

<https://doi.org/10.31891/2307-5732-2025-349-44>

УДК 004.04: 004.04

КИРИЧЕНКО ІРИНА

Харківський Національний Університет Радіоелектроніки

<https://orcid.org/0000-0002-7686-6439>e-mail: iryna.kyrychenko@gmail.com**ОЛІЙНИК АРТЕМ**

Харківський Національний Університет Радіоелектроніки

<https://orcid.org/0009-0008-8613-964X>e-mail: oleynikk.artem@gmail.com

ДОСЛІДЖЕННЯ МОДЕЛЕЙ ОПТИМІЗАЦІЇ ВИКОРИСТАННЯ МЕДИЧНОГО ОБЛАДНАННЯ У КРИЗОВИХ УМОВАХ

Розглянуто алгоритми та моделі оптимізації управління системами охорони здоров'я для ефективного використання медичного обладнання в кризових умовах. У дослідженні запропоновано комплексний підхід, який поєднує методи лінійного цілочислового програмування (MILP), евристичні алгоритми, такі як генетичний алгоритм та оптимізація рою частинок (PSO), а також стохастичні методи. Основною метою цього дослідження є підвищення ефективності розподілу ресурсів у медичних закладах під час надзвичайних ситуацій, забезпечуючи оптимальний розподіл критично важливого медичного обладнання у відповідь на динамічно мінливі потреби.

Ключові слова: оптимізація ресурсів, медичне обладнання, кризові ситуації, алгоритми управління, оцінка ефективності, MapReduce.

KYRYCHENKO IRYNA**OLIINYK ARTEM**

Kharkiv National University of Radio Electronics

RESEARCH OF MODELS FOR OPTIMIZING THE USE OF MEDICAL EQUIPMENT IN CRISIS CONDITIONS

The algorithms and models for optimizing the management of healthcare systems for the efficient use of medical equipment in crisis conditions are considered. The study proposes an integrated approach that combines linear integer programming (MILP) methods, heuristic algorithms such as the genetic algorithm and particle swarm optimization (PSO), as well as stochastic methods. The key objective of this research is to improve the efficiency of resource allocation in medical facilities during emergency situations, ensuring optimal distribution of critical medical equipment in response to dynamically changing demands.

The study evaluates the performance of different optimization techniques based on key criteria: computational speed, resource allocation quality, and adaptability to unpredictable changes. The integration of parallel computing through the MapReduce framework significantly enhances processing efficiency, making it suitable for real-time crisis response scenarios.

The proposed models are tested under simulated crisis conditions, including pandemic scenarios, natural disasters, and supply chain disruptions. The results indicate that hybrid optimization methods, combining deterministic and heuristic approaches, provide a balance between computational efficiency and optimal decision-making.

The study highlights the importance of resilience in healthcare systems and the need for flexible optimization models that can dynamically adjust to sudden changes in medical equipment demand. The findings contribute to the development of advanced decision-support systems for emergency medical resource management, ensuring that healthcare institutions can respond effectively to critical shortages. Future research will focus on expanding these models to optimize pharmaceutical distribution and emergency medical staffing.

Keywords: resource optimization, medical equipment, crisis situations, management algorithms, performance assessment, MapReduce.

Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями

Сучасні кризи, такі як пандемії, військові конфлікти та природні катастрофи, створюють значний тиск на системи охорони здоров'я, вимагаючи їхньої максимальної ефективності. Під час пандемії COVID-19 багато країн зіткнулися з дефіцитом апаратів штучної вентиляції легень (ШВЛ), що призвело до зростання смертності. Наприклад, у Південній Кореї через нестачу ліжок у відділеннях інтенсивної терапії пацієнти змушені були чекати вдома, що призвело до численних смертей [1].

У США під час першої хвилі пандемії медичні працівники стикнулися з браком масок з високим ступенем захисту, що призвело до рекомендацій щодо повторного використання одноразових масок у кризових ситуаціях. У деяких лікарнях Нью-Йорка використовували одні й ті самі апарати ШВЛ для двох пацієнтів, що підвищувало ризик інфікування [2].

У кризових ситуаціях, таких як пандемії, військові конфлікти та природні катастрофи, системи охорони здоров'я стикаються з низкою серйозних викликів. Непередбачувані навантаження на медичні заклади вимагають швидкої адаптації та ефективного управління ресурсами. Дефіцит медичного обладнання та матеріалів ускладнює надання своєчасної та якісної допомоги пацієнтам. Крім того, динамічні зміни в потребах пацієнтів та умовах роботи медичних установ потребують гнучких підходів до управління та розподілу ресурсів. Вирішення цих проблем є критично важливим для забезпечення стійкості та ефективності системи охорони здоров'я під час криз.

Дослідження підтверджують, що оптимізація розподілу медичних ресурсів під час криз є ключовим фактором у зниженні смертності та підвищенні ефективності лікування. Зокрема, у роботі "Optimal Resource and Demand Redistribution for Healthcare Systems Under Stress from COVID-19" автори

аналізують моделі перерозподілу пацієнтів та ресурсів між медичними закладами для зменшення навантаження на окремі лікарні та підвищення загальної ефективності системи охорони здоров'я. Результати показали, що оптимальний перерозподіл може значно знизити потребу в додаткових ресурсах та зменшити ризик перевантаження окремих лікарень [3].

Наукова новизна даного дослідження полягає в розробці гібридного підходу, який поєднує методи лінійного цілочислового програмування (MILP), евристичні алгоритми (генетичний алгоритм, алгоритм рою часток) та стохастичні методи для оптимізації розподілу медичного обладнання в умовах криз. На відміну від існуючих підходів, які часто орієнтовані на статичні умови або окремі аспекти управління ресурсами, запропонована модель враховує динамічні зміни попиту, обмеженість ресурсів та необхідність швидкого прийняття рішень. Крім того, використання паралельних обчислень MapReduce дозволяє значно прискорити процес оптимізації, що є критично важливим у кризових ситуаціях.

Аналіз досліджень та публікацій

Існуючі підходи до управління ресурсами охорони здоров'я здебільшого базуються на статистичних моделях або стандартних методах логістики, які часто не враховують унікальних умов кризових ситуацій. Більшість досліджень свідчать, що логістичні моделі можуть адаптуватися до динамічних змін попиту в умовах кризи за допомогою нових технологій, таких як штучний інтелект, динамічні моделі прийняття рішень, та інноваційні стратегії управління, хоча деякі дослідження вказують на необхідність подальшої адаптації для підвищення ефективності.[4]

За умов пандемії COVID-19 особливо чітко проявилися обмеження традиційних логістичних моделей: вони переважно не враховували швидке зростання й коливання попиту, а також складну взаємодію між різними закладами охорони здоров'я. Потреба в оперативній координації призвела до активного впровадження підходів, заснованих на штучному інтелекті та сучасних методах математичного моделювання, здатних враховувати різноманітні обмеження (ресурсні, часові, просторові) і вчасно коригувати рішення. Зокрема, дослідження свідчать, що пандемія COVID-19 виявила обмеження традиційних логістичних моделей, підкресливши необхідність впровадження стійких, гнучких та технологічно просунутих рішень для покращення управління ланцюгами постачання, логістики охорони здоров'я та забезпечення безперервності медичних послуг [5].

