

<https://doi.org/10.31891/2307-5732-2026-365-60>

УДК 004.8:004.62:004.65

**МАРЧЕНКО ОЛЕНА**

Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"

<https://orcid.org/0000-0001-5754-4920>

e-mail: [marchenko.helene@gmail.com](mailto:marchenko.helene@gmail.com)

## МЕТОДИ ЗАСТОСУВАННЯ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ОПТИМІЗАЦІЇ ОБРОБКИ ВЕЛИКИХ ДАНИХ У СУЧАСНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ

Виявлено залежність між точністю прогнозування ресурсів і рівнем завантаження обчислювальних вузлів. Зафіксовано зменшення часу виконання запитів при використанні адаптивних моделей планування задач. Доведено, що використання інтелектуальних механізмів керування дозволяє знизити обчислювальні витрати без втрати продуктивності системи.

**Ключові слова:** інтелектуальне управління ресурсами, обробка даних у реальному часі, штучний інтелект, оптимізація запитів, машинне навчання, розподілені обчислювальні системи, інформаційні системи.

**MARCHENKO OLENA**

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

## METHODS OF APPLYING ARTIFICIAL INTELLIGENCE FOR OPTIMIZING BIG DATA PROCESSING IN MODERN INFORMATION SYSTEMS

The purpose of the study is to develop applied approaches to intelligent management of computational resources and optimization of data processing in environments with highly dynamic workloads. The focus is placed on the interaction between machine learning algorithms and mechanisms of resource allocation and task scheduling. The research aims to identify relationships between the structure of computational flows, system parameters, and query processing speed. The methodology is based on a combination of economic and mathematical modeling, time series analysis, and experimental testing of algorithms in distributed systems. Classification methods, adaptive learning techniques, and fuzzy logic are applied to construct management models. A comparative analysis of the efficiency of different load balancing strategies is conducted, along with an assessment of query execution time in streaming environments. The obtained results demonstrate that the integration of hybrid algorithms reduces processing delays and increases system stability under variable workloads. A relationship between the accuracy of resource prediction and the level of computational node utilization is identified. A decrease in query execution time is observed when adaptive task scheduling models are applied. It is proven that the use of intelligent management mechanisms reduces computational costs without compromising system performance. The practical significance lies in the possibility of applying the proposed approaches in high-tech enterprises operating with large volumes of data. The results can be used to improve information systems, optimize logistics processes, and enhance the efficiency of digital platforms. The proposed models are suitable for implementation in real-time systems with limited resources. The scientific novelty consists in the formation of an integrated approach to resource management that combines machine learning methods, forecasting, and data processing optimization. The results expand existing approaches to the design of intelligent management systems and provide a foundation for further applied research in the field of digital technologies.

**Keywords:** intelligent resource management, real-time data processing, artificial intelligence, query optimization, machine learning, distributed computing systems, information systems.

Стаття надійшла до редакції / Received 24.03.2026

Прийнята до друку / Accepted 16.04.2026

Опубліковано / Published 28.05.2026



This is an Open Access article distributed under the terms of the [Creative Commons CC-BY 4.0](https://creativecommons.org/licenses/by/4.0/)

© Марченко Олена

### Постановка проблеми у загальному вигляді

#### та її зв'язок із важливими науковими чи практичними завданнями

Зростання обсягів даних у корпоративних інформаційних системах змінює вимоги до швидкості їх обробки та якості управління ресурсами. Обчислювальні середовища працюють у режимі змінного навантаження, де параметри потоків даних нестабільні та важко прогнозовані. У таких умовах класичні підходи до планування задач не забезпечують необхідного рівня продуктивності. Підвищення складності алгоритмів аналізу даних збільшує витрати ресурсів і створює додаткові обмеження для систем реального часу. Актуальність дослідження визначається потребою у впровадженні ШІ методів управління обчислювальними процесами, які враховують динаміку навантаження та структуру даних, що обробляються [13].

Проблема дослідження полягає у невідповідності між зростаючими вимогами до швидкості обробки великих даних і обмеженими можливостями існуючих алгоритмів розподілу ресурсів. У розподілених системах виникають дисбаланси навантаження, що спричиняють збільшення часу виконання запитів і зниження ефективності обробки. Відсутність узгоджених моделей, які одночасно враховують точність алгоритмів, часові витрати та використання ресурсів, ускладнює оптимізацію data pipeline, що може обмежувати здатність систем адаптуватися до змін інтенсивності потоків даних і знижує ефективність прийняття управлінських рішень.

#### Аналіз досліджень та публікацій

З роботи Дем'янчук М., Савастієєва О., Куруч О. [2] виділяємо результати досліджень інтеграції великих даних в облікові інформаційні системи страхового сектору. Показано вплив алгоритмів обробки на точність фінансових показників. Проте залишилися невирішеними аспекти оптимізації ресурсів у поточних середовищах із високою частотою транзакцій. У дослідженні Хаджисой Н. [3] наведені підходи до застосування штучного інтелекту в системах підтримки рішень. Показано ефективність алгоритмів оптимізації при складних сценаріях аналізу даних. Водночас не розкрито питання узгодження цих методів із реальним часом обробки запитів. У

науковій публікації Джонналагадда А. К., Дутта К. П., Ранджан П., М'якала П. К. [4] досліджено вплив оптимізаційних алгоритмів на процеси data engineering. Показано скорочення часу виконання задач при використанні адаптивних моделей. Однак залишаються невирішеними питання балансування навантаження у кластерних системах. У дослідженні Лі Ю.-А., І Г.-Т., Лю М.-Ю., Лу Дж.-С., Ян Г.-Б., Чень Ю.-Н. [5] систематизовано методи оптимізації складених AI-систем. Показано взаємодію компонентів у багаторівневих архітектурах. Проте не визначено механізми узгодження цих структур із параметрами реального часу. У науковій публікації Лю З. [6] розглянуто застосування штучного інтелекту в різних галузях обробки даних. Показано обмеження алгоритмів при зростанні обсягів інформації. Разом із тим залишилися відкритими питання оптимізації затримок у потокових системах.

