

В. І. Скіцько,
к. е. н., доцент, доцент кафедри математичного моделювання та статистики,
Київський національний університет ім. Вадима Гетьмана
ORCID ID: <https://orcid.org/0000-0002-6290-9194>
М. Ю. Войніков,
аспірант, Київський національний університет ім. Вадима Гетьмана
ORCID ID: <https://orcid.org/0000-0001-7961-5312>

DOI: 10.32702/2306-6814.2025.2.157

ІННОВАЦІЙНІ ПІДХОДИ У ВИРІШЕННІ ЗАДАЧ УПРАВЛІННЯ ЛАНЦЮГАМИ ПОСТАВОК

V. Skitsko,
PhD in Economics, Associate Professor, Associate Professor of the Department of Mathematical Modeling
and Statistics, Kyiv National Economic University named after V. Hetman
M. Voinikov,
Postgraduate student, Kyiv National Economic University named after V. Hetman

INNOVATIVE APPROACHES TO SOLVING SUPPLY CHAIN MANAGEMENT PROBLEMS

У статті розглянуто інноваційні підходи до вирішення оптимізаційних задач управління ланцюгами поставок шляхом застосування еволюційних алгоритмів для їх розв'язання. Інновації у цій сфері полягають у застосуванні сучасних методів оптимізації, здатних швидко адаптуватися до змінних умов ринку, вирішувати складні оптимізаційні задачі та враховувати обмеження реальних бізнес-процесів. Сформовано методологічний підхід до адаптації задач управління ланцюгами поставок для застосування еволюційних алгоритмів, який охоплює визначення типу параметрів задачі, способу кодування рішень, вибір еволюційного алгоритму, формування початкової популяції та налаштування генетичних операторів із врахуванням обмежень. Практичне застосування запропонованого підходу продемонстровано на прикладі задачі комівояжера, яка належить до класу NP-складних задач оптимізації. Впровадження таких підходів потребує певних інвестицій в обчислювальні ресурси та адаптацію бізнес-процесів, проте дозволяє суттєво знизити витрати, підвищити швидкість прийняття рішень та покращити якість обслуговування клієнтів. Перспективи подальших досліджень охоплюють розробку багатокритеріальних методів оптимізації та гібридних алгоритмів для підвищення ефективності та швидкості пошуку рішень у динамічних умовах бізнесу.

The article explores innovative approaches to solving supply chain management problems by applying evolutionary algorithms to optimize logistics processes. These methods provide flexibility in adapting to dynamic market conditions and addressing real-world business constraints, where traditional optimization methods are often less effective due to their complexity. Evolutionary algorithms are especially effective in handling complex supply chain management tasks, including inventory management, transportation planning, and resource allocation. The proposed methodological approach involves key steps such as selecting the problem type and its parameters, defining an appropriate solution encoding, choosing an appropriate evolutionary algorithm, generating an initial population, and customizing genetic operators to ensure that solutions remain

feasible within given constraints. One of the notable contributions of the article is the structured process of forming an initial population that respects problem constraints, ensuring that initial solutions are feasible and diverse to reduce the risk of premature convergence to suboptimal solutions and accelerate the search for near-optimal results. The article also discusses various strategies for handling constraints during the mutation and crossover phases of the algorithm. These strategies include using penalty functions for soft constraints and corrective procedures to ensure that generated offspring remain within the feasible solution space. Practical implementation of the methodology is demonstrated using the traveling salesman problem, an NP-hard combinatorial optimization task that requires finding the shortest route connecting multiple destinations. However, implementing these approaches requires certain investments in computational resources and adjustments to business processes, which are justified by long-term benefits such as increased flexibility, risk mitigation, and enhanced customer service quality. The study concludes by emphasizing future research directions, including the development of hybrid algorithms and multi-objective optimization methods to improve the speed and accuracy of finding optimal solutions in rapidly changing business environments.

Ключові слова: управління ланцюгами поставок, еволюційні алгоритми, генетичний алгоритм, генетичне програмування, еволюційні стратегії, еволюційне програмування, диференціальна еволюція, оптимізація.

Key words: supply chain management, evolutionary algorithms, genetic algorithm, genetic programming, evolutionary strategies, evolutionary programming, differential evolution, optimization.

ПОСТАНОВКА ПРОБЛЕМИ У ЗАГАЛЬНОМУ ВИГЛЯДІ ТА ЇЇ ЗВ'ЯЗОК ІЗ ВАЖЛИВИМИ НАУКОВИМИ ЧИ ПРАКТИЧНИМИ ЗАВДАННЯМИ

Управління ланцюгами поставок — це комплексна система, яка охоплює низку сфер, включаючи закупівлі, виробництво, транспортування, управління запасами, розподіл ресурсів та планування постачання. У сучасних умовах, коли конкуренція на ринках посилюється, а клієнти очікують більш швидкого та якісного обслуговування [11], виникає потреба в оптимізації всіх етапів ланцюга поставок. Зокрема, підвищення швидкості доставки, зниження рівня запасів та витрат, ефективний розподіл ресурсів і планування логістичних маршрутів відіграють важливу роль у забезпеченні конкурентоспроможності компаній.