Інше дослідження підкреслює важливість перерозподілу попиту та ресурсів між медичними закладами для зменшення необхідності створення додаткових потужностей під час криз. Автори розробили серію лінійних та змішаних цілочисельних моделей програмування для вирішення проблеми перерозподілу попиту та ресурсів, враховуючи невизначеність попиту та операційні обмеження. Моделі були перевірені на даних про госпіталізації під час COVID-19 у різних штатах США, що показало значне зменшення потреби у додаткових потужностях при впровадженні запропонованих підходів [6].

Використання штучного інтелекту (ШІ) у сфері охорони здоров'я також набуває все більшого поширення. ШІ-технології застосовуються для персоналізації лікування, розробки лікарських засобів, аналізу результатів медичних обстежень, діагностики та телемедицини. Зокрема, алгоритми ШІ можуть аналізувати медичні дані та передбачати індивідуальний ризик розвитку захворювань, що дозволяє лікарям розробляти оптимальні стратегії лікування та моніторингу здоров'я [7]. Варто зазначити, що окрім використання ШІ одним із перспективних напрямків у контексті цифрової медицини є використання блокчейн-сховищ для безпечного зберігання та передачі медичних зображень, зокрема, рентгеновських знімків та даних МРТ. В цьому контексті застосування різних алгоритмів, дозволяє оптимізувати обсяг збережених даних, зберігаючи баланс між якістю зображення та ефективністю обробки [8].

Однак, впровадження ШІ в управління медичними ресурсами стикається з низкою викликів. Серед них – необхідність стандартизації та якості даних, етичні питання, пов'язані з конфіденційністю пацієнтів, а також значне енергоспоживання, пов'язане з роботою ШІ-систем.

Таким чином, сучасні дослідження підкреслюють необхідність розробки нових моделей та алгоритмів управління медичними ресурсами, які б враховували динамічні зміни попиту та забезпечували гнучкість і адаптивність у кризових умовах.

Формулювання цілей статті

Метою роботи є: аналіз алгоритмів і моделей для ефективного управління медичними ресурсами в кризових умовах.

Виклад основного матеріалу

У цьому дослідженні запропоновано комплексний підхід до оптимізації розподілу медичного обладнання, що поєднує різні методи – детерміновані, стохастичні та адаптивні. Основна мета такого підходу – забезпечити максимально ефективне використання ресурсів у кризових умовах.

Одним із ключових методів оптимізації є лінійне цілочислове програмування (MILP, Mixed-Integer Linear Programming). Це математичний підхід, який дає змогу знаходити найкращі рішення для задач, де присутні лінійні обмеження та цілочислові змінні. У випадку розподілу медичного обладнання цей метод використовується для чіткої математичної формалізації проблеми.

Для реалізації MILP у рамках дослідження використовувався Python та бібліотека PuLP, яка надає зручний інтерфейс для створення та розв'язання задач лінійного програмування. Зокрема, для формулювання цільової функції та обмежень застосовувалися вбудовані функції PuLP, що дозволяють

швидко налаштувати задачу оптимізації та отримати рішення навіть для великих обсягів даних [9].

Основна ціль MILP – максимізувати ефективність розподілу обладнання, що відображається у цільовій функції:

$$Z = \max_{x_{ij}} \sum_{i=1}^N \sum_{j=1}^M \omega_{ij} x_{ij}, \quad (1)$$

де x_{ij} — кількість обладнання типу i , призначеного лікарні j ; ω_{ij} — ваговий коефіцієнт, що враховує пріоритетність; N, M — кількість типів обладнання та лікарень відповідно.

При цьому метод враховує низку важливих обмежень. По-перше, ресурсні обмеження, які гарантують, що загальна кількість обладнання, виділена всім лікарням, не перевищує доступний запас:

$$\sum_{j=1}^M x_{ij} \leq R_i \quad \forall i = 1, \dots, N \quad (2)$$

По-друге, мінімальні потреби лікарень, які визначають, що кожна лікарня повинна отримати не менше ніж необхідну їй кількість обладнання:

$$\sum_{i=1}^N x_{ij} \geq D_j \quad \forall j = 1, \dots, M \quad (3)$$

Хоча лінійне цілочислове програмування (MILP) дозволяє отримувати математично обґрунтовані оптимальні рішення, його застосування може бути обмежене значними обчислювальними витратами, особливо у випадках великомасштабних задач [10]. У таких умовах доцільно використовувати евристичні алгоритми, які здатні швидко знаходити розв'язки навіть за наявності великої кількості змінних та складних обмежень.

Евристичні методи, такі як генетичні алгоритми (GA) та алгоритм рою часток (PSO), широко застосовуються у задачах оптимізації, що включають велику кількість параметрів і складні обмеження [11]. На відміну від детермінованих підходів, ці алгоритми не гарантують пошук абсолютно оптимального рішення, однак дозволяють отримати розв'язки високої якості у значно коротші строки.

У рамках дослідження для реалізації генетичних алгоритмів (GA) використовувався Python і бібліотека DEAP, яка є потужним інструментом для реалізації еволюційних алгоритмів. Ця бібліотека дозволяє моделювати популяцію можливих рішень і застосовувати оператори кросовера, мутації та відбору, що є важливими для досягнення високоякісних розв'язків при складних обмеженнях [12].

Генетичні алгоритми базуються на принципах природного добору та імітують механізми біологічної еволюції. Вони працюють із популяцією можливих рішень, які поступово покращуються шляхом використання операторів кросовера та мутації. Основні етапи застосування генетичних алгоритмів у цьому дослідженні включають:

1. Кодування рішень: Кожен потенційний розподіл ресурсів подається у вигляді вектора хромосом:

$$x = (x_{11}, x_{12}, \dots, x_{NM}); \quad (4)$$

де x_{ij} – розподіл обладнання між лікарнями.

2. Фітнес-функція: Відповідає цільовій функції Z , при цьому враховуються штрафи за порушення обмежень, щоб забезпечити відповідність знайденого рішення заданим критеріям.
3. Оператори генетичного алгоритму:
 - Кросовер: Комбінація рішень від "батьківських" хромосом для створення нових варіантів розподілу ресурсів. Наприклад, використовується однорідна рекомбінація.
 - Мутація: Випадкові зміни в розподілі обладнання x_{ij} у допустимих межах, що дозволяє уникнути локальних екстремумів.
 - Відбір: Використовуються стратегії турнірного відбору або рулетки, що визначають, які рішення переходять у наступне покоління.

