-70.12.
118. Yakovlev S., Havryliuk Y., Matsyi O., Hulianytskyi A., Kirpich A. A neural network-driven adaptive approach for maximum coverage location with

restricted zones. *CEUR Workshop Proceedings*. 2025. Vol. 4164. P. 312–322. URL: <https://ceur-ws.org/Vol-4164/paper20.pdf>.

119. Kingma D. P., Welling M. Auto-encoding variational Bayes. *Proceedings of the 2nd International Conference on Learning Representations (ICLR 2014)*. Banff, 2014.

120. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. 2nd ed. Cambridge, MA : MIT Press, 2018. 552 p.

121. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal policy optimization algorithms. *arXiv preprint*. 2017. arXiv:1707.06347.

122. Гаврилюк Є. А., Коробчинський К. П. UML-орієнтована інформаційна технологія для неперервних задач максимального покриття з об'єктами довільної форми. *Вісник Харківського національного університету імені В. Н. Каразіна. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2025. Вип. 67. С. 18-34. DOI: 10.26565/2304-6201-2025-67.

123. Hwang C. L., Yoon K. Multiple Attribute Decision Making: Methods and Applications. New York : Springer, 1981. 259 p. DOI: 10.1007/978-3-642-48318-9.

124. Behzadian M., Khanmohammadi Otaghsara S., Yazdani M., Ignatius J. A state-of-the-art survey of TOPSIS applications. *Expert Systems with Applications*. 2012. Vol. 39, No. 17. P. 13051–13069. DOI: 10.1016/j.eswa.2012.05.056.

125. Saaty T. L. The Analytic Hierarchy Process. New York : McGraw-Hill, 1980. 287 p. DOI: 10.21236/ADA214804.

126. Saaty T. L. Decision making with the analytic hierarchy process. *International Journal of Services Sciences*. 2008. Vol. 1, No. 1. P. 83–98. DOI: 10.1504/IJSSCI.2008.017590.

127. Richards M., Ford N. Fundamentals of Software Architecture. Sebastopol, CA : O'Reilly Media, 2020. 419 p.

128. Bass L., Clements P., Kazman R. Software Architecture in Practice. 4th ed. Boston : Addison-Wesley, 2022. 456 p.

ДОДАТОК А

Список публікацій здобувача за темою дисертації

Статті у наукових фахових виданнях України

1. Гаврилюк Є. А., Коробчинський К. П. UML-орієнтована інформаційна технологія для неперервних задач максимального покриття з об'єктами довільної форми. *Вісник Харківського національного університету імені В. Н. Каразіна. Серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2025. Вип. 67. С. 18–34. DOI: 10.26565/2304-6201-2025-67.

(Особистий внесок здобувача: розробка UML-орієнтованої інформаційної технології для неперервних задач максимального покриття з об'єктами довільної форми, побудова комплексу UML-діаграм (варіантів використання, класів, діяльності, послідовностей, компонентів, композитної структури, станів та розгортання), формування тривірневої модульної архітектури інформаційної системи, програмна реалізація модулів просторової аналітики та оптимізації покриття, підготовка матеріалів статті. Відповідні результати наведені в теоретичній та практичній частинах роботи.

Особистий внесок Коробчинського К. П.: спільна постановка задач та її аналіз, формування загальної концепції дослідження, валідація результатів дослідження.

2. Гаврилюк Є. А. Нейронно-керований гібридний метод для оптимального покриття території з обмеженням на розташування. *Системи та технології*. 2025. Т. 70, № 2. С. 112–118. DOI: 10.32782/2521-6643-2025-2-70.12.

3. Гаврилюк Є. А. Гібридні ройові метаевристики для неперервних задач максимального покриття зі складними формами об'єктів. *Проблеми керування та інформатики*. 2026. № 1. С. 14–33. DOI: 10.34229/1028-0979-2026-1-2.

4. Гаврилук Є. А., Струков В. М., Узлов Д. Ю. Інтегрований фреймворк на основі методів штучного інтелекту для задач неперервного покриття з геометричними обмеженнями. *Системні технології*. 2026. Т. 1, № 162. С. 160-176. DOI: 10.34185/1562-9945-5-162-2026-18.

