

**АНАЛІЗ АЛГОРИТМІВ РОЙОВОГО ІНТЕЛЕКТУ  
ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧ МАРШРУТИЗАЦІЇ  
ТРАНСПОРТНИХ ЗАСОБІВ**

**Вступ.** Задача маршрутизації транспортних засобів (англ. Vehicle Routing Problem, VRP) визначається як проблема розробки оптимальних маршрутів для парку транспортних засобів, які обслуговують набір клієнтів з одного або кількох депо, з урахуванням різних обмежень та метою мінімізації витрат або максимізації ефективності [1, 2]. VRP має широке застосування в різних секторах економіки, включаючи логістику, транспорт, електронну комерцію, управління відходами, громадський транспорт, охорону здоров'я та виїзне обслуговування. Ефективна оптимізація маршрутів має значний вплив на зниження операційних витрат, покращення якості обслуговування та досягнення цілей сталого розвитку. VRP належить до класу NP-складних задач, що означає високу обчислювальну складність пошуку оптимальних рішень, особливо для її варіацій великого масштабу. У зв'язку з цим, для розв'язання VRP часто застосовуються наближені методи, серед яких особливе місце займають алгоритми ройового інтелекту [1]. З метою дослідити, чи існує скінченний набір властивостей та чи існує кореляція між наборами ефективних параметрів алгоритмів ройового інтелекту, що робить їх ефективними для розв'язання VRP, проведений аналіз результатів наукових праць присвячених розв'язанню VRP.

Існує багато варіантів VRP, які визначаються різними факторами, що властиві задачам у реальному світі, такими як обмеження на вантажопідйомність (Capacitated VRP, CVRP), часові вікна (Vehicle Routing Problem with Time Windows, VRPTW), забір та доставка (Vehicle Routing Problem with Pickup and Delivery, VRPPD), наявність кількох депо (Multi-Depot VRP, MDVRP) та інші [1]. Важливо зазначити, що VRP можна розглядати як узагальнення задачі комівояжера (Traveling Salesman Problem, TSP) [3].

**Огляд літературних джерел.** У роботах [3, 4] проведено великий обсяг аналізу статей, присвячених дослідженню різних видів VRP та алгоритмів їхнього розв'язання. Статті для дослідження були обрані використовуючи базу даних LIMO з вимогою включення

*Проведений аналіз результатів дослідження наукових джерел, що досліджують задачу маршрутизації транспортних засобів та методи її розв'язання алгоритмами ройового інтелекту (PI). Досліджено наявність спільних рис алгоритмів PI, що роблять їх ефективними для розв'язування VRP та відмічена кореляція наборів оптимальних параметрів від варіанту VRP який розв'язується. Намічені подальші перспективні напрямки досліджень.*

**Ключові слова:** задача маршрутизації транспортних засобів, ройовий інтелект, метаевристичні методи, логістика.

«vehicle routing» в заголовок, з типом статті: стаття англійською мовою у періодичному виданні.

Книжки, дисертації, збірники і доповіді на конференціях були виключені з дослідження. 21 журнал був відранжований за його фактором впливу у області Operational Research та Transportation та обрані 9 найвпливовіших. Подальше виключення проблем інвентарної маршрутизації, та ін. дало результуючий набір у 144 статті (див. табл. 1).

ТАБЛИЦЯ 1. Журнали та статті відібрані для дослідження

Журнал	Фактор впливу	Кількість відібраних статей
Expert systems with applications	2.203	28
Transportation Research	1.957	16
European Journal of Operational Research	1.815	36
Computers & Operations Research	1.720	34
Decision Support systems	1.687	4
Computers & Industrial Engineering	1.589	10
Transportation science	1.507	15
Applied Mathematics and Computation	1.317	1

Пропонується авторами наступна таксономія джерел, що є адаптованою версією таксономії, запропонованої Eksioglu в 2009. Вони виділяють п'ять основних тем (тип дослідження, характеристики сценарію, фізичні характеристики проблеми, інформаційні характеристики та характеристики даних), кожна зі своїми власними детальними категоріями та підкатегоріями (див. табл. 2).

Тип дослідження (англ. Type of study) визначає статтю відповідно до змісту та може бути розділений на чотири категорії. Перша категорія, що складається з теоретичних статей, сприяє загальному розумінню VRP у всіх її аспектах. Друга категорія відноситься до методів, що були застосовані для розв'язування VRP. Її підкатегоріями є точні методи – вони дозволяють знайти глобальний оптимум, однак часто дорогі в обчислювальному сенсі, тому що VRP (як і багато її варіацій) є NP-складними. Як результат – багато авторів пропонують різноманітні евристики. Дослідники пропонують у своїй таксономії, розрізнити класичні евристики та метаевристики. Статті, які пропонують методи розв'язання в реальному часі, для вирішення динамічних або онлайнних VRP, класифікуються в останню підкатегорію. Третя категорія включає всі статті, що документують реалізацію підходів для розв'язання задач VRP, що мають прямі відповідники у реальному світі, тоді як остання категорія включає дослідження, огляди та мета-дослідження, які стосуються VRP.

Характеристики сценарію (англ. Scenario characteristics) задають список умов для VRP, що впливає на розв'язок не напряму. Проблеми фізичних характеристик (англ. Problem Physical Characteristics) виражаються через умови й обмеження задачі та напряму впливають на розв'язок, тому винесені в окрему категорію.

Інформаційні характеристики (англ. Informational Characteristics) – ця категорія розрізняє еволюцію, якість, доступність і способи обробки інформації. Еволюція інформації стосується того, як вхід VRP розкривається у процесі маршрутизації. Коли вся інформація відома заздалегідь, VRP визначається як статична; однак часто вхідні дані розкриваються або оновлюються у процесі дослідження (частково динамічного).

Характеристики даних (англ. Data Characteristics) локалізують методи розв'язання, що можна перевірити на даних реального світу, синтетичних даних (таких як дані порівняння чи екземпляри, створені автором), або на їх комбінації.