Таким чином, застосування генетичних алгоритмів у процесі оптимізації дозволяє швидко адаптувати розподіл ресурсів до змінних умов та знаходити ефективні та гнучкі рішення навіть у випадках великої невизначеності. Хоча генетичні алгоритми (GA) демонструють високу ефективність у пошуку розв'язків, їхнім недоліком є потреба в значній кількості ітерацій для досягнення оптимального результату. Щоб прискорити процес оптимізації та покращити адаптивність системи до змінних умов, доцільно застосовувати метод рою часток (PSO), який ефективно працює у динамічних середовищах [13].

Метод рою часток (Particle Swarm Optimization, PSO) – це евристичний алгоритм глобальної оптимізації, що базується на імітації поведінки зграї птахів або рою комах. Основна ідея цього підходу полягає в тому, що кожен потенційний розв'язок (частка) рухається в просторі можливих рішень та коригує свою траєкторію на основі власного найкращого досвіду та колективного досвіду всього рою. Завдяки цьому алгоритм здатний швидко адаптуватися до змін у середовищі та знаходити ефективні рішення навіть за складних обмежень.

Для експерименту використовується функція "fitness", яка дозволяє обчислювати якість рішень

на основі цільової функції Z , де враховуються штрафи за порушення обмежень, щоб забезпечити відповідність знайденого рішення заданим критеріям.

Метод рою часток (PSO) був реалізований через бібліотеку PySwarms, що надає прості вектори для налаштування параметрів рою та налаштування поведінки часток у просторі пошуку [14]. Завдяки використанню цієї бібліотеки, вдалося створити ефективне середовище для пошуку оптимальних рішень у динамічних умовах.

У рамках PSO кожна частка представляє можливий варіант розподілу ресурсів, а її рух визначається такою формулою:

$$v_{k+1} = w \cdot v_k + c_1 r_1 (p_{\text{best}} - x_k) + c_2 r_2 (g_{\text{best}} - x_k) \quad (5)$$

де w — коефіцієнт інерції, що впливає на швидкість оновлення розв'язку; c_1, c_2 — коефіцієнти навчання, які визначають, наскільки сильно частка орієнтується на свій найкращий досвід (p_{best}) та найкращий знайдений розв'язок у всьому рої (g_{best}), r_1, r_2 — випадкові значення, що додають стохастичний елемент до пошуку.

Головна перевага PSO — здатність ефективно обробляти великомасштабні задачі оптимізації, при цьому швидко знаходячи рішення з урахуванням змінних умов, таких як коливання попиту чи обмеження ресурсів [15]. Завдяки цьому PSO широко застосовується у складних розподільчих системах, дозволяючи динамічно адаптувати управління медичними ресурсами навіть у кризових ситуаціях.

У кризових ситуаціях, коли попит на медичні ресурси може змінюватися непередбачувано, доцільно використовувати стохастичні методи оптимізації, що дозволяють адаптувати рішення до невизначеності.

Стохастична оптимізація — це підхід, що враховує невизначеність у вхідних даних, наприклад, непередбачувані зміни попиту, відмови обладнання або логістичні затримки. У контексті управління медичними ресурсами стохастичні методи дозволяють враховувати різні сценарії розвитку подій та забезпечувати гнучкість і стійкість системи охорони здоров'я.

Основними підходами до стохастичної оптимізації є метод Монте-Карло і стохастичне програмування [16]. Метод Монте-Карло використовується для моделювання невизначеності шляхом генерації тисяч можливих сценаріїв. Кожен сценарій базується на ймовірнісних розподілах параметрів (наприклад, нормальний розподіл для прогнозування змін попиту). Для кожного сценарію розраховується оптимальний розподіл ресурсів, після чого отримані результати усереднюються для формування стійкої стратегії. В той час як стохастичне програмування це математичний підхід, який формалізує проблему з урахуванням випадкових параметрів. Оптимальне рішення вибирається на основі математичного очікування цільової функції:

$$\min_x E_{\xi}[L(x, \xi)] \quad (6)$$

де x — рішення, яке потрібно знайти; ξ — випадкові фактори (наприклад, коливання попиту або технічні збої); $L(x, \xi)$ — функція втрат або витрат у різних сценаріях; E_{ξ} — математичне очікування втрат у середньому по всіх можливих сценаріях.

Стохастичні методи дозволяють гнучко адаптуватися до динамічних змін, що є критично важливим у кризових ситуаціях, коли точність прогнозування обмежена, а необхідність приймати швидкі та ефективні рішення є першочерговою [17].

У випадку коли рішення приймаються на різних рівнях управління, ефективним підходом стають мультиагентні моделі, що дозволяють змодельовувати взаємодію різних учасників медичної системи та оптимізувати їхні дії в умовах кризових ситуацій.

Для реалізації стохастичного програмування використано бібліотеку PySP, яка дозволяє реалізувати методи стохастичної оптимізації, зокрема моделювання різних сценаріїв розвитку подій та обчислення математичних очікувань. Бібліотека забезпечує зручну інтеграцію з іншими бібліотеками Python і допомагає швидко запускати моделі на основі стохастичних сценаріїв [18].

Мультиагентні моделі — це підхід, що імітує взаємодію незалежних учасників системи охорони здоров'я, таких як лікарні, логістичні центри, пацієнти та постачальники обладнання. Кожен агент має свої локальні цілі та приймає рішення, орієнтуючись на доступні дані, що дозволяє адаптивно розподіляти ресурси в динамічних умовах.

Основні компоненти мультиагентного підходу:

- Локальні цілі: Наприклад, кожна лікарня прагне максимізувати свою забезпеченість медичним обладнанням, зважаючи на прогнозовані потреби;
- Механізми взаємодії: Агенти обмінюються інформацією через центральний модуль управління, що дозволяє оперативно коригувати розподіл ресурсів;
- Автоматичне перерозподілення: При раптовому збільшенні попиту (наприклад, при зростанні кількості важкохворих) агенти автономно оптимізують використання наявних ресурсів.

Для балансування між різними критеріями прийняття рішень використовується функція корисності:

$$U = \alpha \cdot \text{Ефективність} + \beta \cdot \text{Швидкість} + \gamma \cdot \text{Етика} \quad (7)$$

де α – коефіцієнт, що відображає важливість ефективного використання обладнання; β – вплив швидкості ухвалення рішень (наприклад, для критично хворих пацієнтів); γ – параметр, що враховує етичні аспекти розподілу ресурсів (рівність доступу, пріоритети).

Завдяки своїй децентралізованій структурі мультиагентні моделі відображають реальні процеси прийняття рішень у кризових умовах, забезпечуючи більш гнучке та адаптивне управління системою охорони здоров'я [19].