Оптимізація таких процесів потребує ефективних методів управління, здатних враховувати змінні умови ринку та специфічні обмеження. Більшість наведених оптимізаційних задач належить до класу NP-складних, тобто для їхнього точного розв'язання необхідно значно більше обчислювальних ресурсів та часу, ніж є в розпорядженні бізнесу, тому виникає потреба пошуку та розробки інноваційних методів до розв'язання задач, які дозволяють отримати точні або наближені розв'язки за обмежений час.

Інноваційні підходи до вирішення задач управління ланцюгами поставок спрямовані на оптимізацію різних сфер цієї системи з мінімальними витратами обчислювальних ресурсів та зменшенням часу на пошук рішення. Основним вектором розвитку таких підходів є використання алгоритмів штучного інтелекту, які допомагають автоматизувати складні процеси прийняття рішень та пошуку оптимальних рішень.

Особливої уваги заслуговують алгоритми колективного штучного інтелекту, які імітують природні процеси, зокрема еволюційні алгоритми. Ці алгоритми використовуються для оптимізації задач, що мають великий простір допустимих рішень, де традиційні методи можуть бути неефективними або надто ресурсозатратними. Еволюційні алгоритми мають здатність адаптуватися до різних типів задач і ефективно враховувати обмеження, що робить їх універсальним інструментом для задач управління ланцюгами поставок.

Використання еволюційних алгоритмів у задачах управління ланцюгами поставок потребує певних інвестицій в обчислювальні ресурси та адаптацію бізнес-процесів. Однак ці інвестиції мають довгостроковий ефект у вигляді зниження операційних витрат, підвищення швидкості прийняття рішень та покращення якості обслуговування клієнтів. Ефективність таких алгоритмів у бізнес-середовищі визначається їхньою здатністю швидко адаптуватися до змінних умов, враховувати жорсткі обмеження та мінімізувати ризики. Здатність еволюційних алгоритмів ефективно розв'язувати складні оптимізаційні задачі відкриває нові можливості для підвищення гнучкості та швидкості логістичних процесів, що є критично важливим для сучасних компаній.

Однак для забезпечення ефективності використання еволюційних алгоритмів важливо сформувати методологічний підхід для адаптації задач управління ланцюгами поставок до специфіки цих алгоритмів. Це передбачає визначення типів задач, способів кодування рішень і налаштування генетичних операторів з урахуванням обмежень та особливостей конкретного бізнес-процесу. Такий підхід уможливорює створення ефективних механізмів для пошуку оптимальних рішень у реаль-

них умовах із мінімальними витратами часу та ресурсів, що дозволяє підвищити ефективність логістичних процесів і приймати обґрунтовані управлінські рішення.

АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ І ПУБЛІКАЦІЙ

У науковій літературі висвітлено численні дослідження, спрямовані на розробку та застосування еволюційних алгоритмів для вирішення різноманітних задач оптимізації, зокрема у сфері управління ланцюгами поставок. У цих роботах основний акцент зроблено на застосуванні окремих алгоритмів до конкретних оптимізаційних задач із врахуванням специфіки відповідних бізнес-процесів та обмежень.

Зокрема, у роботі [1] розглядаються числові приклади оптимізації на основі наборів даних, пропонуються методи створення початкової популяції та генетичні оператори схрещування. У дослідженні [2] порівнюють генетичний алгоритм із класичними методами побудови мінімального кістяка графа, такими як алгоритм Прима. Дослідження [5] акцентує увагу на розробці методів створення початкової популяції для задачі комівояжера та тестуванні швидкості виконання генетичних алгоритмів.

Окремі дослідження зосереджуються на гібридних методах. Наприклад, у [3] пропонують метод вирішення задачі маршрутизації транспорту з врахуванням додаткових обмежень, який поєднує генетичний алгоритм із класичними підходами. Аналіз різних методів обчислювального інтелекту для вирішення задач комівояжера розглядається, зокрема, у роботі [4] розглядаються мурашині алгоритми, генетичні алгоритми та штучні нейронні мережі.

Значний внесок у становлення та розвиток теоретичних основ еволюційних алгоритмів зробили західні науковці, чиї праці стали фундаментом для подальших досліджень у цій галузі. Зокрема, роботи [6, 7, 8, 9, 10] заклали основи сучасної теорії еволюційних алгоритмів, досліджуючи їх математичні моделі, механізми адаптації та генетичні оператори.

Водночас аналіз наукових публікацій показує, що хоча у багатьох роботах розглядаються окремі аспекти застосування еволюційних алгоритмів для вирішення конкретних задач, бракує комплексного методологічного підходу, який дозволяв би адаптувати широкий спектр задач управління ланцюгами поставок до специфіки еволюційних алгоритмів. Таким чином, актуальним залишається створення універсальної методологічного підходу адаптації задач управління ланцюгами поставок до еволюційних алгоритмів, яка враховувала б різноманітні типи задач, способи представлення рішень та специфічні обмеження бізнес-процесів.

ФОРМУЛЮВАННЯ ЦІЛЕЙ СТАТТІ (ПОСТАНОВКА ЗАВДАННЯ)

Метою статті є формування методологічного підходу до адаптації задач управління ланцюгами поставок для застосування еволюційних алгоритмів, що дозволить підвищити ефективність вирішення задач оптимізації в реальних умовах. Такий підхід передбачає забезпечення правильного вибору алгоритму залежно від типу параметрів задачі, способів кодування рішень та

налаштування генетичних операторів із урахуванням специфіки бізнес-процесів у вигляді обмежень.