ТАБЛИЦЯ 2. Запропонована таксономія статей про VRP

<ul style="list-style-type: none"> <li>1. Type of study             <ul style="list-style-type: none"> <li>1.1. Theory</li> <li>1.2. Applied methods                 <ul style="list-style-type: none"> <li>1.2.1. Exact methods</li> <li>1.2.2. Classical Heuristics</li> <li>1.2.3. Metaheuristics</li> <li>1.2.4. Simulation</li> <li>1.2.5. Real-time solution methods</li> </ul> </li> <li>1.3. Implementation documented</li> <li>1.4. Survey, review or meta-research</li> </ul> </li> <li>2. Scenario Characteristics             <ul style="list-style-type: none"> <li>2.1. Number of stops on route                 <ul style="list-style-type: none"> <li>2.1.1. Known (deterministic)</li> <li>2.1.2. Partially known, partially probabilistic</li> </ul> </li> <li>2.2. Load splitting constraint                 <ul style="list-style-type: none"> <li>2.2.1. Splitting allowed</li> <li>2.2.2. Splitting not allowed</li> </ul> </li> <li>2.3. Customer service demand quantity                 <ul style="list-style-type: none"> <li>2.3.1. Deterministic</li> <li>2.3.2. Stochastic</li> <li>2.3.3. Unknown</li> </ul> </li> <li>2.4. Request times of new customers                 <ul style="list-style-type: none"> <li>2.4.1. Deterministic</li> <li>2.4.2. Stochastic</li> <li>2.4.3. Unknown</li> </ul> </li> <li>2.5. Onsite service/waiting times                 <ul style="list-style-type: none"> <li>2.5.1. Deterministic</li> <li>2.5.2. Dependent</li> <li>2.5.3. Stochastic</li> <li>2.5.4. Unknown</li> </ul> </li> <li>2.6. Time window structure                 <ul style="list-style-type: none"> <li>2.6.1. Soft time windows</li> <li>2.6.2. Strict time windows</li> <li>2.6.3. Mix of both</li> </ul> </li> <li>2.7. Time horizon                 <ul style="list-style-type: none"> <li>2.7.1. Single period</li> <li>2.7.2. Multi period</li> </ul> </li> <li>2.8. Backhauls                 <ul style="list-style-type: none"> <li>2.8.1. Nodes request simultaneous pickups and deliveries</li> <li>2.8.2. Nodes request either linehaul or backhaul service, but not both</li> </ul> </li> <li>2.9. Node/Arc covering constraints                 <ul style="list-style-type: none"> <li>2.9.1. Precedence and coupling constraints</li> <li>2.9.2. Subset covering constraints</li> <li>2.9.3. Recourse allowed</li> </ul> </li> </ul> </li> <li>3. Problem Physical Characteristics             <ul style="list-style-type: none"> <li>3.1. Transportation network design                 <ul style="list-style-type: none"> <li>3.1.1. Directed network</li> <li>3.1.2. Undirected network</li> </ul> </li> <li>3.2. Location of addresses (customers)                 <ul style="list-style-type: none"> <li>3.2.1. Customer on nodes</li> <li>3.2.2. Arc routing instances</li> </ul> </li> <li>3.3. Number of points of origin                 <ul style="list-style-type: none"> <li>3.3.1. Single origin</li> </ul> </li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>3.3.2. Multiple origin</li> <li>3.4. Number of points of loading/unloading facilities (depot)             <ul style="list-style-type: none"> <li>3.4.1. Single depot</li> <li>3.4.2. Multiple depots</li> </ul> </li> <li>3.5. Time window type             <ul style="list-style-type: none"> <li>3.5.1. Restriction on customers</li> <li>3.5.2. Restriction on depot/hubs</li> <li>3.5.3. Restriction on drivers/vehicle</li> </ul> </li> <li>3.6. Number of vehicles             <ul style="list-style-type: none"> <li>3.6.1. Exactly vehicles</li> <li>3.6.2. Up to vehicles</li> <li>3.6.3. Unlimited number of vehicles</li> </ul> </li> <li>3.7. Capacity consideration             <ul style="list-style-type: none"> <li>3.7.1. Capacitated vehicles</li> <li>3.7.2. Uncapacitated vehicles</li> </ul> </li> <li>3.8. Vehicle homogeneity (Capacity)             <ul style="list-style-type: none"> <li>3.8.1. Similar vehicles</li> <li>3.8.2. Load-specific vehicles</li> <li>3.8.3. Heterogeneous vehicles</li> <li>3.8.4. Customer-specific vehicles</li> </ul> </li> <li>3.9. Travel time             <ul style="list-style-type: none"> <li>3.9.1. Deterministic</li> <li>3.9.2. Function dependent (of current time)</li> <li>3.9.3. Stochastic</li> <li>3.9.4. Unknown</li> </ul> </li> <li>3.10. Objective             <ul style="list-style-type: none"> <li>3.10.1. Travel time dependent</li> <li>3.10.2. Distance dependent</li> <li>3.10.3. Vehicle dependent</li> <li>3.10.4. Function of lateness</li> <li>3.10.5. Implied hazard/risk related</li> <li>3.10.6. Other</li> </ul> </li> <li>4. Information Characteristics             <ul style="list-style-type: none"> <li>4.1. Evolution of information                 <ul style="list-style-type: none"> <li>4.1.1. Static</li> <li>4.1.2. Partially dynamic</li> </ul> </li> <li>4.2. Quality of information                 <ul style="list-style-type: none"> <li>4.2.1. Known (Deterministic)</li> <li>4.2.2. Stochastic</li> <li>4.2.3. Forecast</li> <li>4.2.4. Unknown (Real-time)</li> </ul> </li> <li>4.3. Availability of information                 <ul style="list-style-type: none"> <li>4.3.1. Local</li> <li>4.3.2. Global</li> </ul> </li> <li>4.4. Processing of information                 <ul style="list-style-type: none"> <li>4.4.1. Centralized</li> <li>4.4.2. Decentralized</li> </ul> </li> </ul> </li> <li>5. Data Characteristics             <ul style="list-style-type: none"> <li>5.1. Data used                 <ul style="list-style-type: none"> <li>5.1.1. Real-world data</li> <li>5.1.2. Synthetic data</li> <li>5.1.3. Both real and synthetic data</li> </ul> </li> <li>5.2. No data used</li> </ul> </li> </ul>
---	--

Частково, результати аналізу були презентовані на XXVII Міжнародному семінарі «Комбінаторні конфігурації та їх застосування» [5]. Результати отримані авторами [3, 4] класифікації 144 статей у розрізі проблем та фізичних характеристик наведені у табл. 3. Оскільки кілька факторів зазвичай комбінуються (наприклад, часові вікна, кілька депо та місткість транспортних засобів), категорії не виключають одна одну.

ТАБЛИЦЯ 3. Огляд статей у розрізі варіантів VRP в абсолютних та відносних значеннях

Варіація	Кількість статей	Відносна представленість
CVRP (Capacitated)	128	88.89 %
VRPTW (Time Window)	57	39.58 %
HVRP (Heterogeneous)	27	18.75 %
MDVRP (Multi Depot)	18	12.50 %
VRPPB (Backhauls)	17	11.81 %
SDVRP (Split Deliveries)	16	11.11 %
DVRP (Dynamic)	15	10.42 %
PVRP (Periodic)	14	9.72 %
VRPSD (Stochastic Demands)	13	9.03 %
VRRSPD (Simultaneous Pickup and Delivery)	12	8.33 %
OVRP (Open)	9	6.25 %
TDVRP (Time Dependent)	7	4.86 %
MCVRP (Multi-Compartment)	5	3.47 %
CCVRP (Cumulative)	3	2.08 %

Дослідники зазначають, що для відкритої VRP (Open VRP, OVRP) на практиці проблема виникає, коли автопарк не належить самій компанії або коли наявний автопарк не в змозі задовольнити попит своїх клієнтів і частина діяльності з перевезення вантажу передається сторонньому логістичному постачальнику (3PL). Окрім того, OVRP може бути представлена у варіації забору та доставки, коли після доставки товарів заданим клієнтам транспортні засоби забирають товари від тих самих клієнтів, але у зворотному порядку. Також у дослідженні зауважено, що у реальному житті OVRP часто застосовують для моделювання під час доставки посилок та газет додому, маршрутизації шкільних автобусів, маршрутизації матеріалів вугільних шахт або перевезення небезпечних матеріалів.

В роботах [3, 4] виокремлюють факт того, що Sariklis та Powell першими, для розв'язання OVRP, ще у 2000 році, запропонували як метод розв'язання – евристику, засновану на процедурі мінімального охоплюючого дерева зі штрафами. З 2000 року були запропоновані декілька нових евристик та метаевристик, таких як заборонений пошук, детермінований відпал, пошук у великих околицях та розгалуження та розрізання та PSO для розв'язання OVRP. Також автори зазначають, що загалом, абсолютна більшість статей, що розглядають OVRP містять обмеження щодо місткості транспортних засобів; окрім того, більше половини статей містять обмеження щодо відстані (або часу) та припускають однорідний парк транспортних засобів, та детерміновані вимоги. В роботах наголошується, що усі запропоновані у статтях методи розв'язання протестовані на еталонних прикладах, зібраних з літератури або згенерованих авторами.

Авторами [3] зазначається, що еволюція технологій реального часу, таких як інтелектуальні системи трансформації (англ. Intellectual Transformation Systems, ITS), передові системи управління автопарком (англ. Automated Freight Management Systems, AFMS) та глобальні системи позиціонування (англ. Global Positioning System, GPS), зробила динамічну задачу маршрутизації транспортних засобів (Dynamic VRP, DVRP) досить потужною темою останніх років (10,42 % класифікованих статей обговорюють DVRP). У DVRP (також відомій як онлайн – або VRP реального часу) вхідні дані

постійно розкриваються або оновлюються (наприклад, нові запити клієнтів, що надходять у будь-яку точку маршруту транспортного засобу). Як зазначають автори невизначеність щодо попиту, часу подорожі та часу запиту є найбільш поширеною в DVVRP. Невизначеність щодо часу обслуговування на місці або часу очікування зазвичай не вивчається. Окремо відмічається, що для DVVRP немає контрольних прикладів для об'єктивного тестування та порівняння запропонованих методів рішення.

Дослідники помітили, що більшість VRP припускають, що час подорожі між депо та клієнтами є детермінованим та постійним або ж дорівнює відстані між клієнтами. У реальному житті час подорожі майже ніколи не буває постійним, і залежить від багатьох факторів (наприклад, затори, якість дорожнього покриття тощо), що вочевидь впливає на вартість перевезень через, наприклад, збільшення споживання палива. Залежна від часу VRP (Time Dependent VRP, TDVRP) припускає, що час подорожі це функція часу, що дозволяє визначити вплив різних факторів (наприклад, заторів) на загальну тривалість маршруту, кількість транспортних засобів та вартість перевезень.