У мультиагентній моделі для обробки взаємодії агентів було використано бібліотеку Mesa, що дозволяє моделювати агентні системи за допомогою простих і гнучких інструментів для налаштування поведінки агентів та їх взаємодії в межах складних умов [20]. Моделювання різних агентів охоплює лікарні, логістичні центри та пацієнтів, дозволяючи оцінити ефективність взаємодії в реальних умовах.

Для прискорення обчислень у великомасштабних задачах використовується MapReduce — модель розподіленої обробки даних, яка дозволяє зменшити час розрахунків в декілька разів [21]. Використання технології MapReduce для обробки великих даних було реалізовано через бібліотеку Dask, що дозволяє масштабувати обчислення і забезпечити паралельну обробку даних на декількох вузлах. Це забезпечує ефективність при розв'язанні задач із великою кількістю змінних і даних, зменшуючи час розрахунків та покращуючи масштабованість у великих системах. Її реалізація включає в себе операції Map та Reduce.

Операція Map — Задача розбивається на підзадачі (наприклад, оптимізація для окремого регіону) і кожен вузол кластера обробляє свою підзадачу паралельно.

Операція Reduce — результати з усіх вузлів агрегуються в єдине рішення.

Наприклад, для генетичного алгоритму це означає паралельну обробку різних частин популяції.

Запропоновані методи дозволяють ефективно вирішувати задачу розподілу медичного обладнання в умовах криз, поєднуючи точність детермінованих моделей, гнучкість евристик та адаптивність мультиагентних систем. Використання паралелізації забезпечує масштабованість, що критично важливо для оперативного реагування у великих системах охорони здоров'я.

Нижче наведено інтегрований опис структури та ключових аспектів наукового результату, що був отриманий при проведенні експериментальної перевірки методів оптимізації управління системами охорони здоров'я для ефективного використання медичного обладнання у кризових умовах. Експеримент було сплановано відповідно до класичного підходу: від визначення цілей і вибору моделей до кодування факторів, формування факторного плану, перевірки відновлюваності результатів та аналізу адекватності отриманих моделей.

В ході дослідження здійснювалося оцінювання ефективності різних підходів до оптимізації, включно з лінійними і цілочисловими оптимізаційними моделями (MILP), евристичними алгоритмами (GA), стохастичними методами (PSO) та їх гібридними варіаціями. Одночасно перевірявся вплив застосування паралельних обчислювальних технологій MapReduce. Експеримент складався з декількох етапів. Спершу було сформульовано мету та обрано критерії ефективності, серед яких якість розподілу обладнання (значення цільової функції), час отримання рішень, здатність алгоритмів адаптуватися до динамічних змін та мінімальний обсяг даних, необхідний для досягнення заданого рівня ефективності. Потім було визначено моделі для порівняння (MILP, GA, гібриди MILP+GA, PSO), а також різні умови експерименту, що включали варіювання обсягу вхідних даних (від 5000 до 10000 записів), наявність динамічних подій (раптове збільшення попиту на медичне обладнання або його частковий вихід з ладу), а також масштабування кількості вузлів у MapReduce-кластері (2, 3 або 4 вузли).

Кодування факторів здійснювалося шляхом нормування параметрів, що дало змогу сформувати повний факторний план експериментів, у якому перевірялася робота кожного алгоритму за різних комбінацій умов. З метою забезпечення статистичної надійності результати дослідів оцінювалися за середніми значеннями показників та їх стандартними відхиленнями, оскільки кожний експеримент запускався кілька разів. Перевірка адекватності моделей та отриманих результатів проводилася із застосуванням статистичних критеріїв та методів крос-валідації, що дозволило переконатися в коректності обраних підходів до оптимізації.

Вибір генетичних алгоритмів з-поміж інших евристичних підходів (наприклад, табу-пошуку, симульованого відпалу тощо) був зумовлений тим, що GA добре зарекомендували себе у задачах з великою кількістю змінних і складними обмеженнями, де структура пошукового простору не завжди дозволяє ефективно застосовувати локальні пошукові методи. Крім того, GA забезпечує гнучку адаптацію завдяки механізмам кросовера й мутації, що важливо в умовах динамічних змін (часткових або непередбачуваних відмов обладнання, стрибків у попиті тощо). Можливість паралелізації операцій з окремими хромосомами додатково спрощує масштабування на великі дані.

Проведені експерименти показали, що лінійні оптимізаційні моделі MILP при великих обсягах даних (наприклад, 10000 записів) забезпечують майже оптимальний розподіл, але потребують значно більше часу. Евристичні та стохастичні методи, такі як GA та PSO, працювали швидше. Результати часу обчислень для послідовних версій алгоритмів представлені на рис. 1.

Застосування MapReduce дозволило суттєво скоротити час обчислень. В таблиці 1 наведено приклади прискорення при використанні 3 та 4 вузлів MapReduce для GA, гібридної моделі та PSO.

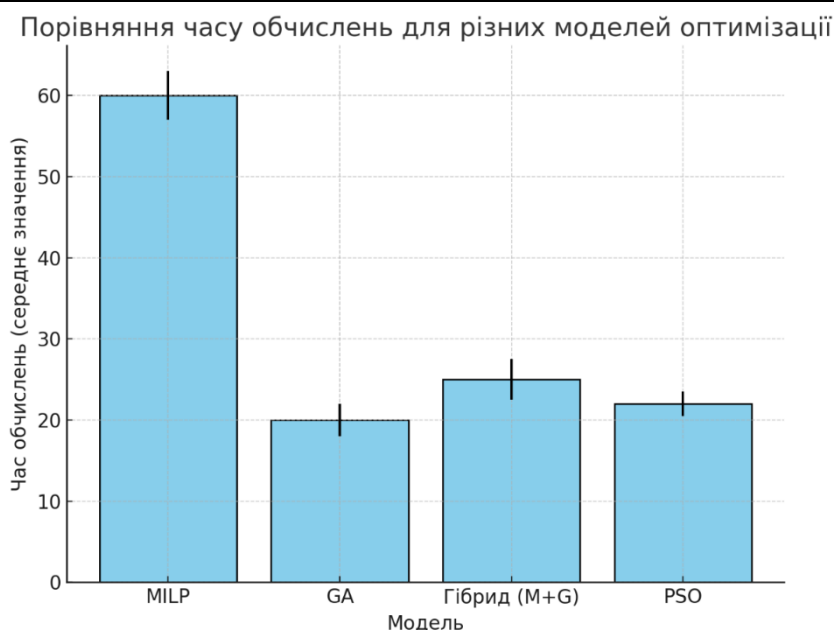


Рис. 1. Порівняння часу обчислень різних моделей

Таблиця 1

Модель	Початковий час, с	3 вузли (с)	Прискорення	4 вузли (с)	Прискорення
GA	20	7	~2.9	5	~4
Гібрид (M+G)	25	9	~2.8	6.5	~3.85
PSO	22	8	~2.75	5.5	~4

При порівнянні якості отриманих рішень MILP забезпечував майже оптимальне значення цільової функції (0.98 від максимуму), тоді як GA і PSO демонстрували дещо нижчу точність (0.93 та 0.92 відповідно), а гібридний підхід був ближчим до MILP (0.95). Ці дані проілюстровану на рис. 2.