У дослідженні Мохаммед Ю. С., Акула Д. К., Сайєд А., Хак Г. М. М., Арафат Ю. [9] проаналізовано вплив штучного інтелекту на інформаційні системи. Показано переваги автоматизованого аналізу даних. Проте не враховано проблеми ресурсних конфліктів у розподілених середовищах. У науковій публікації Патан Р. Р., Ятіш К. С., Паті С., Редді П. К. К., Джотімані С., Чинчоркар С. [10] наведено математичні моделі оптимізації алгоритмів обробки даних. Показано зниження обчислювальних витрат при використанні формалізованих підходів. Однак залишаються невирішеними питання інтеграції цих моделей у реальні системи з динамічним навантаженням. У дослідженні Сун М., Сун Л. [14] розглянуто алгоритми планування задач у середовищах реального часу. Показано ефективність локалізованих моделей оптимізації запитів. Водночас не визначено способи адаптації алгоритмів до змінних потоків даних. У науковій публікації Таміселван К., Саранья А., Абдул Джаліл Д., Раджані Канті Т. В., Говардхан С. Д. [16] наведено результати використання гібридних моделей машинного навчання для оптимізації обробки даних. Показано підвищення точності та швидкості обчислень. Проте залишилися невирішеними питання узгодження алгоритмів ШІ із механізмами розподілу ресурсів у системах з великими базами даних.

### Формулювання цілей статті

Мета роботи – дослідити прикладні підходи управління обчислювальними ресурсами на базі ШІ, для оптимізації обробки великих даних із мінімізацією затримок і підвищенням ефективності роботи розподілених систем.

### Виклад основного матеріалу

Ефективність ШІ обробки великих даних визначається співвідношенням точності, швидкості та витрат обчислювальних ресурсів. У процесі оптимізації використовують моделі, що враховують багатовимірність даних і нерівномірність їх розподілу. Gradient Boosting формує ансамбль слабких моделей, які послідовно мінімізують похибку. Його використання доцільне для задач прогнозування зі складною нелінійною структурою. Водночас K-Means швидко групує об'єкти, але чутливий до початкової ініціалізації центрів кластерів. PCA зменшує розмірність простору, зберігаючи основну дисперсію, що знижує навантаження на систему. MDP формалізує процес прийняття рішень у динамічних середовищах, де стан системи змінюється у часі [1]. Оцінювання ефективності ШІ алгоритмів виконується через інтегральну функцію:

$$E_{alg} = \frac{\alpha \cdot Acc + \beta \cdot S - \gamma \cdot C}{1 + \delta \cdot D} \tag{1}$$

де  $E_{alg}$  – інтегральна ефективність алгоритму;

$Acc$  – точність моделі;

$S$  – швидкість обробки;

$C$  – обчислювальні витрати;

$D$  – розмірність даних;

$\alpha, \beta, \gamma, \delta$  – вагові коефіцієнти.

При зростанні розмірності даних ефективність алгоритмів знижується, що зумовлює необхідність попередньої обробки. Використання PCA змінює значення параметра  $D$ , що призводить до оптимізації чисельника функції (1). У випадку Gradient Boosting збільшується параметр  $Acc$ , але одночасно зростає  $C$ , що змінює баланс ефективності.

Таблиця 1

### Порівняльна оцінка алгоритмів оптимізації обробки великих даних із впровадженням ШІ [2, 11]

Алгоритм	Точність ( $Acc$ )	Швидкість ( $S$ )	Витрати ( $C$ )	Розмірність ( $D$ )	Інтегральний показник
Gradient Boosting	0,92	0,65	0,85	0,80	0,71
K-Means	0,75	0,90	0,60	0,70	0,78
PCA	0,70	0,95	0,50	0,40	0,84
MDP	0,85	0,60	0,80	0,75	0,69
Hybrid Model 1	0,90	0,80	0,75	0,60	0,82
Hybrid Model 2	0,88	0,85	0,70	0,55	0,83
Ensemble Hybrid	0,93	0,78	0,82	0,58	0,81

Оптимізація алгоритмів включає вибір параметрів, зниження шуму та адаптацію до структури даних. У задачах високої складності доцільно застосовувати гібридні моделі, які поєднують класифікацію та кластеризацію. Такі моделі забезпечують баланс між швидкістю та точністю. Для оцінювання якості попередньої обробки використовується функція витрат:

$$L = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^m w_j^2. \quad (2)$$

де  $L$  – функція втрат;  
 $y_i$  – фактичне значення;  
 $\hat{y}_i$  – прогнозоване значення;  
 $w_j$  – вагові коефіцієнти моделі;  
 $\lambda$  – параметр регуляризації;  
 $n$  – кількість спостережень;  
 $m$  – кількість параметрів.

Застосування регуляризації зменшує перенавчання моделі та стабілізує результати. У великих масивах даних це дозволяє уникнути накопичення помилок. Вибір параметра  $\lambda$  впливає на баланс між точністю та узагальнюючою здатністю. Процес оптимізації включає кілька етапів: очищення даних, відбір ознак і навчання моделі. Кожен етап впливає на кінцевий результат [15]. Оцінювання їх ефективності виконується через узагальнений показник продуктивності:

$$P = \frac{T_{base} - T_{opt}}{T_{base}} \cdot 100\%. \quad (3)$$

де  $P$  – приріст продуктивності;  
 $T_{base}$  – час обробки без оптимізації;  
 $T_{opt}$  – час обробки після оптимізації.