Усі досліджені роботи, що стосуються TDVRP, задовольняють властивість відому як властивість «перший прийшов – перший вийшов» (FIFO), яка стверджує, що транспортний засіб, який раніше відправляється від певного клієнта, раніше прибуде до місця призначення. Авторами зазначається, що подорожі, які залежать від часу, у досліджених ними роботах, моделюються за прикладом, де робочий день розділено на кілька періодів, і кожному часовому інтервалу присвоюється постійна швидкість руху, в результаті чого швидкість – це ступінчаста функція часу відправлення (виїзду) для всіх дуг. Чим більша кількість часових інтервалів, тим реалістичнішою буде модель, оскільки швидкості руху змінюватимуться більш поступово, а не різко. Час подорожі між двома клієнтами тоді залежить від часу відправлення (виїзду) від першого клієнта та швидкості, що залежить від часу, на дузі що пов'язує двох клієнтів. Також зазначається, що усі проаналізовані статті про TDVRP включають фактор часових вікон: або м'які (2 статті), або жорсткі (5 статей). Більшість цих часових вікон обмежені клієнтами, тоді як деякі обмежені депо [3, 4].

Цікавий аспект, зазначений дослідниками, – той, що в усіх досліджених статтях TDVRP є детермінованими, і лише одна стаття розглядає TDVRP, що враховує непередбачені події. Також у [1, 3, 4, 6] зазначається, що всі проаналізовані статті розглядають задачі з одним періодом та одним депо. У проаналізованих авторами статтях алгоритми були протестовані або на синтетичних даних, або на комбінації синтетичних та реальних даних і всього 1 стаття розкриває розв'язання TDVRP з часовими вікнами точним методом, що був протестований на тестовому наборі Solomon зі швидкостями, отриманими з реального життя.

Результати досліджень [3, 4] за найбільш популярними методами, що розглядаються в літературі, основним об'єктом дослідження якої є VRP, наведені в табл. 4. Як видно з таблиці – основна увага віддається метаевристичним підходам. Точні методи та класичні евристики зустрічаються мало, причиною чого є їхні недоліки (перші – обчислювально дорогі для складних та великих випадків, а другі – схильні до застрягання у локальних оптимумах). Моделювання та методи розв'язання в реальному часі зустрічаються на диво зрідка. Враховуючи їхню важливість у розв'язанні реалістичних задач, що потребують обчислень у реальному часі, можна прогнозувати зростання представлення цих методів у літературі.

ТАБЛИЦЯ 4. Огляд статей у розрізі методів розв'язання VRP в абсолютних та відносних значеннях

Застосований метод розв'язання	Кількість статей	Відносна представленість
Метаевристика	104	72.22 %
Класична еристика	26	18.06 %
Точні методи	20	13.89 %
Симуляції	7	4.86 %
Методи розв'язання в реальному часі	6	4.17 %

**Аналіз використання алгоритмів ройового інтелекту для розв'язання VRP.** Для розв'язання VRP, як видно, використовуються різні методології. Точні алгоритми, такі як метод гілок та меж, метод гілок та відтинання, динамічне програмування, гарантують знаходження оптимального рішення, проте їх застосування обмежене задачами невеликого розміру через високу обчислювальну складність. Евристичні алгоритми, включаючи конструктивні (наприклад, метод найближчого сусіда, алгоритм заощаджень, алгоритм замітання) та алгоритми локального пошуку (наприклад, 2-орт, 3-орт), здатні швидко знаходити прийнятні рішення, але без гарантії оптимальності. Метаевристичні алгоритми це стратегії високого рівня, які керують евристичним пошуком для уникнення локальних оптимумів та ефективнішого дослідження простору рішень [1].

Першочерговий виклик – це обчислювальна складність (NP-складність), яка обмежує застосування точних методів для великомасштабних задач [6]. Ще один виклик це необхідність врахування різноманітних обмежень реального світу, таких як вантажопідйомність транспортних засобів, часові вікна, динамічні умови тощо. Варто зазначити, що важливо збалансувати фази дослідження (exploration) та експлуатації (exploitation) у метаевристичних алгоритмах, щоб уникнути локальних оптимумів та забезпечити збіжність, як і час, за який працюють алгоритми для великомасштабних варіантів VRP з реального світу, часто є не прийнятним, і їх прискорення, як правило, є не тривіальною задачею, що вимагає суттєвих інвестицій [1]. Залежність алгоритму від параметрів та складність їх оптимального налаштування це також значна проблема [6].

Ройовий інтелект (PI) визначається як обчислювальна та поведінкова парадигма, що черпає натхнення у колективній поведінці децентралізованих, самоорганізованих систем у природі. Основні принципи PI включають самоорганізацію, відсутність централізованого управління, колективну поведінку та емерджентність. Ці принципи дозволяють системам PI бути стійкими, гнучкими та масштабованими, що дозволяє використовувати їх для розв'язування обчислювально складних задач, таких як VRP, для яких знаходження оптимального рішення за поліноміальний час є малоімовірним [7]. Алгоритми PI здатні досліджувати великі простори рішень і часто знаходять майже оптимальні рішення з меншими обчислювальними витратами порівняно з традиційними методами [8].

Алгоритми PI зазвичай відносяться до класу популяційних алгоритмів (таких, що базуються на популяції) разом з генетичними алгоритмами, що здебільшого визначаються природними системами на яких вони базуються [9]. Основні сімейства алгоритмів включають мурашині алгоритми: алгоритм мурашиної колонії (Ant Colony Optimization, ACO), алгоритм систем мурашиних колоній (Ant Colony Systems, ACS), що імітують поведінку мурах у пошуках їжі, використовуючи феромонний слід; бджолині алгоритми, такі як алгоритм штучної бджолиної колонії (Artificial Bee Colony, ABC) та алгоритм бджіл (Bees Algorithm), що відтворюють поведінку медоносних бджіл під час збору нектару та управління вуликом [10]; частинкові алгоритми, найбільш відомий це алгоритм рою частинок (Particle Swarm Optimization, PSO), моделюють соціальну поведінку пташиних зграй або рибних косяків, де частинки рухаються в просторі рішень, керуючись власним досвідом та досвідом сусідів; алгоритми оптимізації китів (Whale Optimization Algorithm, WOA) і його різновиди, що мають у своїй основі унікальну соціальну поведінку горбатих китів під час полювання тощо. Найвідоміші представники наведені у табл. 5.

Як видно з таблиці, на ряду з добре відомими ACO та ABC існує багато інших популярних, алгоритмів PI: алгоритм світлячків (Firefly Algorithm, FA), бере за основу поведінку світлячків [11], алгоритм кажанів (Bat Algorithm, BA), експлуатує ехолокацію кажанів [12], вовчі алгоритми, такі як алгоритм вовчої зграї (Wolf Pack Algorithm, WPA) або алгоритм сірих вовків (Gray Wolf Optimization, GWO) моделюють полювання сірих вовків [13], алгоритм, натхнений голубами (Pigeon-Inspired Optimization, PIO), базується на поведінці голубів при поверненні додому [14] тощо. Кожне з цих сімейств містить алгоритми, що були застосовані до широкого спектру VRP.