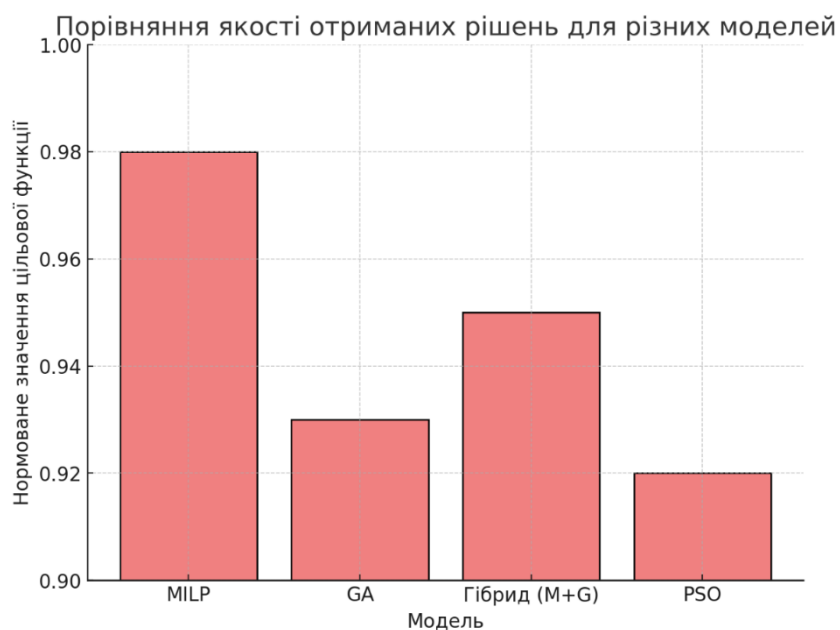


Рис. 2. Порівняння якості отриманих рішень різних моделей

Здатність алгоритмів адаптуватися до динамічних змін, наприклад, різкого збільшення попиту на ШВЛ, була найнижчою у MILP (1–2 бали з 5), дещо кращою у GA (3–4 бали) та PSO (4–5 балів), тоді як гібридний метод демонстрував стабільні 4 бали. Аналіз мінімального обсягу даних, необхідного для досягнення 90% від максимального рівня ефективності, вказав, що MILP вимагав близько 8000 записів, GA – 7000, гібрид – 6800, а PSO – 7500 записів, що свідчить про відносну економію даних у випадку евристичних підходів. Всі ці результати узагальнено в таблиці 2.

Таблиця 2

Модель	Мін. обсяг даних	Адаптивність (бали)
MILP	8000	1–2
GA	7000	3–4
Гібрид (M+G)	6800	4
PSO	7500	4–5

Адаптивність у таблицях виражена в балах від 1 до 5. Цей показник оцінювався за методикою, яка порівнює швидкість повторної оптимізації (або доналаштування рішення) після різкого зростання попиту чи виходу обладнання з ладу, а також відсоток втрати ефективності під час перехідного періоду. На основі оцінки реакції алгоритму йому присвоювався бал від 1 до 5. Один бал означає, що алгоритм практично не зміг адаптуватися до змін, його рішення залишалися неадекватними новим умовам. В той же час, п'ять балів означає, що алгоритм продемонстрував високу здатність до адаптації, швидко та ефективно перерозподіливши ресурси для вирішення проблеми, що виникла.

Важливо зазначити, що оцінка адаптивності може мати деякий суб'єктивний характер. Для підвищення об'єктивності оцінки бажано залучати кількох експертів та усереднювати їх оцінки.

Отримані результати засвідчили, що евристичні, стохастичні та гібридні методи, особливо при використанні паралелізації за допомогою MapReduce, здатні наблизитися за точністю до MILP, проте значно перевершують його за швидкістю обчислень. Це має критичне значення в кризових умовах, коли необхідні швидкі рішення. Крім того, GA та гібриди потребували менше даних для досягнення цільової ефективності, що робить їх привабливими за умов обмеженої інформації.

Для перевірки ефективності алгоритмів та моделей оптимізації використання медичного обладнання у кризових умовах використовувалися синтетично згенеровані дані, оскільки повні реальні дані про лікарні та пацієнтів були недоступні через етичні та конфіденційні обмеження. Генерація передбачала створення 50 віртуальних лікарень у п'яти регіонах, де для кожної лікарні було задано початкову кількість обладнання (ШВЛ, МРТ, УЗД, рентген). Потоки з 10 000 пацієнтів імітувалися випадковим чином, із урахуванням пуассонівського розподілу прибуття і різних рівнів критичності. Для реалізації цих сценаріїв використовували середовище Python (бібліотеки NumPy, Pandas) та спеціально розроблені скрипти. Було синтетично сформовано низку даних, що імітують роботу системи охорони здоров'я протягом 30-денного періоду. Дані охоплюють три основні аспекти: наявність та характеристики обладнання у лікарнях, потоки пацієнтів із різними медичними потребами та події, які змінюють доступність ресурсів під час кризових періодів.

Насамперед було створено набір даних про 50 лікарень, розташованих у п'яти різних регіонах (по 10 лікарень на регіон). Кожній лікарні притаманна своя початкова кількість обладнання: апаратів штучної вентиляції легень (від 5 до 20 одиниць), МРТ (1–3 одиниці), ультразвукових систем (3–10 одиниць) та рентгеновських апаратів (2–6 одиниць). Ці дані задають базову інфраструктурну основу, демонструючи неоднорідність розподілу ресурсів та створюючи умови для різного ступеня навантаження на медичні заклади в залежності від їхнього географічного розташування.

Другий важливий компонент – інформація про 10 000 пацієнтів, які з'являються у лікарнях протягом 30 днів згідно з пуассонівським розподілом прибуття. Для кожного пацієнта зазначається день та година приходу, регіон та тип необхідного обладнання (апарат ШВЛ, МРТ, УЗД чи рентген), а також критичність стану (від 1 до 5). Тривалість процедур для різних типів обладнання розподілена випадково: для ШВЛ – у середньому близько 5 годин, для МРТ – 1.5 години, для УЗД – 0.75 години та для рентгену – 0.75 години. Така деталізація дозволяє моделювати реалістичні ситуації, коли одночасно може зростати попит на певні послуги, а рівень критичності пацієнтів змушує визначати пріоритети у використанні дефіцитних ресурсів.