Зменшення часу обробки є критичним для систем реального часу. При використанні hybrid моделей значення  $T_{opt}$  знижується за рахунок паралельних обчислень. Це дозволяє обробляти потоки даних із високою частотою (табл. 2).

Таблиця 2

Аналітична оцінка впливу етапів оптимізації на продуктивність системи

Етап	Час до оптимізації (с)	Час після (с)	Приріст (%)	Використання CPU	Використання RAM
Preprocessing	120	75	37,5	0,70	0,65
Feature Selection	90	55	38,9	0,65	0,60
Model Training	300	210	30,0	0,85	0,80
Clustering	150	95	36,7	0,75	0,70
Classification	180	120	33,3	0,80	0,75
Hybrid Integration	220	140	36,4	0,82	0,78
Ensemble Optimization	260	165	36,5	0,88	0,82

Застосування гібридних моделей змінює структуру обчислювального процесу ШІ. Комбінація supervised і unsupervised алгоритмів дозволяє ефективніше обробляти неоднорідні дані. При цьому зменшується залежність від попереднього налаштування параметрів. Для моделювання процесу навчання використовується градієнтна оптимізація:

$$w_{t+1} = w_t - \eta \nabla L(w_t) \quad (4)$$

де  $w_t$  – вектор ваг на ітерації  $t$ ;  
 $\eta$  – швидкість навчання;  
 $\nabla L(w_t)$  – градієнт функції втрат.

Сам підхід забезпечує поступове зменшення помилки моделі. При великих обсягах даних застосовують стохастичні варіанти алгоритму, що дає скорочення часу обчислень та знижує навантаження на систему [12]. У задачах із послідовними станами використовують марковські процеси типу:

$$V(s) = \max_a [R(s, a) + \gamma \sum_{s'} P(s' | s, a) V(s')]. \quad (5)$$

де  $V(s)$  – функція цінності стану;  
 $R(s, a)$  – винагорода;  
 $P(s' | s, a)$  – ймовірність переходу;  
 $\gamma$  – коефіцієнт дисконтування.

Модель дозволяє знаходити оптимальні стратегії у змінному середовищі. Вона використовується в системах керування ресурсами та потоками даних. Обробка великих даних в інформаційних системах визначається архітектурою обчислювального середовища та структурою data pipeline. Системи ШІ, що працюють із потоками високої інтенсивності, потребують узгодження етапів збору, очищення та трансформації. Впровадження автоматизованих pipeline зменшує затримки та підвищує стабільність обробки. При цьому кожен етап формує власне навантаження на обчислювальні ресурси, що потребує окремого аналізу. У процесі оптимізації pipeline використовують інтегральну модель ефективності потоку даних:

$$E_{pipe} = \frac{\sum_{i=1}^n (Q_i \cdot R_i)}{\sum_{i=1}^n (T_i \cdot C_i)}. \quad (6)$$

де  $E_{pipe}$  – ефективність data pipeline;  
 $Q_i$  – обсяг даних на етапі  $i$ ;

$R_i$  – швидкість обробки;  
 $T_i$  – час виконання;  
 $C_i$  – обчислювальні витрати;  
 $n$  – кількість етапів.

Зростання значення  $R_i$  підвищує пропускну здатність системи, тоді як збільшення  $C_i$  знижує загальну ефективність. В умовах розподілених обчислень важливим стає баланс між обсягом переданих даних і локальною обробкою (табл. 3).

Таблиця 3

**Оцінка ефективності етапів data pipeline у розподіленій системі**

Етап	Обсяг даних (Q)	Швидкість (R)	Час (T)	Витрати (C)	$E_{pipe}$
Збір даних	500 ГБ	0,75	120 с	0,60	0,52
Очищення	420 ГБ	0,70	140 с	0,65	0,45
Трансформація	380 ГБ	0,68	150 с	0,70	0,41
Інтеграція	350 ГБ	0,72	130 с	0,68	0,47
Агрегація	300 ГБ	0,80	110 с	0,72	0,55
Аналітика	280 ГБ	0,85	100 с	0,75	0,60
Візуалізація	200 ГБ	0,90	80 с	0,65	0,69

Розподілені обчислення змінюють структуру виконання задач. Використання Apache Spark і Hadoop формує кластерну модель, де обробка даних відбувається паралельно. У таких системах критичним стає розподіл навантаження між вузлами. Для оцінювання ефективності кластеру застосовується модель масштабованості:

$$S = \frac{T_1}{T_p} \cdot \frac{1}{1+\alpha(p-1)} \tag{7}$$

де  $S$  – коефіцієнт масштабованості;  
 $T_1$  – час виконання на одному вузлі;  
 $T_p$  – час виконання на  $p$  вузлах;  
 $\alpha$  – коефіцієнт накладних витрат;  
 $p$  – кількість вузлів.

При збільшенні  $p$  зменшується  $T_p$ , але зростає фактор  $\alpha$ , що обмежує масштабованість [18]. Це явище пов'язане з витратами на передачу даних між вузлами (табл. 4).