ТАБЛИЦЯ 5. Найвідоміші представники алгоритмів ройового інтелекту

Алгоритм	Джерела натхнення	Ключові характеристики	Типове застосування
Алгоритм мурашиної колонії (ACO)	Поведінка мурах при пошуку їжі	Феромонні сліди, імовірнісне конструювання шляху, позитивний зворотний зв'язок.	Задача комівояжера, VRP [15, 16], оптимізація мереж
Алгоритм рою часток (PSO)	Зграї птахів, косяки риб	Частинки, що рухаються в просторі пошуку, керуючись особистим та глобальним найкращими положеннями.	Навчання нейронних мереж, оптимізація портфелів, інженерний дизайн, VRP [9]
Алгоритм штучної бджолоїної колонії (ABC)	Поведінка медоносних бджіл	Працюючі бджоли, бджоли-спостерігачі, бджоли-розвідники, кількість нектару як показник якості рішення.	Оптимізація ланцюгів поставання, кластеризація, машинне навчання, VRP [17]
Алгоритм світлячків (FA)	Світіння світлячків	Привабливість залежно від яскравості та відстані, рух до яскравіших світлячків.	Обробка зображень, інженерний дизайн, мультимодальна оптимізація, VRP [18]
Алгоритм кажанів (BA)	Ехолокація кажанів	Налаштування частоти, контроль гучності, швидкість випромінювання імпульсів	Безперервна та комбінаторна оптимізація, обробка зображень, кластеризація, VRP [19]
Алгоритм вовчої зграї (WPA)	Поведінка вовків при полюванні	Поведінка розвідки, скликання, оточення, вибір лідера, виживання сильніших	Оптимізація функцій, робототехніка, управління ресурсами, VRP [20]
Алгоритм, натхнений голубами (PIO)	Поведінка голубів при поверненні додому	Оператор карти та компаса, оператор орієнтирів	Планування маршрутів БПЛА, обробка зображень, оптимальне управління, VRP [21]
Алгоритм оптимізації китів (WOA)	Поведінка горбатих китів при полюванні	Атака бульбашковою сіткою, спіральна атака, консолідація агентів (оболога)	Оптимізації розмірів та стану конденсаторів у радіальних системах розподілу, планування потокового цеху, VRP [22–24]

Алгоритм мурашиної колонії (ACO) базується на тому, як мурахи знаходять найкоротший шлях до джерела їжі, залишаючи феромонні сліди, ACO використовує штучних мурах, які імовірно будують рішення на графі, що представляє задачу [15, 25]. Штучні мурахи відкладають феромони на пройдених ребрах, і концентрація феромонів впливає на вибір шляху наступними мурахами, створюючи позитивний зворотний зв'язок, який сприяє коротшим та ефективнішим маршрутам [26, 27]. ACO особливо сильний у розв'язанні комбінаторних задач оптимізації, таких як задача комівояжера та її узагальнення, задача маршрутизації транспортних засобів. Однак він може бути обчислювально інтенсивним і потребує ретельного налаштування параметрів [26].

Алгоритм рою часток (PSO) за головну ідею має соціальну поведінку пташиних зграй або рибних косяків. У PSO кожне потенційне рішення представлено частинкою в багатовимірному просторі пошуку. Ці частинки рухаються у просторі, коригуючи свою швидкість та положення на основі власного найкращого знайденого рішення (особистий найкращий результат) та найкращого рішення, знайденого всім роєм (глобальний найкращий результат). PSO відомий своєю простотою, легкістю реалізації та ефективністю у розв'язанні задач безперервної оптимізації. Він успішно застосовувався до різних варіантів VRP [28]. Слабка сторона PSO – його потенціал передчасної збіжності до локальних оптимумів, особливо у складних просторах пошуку [29].

Алгоритм штучної бджолоїної колонії (ABC) імітує поведінку медоносних бджіл під час збору їжі. Колонія складається з трьох типів бджіл: працюючі бджоли, бджоли-спостерігачі та бджоли-розвідники. Працюючі бджоли досліджують джерела їжі (рішення), бджоли-спостерігачі обирають джерела їжі для експлуатації на основі інформації, наданої працюючими бджолами, а бджоли-розвідники шукають нові джерела їжі [30]. Якість джерела їжі (рішення) представлена кількістю нектару (придатністю). ABC відомий своїм балансом між дослідженням та експлуатацією та здатністю розв'язувати складні задачі оптимізації з меншою кількістю параметрів управління порівняно з деякими іншими алгоритмами PI [26, 30]. Він ефективно використовувався для розв'язання VRP [10], однак автори зазначають, що слабкими сторонами алгоритму ABC є передчасна збіжність та застрягання в локальних оптимумах у результаті надмірної експлуатації та втрати різноманітності популяції; труднощі з обробкою жорстких обмежень, таких як обмеження ємності та часу та складність генерації сусідніх рішень; повільна збіжність на пізніх стадіях пошуку та проблеми з представленням рішень і чутливість до налаштувань параметрів.

Алгоритм світлячків (FA) базується на поведінці світлячків, що блимають. У FA кожен світлячок представляє потенційне рішення, а його яскравість відповідає якості рішення [16]. Світлячки приваблюються до яскравіших світлячків, і їх привабливість зменшується з відстанню. FA особливо ефективний у мультимодальних задачах оптимізації, де існує кілька локальних оптимумів, завдяки своїй здатності поділитися на підгрупи навколо різних оптимумів [31]. Він застосовувався до VRP, але може бути чутливим до налаштування параметрів. Основні слабкості FA, у контексті розв'язання VRP, що зазначаються дослідниками, є передчасна збіжність та застрягання у локальних оптимумах у результаті надмірного тяжіння до найяскравішого світлячка та втрати різноманітності популяції; проблеми з балансом між розвідкою та експлуатацією; труднощі з обробкою обмежень VRP через його погану пристосованість до дискретних задач; повільна збіжність у великомасштабних задачах [32].

Алгоритм кажанів (BA) – метаевристичний алгоритм, в основу якого покладений той самий принцип ехолокації, яким користуються кажани. Кажани видають звукові імпульси та використовують відлуння для виявлення здобичі та перешкод. У BA віртуальні кажани літають випадково з різною швидкістю, частотою та гучністю [33]. Вони коригують ці параметри, наближаючись до своєї цілі (оптимального рішення). BA ефективний для розв'язання як безперервних, так і комбінаторних задач оптимізації та застосовувався до VRP. Він має менше параметрів управління порівняно з деякими іншими алгоритмами, що полегшує його реалізацію. Серед слабких сторін алгоритму в контексті VRP зазначають надто швидку збіжність та застрягання у локальних оптимумах через сильну фазу експлуатації та швидку втрату різноманітності популяції; незбалансованість між розвідкою та експлуатацією; проблеми адаптації до дискретних задач через його неперервну природу (BA був розроблений для неперервних задач оптимізації) і як наслідок – складність кодування/декодування (необхідно використовувати складні схеми кодування для перетворення дискретних маршрутів у неперервні вектори і навпаки); неефективна обробка жорстких обмежень VRP через відсутність вбудованих механізмів та сильну залежність від штрафних функцій [19, 31].

Алгоритми сірих вовків (GWO) та вовчої зграї (WPA) імітують поведінку сірих вовків під час полювання, включаючи розвідку, скликання та оточення здобичі [34]. Алгоритм підтримує соціальну ієрархію з альфа-, бета- та омега-вовками, кожен з яких виконує певні ролі в процесі полювання [35]. Ці алгоритми відомі своєю глобальною збіжністю та обчислювальною стійкістю, особливо для задач високої розмірності та дискретних задач. Він застосовувався до різних задач оптимізації, включаючи VRP [36]. Слабкими сторонами GWO та WPA, на мою думку, є ті самі слабкості, що і FA, до яких варто додати слабкість фази пошуку, та проблеми зі збіжністю алгоритму на пізніх етапах. Це підтверджують автори [23, 37].

Алгоритм, натхненний голубами (PIO) заснований на поведінці голубів при поверненні додому, які використовують комбінацію магнітного поля Землі, сонця та орієнтирів для навігації. Алгоритм

використовує два основні оператори: оператор карти та компаса для глобального пошуку та оператор орієнтирів для локального уточнення. РІО відомий своєю швидкою збіжністю та застосовувався до різних задач оптимізації, включаючи планування маршрутів для БПЛА та VRP [11, 38 – 40]. Автори цих досліджень зазначають що РІО має ряд слабких сторін у контексті розв'язування VRP: схильність до локальних оптимумів, особливо для складних багато-пікових завдань, через низьку мінливість та різноманітність популяції; проблеми з багатовимірними та дискретними просторами; сильна залежність від параметрів (наприклад, кількість голубів, фактори навчання, радіус навігації тощо); складність обробки складних обмежень VRP, таких як часові вікна, обмеження вантажопідйомності, наявність кількох депо, час на обслуговування, перерви водіїв тощо; обчислювальна складність для VRP великого масштабу.