Третій аспект даних пов'язаний із кризовими подіями. На 10–15 дні симуляції спостерігається штучно створений період підвищеного попиту на апарати ШВЛ, який збільшується приблизно на 50%. Це моделює ситуацію, коли різке погіршення епідеміологічної ситуації призводить до різкого зростання кількості пацієнтів, що потребують інтенсивної терапії. На 20–21 дні в деяких лікарнях випадково вибране МРТ-обладнання виходить з ладу (близько 10% від загальної кількості), що відображає технічні неполадки чи дефіцит витратних матеріалів. Інформація про такі зміни доступності обладнання зберігається у додаткових даних, де вказано день, ідентифікатор лікарні, тип обладнання та величину змін у його доступності.

Загалом, сформовані дані дають цілісне уявлення про роботу медичної системи у динамічних і кризових умовах, враховуючи розподіл ресурсів по лікарнях, масовий та нерівномірний потік пацієнтів, а також непередбачувані події, що впливають на доступність обладнання. Завдяки цьому можна досліджувати різні моделі оптимізації, перевіряти їх стійкість до стресових ситуацій та здатність забезпечити максимальну ефективність використання обмежених медичних ресурсів.

Особливості обраних наборів даних:

— Реалістичність моделювання: Дані створені з урахуванням реальних сценаріїв кризових ситуацій у системах охорони здоров'я, таких як пандемії чи техногенні катастрофи;

- Гетерогенність даних: Набори даних включають різні типи інформації — від кількості медичного обладнання до характеристик пацієнтів та змін у доступності обладнання;
- Динамічність: Дані враховують зміни у системі протягом часу, що дозволяє моделювати адаптацію системи до кризових умов;
- Масштабованість: Кількість лікарень, пацієнтів та обладнання дозволяє проводити аналіз на різних рівнях — від локальних до загальнонаціональних;
- Випадковість розподілу: Початкові обсяги обладнання та прибуття пацієнтів мають елементи випадковості, що імітує реальні умови роботи системи охорони здоров'я.

Для узагальнення та порівняння отриманих результатів експерименту, пов'язаного з оптимізацією розподілу медичного обладнання в кризових умовах, можна виділити кілька ключових метрик ефективності:

Час обчислень: Відображає середній час (у секундах) на знаходження розв'язку відповідним методом. Менший час обчислень є кращим, оскільки це критично в умовах, коли рішення потрібно ухвалювати швидко.

Якість розподілу: Нормоване значення цільової функції (від 0 до 1), яке показує, наскільки близьким є отриманий розподіл до теоретично можливого максимуму. Чим більше це значення, тим ефективніше використовується обладнання.

Адаптивність: Узагальнений показник (у балах від 1 до 5), що характеризує здатність алгоритму (або моделі) підлаштовуватися до динамічних змін, наприклад різкого збільшення попиту на певний тип обладнання чи зменшення доступних ресурсів.

Мінімальний обсяг даних: Кількість записів (у тисячах), необхідна алгоритму для досягнення 90% від максимальної можливої ефективності. Менше значення свідчить про те, що алгоритм може працювати якісно навіть при відносно обмеженій інформації.

Для формальної оцінки зручно переходити до нормованих показників, щоб усі метрики знаходилися в діапазоні від 0 до 1 і були орієнтовані на максимізацію. Тобто, що більше значення метрики, то краще. Для цього:

Для метрик, де більше = краще (якість, адаптивність), застосовуємо:

$$V_{norm} = \frac{V - V_{min}}{V_{max} - V_{min}}; \quad (8)$$

Для метрик, де менше = краще (час обчислень, обсяг даних), зручно використати:

$$V_{norm} = 1 - \frac{V - V_{min}}{V_{max} - V_{min}} \quad (9)$$

Після обчислення нормованих значень можна об'єднати їх у зведену таблицю. Нижче подано таблицю 3 де всі показники переведено у шкалу [0; 1].

Таблиця 3

Модель	Нормований час	Нормована якість	Нормована адаптивність	Норм. обсяг даних
MILP	0.00	1.00	0.00	0.70
GA	0.70	0.95	0.67	0.21
Гібрид (M+G)	0.65	0.97	0.83	0.25
PSO	0.75	0.94	0.92	0.42

Отримані дані представлені у графічному вигляді на рис. 3.

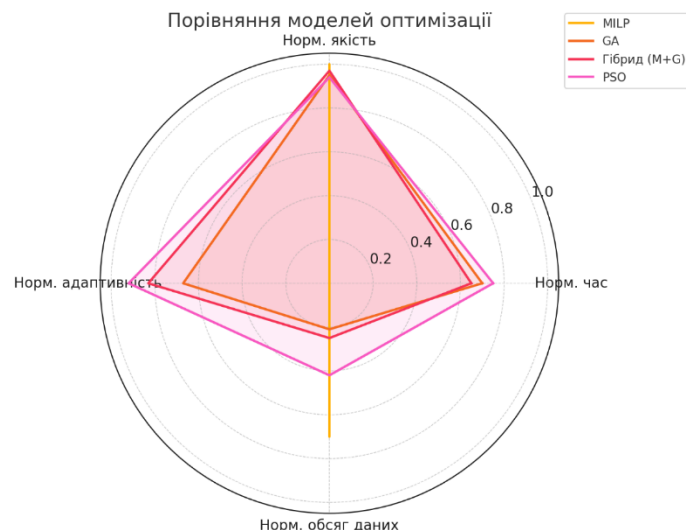


Рис. 3. Радарна діаграма, яка порівнює чотири моделі оптимізації за різними критеріями

На основі нормованих показників можна застосувати низку формальних методів багатокритеріальної оптимізації, наприклад метод парето-оптимальності для визначення моделей, які не можна одночасно покращити за всіма ознаками, не погіршуючи якусь із них [22], а також розглянути адитивну згортку, де за допомогою вагових коефіцієнтів можна підкреслити пріоритетність часу обчислень, адаптивності чи якості. Формула лінійної функції корисності для адитивної згортки наведена нижче:

$$U = \alpha \cdot (\text{Нормований час}) + \beta \cdot (\text{Нормована якість}) + \gamma \cdot (\text{Нормована адаптивність}) + \delta \cdot (\text{Норм. обсяг даних}) \quad (9)$$

де $\alpha, \beta, \gamma, \delta$ – вагові коефіцієнти, можна отримати єдине інтегральне числове значення.

У кризових умовах важливіше мінімізувати час і мати високу адаптивність, тому можна обрати $\alpha = 0.3, \beta = 0.2, \gamma = 0.3, \delta = 0.2$. На практиці вибір ваг залежить від конкретних пріоритетів системи охорони здоров'я.

За результатами проведених обчислень можна визначити найкращі компромісні методи залежно від конкретної ситуації: чи є найбільш критичним фактором точність розподілу, чи швидкість отримання рішення, чи адаптивність до динамічних подій.