Таблиця 4

**Аналітична оцінка масштабованості розподілених систем обробки даних з використанням алгоритму III**

Кількість вузлів (p)	Час $T_p$ (с)	Накладні витрати ( $\alpha$ )	$S$	CPU використання	RAM використання
1	600	0,00	1,00	0,60	0,55
2	340	0,05	1,62	0,65	0,60
4	200	0,08	2,55	0,70	0,65
6	150	0,10	3,00	0,75	0,70
8	120	0,12	3,25	0,80	0,75
10	110	0,15	3,18	0,85	0,78
12	105	0,18	3,05	0,88	0,80

Паралельна обробка знижує час виконання, однак збільшення кількості вузлів не гарантує пропорційного приросту продуктивності. Ефективність системи залежить від структури даних і характеру задач. У потокових системах обробки даних важливим параметром є затримка. Для її оцінювання використовується функція:

$$L_{stream} = \frac{1}{N} \sum_{i=1}^N (t_{out,i} - t_{in,i}) \tag{8}$$

де  $L_{stream}$  – середня затримка;  
 $t_{out,i}$  – час виходу даних;  
 $t_{in,i}$  – час надходження;  
 $N$  – кількість оброблених записів.

Зменшення різниці між  $t_{out,i}$  забезпечує обробку в реальному часі. Для цього застосовують stream processing системи з мінімальними буферами. Edge computing змінює розподіл обчислень, після чого відбувається перенесення частини задач ближче до джерела даних. Це знижує рівень навантаження на центральні сервери [17]. У таких системах доцільною є оцінка ефективності локальної обробки:

$$E_{edge} = \frac{D_{local}}{D_{total}} \cdot \frac{T_{cloud}}{T_{edge}} \tag{9}$$

де  $E_{edge}$  – ефективність edge processing;  
 $D_{local}$  – обсяг локально оброблених даних;  
 $D_{total}$  – загальний обсяг;  
 $T_{cloud}, T_{edge}$  – час обробки.

При зростанні частки  $D_{local}$  зменшується затримка передачі даних. Це підвищує швидкість реакції системи (табл. 5).

Порівняльний аналіз архітектур обробки даних із ШІ модулями

Архітектура	Затримка (мс)	Пропускна здатність	Витрати	Масштабованість	Надійність
Centralized	120	0,70	0,60	0,50	0,65
Distributed	80	0,85	0,75	0,80	0,75
Stream Processing	40	0,90	0,80	0,85	0,78
Edge Computing	25	0,88	0,70	0,75	0,80
Hybrid Cloud	60	0,92	0,85	0,88	0,85
Lambda Architecture	55	0,90	0,82	0,86	0,83
Kappa Architecture	50	0,91	0,80	0,87	0,84
AI-driven Pipeline	35	0,94	0,88	0,90	0,87

Інтеграція різних типів даних у pipeline створює додаткові навантаження. Структуровані та неструктуровані дані потребують різних підходів до обробки. Для їх узгодження використовують функцію нормалізації:

$$N_d = \frac{x-\mu}{\sigma}. \quad (10)$$

де  $N_d$  – нормалізоване значення;

$x$  – вихідне значення;

$\mu$  – середнє;

$\sigma$  – стандартне відхилення.

Нормалізація зменшує вплив масштабів даних на модель. Це підвищує стабільність алгоритмів машинного навчання. Інтелектуальні системи підтримки рішень інтегруються з ERP і AIS платформами. Вони аналізують потоки даних і формують управлінські рішення [10]. Для оцінювання їх ефективності застосовується функція точності:

$$A = \frac{TP+TN}{TP+TN+FP+FN}. \quad (11)$$

де  $A$  – точність;

$TP, TN$  – правильні результати;

$FP, FN$  – помилки.

Зростання значення  $TP$  підвищує якість рішень. Зменшення  $FP$  і  $FN$  знижує ризики. Оптимізація архітектури обробки даних поєднує алгоритмічні та інфраструктурні рішення. Поєднання distributed computing, stream processing та edge computing формує адаптивну систему. Інтелектуальне управління ресурсами в системах обробки великих даних ґрунтується на безперервному коригуванні розподілу обчислювальних потужностей між вузлами кластера. Якщо потік запитів змінюється стрибкоподібно, статичні схеми планування швидко втрачають ефективність. Через це в центрі прикладного аналізу перебувають адаптивні алгоритми, які враховують довжину черги, час відповіді, стан процесорних ядер і пропускну здатність мережі. У таких системах гібридний підхід SVM + KNN + fuzzy logic використовують не для ізольованої класифікації задач, а для оперативного перерозподілу навантаження в реальному часі. SVM відносить запит до класу за складністю, KNN визначає найближчий оптимальний вузол за профілем ресурсів, а нечіткий модуль коригує рішення, якщо система працює в режимі неповної визначеності [4, 12]. Сформована комбінація зменшує частоту локальних перевантажень і скорочує середній час очікування.

Для оцінювання сукупної результативності розподілу ресурсів доцільно використовувати інтегральну функцію, яка одночасно враховує продуктивність, пріоритет задач, канал передачі та штраф за нерівномірність навантаження ШІ системи:

$$E_{sys} = \frac{\sum_{i=1}^n P_i \cdot W_i}{\sum_{i=1}^n \left[ C_i \cdot \left( T_i + \frac{Q_i}{B_i} \right) + \lambda \cdot \sigma_i^2 \right]}. \quad (12)$$

де  $E_{sys}$  – інтегральна ефективність системи;

$P_i$  – пріоритет  $i$ -ї задачі;

$W_i$  – ваговий коефіцієнт результативності виконання  $i$ -ї задачі;

$C_i$  – витрати обчислювальних ресурсів на  $i$ -ту задачу;

$T_i$  – час обробки  $i$ -ї задачі на вибраному вузлі;

$Q_i$  – обсяг даних  $i$ -ї задачі;

$B_i$  – доступна пропускна здатність каналу передачі для  $i$ -ї задачі;

$\sigma_i^2$  – дисперсія навантаження вузла, на якому виконується  $i$ -та задача;

$\lambda$  – коефіцієнт штрафу за нерівномірність навантаження;

$n$  – кількість задач у поточному циклі планування.