Алгоритм оптимізації китів (WOA імітує поведінку горбатих китів, зокрема їхню унікальну стратегію полювання). Цей алгоритм був запропонований Mirjalili та Lewis у 2016 році і швидко зарекомендував себе як ефективний інструмент для розв'язання різноманітних оптимізаційних задач [24]. WOA імітує поведінку горбатих китів, відому як "маневр годування бульбашковою сіткою". Ця стратегія включає занурення китів на глибину близько 12 метрів. Під час підйому вони рухаються по спіралі, одночасно випускаючи бульбашки різного розміру. Ці бульбашки піднімаються в унісон, утворюючи "спіральну бульбашкову сітку", яка служить для захоплення здобичі та спрямування її до центру. Після того, як здобич сконцентрована в бульбашковому колі, кит розташовує свій рот майже вертикально серед бульбашок, щоб поглинути захоплену здобич. Ця тактика емулюється трьома основними механізмами: спіральний рух, випадкове хижацтво та оточення здобичі [14]. Цей прямий перехід від біологічної поведінки до математичних механізмів алгоритму – ключовий. Цілеспрямоване оточення та спіральний рух китів до здобичі, що відображається в експлуатаційній фазі алгоритму, надає WOA високу ефективність у швидкому знаходженні та уточненні локальних оптимумів. Водночас, механізм випадкового пошуку, що імітує початковий пошук здобичі китами, забезпечує достатнє дослідження простору для уникнення передчасної збіжності до субоптимальних рішень. Такий баланс – не випадковий, а прямий наслідок ефективності біологічної стратегії у природі. Цікавий аспект алгоритму це те, що для WOA спостерігається парадокс простоти та продуктивності. Як зазначають автори [41] в літературі WOA схвалюють за його "операційну простоту", "невелику кількість контрольних параметрів", "просту структуру" та "легкість реалізації" і при цьому алгоритм має "сильну здатність виходити з локального оптимуму", "ефективний баланс між дослідженням та використанням" та "конкурентну швидкість конвергенції". Це представляє парадокс: складні проблеми часто вимагають складних алгоритмів, але WOA досягає високої продуктивності зберігаючи відносну простоту. Вищенаведені особливості зумовлюють ефективне використання WOA та його варіацій [42, 43] для розв'язання VRP [14, 22]. Однак автори [14, 22] зазначають, що незважаючи на свою ефективність WOA має кілька визнаних обмежень: проблеми передчасної збіжності, низьку різноманітність популяції, велику залежність від контрольних параметрів (швидкість конвергенції та загальна продуктивність WOA значно залежать від одного і того ж контрольного параметра), обмежена здатність до виходу з локальних оптимумів у певних сценаріях (у складних ландшафтах пошуку здатність WOA виходити з локальних оптимумів може значно знижуватись, і не є абсолютною, а залежить від конкретної структури проблеми), непридатність для бінарних просторів та складних обмежень без модифікацій.

Як зазначають автори [33] дослідники часто утворюють гібриди, поєднуючи алгоритми РІ з методами локального пошуку, такими як 2-орт або 3-орт, щоб покращити якість знайдених рішень шляхом ефективнішої експлуатації локальної околиці.

Для представлення рішень VRP в алгоритмах РІ використовуються різні схеми кодування, такі як кодування на основі перестановок або представлення шляху. Процес декодування перетворює закодоване рішення знову у дійсний набір маршрутів транспортних засобів. Однак, як зазначають

автори [35] при цьому виникають складності, пов'язані із забезпеченням прийнятності рішень щодо обмежень VRP (наприклад, обмеження вантажопідйомності, часові вікна).

Обмеження VRP враховуються у функції пристосованості та процесі генерації рішень алгоритмів PI. Для обробки жорстких та м'яких обмежень використовуються різні методи, наприклад, штрафні функції за порушення обмежень. Конкретні обмеження, такі як часові вікна, найчастіше інтегруються в правила руху або функції оцінки різних алгоритмів PI [1].

Узагальнюючи вище сказане можна зробити висновок, що досліджені алгоритми володіють рядом спільних властивостей, що роблять їх добре підходящими для розв'язання складних комбінаторних задач оптимізації, таких як задача маршрутизації транспортних засобів. До таких властивостей можна віднести наступні властивості:

- децентралізована та самоорганізована природа ройового інтелекту, що дозволяє ефективно досліджувати великий простір рішень VRP – не покладаючись на централізований механізм управління, окремі агенти в рої можуть незалежно досліджувати різні області простору пошуку, збільшуючи різноманітність рішень та зменшуючи ризик передчасної збіжності;
- ефективність фази експлуатації, що дозволяє їм удосконалювати перспективні рішення після їх виявлення, що призводить до отримання якісних результатів;
- стійкість до локальних оптимумів, що є поширеною проблемою у VRP через його комбінаторну природу;
- рандомізація та різноманітні взаємодії популяції, інтегровані в алгоритми PI надають їм здатності виходити з локальних оптимумів;
- колективний інтелект, що виникає внаслідок локальних взаємодій агентів рою дозволяє динамічно адаптуватись до різних типів обмежень, включаючи вантажопідйомність транспортних засобів, часові вікна та інші реальні обмеження;
- здатність до утворення гібридів дає можливість розробникам підсилити слабкі сторони алгоритму, при цьому зберегти його сильні сторони, що позитивно впливає на отримані результати.

Ці висновки підтверджуються дослідниками у роботах [7, 12 – 14, 16, 29, 33, 42, 44 – 46].

Як зазначають автори [47] порівнюючи продуктивність алгоритму системи мурашиної колонії (ACS) та генетичних алгоритмів (GA) на задачі комівояжера (TSP) (VRP можна розглядати як узагальнення TSP [3]) показали, що хоча АСО може знаходити коротші відстані, GA може запропонувати кращу швидкість збіжності. Дослідження [14, 18] показали, що гібридні алгоритми штучної бджолоїної колонії (ABC) можуть перевершувати базовий ABC та давати кращі результати порівняно з іншими евристичними. Гібридні підходи, що поєднують WOA з іншими алгоритмами, такими як алгоритм сірого вовка (GWO), також продемонстрували чудову продуктивність в оптимізації відстані доставки.

Продуктивність різних алгоритмів PI в контексті розв'язання VRP значно залежить від характеристик конкретної варіації. Наприклад, розмір задачі (кількість клієнтів), жорсткість обмежень (наприклад, вузькі часові вікна, суворі обмеження вантажопідйомності), просторовий розподіл клієнтів (топология) тощо, можуть значно впливати на те, який алгоритм покаже найкращі результати.

Алгоритми ройового інтелекту демонструють різний ступінь чутливості до своїх параметрів при застосуванні до VRP. Деякі алгоритми, такі як алгоритм світлячків, відомі своєю значною чутливістю до вибору параметрів, таких як коефіцієнт поглинання світла та параметр рандомізації, де неправильні налаштування можуть призвести до субоптимальних результатів. Інші алгоритми, такі як PSO, можуть бути чутливими до ваги інерції та коефіцієнтів прискорення, які контролюють баланс між дослідженням та експлуатацією [48].

Налаштування параметрів, відіграє вирішальну роль у продуктивності алгоритмів ройового інтелекту при їх застосуванні до задачі маршрутизації транспортних засобів. Значення, присвоєні таким параметрам, як розмір популяції, вага інерції, коефіцієнти прискорення, швидкість випарову-

вання феромонів, коефіцієнти привабливості та розміри кроків, можуть значно вплинути на здатність алгоритму знаходити якісні рішення ефективно [35]. Правильно налаштовані параметри дозволяють алгоритму ефективно збалансувати дослідження простору рішень та експлуатацію перспективних областей, що приводить до кращої збіжності та покращеної якості рішень [26].

Існують різні методології для підбору параметрів алгоритмів ройового інтелекту. Ручне налаштування передбачає коригування параметрів на основі досвіду та експериментів, що може вимагати (і, зазвичай, вимагає) багато часу та ресурсів, та не завжди призводить до отримання оптимальних результатів. Пошук за сіткою та випадковий пошук можуть систематично дослідити попередньо визначений діапазон значень параметрів, але можуть бути обчислювально не виправданими, особливо для алгоритмів з великою кількістю параметрів [49]. Планування експериментів (Design of Experiments, DOE) забезпечує більш структурований підхід до підбору параметрів, систематично варіюючи значення параметрів відповідно до попередньо визначеного експериментального плану для визначення найбільш впливових параметрів та їх оптимальних налаштувань [50].