Основні завдання дослідження полягають у визначенні ключових характеристик еволюційних алгоритмів, які впливають на їхню ефективність у різних типах параметрів задач; аналізі типів даних і способів кодування рішень, що забезпечують швидке дослідження пошукового простору та врахування обмежень задачі; а також налаштуванні генетичних операторів для пошуку оптимальних рішень із мінімальними витратами часу та обчислювальних ресурсів із забезпеченням задоволення обмежень задачі. Для апробації методологічного підходу доцільним є її застосування на прикладі задач управління ланцюгами поставок, наприклад на задачі комівояжера.

Результати дослідження спрямовані на створення універсальної бази для інтеграції еволюційних алгоритмів у процеси управління ланцюгами поставок, що дозволяє дослідникам та практикам ефективно адаптувати задачі управління ланцюгами поставок до специфіки еволюційних алгоритмів і впроваджувати їх у реальні бізнес-процеси.

ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ ДОСЛІДЖЕННЯ

Еволюційні алгоритми є перспективним інструментом для вирішення складних оптимізаційних задач у сфері управління ланцюгами поставок. На відміну від класичних методів, вони базуються на принципах природного відбору та еволюції, що дозволяє їм досліджувати великий простір допустимих рішень та знаходити наближені оптимальні рішення за обмежений час. Завдяки своїй універсальності, еволюційні алгоритми можуть адаптуватися до специфіки різних задач, враховувати обмеження та працювати з різними типами даних.

До основних еволюційних алгоритмів належать генетичний алгоритм, генетичне програмування, еволюційні стратегії, еволюційне програмування та диференціальна еволюція. Кожен із них має унікальні особливості та підходить для різних типів оптимізаційних задач.

Генетичний алгоритм (ГА) є одним із найпоширеніших представників еволюційних алгоритмів, що базується на моделюванні процесів природного відбору та еволюції [9]. ГА вирізняється адаптивністю та здатністю вирішувати широкий спектр оптимізаційних задач, зокрема комбінаторних. Класичний варіант ГА передбачає бінарне кодування рішень у вигляді послідовностей нулів і одиниць [9], однак можливі й інші типи кодування залежно від специфіки задачі.

Генетичне програмування (ГП) — це різновид еволюційних алгоритмів, який відрізняється здатністю автоматично генерувати програми або математичні моделі для вирішення складних задач. Основною особливістю ГП є представлення рішень у вигляді деревоподібних структур, де вузли відповідають операторам, а листя — змінним або константам [10]. На відміну від інших еволюційних алгоритмів, ГП оптимізує не лише параметри, а й саму структуру рішень, тобто оператори мутації та схрещування застосовуються для зміни та комбінування піддерев [10].

Таблиця 1. Відповідність алгоритмів до типу задач

Алгоритм	Дискретні задачі	Неперервні задачі
Генетичний алгоритм	Так	Потребує адаптацій
Генетичне програмування	Так	Ні
Еволюційні стратегії	Потребує адаптацій	Так
Еволюційне програмування	Потребує адаптацій	Так
Диференціальна еволюція	Ні	Так

Джерело: сформовано на основі [6, 8, 9].

Еволюційні стратегії (ЕС) та еволюційне програмування (ЕП) є алгоритмами, які орієнтовані на оптимізацію параметрів, проте мають різний підхід до реалізації та сфери застосування. ЕС зосереджені на оптимізації неперервних параметрів за допомогою адаптивної мутації та детермінованого відбору, що забезпечує високу ефективність у задачах безперервної оптимізації. Особливістю ЕС є здатність автоматично змінювати масштаби мутацій залежно від поточного стану пошуку, що підвищує їхню адап-

тивність і забезпечує швидке знаходження оптимальних рішень [6, 7].

Натомість ЕП орієнтоване на оптимізацію параметрів фіксованих моделей і систем управління, таких як регресійні моделі або ваги нейронних мереж. Основною відмінністю ЕП є відсутність оператора схрещування та використання стохастичних механізмів селекції, часто турнірного відбору, що сприяє збереженню генетичної різноманітності в популяції [7].

Диференціальна еволюція (ДЕ) — це еволюційний алгоритм, призначений для оптимізації багатовимірних неперервних функцій. Основною особливістю ДЕ є використання мутації на основі різниць між векторами популяції, що дозволяє ефективно досліджувати простір можливих рішень [8]. На відміну від інших еволюційних алгоритмів, де основний акцент робиться на схрещуванні, ДЕ орієнтована на комбінування мутації та рекомбінації для створення нових рішень.