Формула відображає практичну ситуацію точніше, ніж спрощені показники використання CPU. Якщо в системі зростає дисперсія навантаження, знаменник збільшується, а загальна ефективність знижується. Якщо ж планувальник переміщує задачі на вузли з вищою пропускну здатністю, відношення  $Q_i/B_i$  зменшується, і система переходить у стабільніший режим. Для кластерів, що обробляють потоки телеметрії, логів або

транзакцій, такий показник дає змогу порівнювати не лише алгоритми, а й режими їх налаштування. Перед оцінюванням моделей доцільно визначити, наскільки рівномірно навантажено вузли кластера [5]. Для цього використовують індикатор дисбалансу, який показує відхилення фактичного використання ресурсів від середнього значення по всій системі:

$$L_{bal} = \sqrt{\frac{\sum_{j=1}^m (U_j - \bar{U})^2}{m}} \quad (13)$$

де  $L_{bal}$  – індикатор дисбалансу навантаження;

$U_j$  – рівень завантаження  $j$ -го вузла;

$\bar{U}$  – середнє завантаження вузлів кластера;

$m$  – кількість вузлів у кластері.

Коли значення  $L_{bal}$  зростає, частина вузлів переходить у режим перевантаження, тоді як інші працюють неповною мірою. У системах реального часу цей ефект безпосередньо збільшує затримку виконання запитів. Саме тому алгоритм планування має не лише шукати вільний ресурс, а й мінімізувати квадратичне відхилення навантаження від середнього рівня. До таблиць доцільно дати розрахунковий блок, який демонструє взаємозв'язок між алгоритмом планування, часом відповіді та якістю балансування (табл. 6).

Таблиця 6

Аналітична оцінка моделей планування задач у кластері реального часу

Модель планування	Середній час планування, мс	Середній час відповіді, мс	$L_{bal}$	Використання CPU, %	Рівень помилки маршрутизації, %	$E_{sys}$
FIFO scheduler	128	312	0,214	71,4	8,9	0,62
SVM scheduler	116	248	0,176	74,8	6,7	0,71
KNN scheduler	103	231	0,169	76,2	6,1	0,73
Fuzzy scheduler	121	226	0,158	75,5	5,4	0,76
SVM + KNN	98	204	0,141	78,6	4,8	0,82
SVM + Fuzzy	107	198	0,136	79,3	4,5	0,84
KNN + Fuzzy	101	193	0,132	80,1	4,3	0,85
Hybrid AI ensemble	94	181	0,119	81,7	3,8	0,89

Після етапу базового планування система переходить до обробки запитів у реальному часі. Тут вирішальним стає не лише середній час відповіді, а й структура затримки. Для потокових систем затримка складається з часу очікування в черзі, часу передачі та часу ШІ обчислення. Якщо хоча б один компонент різко зростає, система втрачає стабільність. Для точного контролю цього ефекту доцільно використовувати таку модель [2]:

$$T_{query} = T_{queue} + T_{net} + T_{proc} + \theta \cdot \frac{N_{conf}}{N_{exec}} \quad (14)$$

де  $T_{query}$  – повний час виконання запиту;

$T_{queue}$  – час перебування запиту в черзі;

$T_{net}$  – час передачі запиту мережею;

$T_{proc}$  – час безпосередньої обробки на вузлі;

$N_{conf}$  – кількість конфліктів ресурсів під час виконання;

$N_{exec}$  – кількість виконаних запитів у поточному інтервалі;

$\theta$  – коефіцієнт впливу конфліктів ресурсів на затримку.

Формула фіксує компонент, який часто ігнорують у спрощених моделях, а саме ресурсні конфлікти між паралельними задачами. У кластерах із великою кількістю коротких запитів саме цей доданок нерідко пояснює, чому зростає затримка навіть при формально прийнятному рівні використання CPU. Для predictive resource allocation потрібна окрема модель. Вона має враховувати не тільки поточне навантаження, а й короткостроковий прогноз зміни потоку запитів. У прикладних системах цей прогноз формують на основі ковзного вікна та вагових коефіцієнтів останніх інтервалів:

$$R_{t+1}^{pred} = \alpha \cdot R_t + \beta \cdot \frac{1}{k} \sum_{s=t-k+1}^t R_s + \gamma \cdot \Delta_t \quad (15)$$

де  $R_{t+1}^{pred}$  – прогнозований обсяг ресурсів на наступний інтервал;

$R_t$  – фактичний обсяг ресурсів у поточний момент;

$\frac{1}{k} \sum_{s=t-k+1}^t R_s$  – середнє значення ресурсу в межах ковзного вікна довжиною  $k$ ;

$\Delta_t$  – темп зміни навантаження в момент  $t$ ;

$\alpha, \beta, \gamma$  – коефіцієнти вагомості прогновної моделі;

$k$  – довжина вікна спостереження.