(Особистий внесок здобувача: розробка та формалізація інтегрованого AI-базованого фреймворку для задач неперервного покриття з геометричними обмеженнями, реалізація та порівняння ройових і меметичних методів оптимізації, розробка нейромережесвих сурогатних моделей для апроксимації цільової функції покриття та адаптивних штрафних механізмів, проведення обчислювальних експериментів на задачах з різними типами геометричних обмежень, підготовка матеріалів статті. Відповідні результати наведені в теоретичній та практичній частинах роботи.

Особистий внесок Струков В. М.: спільна постановка задач та її аналіз, валідація результатів дослідження.

Особистий внесок Узлов Д. Ю.: спільна постановка задач та її аналіз, формування загальної концепції дослідження та інтеграція AI-фреймворку до задач покриття).

**Наукові праці у наукових фахових виданнях,
що входять до міжнародних наукометричних баз**

5. Uzlov D., Havryliuk Y., Hushchyn I., Strukov V., Yakovlev S. Handling outliers in swarm algorithms: a review. *CEUR Workshop Proceedings*. 2024. Vol. 3777. P. 430-437. *Proceedings of the 4th International Workshop of IT-Professionals on Artificial Intelligence (ProfIT AI 2024)*, Cambridge, MA, USA, September 25–27, 2024). URL: <https://ceur-ws.org/Vol-3777/short13.pdf> (Scopus).

(Особистий внесок здобувача: модифікація методу рою часток, дослідження впливу викидів на збіжність ройових алгоритмів у задачах

покриття, презентація доповіді. Відповідні результати наведені в аналітичній частині роботи.

Особистий внесок Uzlov D.: постановка задачі дослідження, загальна концепція дослідження.

Особистий внесок Hushchyn I.: збір та систематизація літературних джерел, аналіз сучасного стану предмету дослідження.

Особистий внесок Strukov V.: перевірка наукової достовірності отриманих результатів.

Особистий внесок Yakovlev S.: наукове керівництво, планування обчислювальних експериментів, перевірка наукової достовірності отриманих результатів.).

6. Pichugina O., Kirichenko L., Koliechkina L., Skob Yu., Havryliuk Ye. Exploring one linear permutation-based stochastic optimization problem with applications. *Proceedings of the 13th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2025)*, Gliwice, Poland, 4-6 September 2025. DOI: 10.1109/IDAACS68557.2025.11322262 (**Scopus**).

(Особистий внесок здобувача: огляд статистичних методів оптимізації, збір та систематизація літературних джерел, аналіз сучасного стану предмету дослідження.

Особистий внесок Pichugina O.: постановка задачі, математична формалізація задачі комбінаторної оптимізації.

Особистий внесок Kirichenko L.: аналіз стохастичних моделей, перевірка наукової достовірності результатів.

Особистий внесок Koliechkina L.: розробка комбінаторних методів розв'язання.

Особистий внесок Skob Yu.: участь у програмній реалізації та візуалізації результатів.).

7. Yakovlev S., Havryliuk Y., Matsyi O., Hulianytskyi A., Kirpich A. A neural network-driven adaptive approach for maximum coverage location with restricted zones. *CEUR Workshop Proceedings*. 2025. Vol. 4164. P. 312-322. *Proceedings of the 5th International Workshop of IT-Professionals on Artificial Intelligence (ProfIT AI 2025)*, Liverpool, UK, October 15–17, 2025). URL: <https://ceur-ws.org/Vol-4164/paper20.pdf> (**Scopus**).

(Особистий внесок здобувача: розробка та реалізація нейромережевого адаптивного підходу для задачі максимального покриття із забороненими зонами, програмна реалізація гібридного алгоритму, що поєднує ройову оптимізацію з нейромережевим керуванням штрафними функціями, проведення обчислювальних експериментів та аналіз результатів. Відповідні результати наведені в практичній частині роботи.

Особистий внесок Yakovlev S.: наукове керівництво, планування обчислювальних експериментів, перевірка наукової достовірності отриманих результатів.

Особистий внесок Matsyi O.: участь в аналізі результатів, аналіз практичного застосування.

Особистий внесок Hulianytskyi A.: участь у математичному обґрунтуванні методу.

Особистий внесок Kirpich A.: участь у програмній реалізації та тестуванні).

ДОДАТОК Б

Параметри обчислювальних експериментів

У цьому додатку наведено повні параметри обчислювальних експериментів, результати яких обговорено у розділах 2-3.