Для ефективного застосування еволюційних алгоритмів в оптимізації задач управління ланцюгами поставок важливим є вибір придатного алгоритму залежно від специфіки задачі. Окрім цього, важливим аспектом ефективності еволюційних алгоритмів є їх налаштування відповідно до характеристик конкрет-

Таблиця 2. Основні типи кодування рішень в еволюційних алгоритмах

Тип кодування	Опис	Приклад задачі
Бінарне	Представлення змінних у вигляді послідовностей нулів та одиниць. Використовується для дискретних рішень.	Обрати оптимальний набір постачальників: [1, 0, 1, 1, 0] – обрані постачальники 1, 3 і 4; відхилені 2 і 5.
Перестановочне	Представлення рішення у вигляді упорядкованої послідовності елементів. Використовується для задач, для яких важливий порядок.	Визначення оптимального маршруту доставки між пунктами: [5, 2, 1, 3, 4] – маршрут між складами 5, 2, 1, 3, 4.
Векторне	Представлення змінних у вигляді числових значень. Використовується для оптимізації безперервних параметрів.	Визначення значень мінімальних і максимальних запасів для кожного товару: [100, 250, 80] – мінімальні та максимальні запаси товарів.
Комбіноване	Поєднання кількох типів кодування (дискретного та безперервного). Підходить для задач із різнорідними змінними.	Вибір постачальників та управління закупівлями: [1, 0, 50.0, 75.0] – постачальник 1 обраний, 2 відхилений, запаси товарів 50 і 75 од.
Реальне	Представлення змінних у вигляді чисел із плаваючою комою. Використовується для задач із безперервними змінними.	Прогнозування попиту: [1500.0, 1800.5, 2000.0] — прогнозований попит продукції на три наступні місяці.
Символьне	Представлення змінних як послідовності символів або логічних умов. Використовується для автоматизації процесів.	Створення правил автоматизації логістичних процесів: ['if', 'demand > supply', 'order more', 'else', 'wait'] – правило для автоматизації закупівель.
Графове	Представлення рішень у вигляді графів із вузлами та зв'язками між ними. Використовується для складних мереж.	Оптимізація логістичної мережі: {A: [B, C], B: [D], C: [D, E]} – логістична мережа з вузлами та зв'язками.
Програми як дерева	Представлення рішень у вигляді дерева умов і дій. Використовується для автоматичної генерації стратегій.	Автоматична генерація плану поставок у вигляді дерева умов і дій: (if (demand > 1000) (order 500) (order 200)) – дерево умовного плану поставок.

Джерело: сформовано на основі [1, 9].

ної задачі, зокрема вибір кодування, створення початкової популяції, оператори схрещування та мутації. Неправильно підібрані генетичні оператори або спосіб кодування рішень можуть значно уповільнити збіжність алгоритму або призвести до отримання результатів, які не задовольняють бізнес-обмеження. Розгляньмо кроки адаптації еволюційних алгоритмів до задачі.

Крок 1. Аналіз параметрів задачі, вибір способу кодування та підбір алгоритму. Тип параметрів оптимізаційної задачі та спосіб представлення рішень мають значний вплив при виборі еволюційного алгоритму, оскільки різні еволюційні алгоритми застосовують різні оператори схрещування, мутації та селекції, які по-різному впливають на продуктивність оптимізації залежно від характеру параметрів [9]. Залежно від типу параметрів, оптимізаційні задачі поділяються на дискретні та неперервні.

Дискретні задачі передбачають пошук оптимального рішення серед фіксованої множини можливих варіантів. Такі задачі включають вибір, комбінації, перестановки та логічні структури. Неперервні задачі передбачають оптимізацію параметрів у просторі безперервних значень. Такі задачі зазвичай пов'язані з мінімізацією витрат, оптимізацією параметрів виробничих процесів або прогнозуванням попиту.

Для дискретних задач найкраще підходять генетичний алгоритм та генетичне програмування, оскільки вони ефективно працюють із задачами вибору та перестановок; еволюційні стратегії та еволюційне програмування можуть бути ефективними як для дискретних (за умови адаптації операторів), так і для неперервних задач; диференціальна еволюція спеціалізується на оптимізації неперервних функцій і не підходить для задач із дискретними параметрами (Табл. 1).

Вибір еволюційного алгоритму повинен враховувати не лише природу параметрів задачі (дискретних чи неперервних), але й спосіб представлення рішень у вигляді кодування. Формат кодування визначає структуру подання рішень у рамках алгоритму та відіграє важливу роль у забезпеченні ефективності процесу пошуку, оскільки він впливає на здатність алгоритму досліджувати пошуковий простір, враховувати обмеження задачі, ефективно використовувати обчислювальні ресурси та забезпечувати зручну інтерпретацію результатів. Наведемо основні типи кодування рішень, їх ключові характеристики та сфери застосування (Табл. 2).

Генетичний алгоритм ефективний для задач із дискретними параметрами завдяки роботі з перестановками та комбінаціями, але потребує адаптації для неперервних і деревоподібних структур; генетичне програмування оптимізує символічні й графові моделі, але менш ефективно для бінарного та векторного кодування; еволюційні стратегії та еволюційне програмування краще працюють із неперервними параметрами й векторними структурами, забезпечуючи глибокий пошук за рахунок адаптивних мутацій; диференціальна еволюція спеціалізується на задачах із реальним або векторним кодуванням, але менш придатна для дискретних проблем через обмеження операторів у неперервному просторі (Табл. 3).