Висновки з даного дослідження

і перспективи подальших розвідок у даному напрямі

Запропонований гібридний підхід, що поєднує детерміновані, стохастичні та мультиагентні методи, продемонстрував здатність підвищувати адаптивність системи охорони здоров'я та забезпечувати ефективний розподіл медичного обладнання навіть за умов різних змін попиту та обмеженості ресурсів. Завдяки інтеграції методів паралельних обчислень пошук розв'язків суттєво прискорюється, а гнучкість урахування динамічних факторів зростає.

Наукова новизна дослідження полягає у розробці гібридних алгоритмів, які поєднують методи лінійного цілочислового програмування (MILP), евристичні підходи (GA, PSO) та паралельні обчислення (MapReduce) для швидкої оптимізації розподілу ресурсів. Запропонована модель враховує динамічні зміни попиту, обмеженість ресурсів та необхідність швидкого прийняття рішень, що вигідно відрізняє її від традиційних статичних підходів. Додатково, мультиагентні системи дозволяють підвищити автономність процесу прийняття рішень та враховувати специфіку кожного медичного закладу.

Разом із тим, дослідження виявило певні обмеження застосованих методів. Зокрема, евристичні алгоритми GA та PSO можуть давати локально-оптимальні рішення, що може знижувати ефективність у випадку високої складності проблеми. Також, на етапах впровадження алгоритмів у реальні умови можлива поява людських помилок під час налаштування моделей та їх адаптації до нових кризових ситуацій. Крім того, на результати оптимізації можуть впливати технічні обмеження, пов'язані з інфраструктурою обчислювальних систем.

Окрім традиційних методів оптимізації, останні дослідження показують ефективність використання гібридних моделей збереження та індексації даних у великих системах [23]. Це дозволяє зменшити навантаження на обчислювальні ресурси та покращити швидкість доступу до критично важливих даних у кризових умовах.

Перспективи подальших досліджень включають вдосконалення алгоритмів для зменшення ризику локальної оптимізації та підвищення їхньої здатності знаходити глобально-оптимальні рішення. Також варто розширити модель на управління фармацевтичними запасами та вакцинами в умовах кризових ситуацій, а також розробити багаторівневі мультиагентні системи для міжнародної координації, що дозволить оперативніше обмінюватися даними та приймати ефективні рішення у масштабах глобальної охорони здоров'я.

Література

1. Kuhn, A. (2020, March 10). How a South Korean city is changing tactics to tamp down its COVID-19 surge. *NPR*. <https://www.npr.org/sections/goatsandsoda/2020/03/10/812865169/how-a-southkorean-city-is-changing-tactics-to-tamp-down-its-Covid-19-surge>
2. Centers for Disease Control and Prevention. (2019, April 9). Strategies for optimizing the supply of N95 respirators. *CDC*. <https://www.cdc.gov/coronavirus/2019-ncov/hcp/respirators-strategy/index.html>
3. Parker, F., Sawczuk, H., Ganjkanloo, F., Ahmadi, F., & Ghobadi, K. (2020). Optimal resource and demand redistribution for healthcare systems under stress from COVID-19. *arXiv*.
4. Zheng, J., Bao, F., Shen, Z., & Xu, C. (2022). Data-driven dynamic adjustment and optimization model of emergency logistics network in public health. *Risk Management and Healthcare Policy*, 15, 151–169. <https://doi.org/10.2147/RMHP.S350275>
5. Chen, A., Hansoti, B., & Hsu, E. (2022). The COVID-19 pandemic response and its impact on post-pandemic health emergency and disaster risk management in the United States. *Sustainability*. <https://doi.org/10.3390/su142316301>
6. Jana, S., Majumder, R., & Ghose, D. (2023). Critical medical resource allocation during COVID-19 pandemic. *5th World Congress on Disaster Management: Volume III*, 275–283. <https://doi.org/10.4324/9781003341963-37>

7. Artificial intelligence technologies in modern medicine: Implementation and issues. (2024). *Ukrainian Medical Journal*, 163(5). <https://doi.org/10.32471/umj.1680-3051.163.257497>
8. Tereshchenko, G., Kyrychenko, I., Smelyakov, K., & Oliynyk, A. (2024). Analysis of image compression methods for storage in decentralized blockchain repositories. *Bionics of Intelligence*, 1(100), 23–35.
9. Mitchell, S., O'Sullivan, M., & Dunning, I. (2011). *PuLP: A linear programming toolkit for Python*.
10. Camisa, A., Notarnicola, I., & Notarstefano, G. (2020). Distributed primal decomposition for large-scale MILPs. *IEEE Transactions on Automatic Control*, 67, 413–420. <https://doi.org/10.1109/TAC.2021.3057061>
11. Shabir, S. (2016). A comparative study of genetic algorithm and the particle swarm optimization.
12. Kim, J., & Yoo, S. (2018). Software review: DEAP (Distributed Evolutionary Algorithm in Python) library. *Genetic Programming and Evolvable Machines*, 20, 139–142. <https://doi.org/10.1007/s10710-018-9341-4>
13. Abd-El-Wahed, W., Mousa, A., & El-Shorbagy, M. (2011). Integrating particle swarm optimization with genetic algorithms for solving nonlinear optimization problems. *Journal of Computational and Applied Mathematics*, 235, 1446–1453. <https://doi.org/10.1016/j.cam.2010.08.030>
14. Miranda, L. (2018). PySwarms: A research toolkit for particle swarm optimization in Python. *Journal of Open Source Software*, 3, 433. <https://doi.org/10.21105/JOSS.00433>
15. Ghaffari, Z., Nasiri, M., Bozorgi-Amiri, A., & Rahbari, A. (2020). Emergency supply chain scheduling problem with multiple resources in disaster relief operations. *Transportmetrica A: Transport Science*, 16, 930–956. <https://doi.org/10.1080/23249935.2020.1720858>
16. Homem-De-Mello, T., & Bayraksan, G. (2014). Monte Carlo sampling-based methods for stochastic optimization. *Surveys in Operations Research and Management Science*, 19, 56–85. <https://doi.org/10.1016/J.SORMS.2014.05.001>
17. Shmelova, T. (2018). Stochastic methods for estimation and problem solving in engineering: Stochastic methods of decision making in aviation. *IGI Global*, 139–160. <https://doi.org/10.4018/978-1-5225-5045-7.CH006>
18. Watson, J., Woodruff, D., & Hart, W. (2012). PySP: Modeling and solving stochastic programs in Python. *Mathematical Programming Computation*, 4, 109–149. <https://doi.org/10.1007/s12532-012-0036-1>
19. García-Magariño, I., & Gutiérrez, C. (2013). Agent-oriented modeling and development of a system for crisis management. *Expert Systems with Applications*, 40, 6580–6592. <https://doi.org/10.1016/j.eswa.2013.06.012>
20. Masad, D., & Kazil, J. (2015). Mesa: An agent-based modeling framework. *Proceedings of the 14th Python in Science Conference*, 51–58. <https://doi.org/10.25080/MAJORA-7B98E3ED-009>
21. Ramírez-Gallego, S., Fernández, A., García, S., Chen, M., & Herrera, F. (2018). Big data: Tutorial and guidelines on information and process fusion for analytics algorithms with MapReduce. *Information Fusion*, 42, 51–61. <https://doi.org/10.1016/j.inffus.2017.10.001>
22. Laufer-Goldshtein, B., Fisch, A., Barzilay, R., & Jaakkola, T. (2022). Efficiently controlling multiple risks with Pareto testing. *arXiv, abs/2210.07913*. <https://doi.org/10.48550/arXiv.2210.07913>
23. Kyrychenko, I., Tereshchenko, G., & Smelyakov, K. (2024). Optimized indexing method in hybrid image storage model for efficient storage in big data environments. *2024 IEEE 17th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*, Lviv, Ukraine, 1–4. <https://doi.org/10.1109/TCSET64720.2024.10755763>