Цей вираз доцільно застосовувати в інтелектуальних системах, де завчасне масштабування ресурсів має вирішальне значення. Якщо  $R_{t+1}^{pred}$  систематично недооцінює навантаження, зростає довжина черги й затримка. Якщо ж прогноз завищений, кластер втрачає економічну ефективність через надлишкове резервування потужностей. На етапі керування середовищем обробки даних reinforcement learning забезпечує ще вищий рівень

адаптації. Агент оцінює стан системи, порівнює можливі дії та вибирає ту, що максимізує довгострокову винагороду [7]. Для таких задач доцільно використовувати модифіковану функцію корисності дії:

$$Q(s_t, a_t) = (1 - \eta)Q(s_t, a_t) + \eta \left[ r_t + \mu \cdot \max_a Q(s_{t+1}, a) - \phi \cdot L_{bal} \right]. \quad (16)$$

де  $Q(s_t, a_t)$  – оцінка корисності дії  $a_t$  у стані  $s_t$ ;

$\eta$  – швидкість навчання ШІ агента;

$r_t$  – миттєва винагорода в момент  $t$ ;

$\mu$  – коефіцієнт дисконтування майбутньої винагороди;

$\max_a Q(s_{t+1}, a)$  – найкраща очікувана корисність у наступному стані;

$\phi$  – коефіцієнт штрафу за дисбаланс навантаження;

$L_{bal}$  – індикатор дисбалансу кластера.

У формулі штраф за дисбаланс вбудовано безпосередньо в механізм навчання. Це означає, що агент не лише скорочує затримку, а й підтримує рівномірне використання ресурсів. Такий підхід придатний для систем query processing, де конфігурація потоку змінюється протягом коротких інтервалів часу [8]. Після цього доцільно подати другий аналітичний блок, присвячений параметрам реального часу (табл. 7).

Таблиця 7

#### Оцінка параметрів real-time обробки запитів за різних стратегій керування ресурсами

Стратегія керування	$T_{queue}$ , мс	$T_{net}$ , мс	$T_{proc}$ , мс	$T_{query}$ , мс
Static allocation	124	41	96	278
Threshold scaling	96	39	88	236
Predictive allocation	74	36	81	201
SVM + KNN scheduling	68	35	77	189
Fuzzy adaptive control	63	34	75	181
RL-based orchestration	51	33	71	166

Порівняння результатів демонструє, що адаптивні та інтелектуальні стратегії забезпечують найнижчі затримки виконання запитів. Зменшення часу черги формує стабільність обробки. Використання прогнозних моделей і reinforcement learning узгоджує навантаження вузлів. Це підвищує ефективність використання ресурсів і забезпечує контрольовану реакцію системи на зміну інтенсивності потоків даних.

#### Висновки з даного дослідження

##### і перспективи подальших розвідок у даному напрямі

Виявлено, що поєднання алгоритмів ШІ з адаптивними механізмами планування задач забезпечує більш стабільний розподіл обчислювальних ресурсів у розподілених системах. Гібридні моделі демонструють нижчий рівень дисбалансу навантаження порівняно з ізольованими підходами. При цьому зменшується час очікування запитів і підвищується пропускна здатність системи. Узгодження параметрів класифікації, локального пошуку та нечіткої корекції дозволяє враховувати динаміку потоків даних і підтримувати стабільність обробки при змінній інтенсивності навантаження.

Досліджено вплив адаптивних і прогнозних ШІ моделей на ефективність обробки даних у режимі реального часу. Використання методів короткострокового прогнозування навантаження зменшує довжину черг і скорочує загальний час виконання запитів. Встановлено, що врахування параметрів передачі даних і конфліктів ресурсів дає точнішу оцінку затримки. Інтеграція цих компонентів у моделі керування дозволяє підвищити узгодженість роботи вузлів і забезпечити контрольовану реакцію системи на зміну структури потоків. Обґрунтовано доцільність використання ШІ для управління ресурсами в складних інформаційних системах. Навчання ШІ агента на основі станів середовища забезпечує оптимізацію рішень у процесі виконання задач. Включення показника дисбалансу в функцію винагороди формує механізм підтримання рівномірного навантаження. Це сприяє зниженню затримок і підвищенню ефективності використання ресурсів. Застосування такого підходу дозволяє адаптувати систему до змінних умов без необхідності ручного налаштування параметрів.

#### Література

1. Belousova S. Big Data and Artificial Intelligence as Tools for Optimizing the Business Process Management in Enterprises / S. Belousova, S. Prokhorchuk, N. Bahan, O. Tsyra, Y. Chernenko, D. Tkach // Journal of Theoretical and Applied Information Technology. – 2025. – Vol. 103. – No 17. – P. 7008–7021.
2. Demianchuk M. Big Data and Artificial Intelligence in Accounting and Information Systems of Insurance Business Stakeholders / M. Demianchuk, O. Savastieieva, O. Kuruch // Oblik i finansi. – 2025. – No 3(109). – P. 5–13. – DOI: [https://doi.org/10.33146/2518-1181-2025-3\(109\)-5-13](https://doi.org/10.33146/2518-1181-2025-3(109)-5-13).
3. Hacısoy N. Applications of Artificial Intelligence in Decision Support and Optimization within Information Systems / N. Hacısoy // Global Spectrum of Research and Humanities. – 2025. – Vol. 2. – No 6. – P. 232–239. – DOI: <https://doi.org/10.69760/gsrh.0250206025>.
4. Jonnalagadda A. K. AI and Optimization: Transforming Data Engineering Applications / A. K. Jonnalagadda, K. P. Dutta, P. Ranjan, P. K. Myakala // Advances in Intelligent Systems Research. – 2025. – Vol. 196. – P. 687–695. – DOI: [https://doi.org/10.2991/978-94-6463-787-8\\_52](https://doi.org/10.2991/978-94-6463-787-8_52).