Таблиця 3. Відповідність еволюційних алгоритмів до типів кодування

Тип кодування	ГА	ГП	ЕС	ЕП	ДЕ
Бінарне кодування	Так	Ні	Мод	Мод	Мод
Перестановочне кодування	Так	Мод	Мод	Мод	Ні
Векторне кодування	Так	Мод	Так	Так	Так
Комбіноване кодування	Так	Мод	Так	Так	Мод
Реальне (дійсне) кодування	Так	Мод	Так	Так	Так
Символьне кодування	Мод	Так	Ні	Ні	Ні
Графове кодування	Так	Так	Мод	Мод	Мод
Деревоподібне кодування	Мод	Так	Ні	Ні	Мод

Джерело: сформовано на основі [6, 8, 9].

У поясненні до таблиці 3 використовуються такі позначення: "Так" означає, що алгоритм є ефективним для відповідного типу кодування і не потребує модифікації генетичних операторів; "Мод" вказує на можливість застосування алгоритму за умови адаптації операторів для забезпечення належної продуктивності; "Ні" свідчить про непридатність алгоритму для роботи з даним типом кодування через обмеження операторів або специфіку структури представлення рішень.

Наведені критерії вибору еволюційного алгоритму, зокрема тип параметрів задачі та спосіб їх кодування не охоплюють усіх можливих аспектів прийняття рішення, проте забезпечують надійне підґрунтя для первинного відбору методів оптимізації. Іншими факторами є наявність обмежень задачі (часових, ресурсних), структура пошукового простору та його розмірність, а також очікувана точність результату. Зважаючи на сукупність цих факторів, дослідник має можливість адаптувати вибір алгоритму до специфіки задачі управління ланцюгами поставок, що забезпечує баланс між якістю рішень та обчислювальними ресурсами.

Крок 2. Формування початкової популяції з урахуванням обмежень. Формування популяції має враховувати обмеження задачі, що забезпечує коректність та допустимість згенерованих рішень. Обмеження можна класифікувати на два основні типи: м'які обмеження (умови, порушення яких є небажаним) та жорсткі обмеження (умови, порушення яких призводить до неприйнятності рішення).

Забезпечення дотримання обмежень під час створення початкової популяції є важливим для уникнення недопустимих рішень у подальших ітераціях алгоритму. Окрім забезпечення умов, необхідно також підтримувати генетичне різноманіття популяції, щоб уникнути передчасної збіжності алгоритму до локального оптимуму. Для цього можуть застосовуватися такі підходи:

1. Звичайна генерація особин із випадковою складовою для м'яких обмежень. У цьому підході особини генеруються випадковим чином без обов'язкового дотримання всіх обмежень. Порушення м'яких обмежень компенсуються за допомогою штрафних функцій у функції пристосованості;

2. Повторна генерація особини у випадку порушення жорстких обмежень. Цей метод передбачає перевірку кожної згенерованої особини на відповідність жорстким обмеженням. Якщо особина порушує ці обмеження, вона вилучається з популяції, а натомість генерується нова особина;

3. Використання обмеженого простору значень елементів особини. У цьому підході початкові рішення формуються виключно в межах допустимого простору рішень. У дослідженні [1] наведених розширений перелік методів формування початкової популяції, які дозволяють формувати рішення, що входять до області допустимих рішень та мають різноманітний генетичний матеріал.

Крок 3. Застосування генетичних операторів з урахуванням обмежень. Під час застосування генетичних операторів схрещування та мутації в еволюційних алгоритмах виникає ризик виходу особин за області допустимих рішень. Щоб забезпечити входження особин до області допустимих рішень після застосувань до них генетичних операторів, можуть використовуватися різні підходи, зокрема:

1. Генетичні оператори, що забезпечують входження до області допустимих рішень. Цей підхід полягає у виборі та (за потреби) адаптації генетичних операторів, після застосування яких утворюються нащадки, що входять до області допустимих рішень. Обираються оператори мутації, що змінюють особину таким чином, щоб не порушувати обмеження; оператори схрещування можуть комбінувати класичні методи (одноточкове, двоточкове схрещування тощо) із додатковими перевірками та корекціями;

2. Процедура повернення особини до області допустимих рішень. Якщо генетичні оператори не гарантують створення особин, що входять до області допустимих рішень, після їх застосування може використовуватися процедура щодо повернення особини до області допустимих рішень. Ця процедура полягає у модифікації значень елементів особини до тих пір, поки вона не увійде до області допустимих рішень. У більш складних варіантах корекції можуть використовуватися евристичні правила корекції;

3. Штрафні функції для м'яких обмежень. Для обліку м'яких обмежень у генетичних алгоритмах застосовуються штрафні функції.

Правильний вибір генетичних операторів, їх модифікація або застосування процедур корекції дозволяє усунути особини, що не входять до області допустимих рішень. Штрафні функції, своєю чергою, дозволяють ефективно працювати із задачами, що мають м'які обмеження, адаптуючи алгоритм до реальних умов. Таким чином, інтеграція цих підходів забезпечує можливість роботи з обмеженнями задачі.

Крок 4. Вибір типових компонентів алгоритму із загальноприйнятого підходу. Основні компоненти алгоритму, які підлягають налаштуванню, включають метод селекції, цільову функцію, критерій зупинки, розмір популяції та ймовірності застосування генетичних операторів, таких як схрещування та мутація.