Оптимальні налаштування параметрів для певного алгоритму ройового інтелекту часто взаємопов'язані зі специфічними характеристиками варіанту задачі маршрутизації транспортних засобів, що розв'язується. Дослідження [49] вказує на те, що значення параметрів, які забезпечують хорошу продуктивність для однієї і тієї ж варіації VRP, можуть бути неоптимальними для іншої. Наприклад, ідеальна вага інерції для PSO в CVRP може відрізнитися від оптимального значення для VRPTW. Аналогічно, швидкість випаровування феромонів в ACO може потребувати коригування залежно від щільності та розподілу клієнтів у варіанті VRP.

На основі результатів аналізу впливу параметрів та їх налаштувань на ефективність алгоритмів PI в контексті розв'язання VRP можна запропонувати деякі загальні рекомендації щодо вибору відповідного алгоритму PI та прийняттого діапазону значень параметрів для різних варіацій VRP. Для VRP із сильно вираженою комбінаторною природою та акцентом на побудові маршрутів, ACO може бути хорошим початком, з параметрами, налаштованими для балансування дослідження нових маршрутів та експлуатації перспективних. Для VRP, що включають безперервні аспекти або потребують балансу між глобальним та локальним пошуком, PSO з ретельно налаштованими вагою інерції та коефіцієнтами прискорення може бути більш ефективним. Стійкість та баланс ABC можуть зробити його придатним для широкого спектру варіацій VRP, з параметрами, такими як розмір колонії та граничне значення, що коригуються залежно від масштабу задачі.

Такі загальні рекомендації підтверджуються авторами [9, 49, 50].

**Висновки.** VRP, будучи критично важливою задачею у логістиці та транспорті, представляє значні обчислювальні труднощі через свою NP-складність та складність реальних обмежень. Ройовий інтелект, натхненний колективною поведінкою природних систем, пропонує потужну парадигму для розв'язання цих проблем за допомогою децентралізованих, самоорганізованих процесів пошуку.

Аналіз результатів досліджень наукових праць, що фокусуються на VRP, виявив, що CVRP залишається найпоширенішим варіантом серед дослідників (представлений майже у 89 % проаналізованих статей). Автори зазначають, що за останні 25 років з'явився новий варіант CVRP – Cumulative CVRP: замість того, щоб мінімізувати загальну відстань (або час у дорозі) як ціль, він мінімізує суму часу прибуття до клієнтів (цей новий варіант VRP вважається дуже перспективним, незважаючи на те, що представлений у всього 2 % класифікованих статей).

Іншим популярним варіантом VRP серед розглянутих наукових праць є VRP з часовими вікнами (представлений у 39.58 % проаналізованих статей). Часові вікна зазвичай обмежені клієнтами (57 статей) і депо (30 статей). Жорсткі часові вікна залишаються найпопулярнішими (44 статті проти 13 статей, в яких згадуються м'які часові вікна). Gendreau та ін. [51] також надають категоризовану бібліографію, де можна знайти найважливіші метаевристики для VRPTW.

Автори зазначають тенденцію до зростання популярності варіантів VRP, які включають припущення з реальних задач, відкритої VRP, динамічної та залежної від часу VRP. Часто такі параметри реального життя, як транспортування готівки, доставка невеликих пакетів, збір сміття або соціальне законодавство щодо робочого часу водіїв, спонукають дослідників розробити конкретне математичне формулювання, для якого потім пропонується підхід. На жаль, такі підходи зазвичай тісно прив'язані до певної проблеми, а деякі навіть спеціально адаптовані до певних екземплярів тесту.

Результати досліджень статей, основним об'єктом дослідження яких є VRP, вказують на те, що найбільш популярні методи, що висвітлюються в літературі, це метаевристичні підходи (представлені у 72.22 % проаналізованих статей) та класичні евристичні методи (представлені у 18.06 % проаналізованих статей). Точні методи займають 3-тє місце, будучи представленими у 13.89 % проаналізованих статей.

Як наступні кроки передбачено проведення власного дослідження наукових праць присвяченої розв'язанню задачі маршрутизації транспортних засобів, з застосуванням розробок на базі генеративного штучного інтелекту.

Проаналізовані різні алгоритми PI, включаючи ACO, PSO, ABC, FA, BA, GWO(WPA), PIO і WOA та їх варіації включно з гібридами, що були успішно адаптовані та застосовані до різних варіантів VRP. Алгоритми демонструють свою універсальність та ефективність у знаходженні майже оптимальних рішень. Не зважаючи на суттєві відмінності між алгоритмами та підходами, їхні спільні властивості дослідження, експлуатації та стійкості до локальних оптимумів роблять їх добре придатними для складної комбінаторної природи VRP.

Варто зазначити, що всі розглянуті алгоритми, в контексті розв'язання VRP, мають спільні негативні особливості – це схильність до застрягання у локальних оптимумах та значну залежність продуктивності цих алгоритмів від налаштування параметрів і ефективності стратегії їх налаштування. Алгоритми та значення його параметрів сильно взаємопов'язані зі специфічними характеристиками варіанту VRP, що підкреслює необхідність інтегрованих підходів до їх вибору та їх налаштування.

Незважаючи на значний прогрес у застосуванні алгоритмів ройового інтелекту для розв'язання задачі маршрутизації транспортних засобів, залишаються актуальними виклики в ефективному розв'язанні великомасштабних варіантів VRP у реальному часі [1]. Обчислювальна складність залишається головною перешкодою, особливо зі збільшенням розміру задачі та врахуванням більш реалістичних обмежень. Тому майбутні напрямки досліджень можуть бути зосереджені на розробці більш адаптивних та самоналагоджуваних алгоритмів PI, здатних автоматично коригувати свої параметри та стратегії пошуку на основі специфічних характеристик варіації VRP та прогресу пошуку. Інтеграція методів машинного навчання також відкриває перспективні можливості для прогнозування оптимальних налаштувань параметрів, варіацій уточнюючого алгоритму для керування процесом пошуку (динамічна гібридизація) або навіть вибору найбільш підходящого алгоритму PI для заданої варіації VRP. Крім того, застосування PI до задач TDVR, де попит клієнтів або час у дорозі змінюються у реальному часі, виглядає як високо перспективний напрямок подальших досліджень.

**Подяки.** Автор висловлює подяку науковому керівнику, Стецюку Петру Івановичу, за керівництво і допомогу; завідувачу відділу Інституту кібернетики імені В.М. Глушкова НАН України, Гуляницькому Леоніду Федоровичу, за консультацію та допомогу; завідувачу кафедри Інформаційних управляючих систем та технологій УжНУ, Міці Олександровичу Володимировичу, за допомогу та мотивацію; батьку, Єгеру Дмитру Олександровичу, за мотивацію, підтримку та допомогу; аспіранту УжНУ, Тиводару Станіславу Романовичу та студенту УжНУ, Задорожному Богдану Олександровичу, за допомогу та обговорення результатів.