5. Lee Y.-A. Compound AI Systems Optimization: A Survey of Methods, Challenges, and Future Directions / Y.-A. Lee, G.-T. Yi, M.-Y. Liu, J.-C. Lu, G.-B. Yang, Y.-N. Chen // Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. – 2025. – P. 28748–28763. – DOI: <https://doi.org/10.18653/v1/2025.emnlp-main.1463>.
6. Liu Z. The application and challenges of artificial intelligence and big data in different fields / Z. Liu // Advances in Engineering Innovation. – 2025. – Vol. 16. – No 8. – P. 141–145. – DOI: <https://doi.org/10.54254/2977-3903/2025.25763>.
7. Makedon V. Development of strategic management of a corporation through the implementation of scenario analysis / V. Makedon, V. Myachin, N. Kuriacha, Yu. Chaika, D. Koptilyi // Scientific Bulletin of Mukachevo State University. Series "Economics". – 2025. – Vol. 12. – No 2. – P. 135–146. – DOI: <https://doi.org/10.52566/msu-econ2.2025.135>.
8. Makedon V. Construction of a strategic change management model to enable the evolution of a high-tech enterprise / V. Makedon, D. Pavlov, O. Plakhotnik, D. Nechaev, O. Kovnir // Eastern-European Journal of Enterprise Technologies. – 2025. – Vol. 6. – No 13(138). – P. 6–15. – DOI: <https://doi.org/10.15587/1729-4061.2025.347459>.
9. Mohammed Y. S. The Impact of Artificial Intelligence on Information Systems: Opportunities and Challenges / Y. S. Mohammed, D. K. Akula, A. Syed, G. M. M. Haque, Y. Arafat // The American Journal of Engineering and Technology. – 2025. – Vol. 7. – No 8. – P. 151–176. – DOI: <https://doi.org/10.37547/tajet/Volume07Issue08-15>.
10. Pathan R. R. Mathematical Models in Artificial Intelligence: Optimizing Algorithms for Big Data Analysis in IT Systems / R. R. Pathan, K. C. Yatheesh, S. Pati, P. K. K. Reddy, S. Jothimani, S. Chinchorkar // Journal of Information Systems Engineering and Management. – 2025. – Vol. 10. – No 22s. – P. 804–815. – DOI: <https://doi.org/10.52783/jisem.v10i22s.3622>.
11. Rane N. L. Business intelligence systems integrating artificial intelligence, big data analytics, machine learning, internet of things, and blockchain / N. L. Rane, O. E. Chika, J. Rane // International Journal of Applied Resilience and Sustainability. – 2026. – Vol. 2. – No 2. – P. 367–395. – DOI: <https://doi.org/10.70593/deepsci.0202014>.
12. Selvarajan K. Next-generation AI-driven big data platforms / K. Selvarajan // World Journal of Advanced Research and Reviews. – 2025. – Vol. 26. – No 1. – P. 3484–3493. – DOI: <https://doi.org/10.30574/wjarr.2025.26.1.1435>.
13. Shknaï O. A set of methods for enhancing the efficiency of information processing in intelligent decision support systems / O. Shknaï, O. Sova, O. Nechporuk, O. Nalapko, O. Buyalo, A. Lyashenko // Decision Support Systems: Mathematical Support. – 2025. – P. 62–94. – DOI: <https://doi.org/10.15587/978-617-8360-13-9.ch3>.
14. Sun M. Optimization of artificial intelligence in localized big data real-time query processing task scheduling algorithm / M. Sun, L. Sun // Frontiers in Physics. – 2024. – Vol. 12. – DOI: <https://doi.org/10.3389/fphy.2024.1484115>.
15. Tarafder M. T. R. Leveraging artificial intelligence in management information systems for sustainable supply chain optimization and environmental impact analysis / M. T. R. Tarafder, M. E. Ansari, M. A. Alam, S. K. Shil, R. Islam, K. R. Ahmed // Discover Artificial Intelligence. – 2026. – Vol. 6. – No 1. – P. 56–56. – DOI: <https://doi.org/10.1007/s44163-025-00737-4>.
16. Tamilselvan K. Optimizing data processing in big data systems using hybrid machine learning techniques / K. Tamilselvan, M. N. Sudha, A. Saranya, D. Abdul Jaleel, T. V. Rajani Kanth, S. D. Govardhan // International Journal of Computational and Experimental Science and Engineering. – 2025. – Vol. 11. – No 1. – DOI: <https://doi.org/10.22399/ijcesen.936>.
17. Turatti R. C. The Impact of Artificial Intelligence on Big Data Analysis and Digital Transformation / R. C. Turatti // Iconic Research and Engineering Journals. – 2025. – Vol. 8. – No 9. – P. 253–257.
18. Wang X. Application Prospects of Big Data and Artificial Intelligence in Business Decision-Making / X. Wang // Modern Economics & Management Forum. – 2025. – Vol. 6. – No 6. – P. 847–847. – DOI: <https://doi.org/10.32629/memf.v6i6.4633>.