Для серії 1 параметри області покриття наступні (таблиця Б.1):

Таблиця Б.1 – Параметри області покриття для серії 1

	1	2	3	4	5	6	7	8
x	1.5698	2.7404	5.0817	7.4230	8.9838	8.2034	5.0817	1.9600
y	4.0643	1.7230	2.8937	1.3328	3.6741	6.4056	6.7958	5.6251

Параметри кіл, що визначаються лише радіусом, подано у таблиці Б.2:

Таблиця Б.2 – Параметри покриваючих кіл для серії 1

Номер	Радіус	Площа
1	0.402	0.5076
2	0.6184	1.2016
3	0.8395	2.2143
4	0.7939	1.9799
5	0.8124	2.0733
Всього		7.9767

Параметри еліпсів визначаються двома півосьми та кутом поворота (таблиці Б.3):

Таблиця Б.3 – Параметри покриваючих еліпсів для серії 1

	Півось a	Півось b	Кут	Площа
1	0.5319	0.4221	127.64	0.7053
2	0.816	0.6	48.01	1.5381
3	0.564	0.508	30.1	0.9001
4	0.5114	0.4669	82.76	0.75
5	0.8117	0.635	54	1.6193
Всього				5.5129

Параметри для полігонів визначаються подані у таблиці Б.4:

Таблиця Б.4 – Параметри покриваючих полігонів для серії 1

		1	2	3	4	5	6	7	8	9
P1	x	0.6008	0.5060	0.0424	-0.4094	-0.6574	-0.3752	0.0944	0.4308	
	y	-0.0138	0.3878	0.6868	0.4993	0.0806	-0.4910	-0.6392	-0.4731	
P2	x	0.8069	0.1987	-0.7122	-0.6254	0.4789				
	y	-0.3013	0.8422	0.4182	-0.5743	-0.7238				
P3	x	0.8129	-0.6321	-0.4397						
	y	0.0890	0.6387	-0.7314						
P4	x	0.7173	0.6334	-0.0389	-0.7010	-0.7643	-0.1971	0.4740		
	y	0.0715	0.5215	0.7284	0.2501	-0.3034	-0.7306	-0.6650		
P5	x	0.7345	-0.2947	-0.6385						
	y	0.1389	0.6694	-0.4602						
P6	x	0.7756	-0.3969	-0.5613						
	y	0.0082	0.6550	-0.6474						
P7	x	0.5839	0.5219	0.1979	-0.3492	-0.5367	-0.4859	-0.3396	0.1039	0.4853
	y	-0.0485	0.3449	0.5239	0.4838	0.2455	-0.2972	-0.5176	-0.6010	-0.3270
P8	x	0.8480	0.3274	-0.8996	-0.1141					
	y	0.0035	0.8861	-0.2771	-0.9258					
P9	x	0.7213	0.3827	-0.0667	-0.6821	-0.6732	-0.0337	0.5600		
	y	-0.1004	0.6895	0.8284	0.4476	-0.2962	-0.7370	-0.6012		
P10	x	0.9784	0.1614	-0.9107	-0.8243	0.4341				
	y	0.1613	1.0139	0.4083	-0.4650	-0.9154				

Для серії 2 параметри області покриття були наступні (таблиця Б.5):

Таблиця Б.5 – Параметри області покриття для серії 2

	1	2	3	4	5	6	7	8
x	-0.2988	1.4471	4.9390	8.4309	10.7588	9.5948	4.9390	0.2832
y	3.9520	0.4601	2.2061	-0.1218	3.3700	7.4439	8.0258	6.2799

Для серії 2 були взяті ті самі об'єкти, що й для серії 1, та додано 20 нових.

Параметри нових кіл наступні (таблиця Б.6):

Таблиця Б.6 – Параметри нових покриваючих кіл для серії 2

	Радіус	Площа
1	0.7068	1.5695
2	0.4363	0.5981
3	0.9757	2.9906
4	0.9468	2.8161
5	0.627	1.235
Нове		9.2093
Всього		17.186

Параметри нових еліпсів наступні (таблиця Б.7):

Таблиця Б.7 – Параметри нових покриваючих еліпсів для серії 2

	Півось a	Півось b	Кут	Площа
1	0.547	0.4954	0.66	0.8512
2	0.7516	0.698	124.9	1.6481
3	0.8289	0.7522	81.18	1.9588
4	1.1053	0.666	25.13	2.3126
5	0.8967	0.5861	96.44	1.651
Нове				8.4217
Всього				13.9346

Параметри нових полігонів наступні (таблиця Б.8):