## Список літератури

1. Hao D., Mao X., Wei Y., Sun L., Yang Y. Review of research on vehicle routing problems. *International Conference on Smart Transportation and City Engineering (STCE 2023)*. 2024. **13018**. <https://doi.org/10.1117/12.3024185>
2. Гуляницький, Л.Ф., Коткова А.А. До класифікації задач маршрутизації транспортних засобів. *Науковий вісник Ужгородського університету. Серія "Математика і інформатика"*. 2020. **1** (36). С. 73–84. [https://doi.org/10.24144/2616-7700.2020.1\(36\).73-84](https://doi.org/10.24144/2616-7700.2020.1(36).73-84)
3. Braekers K., Ramaekers K., Nieuwenhuys I. The Vehicle Routing Problem: State of the Art Classification and Review. *Computers & Industrial Engineering*. 2016. **99**. P. 300–313. <https://doi.org/10.1016/j.cie.2015.12.007>
4. Tan S., Yeh W. The Vehicle Routing Problem: State-of-the-Art Classification and Review. *Applied Sciences*. 2021. **11** (21). 10295. <https://doi.org/10.3390/app112110295>
5. Єгер М.Д. Результати огляду статей про розв'язання задач маршрутизації транспортних засобів. *Матеріали XXVII міжнародного практичного семінару «Комбінаторні конфігурації та їх застосування»*. 2025. С. 20–26
6. Kumar S., Panneerselvam R. A Survey on the Vehicle Routing Problem and Its Variants. *Intelligent Information Management*. 2012. **4**. P. 66–74. <https://doi.org/10.4236/iim.2012.43010>
7. Yang J., Qu L., Shen Y., Shi Y., Cheng S., Zhao J., Shen X. Swarm Intelligence in Data Science: Applications, Opportunities and Challenges. *Advances in Swarm Intelligence*. 2020. **12145**. P. 3–14. [https://doi.org/10.1007/978-3-030-53956-6\\_1](https://doi.org/10.1007/978-3-030-53956-6_1)
8. Abdelaziz D. The Impact of Swarm Intelligence on Optimization Problems. *Int J Swarm Evol Comput*. 2024. **13** (5). P. 390. <https://doi.org/10.35248/2090-4908.24.13.390>
9. Abadlia H., Belhassen I., Smairi N. Adaptive multi-strategy particle swarm optimization for solving NP-hard optimization problems. *International Journal of Knowledge-Based and Intelligent Engineering Systems*. 2024. **28** (1). P. 195–209. <https://doi.org/10.3233/KES-230137>
10. Shi Y., Meng F., Shen G. A Modified Artificial Bee Colony Algorithm for Vehicle Routing Problems with Time Windows. *Information Technology Journal*. 2012. **11**. P. 1490–1495. <https://doi.org/10.3923/ijtj.2012.1490.1495>
11. Yang X., Xingshi H. Firefly Algorithm: Recent Advances and Applications. *International Journal of Swarm Intelligence*. 2013. **1**. <https://doi.org/10.1504/IJSI.2013.055801>
12. Vasuki A. Nature-Inspired Optimization Algorithms (1st ed.). Chapman and Hall/CRC. 2020. P. 166–180. <https://doi.org/10.1201/9780429289071>
13. Wu H., Zhang F., Wu L. New swarm intelligence algorithm-wolf pack algorithm. *Systems Engineering and Electronics*. 2013. **35**. <https://doi.org/10.3969/j.issn.1001-506X.2013.11.33>
14. Tian A.-Q., Chu S.-C., Pan J.-S., Liang Y. A Novel Pigeon-Inspired Optimization Based MPPT Technique for PV Systems Processes. 2020. **8** (3). P. 356. <https://doi.org/10.3390/pr8030356>
15. Rizzoli A., Montemanni R., Lucibello E., Gambardella L. Ant colony optimization for real-world vehicle routing problems. *Swarm Intelligence*. 2006. **1**. P. 135–151. <https://doi.org/10.1007/s11721-007-0005-x>
16. Гуляницький Л.Ф., Рибальченко О.В. Оптимізація маршрутів при плануванні місій гібридних транспортних систем «дрон+транспортний засіб». *Кибернетика та комп'ютерні технології*. 2023. **3**. С. 44–58. <https://doi.org/10.34229/2707-451X.23.3.4>
17. Mar'if F., Ubaidillah H., Mahmudy W. Hybrid Artificial Bee Colony Algorithm and Improved Simulated Annealing for Capacitated Vehicle Routing Problem. *Knowledge Engineering and Data Science (KEDS)*. 2022. **5** (2). P. 109–121. <https://doi.org/10.17977/UM018V5I22022P109-121>
18. Singh G., Prateek M., Kumar S., Verma M., Singh D., Lee H. Hybrid Genetic Firefly Algorithm-based Routing Protocol for VANETs. *IEEE Access*. 2022. **10**. P. 1. <https://doi.org/10.1109/ACCESS.2022.3142811>
19. Cut E., Xu H., Wong W. Metaheuristics is All You Need. 2025. <https://arxiv.org/html/2411.05797v2> (звернення: 25.06.2025)
20. Mahmudy W.F., Widodo A.W., Haikal A.H. Challenges and Opportunities for Applying Meta-Heuristic Methods in Vehicle Routing Problems: A Review. *Engineering Proceedings*. 2024. **63**. P. 12. <https://doi.org/10.3390/engproc2024063012>
21. Zhang X., Wei Y., Hashim Z. Improvement of Swarm Intelligence Algorithm and Its Application in Logistics Network Routing. *Taiwan Ubiquitous Information, Journal of Network Intelligence*. 2023. **8** (4). P. 1077–1094. [https://bit.kuas.edu.tw/~jni/2023/vol8/s4/02\\_JNI-S-2023-04-014.pdf](https://bit.kuas.edu.tw/~jni/2023/vol8/s4/02_JNI-S-2023-04-014.pdf) (звернення: 25.06.2025)
22. Yu N., Jiang W., Hu R., Qian B., Wang L. Learning Whale Optimization Algorithm for Open Vehicle Routing Problem with Loading Constraints. *Discrete Dynamics in Nature and Society*. 2021. P. 1–14. <https://doi.org/10.1155/2021/8016356>
23. Pham V., Nguyen V., Dang N. Hybrid whale optimization algorithm for enhanced routing of limited capacity vehicles in supply chain management. *Scientific Reports*. 2024. **14**. <https://doi.org/10.1038/s41598-024-51359-2>