Вибір параметрів еволюційного алгоритму базується на комплексному врахуванні специфіки задачі та характеристик обраного алгоритму та може бути здійснений з наявних параметрів в класичній літературі. Коректне налаштування дозволяє підвищити ефективність алгоритму, забезпечити швидкий пошук оптимального рішення та уникнути передчасної збіжності до локальних екстремумів.

Для наочності приведемо приклад застосування методологічного підходу для вирішення задачі комівояжера.

Задача комівояжера належить до класу комбінаторних задач оптимізації. Її суть полягає у визначенні найкоротшого маршруту, яким комівояжер відвідає всі задані міста, побуває в кожному місті лише один раз і повернеться до початкового пункту. Задача є NP-складною, що робить використання еволюційних алгоритмів ефективним підходом до пошуку наближеного рішення.

Крок 1. Аналіз параметрів задачі, вибір способу кодування та підбір алгоритму. Тип даних у задачі комівояжера є дискретним, оскільки рішення представляється як перестановка міст. Для роботи з дискретними параметрами можуть застосовуватися такі еволюційні алгоритми (Табл. 1): генетичний алгоритм, генетичне програмування, еволюційні стратегії та еволюційне програмування.

Одним із ефективних способів кодування рішень для задачі комівояжера є перестановочне кодування, за якого особина представляється у вигляді упорядкованої послідовності міст. Цей підхід має низку переваг для задач комбінаторної оптимізації:

— легка інтерпретація результатів: перестановочне кодування забезпечує природне відображення маршруту, що полегшує аналіз та інтерпретацію знайденого рішення. Кожна особина (хромосома) безпосередньо відображає послідовність міст, яку необхідно відвідати;

— ефективне врахування обмежень задачі: однією з основних вимог задачі комівояжера є відвідування кожного міста лише один раз. Перестановочне кодування дозволяє ефективно враховувати обмеження задачі, зокрема, певні генетичні оператори можуть використовуватися без модифікацій;

— зниження обчислювальної складності: у порівнянні з іншими способами представлення рішень, такими як векторне (матричне) або деревовидне кодування, перестановочне кодування є менш обчислювально затратним. Воно не потребує створення додаткових структур даних або складних алгоритмів перевірки допустимості рішень, що знижує обсяг обчислень і прискорює роботу алгоритму.

З наведених алгоритмів генетичний алгоритм є найкращим вибором для задач із перестановочним кодуванням, оскільки його оператори легко адаптуються до роботи з дискретними даними без необхідності складних модифікацій (Табл. 3). Інші алгоритми, такі як генетичне програмування, еволюційні стратегії та еволюційне програмування, потребують додаткових змін для забезпечення ефективного пошуку в просторі допустимих рішень. Отже, для задачі комівояжера обираємо генетичний алгоритм як найбільш ефективний метод.

Крок 2. Формування початкової популяції з урахуванням обмежень. Формування початкової популяції передбачає створення множини унікальних хромосом (маршрутів), які відповідають обмеженням задачі — кожне місто має бути відвідане лише один раз. Для забезпечення дотримання цієї умови пропонується застосувати метод випадкового генерування перестановок, який забезпечує створення допустимих рішень без необхідності додаткових перевірок на унікальність міст у маршруті. Процес формування хромосом відбувається наступним чином:

1. Із вибірки недоданих міст обирається випадкове місто для додавання до маршруту;

2. Місто, яке було додане до маршруту, виключається з вибірки недоданих міст, щоб уникнути повторного використання;

3. Процес повторюється, доки не будуть додані всі міста до маршруту.

Кожна сформована перестановка міст утворює унікальну хромосому, яка входить до початкової популяції. Процес формування повторюється до досягнення заданого розміру популяції. Таким чином, сформована початкова популяція складається з допустимих рішень, що охоплюють широкий простір можливих маршрутів.

Крок 3. Застосування генетичних операторів з урахуванням обмежень. У задачі комівояжера важливо забезпечити, щоб застосування операторів мутації та схрещування не призводило до створення некоректних маршрутів, де деякі міста відвідуються більше одного разу або не відвідуються взагалі.

Оператор мутації. Для забезпечення збереження маршруту в межах області допустимих рішень застосовується мутація обміну (swap mutation). Цей оператор працює наступним чином:

- випадковим чином обираються два гени (міста) в хромосомі;

- значення цих двох генів міняються місцями.

Застосування цього оператора дозволяє генерувати нові маршрути, які залишаються допустимими, оскільки кожне місто залишається унікально представленим у хромосомі.

Оператор схрещування. Обирається багатоточкове схрещування, яке потребує модифікацій для збереження допустимості рішень:

- від першого батька береться перший та третій сегменти хромосоми;

- другий сегмент заповнюється генами з другого батька, які ще не представлені в хромосомі на аналогічних позиціях;

- невикористані значення генів додаються у випадковому порядку.

Цей підхід дозволяє зберігати всі необхідні обмеження задачі під час генерації нових особин.