Таблиця Б.8 – Параметри нових покриваючих полігонів для серії 2

		1	2	3	4	5	6	7	8	9
P1	x	0.6008	0.5060	0.0424	-0.4094	-0.6574	-0.3752	0.0944	0.4308	
	y	-0.0138	0.3878	0.6868	0.4993	0.0806	-0.4910	-0.6392	-0.4731	
P2	x	0.8069	0.1987	-0.7122	-0.6254	0.4789				
	y	-0.3013	0.8422	0.4182	-0.5743	-0.7238				
P3	x	0.8129	-0.6321	-0.4397						
	y	0.0890	0.6387	-0.7314						
P4	x	0.7173	0.6334	-0.0389	-0.7010	-0.7643	-0.1971	0.4740		
	y	0.0715	0.5215	0.7284	0.2501	-0.3034	-0.7306	-0.6650		
P5	x	0.7345	-0.2947	-0.6385						
	y	0.1389	0.6694	-0.4602						

Продовження таблиці Б.8

		1	2	3	4	5	6	7	8	9
P6	x	0.7756	-0.3969	-0.5613						
	y	0.0082	0.6550	-0.6474						
P7	x	0.5839	0.5219	0.1979	-0.3492	-0.5367	-0.4859	-0.3396	0.1039	0.4853
	y	-0.0485	0.3449	0.5239	0.4838	0.2455	-0.2972	-0.5176	-0.6010	-0.3270
P8	x	0.8480	0.3274	-0.8996	-0.1141					
	y	0.0035	0.8861	-0.2771	-0.9258					
P9	x	0.7213	0.3827	-0.0667	-0.6821	-0.6732	-0.0337	0.5600		
	y	-0.1004	0.6895	0.8284	0.4476	-0.2962	-0.7370	-0.6012		
P10	x	0.9784	0.1614	-0.9107	-0.8243	0.4341				
	y	0.1613	1.0139	0.4083	-0.4650	-0.9154				

Для серії 3 параметри області покриття були наступні (таблиця Б.9):

Таблиця Б.9 – Параметри області покриття для серії 3

	1	2	3	4	5	6	7	8
x	-2.5562	-0.1153	4.7666	9.6485	12.9031	11.2758	4.7666	-1.7426
y	3.8163	-1.0656	1.3754	-1.8792	3.0027	8.6982	9.5119	7.0709

Для серії 3 були взяті ті самі об'єкти, що й для серії 2, та додано 20 нових.

Параметри нових кіл наступні (таблиця Б.10):

Таблиця Б.10 – Параметри нових покриваючих кіл для серії 3

	Радіус	Площа
1	0.4037	0.512
2	0.7312	1.6796
3	0.4188	0.551
4	0.7988	2.0044
5	0.7255	1.6535
6	0.569	1.0173
7	0.7322	1.6845
8	0.6964	1.5235
9	0.3303	0.3427
10	0.6711	1.4148
Нова		12.3831
Всього		29.5692

Параметри нових еліпсів наступні (таблиця Б.11):

Таблиця Б.11 – Параметри нових покриваючих еліпсів для серії 3

	Півось a	Півось b	Кут	Площа
1	1.0347	0.6869	73.7	2.2328
2	0.706	0.6387	174.47	1.4166
3	0.8105	0.7994	6.49	2.0356
14	0.8694	0.4209	88.97	1.1498
5	0.9279	0.8135	94.13	2.3715
6	1.1082	0.4961	23.11	1.727
7	0.8255	0.7291	127.43	1.8909
8	0.5257	0.4491	45.78	0.7418
9	0.8301	0.817	48.76	2.1307
10	0.8388	0.5321	156.46	1.4023
Нова				17.0989
Всього				31.0334

Параметри нових полігонів наступні (таблиця Б.12):

Таблиця Б.12 – Параметри нових покриваючих полігонів для серії 3

		1	2	3	4	5	6	7	8	9	10
P21	x	0.6883	0.451 1	0.170 3	- 0.424 5	- 0.709 3	- 0.660 1	- 0.377 7	0.013 8	0.495 1	
	y	0.0393	0.497 5	0.693 4	0.589 6	0.268 6	- 0.253 7	- 0.684 0	- 0.680 3	- 0.556 1	
P22	x	0.7898	- 0.352 5	- 0.676 5							
	y	- 0.2494	0.737 5	- 0.342 1							
P23	x	0.9792	- 0.281 5	- 0.968 3	- 0.362 5						
	y	0.2485	0.878 4	0.235 1	- 0.833 7						