References

1. Kuhn, A. (2020, March 10). How a South Korean city is changing tactics to tamp down its COVID-19 surge. NPR. <https://www.npr.org/sections/goatsandsoda/2020/03/10/812865169/how-a-southkorean-city-is-changing-tactics-to-tamp-down-its-Covid-19-surge>
2. Centers for Disease Control and Prevention. (2019, April 9). Strategies for optimizing the supply of N95 respirators. CDC. <https://www.cdc.gov/coronavirus/2019-ncov/hcp/respirators-strategy/index.html>
3. Parker, F., Sawczuk, H., Ganjkanloo, F., Ahmadi, F., & Ghobadi, K. (2020). Optimal resource and demand redistribution for healthcare systems under stress from COVID-19. *arXiv*.
4. Zheng, J., Bao, F., Shen, Z., & Xu, C. (2022). Data-driven dynamic adjustment and optimization model of emergency logistics network in public health. *Risk Management and Healthcare Policy*, 15, 151–169. <https://doi.org/10.2147/RMHP.S350275>
5. Chen, A., Hansoti, B., & Hsu, E. (2022). The COVID-19 pandemic response and its impact on post-pandemic health emergency and disaster risk management in the United States. *Sustainability*. <https://doi.org/10.3390/su142316301>
6. Jana, S., Majumder, R., & Ghose, D. (2023). Critical medical resource allocation during COVID-19 pandemic. *5th World Congress on Disaster Management: Volume III*, 275–283. <https://doi.org/10.4324/9781003341963-37>
7. Artificial intelligence technologies in modern medicine: Implementation and issues. (2024). *Ukrainian Medical Journal*, 163(5). <https://doi.org/10.32471/umj.1680-3051.163.257497>
8. Tereshchenko, G., Kyrychenko, I., Smelyakov, K., & Oliynyk, A. (2024). Analysis of image compression methods for storage in decentralized blockchain repositories. *Bionics of Intelligence*, 1(100), 23–35.
9. Mitchell, S., O'Sullivan, M., & Dunning, I. (2011). *PuLP: A linear programming toolkit for Python*.
10. Camisa, A., Notarnicola, I., & Notarstefano, G. (2020). Distributed primal decomposition for large-scale MILPs. *IEEE Transactions on Automatic Control*, 67, 413–420. <https://doi.org/10.1109/TAC.2021.3057061>
11. Shabir, S. (2016). A comparative study of genetic algorithm and the particle swarm optimization.
12. Kim, J., & Yoo, S. (2018). Software review: DEAP (Distributed Evolutionary Algorithm in Python) library. *Genetic Programming and Evolvable Machines*, 20, 139–142. <https://doi.org/10.1007/s10710-018-9341-4>

13. Abd-El-Wahed, W., Mousa, A., & El-Shorbagy, M. (2011). Integrating particle swarm optimization with genetic algorithms for solving nonlinear optimization problems. *Journal of Computational and Applied Mathematics*, 235, 1446–1453. <https://doi.org/10.1016/j.cam.2010.08.030>
14. Miranda, L. (2018). PySwarms: A research toolkit for particle swarm optimization in Python. *Journal of Open Source Software*, 3, 433. <https://doi.org/10.21105/JOSS.00433>
15. Ghaffari, Z., Nasiri, M., Bozorgi-Amiri, A., & Rahbari, A. (2020). Emergency supply chain scheduling problem with multiple resources in disaster relief operations. *Transportmetrica A: Transport Science*, 16, 930–956. <https://doi.org/10.1080/23249935.2020.1720858>
16. Homem-De-Mello, T., & Bayraksan, G. (2014). Monte Carlo sampling-based methods for stochastic optimization. *Surveys in Operations Research and Management Science*, 19, 56–85. <https://doi.org/10.1016/J.SORMS.2014.05.001>
17. Shmelova, T. (2018). Stochastic methods for estimation and problem solving in engineering: Stochastic methods of decision making in aviation. IGI Global, 139–160. <https://doi.org/10.4018/978-1-5225-5045-7.CH006>
18. Watson, J., Woodruff, D., & Hart, W. (2012). PySP: Modeling and solving stochastic programs in Python. *Mathematical Programming Computation*, 4, 109–149. <https://doi.org/10.1007/s12532-012-0036-1>
19. García-Magariño, I., & Gutiérrez, C. (2013). Agent-oriented modeling and development of a system for crisis management. *Expert Systems with Applications*, 40, 6580–6592. <https://doi.org/10.1016/j.eswa.2013.06.012>
20. Masad, D., & Kazil, J. (2015). Mesa: An agent-based modeling framework. *Proceedings of the 14th Python in Science Conference*, 51–58. <https://doi.org/10.25080/MAJORA-7B98E3ED-009>
21. Ramírez-Gallego, S., Fernández, A., García, S., Chen, M., & Herrera, F. (2018). Big data: Tutorial and guidelines on information and process fusion for analytics algorithms with MapReduce. *Information Fusion*, 42, 51–61. <https://doi.org/10.1016/j.inffus.2017.10.001>
22. Laufer-Goldshtein, B., Fisch, A., Barzilay, R., & Jaakkola, T. (2022). Efficiently controlling multiple risks with Pareto testing. *arXiv, abs/2210.07913*. <https://doi.org/10.48550/arXiv.2210.07913>
23. Kyrychenko, I., Tereshchenko, G., & Smelyakov, K. (2024). Optimized indexing method in hybrid image storage model for efficient storage in big data environments. *2024 IEEE 17th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv, Ukraine*, 1–4. <https://doi.org/10.1109/TCSET64720.2024.10755763>