Крок 4. Вибір типових компоненти алгоритму із загальноприйнятого підходу. Для реалізації еволюційного алгоритму застосовуються стандартні компоненти, які не потребують додаткових модифікацій та можуть бути використані відповідно до загальноприйнятих підходів, описаних у класичній літературі:

1. Метод селекції. Застосуємо рангову селекція, яка запобігає передчасній збіжності до локальних оптимумів, оскільки ймовірність відбору особини залежить від її рангу, а не від абсолютного значення цільової функції;

2. Функція пристосованості. У задачі комівояжера відповідає цільовій функції, яка визначає мінімізацію загальної довжини маршруту, який необхідно пройти комівояжеру;

3. Критерій зупинки алгоритму. Стандартний підхід передбачає використання обмеження на кількість поколінь, після досягнення якого алгоритм припиняє роботу. Для задачі комівояжера встановимо початкову

покоління — 700, проте це значення може бути скориговане залежно від результатів тестування;

4. Ймовірності схрещування та мутації. Ймовірність схрещування — 0.8, ймовірність мутації — 0.3. Ці параметри можуть бути змінені залежно від отриманих результатів оптимізації;

5. Розмір популяції. Вибір розміру популяції залежить від складності задачі та розміру пошукового простору. Для задачі комівояжера рекомендується використовувати популяцію з 150—200 особин [1]. Більший розмір популяції може збільшити якість рішень, проте потребуватиме більше обчислювальних ресурсів.

Ці параметри є базовими налаштуваннями алгоритму та можуть бути використані без додаткової адаптації. Проте після початкових запусків алгоритму їх значення рекомендовано тестувати та коригувати для досягнення кращих результатів оптимізації.

Приклад задачі комівояжера із застосуванням еволюційних алгоритмів демонструє практичність запропонованої методологічного підходу. Визначено алгоритм і спосіб кодування рішень, метод формування початкової популяції та адаптовані генетичні оператори для забезпечення допустимості рішень. Зазначені налаштування дозволяють проводити тестування алгоритму та його подальше впровадження в бізнес-процеси для розв'язання задач оптимізації маршрутів.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ РОЗВІДОК У ДАНОМУ НАПРЯМІ

У статті запропоновано методологічний підхід для адаптації задач управління ланцюгами поставок із застосуванням еволюційних алгоритмів, що передбачає визначення типу параметрів задачі, способу кодування рішень, вибір відповідного алгоритму, формування початкової популяції та налаштування генетичних операторів з урахуванням обмежень задачі. Зазначений підхід дозволяє ефективно адаптувати різні типи оптимізаційних задач, які виникають у сфері управління ланцюгами поставок, до специфіки еволюційних алгоритмів, забезпечуючи ефективне підґрунтя для подальших досліджень та впровадження реалізованих алгоритмів у реальні системи управління ланцюгами поставок.

Особливу увагу було приділено врахуванню жорстких та м'яких обмежень, що є важливими для забезпечення практичної застосовності алгоритмів у реальних бізнес-процесах. Практичну реалізацію методологічного підходу продемонстровано на прикладі задачі комівояжера, яка є типовою для комбінаторної оптимізації у сфері логістики. Зокрема, розглянуто формування початкової популяції з дотриманням обмежень, вибір перестановочного кодування, адаптацію операторів схрещування та мутації, а також налаштування основних параметрів алгоритму.

Застосування еволюційних алгоритмів у задачах управління ланцюгами поставок дозволяє знижувати операційні витрати, оптимізувати логістичні процеси та підвищувати ефективність прийняття управлінських рішень. Використання запропонованого методологічного підходу сприяє забезпеченню збалансованого вирішення широкого спектра оптимізаційних задач в умовах динамічного ринкового середовища. При цьому впровадження еволюційних алгоритмів потребує інвес-

тицій в обчислювальні ресурси та адаптацію бізнес-процесів, однак ці інвестиції є стратегічно виправданими. Вони дозволяють компаніям підвищити гнучкість управління, швидше реагувати на зміни у попиту та пропозиції, знизити ризики та підвищити загальну конкурентоспроможність у довгостроковій перспективі.

Перспективи подальших досліджень полягають у розширенні методологічного підходу для багатокритеріальних задач оптимізації, розробці гібридних алгоритмів, що поєднують еволюційні та класичні методи оптимізації.

Література:

1. Генетичний алгоритм як засіб розв'язання оптимізаційних задач / Я. Пиріг та ін. Інфокомунікаційні технології та електронна інженерія. 2023. Т. 3, № 2. С. 95—107. URL: <https://doi.org/10.23939/ictee2023.02.095>.
2. Господінов А. М., Смирнов С. А. Генетичний алгоритм для розв'язання задачі комівояжера. Теоретичні і прикладні проблеми фізики, математики та інформатики: Матеріали XVI всеукр. науково-практ. конференції студентів, аспірантів та молодих вчен., м. Київ, 26—27 квіт. 2018 р. С. 19—21. URL: <https://ela.kpi.ua/server/api/core/bitstreams/512b9bd2-1825-4a62-990a-c89d58c0c1af/content>.
3. Іващенко Г. С., Склярів А. С., Барковська О. Ю. Гібридний метод рішення задачі маршрутизації транспорту з урахуванням додаткових обмежень. Системи управління, навігації та зв'язку. Збірник наукових праць. 2023. Т. 1, № 71. С. 31—35. URL: <https://doi.org/10.26906/sunz.2023.1.031>.
4. Методи рішення задачі комівояжера на основі обчислювального інтелекту / Г. С. Іващенко та ін. Системи управління, навігації та зв'язку. Збірник наукових праць. 2024. Т. 2, № 76. С. 99—105. URL: <https://doi.org/10.26906/sunz.2024.2.099>.
5. Розв'язання задачі комівояжера на основі еволюційного моделювання / А. О. Олійник та ін. Реєстрація, зберігання і обробка даних. 2019. Т. 21, № 3. С. 31—41. URL: <https://doi.org/10.35681/1560-9189.2019.21.3.183550>.
6. Beyer H. G., Schwefel H. P. Evolution strategies — A comprehensive introduction. Natural Computing. 2002. Vol. 1, no. 1. P. 3—52. URL: <https://doi.org/10.1023/a:1015059928466>.
7. Brabazon A., O'Neill M., McGarraghy S. Evolution strategies and evolutionary programming. Evolutionary Computation. 2000. P. 295—316. URL: <https://doi.org/10.1201/9781482273960-17>.
8. Feoktistov V. Differential Evolution. Differential Evolution. Boston, MA. 2006. P. 1—24. URL: https://doi.org/10.1007/978-0-387-36896-2_1.
9. Holland J. H. Adaptation in Natural and Artificial Systems. The MIT Press, 1992. URL: <https://doi.org/10.7551/mitpress/1090.001.0001>.
10. Koza J. Genetic programming as a means for programming computers by natural selection. Statistics and Computing. 1994. Vol. 4, no. 2. P. 87—112. URL: <https://doi.org/10.1007/bf00175355>.
11. The Last Mile Mandate: a consumer survey with key insights. FarEye. URL: <https://fareye.com/resources/blogs/the-last-mile-mandate-survey>.

References:

1. Pyrih, Ya., Klymash, M., Pyrih, Yu., and Lavriv, O. (2023), "Genetic algorithm as a tool for solving optimisation problems", Infokomunikatsijni tehnologii ta elektronna inzheneriia, vol. 3 (2), pp. 95—107. <https://doi.org/10.23939/ictee2023.02.095>.
2. Hospodinov, A. M. and Smyrnov, S. A. (2018), "Genetic algorithm for solving the traveling salesman problem", Teoretychni i prykladni problemy fizyky, matematyky ta informatyky: Materialy XVI vseukr. naukovo-prakt. konferentsii studentiv, aspirantiv ta molodykh vchen. [Theoretical and applied problems of physics, mathematics and computer science: Materials of the XVI All-Ukrainian scientific and practical conference of students, postgraduates and young scientists.], KPI im. Ihoria Sikors'koho, FTI, Kyiv, Ukraine, pp. 19—21, available at: <https://ela.kpi.ua/server/api/core/bitstreams/512b9bd2-1825-4a62-990a-c89d58c0c1af/content> (Accessed 05 Jan 2025).
3. Ivaschenko, H. S., Skliarov, A. S., and Barkovs'ka, O. Yu. (2023), "A hybrid approach for solving the vehicle routing problem with additional constraints", Systemy upravlinnia, navihatsii ta zv'iazku. Zbirnyk naukovykh prats', vol. 1 (71), pp. 31—35. <https://doi.org/10.26906/sunz.2023.1.031>.
4. Ivaschenko, H. S., Onyschenko, O. I., Bondarenko, M. E. and Zdoryk, N. V. (2024), "Methods of solving the traveling salesman problem based on computational intelligence", Systemy upravlinnia, navihatsii ta zv'iazku. Zbirnyk naukovykh prats', vol. 2 (76), pp. 99—105. <https://doi.org/10.26906/sunz.2024.2.099>.
5. Oliynyk, A. O., Fedorchenko, Ye. M., Stepanenko, O. O., and Rud', M. S. (2019), "Solving the traveling salesman problem based on evolutionary modeling", Reiestratsiia, zberihannia i obrobka danykh, vol. 21 (3), pp. 31—41. <https://doi.org/10.35681/1560-9189.2019.21.3.183550>.
6. Beyer, H. G. and Schwefel, H. P. (2002), "Evolution strategies — A comprehensive introduction", Natural Computing, vol. 1, no. 1, pp. 3—52. <https://doi.org/10.1023/a:1015059928466>.
7. Brabazon, A., O'Neill, M. and McGarraghy, S. (2000), "Evolution strategies and evolutionary programming", Evolutionary Computation, pp. 295—316. <https://doi.org/10.1201/9781482273960-17>.
8. Feoktistov, V. (2006), Differential Evolution. In Search of Solutions, Differential Evolution, Boston, MA, pp. 1—24. https://doi.org/10.1007/978-0-387-36896-2_1.
9. Holland, J. H. (1992), Adaptation in Natural and Artificial Systems, The MIT Press, Cambridge, USA. <https://doi.org/10.7551/mitpress/1090.001.0001>.
10. Koza, J. (1994), "Genetic programming as a means for programming computers by natural selection", Statistics and Computing, vol. 4, no. 2. p. 87—112. <https://doi.org/10.1007/bf00175355>.
11. FarEye (2024), "The Last Mile Mandate: a consumer survey with key insights", available at: <https://fareye.com/resources/blogs/the-last-mile-mandate-survey> (Accessed 05 Jan 2025).

Стаття надійшла до редакції 10.01.2025 р.