Продовження таблиці Б.12

		1	2	3	4	5	6	7	8	9	10
P24	x	0.687 0	0.561 9	0.237 0	- 0.121 1	- 0.516 5	- 0.648 9	- 0.450 2	- 0.119 9	0.173 2	0.544 9
	y	- 0.054 0	0.412 7	0.580 8	0.690 1	0.376 2	0.020 5	- 0.460 4	- 0.637 5	- 0.699 4	- 0.381 0
P25	x	0.510 8	0.204 5	- 0.177 4	- 0.574 2	- 0.344 7	0.352 2				
	y	- 0.075 9	0.451 6	0.505 9	0.063 6	- 0.415 9	- 0.432 2				
P26	x	0.592 6	0.297 6	- 0.475 4	- 0.401 4	0.001 5					
	y	0.188 5	0.559 7	0.456 5	- 0.527 1	- 0.585 5					
P27	x	0.563 1	0.337 2	- 0.279 4	- 0.601 7	- 0.439 4	0.231 5				
	y	- 0.125 2	0.518 3	0.499 1	- 0.069 3	- 0.424 6	- 0.535 0				
P28	x	0.752 2	- 0.647 1	- 0.696 6							
	y	- 0.330 0	0.481 8	- 0.528 2							
P29	x	0.551 0	0.341 5	- 0.110 4	- 0.410 3	- 0.463 8	- 0.181 5	0.343 3			
	y	0.086 0	0.450 3	0.526 0	0.358 0	- 0.316 5	- 0.494 3	- 0.403 8			
P30	x	0.753 5	0.448 0	- 0.574 5	- 0.505 7	0.122 8					
	y	- 0.034 4	0.549 8	0.469 2	- 0.513 6	- 0.674 1					

Продовження таблиці Б.12

		1	2	3	4	5	6	7	8	9	10
P31	x	0.924 1	0.828 9	0.026 4	- 0.660 4	- 0.971 2	- 0.736 9	0.067 8	0.866 9		
	y	- 0.136 5	0.652 2	0.922 6	0.757 3	0.080 2	- 0.691 8	- 0.998 3	- 0.569 5		
P32	x	0.577 7	0.528 2	0.167 1	- 0.204 5	- 0.503 3	- 0.546 3	- 0.482 3	- 0.103 5	0.088 3	0.522 2
	y	0.024 0	0.256 2	0.527 5	0.572 1	0.345 0	0.092 1	- 0.387 4	- 0.537 8	- 0.576 7	- 0.341 2
P33	x	0.819 3	0.541 5	- 0.046 3	- 0.746 9	- 0.602 3	- 0.335 1	0.490 6			
	y	0.103 4	0.452 9	0.819 3	0.178 6	- 0.467 1	- 0.705 2	- 0.564 7			
P34	x	0.930 8	0.038 7	- 0.770 2	- 0.721 7	0.429 6					
	y	0.308 9	1.014 3	0.556 4	- 0.542 1	- 0.846 6					
P35	x	0.647 5	0.452 1	- 0.146 8	- 0.509 8	- 0.550 3	0.008 2	0.372 7			
	y	0.037 0	0.472 1	0.604 3	0.285 9	- 0.359 8	- 0.615 4	- 0.576 2			
P36	x	0.870 1	- 0.784 8	- 0.215 0							
	y	0.321 5	0.526 2	- 0.938 6							
P37	x	0.977 0	0.189 2	- 0.922 2	- 0.938 4	0.124 9					
	y	0.108 2	0.953 9	0.343 6	- 0.296 6	- 0.936 9					

Продовження таблиці Б.12

		1	2	3	4	5	6	7	8	9	10
P38	x	0.715 1	0.368 0	0.145 6	- 0.545 0	- 0.688 1	- 0.477 4	- 0.029 9	0.389 2		
	y	0.051 8	0.575 4	0.642 4	0.359 0	0.066 2	- 0.480 6	- 0.710 2	- 0.514 2		
P39	x	1.032 0	0.745 7	- 0.289 0	- 0.774 3	- 0.829 5	- 0.321 2	0.803 7			
	y	0.275 7	0.735 8	0.935 2	0.611 0	- 0.536 3	- 0.914 8	- 0.591 5			
P40	x	0.638 5	0.455 6	- 0.101 1	- 0.475 4	- 0.649 2	- 0.225 0	0.454 3			
	y	0.142 5	0.420 4	0.661 1	0.403 5	- 0.158 8	- 0.557 6	- 0.521 3			

Нижче також наведено опис параметрів для кожного алгоритму та їх значення, окремо для серій із 20, 40 і 80 об'єктів. Параметри, що залежать від розміру задачі (популяція, кількість ітерацій), масштабуються зі збільшенням кількості об'єктів; решта параметрів – сталі, оскільки вони відповідають за баланс дослідження/експлуатації і не залежать від розмірності.