24. Mirjalili S., Lewis A. The Whale Optimization Algorithm. *Advances in Engineering Software*. 2016. **95**. P. 51–67. <https://doi.org/10.1016/j.advengsoft.2016.01.008>
25. Dorigo M. Ant colony optimization. *Scholarpedia*. 2007. **2** (3). P. 1461. <http://doi.org/10.4249/scholarpedia.1461>
26. Dorigo M., Birattari M., Stützle T. Ant Colony Optimization. *Computational Intelligence Magazine, IEEE*. 2006. **1**. P. 28–39. <https://doi.org/10.1109/MCI.2006.329691>
27. Krentz T., Greenhagen C., Roggow A., Desmond D., Khorbotly S. A modified ant colony optimization algorithm for implementation on multi-core robots. *Swarm/Human Blended Intelligence Workshop (SHBI)*. 2015. P. 1–6. <http://doi.org/10.1109/SHBI.2015.7321683>
28. Alinezhad H., Yaghoubi S., Hosseini-Motlagh S.-M., Allahyari S., Saghafi M. An Improved Particle Swarm Optimization for a Class of Capacitated Vehicle Routing Problems. *International Journal of Transportation Engineering*. 2018. **5**. P. 331–347. [https://www.researchgate.net/publication/321299808\\_An\\_Improved\\_Particle\\_Swarm\\_Optimization\\_for\\_a\\_Class\\_of\\_Capacitated\\_Vehicle\\_Routing\\_Problems](https://www.researchgate.net/publication/321299808_An_Improved_Particle_Swarm_Optimization_for_a_Class_of_Capacitated_Vehicle_Routing_Problems) (звернення: 25.06.2025)
29. Wei J., Gu Y., Lu B., Cheong N. RWOA: A novel enhanced whale optimization algorithm with multi-strategy for numerical optimization and engineering design problems. *PLoS ONE*. 2025. **20** (4). <https://doi.org/10.1371/journal.pone.0320913>
30. Salehahmadi Z., Manafi A. How can bee colony algorithm serve medicine? *World J Plast Surg*. 2014. **3** (2). P. 87–92. <https://pmc.ncbi.nlm.nih.gov/articles/PMC4236990> (звернення: 25.06.2025)
31. Johari N., Zain A., Mustaffa N., Udin A. Firefly Algorithm for Optimization Problem. *Applied Mechanics and Materials*. 2013. **421**. P. 512–517. <https://doi.org/10.4028/www.scientific.net/AMM.421.512>
32. O M.Z., Mu`azu M.B., Adedokun A.E., Tijani S.A., Bello I.A. Optimized Model Simulation of a Capacitated Vehicle Routing problem based on Firefly Algorithm. *Covenant Journals Informatics and Communication Technology*. 2018. **6** (2). <https://journals.covenantuniversity.edu.ng/index.php/cjict/article/view/1177> (звернення: 25.06.2025)
33. Wei X., Xiao Z., Wang Y. Solving the Vehicle Routing Problem with Time Windows Using Modified Rat Swarm Optimization Algorithm Based on Large Neighborhood Search. *Mathematics*. 2024. **12**. P. 1702. <https://doi.org/10.3390/math12111702>
34. Wu H.-S., Zhang F.-M. Wolf Pack Algorithm for Unconstrained Global Optimization. *Mathematical Problems in Engineering*. 2014. P. 1–17. <https://doi.org/10.1155/2014/465082>
35. Geetha S., Poonthalir G., Vanathi P.T. A Hybrid Particle Swarm Optimization with Genetic Operators for Vehicle Routing Problem. *Journal of Advances in Information Technology*. 2010. **1** (4). P. 181–188. <https://doi.org/10.4304/jait.1.4.181-188>
36. Zhang H., Xiao L., Chao M., Liangzhong C. An Elite Wolf Pack Algorithm Based on the Probability Threshold for a Multi-UAV Cooperative Reconnaissance Mission. *Drones*. 2024. **8** (9). P. 513. <https://doi.org/10.3390/drones8090513>
37. Wu H., Xiao R. Flexible Wolf Pack Algorithm for Dynamic Multidimensional Knapsack Problems. *Research (Wash D C)*. 2020. **2020**. <https://doi.org/10.34133/2020/1762107>
38. Li C., Duan H. Target detection approach for UAVs via improved Pigeon-inspired Optimization and Edge Potential Function. *Aerospace Science and Technology*. 2014. **39**. P. 352–360. <https://doi.org/10.1016/j.ast.2014.10.007>
39. Yang Z., Duan H., Fan Y., Deng Y. Automatic Carrier Landing System multilayer parameter design based on Cauchy Mutation Pigeon-Inspired Optimization. *Aerospace Science and Technology*. 2018. **79**. <https://doi.org/10.1016/j.ast.2018.06.013>
40. Hu C., Xia Y., Zhang J. Adaptive Operator Quantum-Behaved Pigeon-Inspired Optimization Algorithm with Application to UAV Path Planning. *Algorithms*. 2018. **12**. P. 3. <https://doi.org/10.3390/a12010003>
41. Yunpeng M., Wang X., Meng W. A Reinforced Whale Optimization Algorithm for Solving Mathematical Optimization Problems. *Biomimetics*. 2024. **9** (9). P. 576. <https://doi.org/10.3390/biomimetics9090576>
42. Mohammed H.M., Umar S.U., Rashid T.A. A Systematic and Meta-Analysis Survey of Whale Optimization Algorithm. *Computational Intelligence and Neuroscience*. 2019. **8718571**. <https://doi.org/10.1155/2019/8718571>
43. Junhao W., Gu Y., Yan Y., Li Z., Lu B., Pan S., Cheong N. LSEWOA: An Enhanced Whale Optimization Algorithm with Multi-Strategy for Numerical and Engineering Design Optimization Problems. *Sensors*. 2025. **25** (7). P. 2054. <https://doi.org/10.3390/s25072054>
44. Jedrzejowicz P., Wierzbowska I. Parallelized Swarm Intelligence Approach for Solving TSP and JSSP Problems. *Algorithms*. 2020. **13** (6). P. 142. <https://doi.org/10.3390/a13060142>
45. Stamadianos T., Taxidou A., Marinaki M., Marinakis Y. Swarm intelligence and nature inspired algorithms for solving vehicle routing problems: a survey. *Operational Research*. 2024. **24**. P. 47. <https://doi.org/10.1007/s12351-024-00862-5>
46. Shen Y., Liu M., Yang J., Shi Y. Middendorf M., A Hybrid Swarm Intelligence Algorithm for Vehicle Routing Problem With Time Windows. *IEEE Access*. 2020. **8**. P. 93882–93893. <https://doi.org/10.1109/ACCESS.2020.2984660>

47. Alexander A., Sriwindono H. The Comparison of Genetic Algorithm and Ant Colony Optimization in Completing Travelling Salesman Problem. *ICSTI EAI*. 2020. <https://doi.org/10.4108/eai.20-9-2019.2292121>
48. Zixuan X., Xueyu H., Wenwen L. Subpopulation Particle Swarm Optimization with a Hybrid Mutation Strategy. *Computational Intelligence and Neuroscience*. 2022. **9599417**. <https://doi.org/10.1155/2022/9599417>
49. Joy G., Huyck C., Yang X.-S. Parameter Tuning of the Firefly Algorithm by Three Tuning Methods: Standard Monte Carlo, Quasi-Monte Carlo and Latin Hypercube Sampling Methods. 2025. <https://doi.org/10.48550/arXiv.2504.18545>
50. Abdolhosseinzadeh M., Alipour M.-M. Design of experiment for tuning parameters of an ant colony optimization method for the constrained shortest Hamiltonian path problem in the grid networks. *Numerical Algebra, Control and Optimization*. 2021. **11** (2). P. 321–332. <https://goi.org/10.3934/naco.2020028>
51. Gendreau M., Potvin J.-Y., Bräysy O., Hasle G., Løkketangen A. Metaheuristics for the Vehicle Routing Problem and Its Extensions: A Categorized Bibliography. *The Vehicle Routing Problem: Latest Advances and New Challenges*. 2008. **43**. P. 143–169. <https://doi.org/10.1007/978-0-387-77778-8>

Одержано 30.06.2025

**Єгер Максим Дмитрович**,  
аспірант Ужгородського національного університету.  
[maksym.yeher@uzhnu.edu.ua](mailto:maksym.yeher@uzhnu.edu.ua)

MSC 90B20, 90C27, 90C59, 68T20

**Maksym Yeher**

## Analysis of Swarm Intelligence Algorithms Used for Solving Vehicle Routing Problems

*Uzhhorod National University, Ukraine*

Correspondence: [maksym.yeher@uzhnu.edu.ua](mailto:maksym.yeher@uzhnu.edu.ua)

**Introduction.** The Vehicle Routing Problem (VRP), first formulated by Danzig and Ramseur in 1959, has remained one of the most popular research subjects to date. This popularity stems from numerous factors, including its wide applicability across various economic sectors. VRP belongs to the class of NP-hard problems, implying high computational complexity in finding optimal solutions, especially for large-scale variations. Over the past 25 years, approaches to its classification and solution have evolved significantly, driven by real-world requirements and constraints, as well as advancements in optimization methods and computational power.

This article analyzes research findings from studies focused on VRP, confirming a substantial shift in researchers' attention towards metaheuristic approaches. It examines application of the most popular swarm intelligence algorithms and their variations, including hybrids, for solving VRP, and what makes them successful. Furthermore, the study investigates the correlation between sets of algorithm parameters.

**The purpose of the paper is to** investigate usage of swarm intelligence algorithms for solving the Vehicle Routing Problem. Paper attempts to determine what makes them effective for solving VRPs (if such) and how this is related to their parameter set. In addition, the study explores whether there is a correlation between the parameter sets of SI algorithms considered effective for VRPs.

**Results.** An analysis of the results of research articles on VRP was conducted, which made it possible to identify the most popular variations of VRP and rank the methods for solving them. A comprehensive analysis of the most popular SI algorithms, including their variations and hybrids, for solving VRP was conducted. Their strengths and weaknesses were analyzed, and algorithmic features that make them effective in solving VRP were identified. A correlation analysis was conducted between the optimal parameter sets of algorithms and a strong dependence of the optimal parameter sets on the specific variation of VRP being solved was revealed.

**Conclusions.** The analysis of the literature reveals that the Capacitated Vehicle Routing Problem (CVRP) remains the most prevalent VRP variant among researchers. Another popular variant is the Vehicle Routing Problem with Time Windows (VRPTW). Overall, there is an increasing trend in the popularity of VRP variants that incorporate real-world assumptions: Open VRP (OVRP), Dynamic VRP (DVRP), and Time-Dependent VRP (TDVRP). Often, real-life parameters such as cash transportation, small parcel delivery, waste collection, or social legislation regarding drivers' working hours, prompt researchers to develop narrow mathematical models. Unfortunately, these models are typically hard-wired to a specific problem, and some are even specifically

adapted to particular test instances. The most popular methods studied in the literature are metaheuristic methods, classical heuristic methods, and exact methods.

Various Swarm Intelligence (SI) algorithms were analyzed. Their shared properties of exploration, exploitation, and resistance to local optima make them well-suited for the complex combinatorial nature of VRP. However, the choice of algorithm and its parameters is strongly interrelated with the specific VRP variation, emphasizing the need for integrated approaches to their selection and tuning. Despite significant progress, challenges remain in effectively solving large-scale real-world VRPs.

**Keywords:** Vehicle Routing Problem, swarm intelligence, metaheuristic methods, logistics.