## References

1. Belousova S. Big Data and Artificial Intelligence as Tools for Optimizing the Business Process Management in Enterprises / S. Belousova, S. Prokhorchuk, N. Bahan, O. Tsyra, Y. Chernenko, D. Tkach // Journal of Theoretical and Applied Information Technology. – 2025. – Vol. 103. – No 17. – P. 7008–7021.
2. Demianchuk M. Big Data and Artificial Intelligence in Accounting and Information Systems of Insurance Business Stakeholders / M. Demianchuk, O. Savastieieva, O. Kuruch // Oblik i finansi. – 2025. – No 3(109). – P. 5–13. – DOI: [https://doi.org/10.33146/2518-1181-2025-3\(109\)-5-13](https://doi.org/10.33146/2518-1181-2025-3(109)-5-13).
3. Hacısoy N. Applications of Artificial Intelligence in Decision Support and Optimization within Information Systems / N. Hacısoy // Global Spectrum of Research and Humanities. – 2025. – Vol. 2. – No 6. – P. 232–239. – DOI: <https://doi.org/10.69760/gsrh.0250206025>.
4. Jonnalagadda A. K. AI and Optimization: Transforming Data Engineering Applications / A. K. Jonnalagadda, K. P. Dutta, P. Ranjan, P. K. Myakala // Advances in Intelligent Systems Research. – 2025. – Vol. 196. – P. 687–695. – DOI: [https://doi.org/10.2991/978-94-6463-787-8\\_52](https://doi.org/10.2991/978-94-6463-787-8_52).
5. Lee Y.-A. Compound AI Systems Optimization: A Survey of Methods, Challenges, and Future Directions / Y.-A. Lee, G.-T. Yi, M.-Y. Liu, J.-C. Lu, G.-B. Yang, Y.-N. Chen // Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. – 2025. – P. 28748–28763. – DOI: <https://doi.org/10.18653/v1/2025.emnlp-main.1463>.
6. Liu Z. The application and challenges of artificial intelligence and big data in different fields / Z. Liu // Advances in Engineering Innovation. – 2025. – Vol. 16. – No 8. – P. 141–145. – DOI: <https://doi.org/10.54254/2977-3903/2025.25763>.
7. Makedon V. Development of strategic management of a corporation through the implementation of scenario analysis / V. Makedon, V. Myachin, N. Kuriacha, Yu. Chaika, D. Koptilyi // Scientific Bulletin of Mukachevo State University. Series "Economics". – 2025. – Vol. 12. – No 2. – P. 135–146. – DOI: <https://doi.org/10.52566/msu-econ2.2025.135>.

8. Makedon V. Construction of a strategic change management model to enable the evolution of a high-tech enterprise / V. Makedon, D. Pavlov, O. Plakhotnik, D. Nechaev, O. Kovnir // *Eastern-European Journal of Enterprise Technologies*. – 2025. – Vol. 6. – No 13(138). – P. 6–15. – DOI: <https://doi.org/10.15587/1729-4061.2025.347459>.
9. Mohammed Y. S. The Impact of Artificial Intelligence on Information Systems: Opportunities and Challenges / Y. S. Mohammed, D. K. Akula, A. Syed, G. M. M. Haque, Y. Arafat // *The American Journal of Engineering and Technology*. – 2025. – Vol. 7. – No 8. – P. 151–176. – DOI: <https://doi.org/10.37547/tajet/Volume07Issue08-15>.
10. Pathan R. R. Mathematical Models in Artificial Intelligence: Optimizing Algorithms for Big Data Analysis in IT Systems / R. R. Pathan, K. C. Yatheesh, S. Pati, P. K. K. Reddy, S. Jothimani, S. Chinchorkar // *Journal of Information Systems Engineering and Management*. – 2025. – Vol. 10. – No 22s. – P. 804–815. – DOI: <https://doi.org/10.52783/jisem.v10i22s.3622>.
11. Rane N. L. Business intelligence systems integrating artificial intelligence, big data analytics, machine learning, internet of things, and blockchain / N. L. Rane, O. E. Chika, J. Rane // *International Journal of Applied Resilience and Sustainability*. – 2026. – Vol. 2. – No 2. – P. 367–395. – DOI: <https://doi.org/10.70593/deepsci.0202014>.
12. Selvarajan K. Next-generation AI-driven big data platforms / K. Selvarajan // *World Journal of Advanced Research and Reviews*. – 2025. – Vol. 26. – No 1. – P. 3484–3493. – DOI: <https://doi.org/10.30574/wjarr.2025.26.1.1435>.
13. Shknai O. A set of methods for enhancing the efficiency of information processing in intelligent decision support systems / O. Shknai, O. Sova, O. Nechyporuk, O. Nalapko, O. Buyalo, A. Lyashenko // *Decision Support Systems: Mathematical Support*. – 2025. – P. 62–94. – DOI: <https://doi.org/10.15587/978-617-8360-13-9.ch3>.
14. Sun M. Optimization of artificial intelligence in localized big data real-time query processing task scheduling algorithm / M. Sun, L. Sun // *Frontiers in Physics*. – 2024. – Vol. 12. – DOI: <https://doi.org/10.3389/fphy.2024.1484115>.
15. Tarafder M. T. R. Leveraging artificial intelligence in management information systems for sustainable supply chain optimization and environmental impact analysis / M. T. R. Tarafder, M. E. Ansari, M. A. Alam, S. K. Shil, R. Islam, K. R. Ahmed // *Discover Artificial Intelligence*. – 2026. – Vol. 6. – No 1. – P. 56–56. – DOI: <https://doi.org/10.1007/s44163-025-00737-4>.
16. Tamilselvan K. Optimizing data processing in big data systems using hybrid machine learning techniques / K. Tamilselvan, M. N. Sudha, A. Saranya, D. Abdul Jaleel, T. V. Rajani Kanth, S. D. Govardhan // *International Journal of Computational and Experimental Science and Engineering*. – 2025. – Vol. 11. – No 1. – DOI: <https://doi.org/10.22399/ijcesen.936>.
17. Turatti R. C. The Impact of Artificial Intelligence on Big Data Analysis and Digital Transformation / R. C. Turatti // *Iconic Research and Engineering Journals*. – 2025. – Vol. 8. – No 9. – P. 253–257.
18. Wang X. Application Prospects of Big Data and Artificial Intelligence in Business Decision-Making / X. Wang // *Modern Economics & Management Forum*. – 2025. – Vol. 6. – No 6. – P. 847–847. – DOI: <https://doi.org/10.32629/memf.v6i6.4633>.