Для PSO ці параметри наступні (таблиця Б.13):

Таблиця Б.13 – Параметри PSO

Параметр	Серія 1	Серія 2	Серія 3
Кіл-ть частинок	50	100	150
Макс. ітерацій	150	200	500
ω	0.72	0.72	0.72
$c1$	1.496	1.496	1.496
$c2$	1.496	1.496	1.496

Для ABC ці параметри наступні (таблиця Б.14):

Таблиця Б.14 – Параметри ABC

Параметр	Серія 1	Серія 2	Серія 3
Кіл-ть частинок	30	50	100
Макс. ітерацій	150	200	500
Ліміт	90	120	180
Збурення	1.0	1.0	1.0

Значення метепараметрів для алгоритму FA подані у таблиці Б.15:

Таблиця Б.15 – Параметри FA

Параметр	Серія 1	Серія 2	Серія 3
Кіл-ть частинок	30	50	100
Макс. ітерацій	150	200	500
Alpha	0.25	0.25	0.25
Beta0	1.0	1.0	1.0
Gamma	0.05	0.005	0.001
Alpha_decay	0.98	0.98	0.98

Метепараметри для алгоритму FSS подані у таблиці Б.16:

Таблиця Б.16 – Параметри FSS

Параметр	Серія 1	Серія 2	Серія 3
Кіл-ть частинок	30	50	80
Макс. ітерацій	150	200	500
Step_ind_init	0.1	0.1	0.1
Step_ind_final	0.001	0.001	0.001
Step_vol	0.01	0.01	0.01
w_init	1.0	1.0	1.0
w_scale	500.0	500.0	500.0

Принцип масштабування: зі збільшенням кількості об'єктів зростає розмірність задачі (2 параметри на коло, 3 – на еліпс/полігон), тому розмір популяції та кількість ітерацій збільшуються, щоб алгоритм встигав досліджувати простір пошуку.

Конфігурації заборонених зон для кожної серії представлені у таблиці Б.17:

Таблиця Б.17 – Конфігурація заборонених зон

Параметр	Серія 1	Серія 2	Серія 3
Радус зони 1	0.881	0.930	1.047
Радус зони 2	0.619	0.688	0.775
Радус зони 3	0.435	0.483	0.545

Ці заборонені зони являють собою ускладнення задачі покриття, де центри об'єктів покриття не можуть бути розміщені у заборонених зонах. Конфігурації заборонених зон були підібрані таким чином, щоб покриття було оптимальним та теоретично максимально можливим за заданих заборонених зон у кожній серії експериментів.

Онлайн сервіс створення та перевірки кваліфікованого та удосконаленого електронного підпису

ПРОТОКОЛ
створення та перевірки кваліфікованого та удосконаленого електронного підпису

Дата та час: 16:22:35 26.05.2026

Назва файлу з підписом: Havryliuk_thesis.pdf.asice
Розмір файлу з підписом: 2.4 МБ

Перевірені файли:
Назва файлу без підпису: Havryliuk_thesis.pdf
Розмір файлу без підпису: 3.0 МБ

Результат перевірки підпису: Підпис створено та перевірено успішно. Цілісність даних підтверджено

Підписувач: Гаврилук Єгор Андрійович
П.І.Б.: Гаврилук Єгор Андрійович
Країна: Україна
РНОКПП: 3624703535
Час підпису (підтверджено кваліфікованою позначкою часу для підпису від Надавача): 16:22:34
26.05.2026
Сертифікат виданий: КНЕДП ДПС
Серійний номер: 3FAA9288358EC00304000000AF7235003A54EA00
Алгоритм підпису: ДСТУ 4145
Тип підпису: Удосконалений
Тип контейнера: Підпис та дані в архіві (розширений) (ASiC-E)
Формат підпису: З повними даними для перевірки (XAdES-B-LT)
Сертифікат: Кваліфікований

Версія від: 2026.04.06 13:00