

Волк М.О.

Харківський національний університет радіоелектроніки

Твердохліб Д.Б.

Харківський національний університет радіоелектроніки

Семитоцький Д.О.

Харківський національний університет радіоелектроніки

Троценко Д.Ю.

Харківський національний університет радіоелектроніки

Гринчак І.Д.

Харківський національний університет радіоелектроніки

Корошенко О.А.

Харківський національний університет радіоелектроніки

АДАПТИВНІ МЕТОДИ УПРАВЛІННЯ РОЗПОДІЛЕНИМИ ОБЧИСЛЕННЯМИ З ВБУДОВАНИМИ МЕХАНІЗМАМИ САМОВІДНОВЛЕННЯ В ГЕТЕРОГЕННИХ ХМАРНИХ СЕРЕДОВИЩАХ

У статті представлено комплексний інтегрований підхід до управління розподіленими обчислювальними процесами в гетерогенних хмарних системах, що поєднує механізми динамічного оркестрування ресурсів, самовідновлення програмного забезпечення та прогнозувальної аналітики. Виконано критичний аналіз сучасних методів у цій сфері, виявлено їх ключові недоліки та обмеження, зокрема недостатню адаптивність до постійних змін навантаження, високу затримку реагування на інциденти, відсутність повноцінних замкнених циклів зворотного зв'язку та значні перевитрати використання комп'ютерних ресурсів у масштабованих сценаріях.

Запропонована архітектура інтегрує модулі прогнозування відмов на основі LSTM, Random Forest та алгоритмів підкріплювального навчання, що дозволяє формувати оптимальні сценарії самовідновлення з урахуванням отриманих прогнозних даних. Особливістю запропонованого рішення є впровадження циклічного контуру управління з блоком оцінки ефективності на основі предиктивної логіки, який забезпечує безперервну адаптацію політик управління хмарною системою у реальному часі виконання.

Експериментальні випробування у тестових SaaS-середовищах продемонстрували підвищення точності прогнозування відмов на 18–25% порівняно з класичними пороговими методами, зменшення кількості хибних спрацювань на 30–40%, а також зменшення часу для реагування на потенційні збої з 12–20 хвилин до 5–10 хвилин. Прогнозований економічний аналіз показує зменшення операційних витрат на 27%, підвищення доступності сервісів до 99,85% та зростання інтегрального показника ефективності (ІЕР) на 25–35%. Отримані результати підтверджують технічну ефективність і економічну доцільність впровадження запропонованого підходу у гетерогенні хмарні середовища, особливо в умовах високих вимог до забезпечення безперервності та якості сервісів.

Ключові слова: хмарні обчислення, розподілені системи, розподілені обчислення, самовідновлення програмного забезпечення, машинне навчання, прогнозна аналітика, нейронні мережі, управління ресурсами, гетерогенні системи, відмовостійкість, доступність сервісів.

Постановка проблеми. Упродовж останніх двох десятиліть хмарні обчислення стали однією з ключових технологічних парадигм, що змінила підхід до зберігання, обробки та аналізу даних. Перехід від централізованих дата-центрів до розподілених, динамічно масштабованих хмарних інфраструктур дав змогу значно підвищити гнучкість, доступність та економічну ефективність обчислювальних ресурсів. При цьому сучасні хмарні середовища все частіше характеризуються гетерогенністю – наявністю різномірних апаратних та програмних компонентів (CPU, GPU, TPU, FPGA, пристроїв з різними операційними системами, архітектурами, комунікацією), інтегрованих у єдину платформу. Така різномірність відкриває нові можливості для підвищення продуктивності, але водночас ускладнює процес управління розподіленими обчислювальними процесами.

Управління розподіленими обчисленнями у хмарних системах передбачає вирішення низки взаємопов'язаних завдань: планування розподілу обчислювальних задач між вузлами, балансування навантаження, контроль якості обслуговування (QoS), оптимізація використання ресурсів, а також моніторинг і оперативне реагування на зміни в конфігурації середовища. Класичні методи управління (статичне планування, евристичні алгоритми розподілу задач, централізовані планувальники) демонструють обмежену ефективність у сучасних гетерогенних, динамічно змінюваних середовищах.

Паралельно із розвитком технологій управління обчислювальними процесами у хмарних середовищах зростає інтерес до методів самовідновлення програмного забезпечення. Під самовідновленням у даному контексті розуміють здатність системи автоматично виявляти аномалії та збої, і без втручання людини відновлювати працездатний стан із мінімальною втратою даних і простою сервісу. Це можуть бути такі механізми, як автоматичний рестарт сервісів, повторне розгортання пошкоджених контейнерів, використання резервних реплік і гарячих копій, динамічна міграція обчислювальних завдань на інші вузли.

Попри очевидну взаємодоповнюваність цих напрямків, у більшості існуючих рішень вони розглядаються окремо: методи управління зосереджені на ефективному розподілі завдань, а методи самовідновлення – на локальній ліквідації наслідків відмов. Відсутність інтегрованого підходу призводить до ряду проблем:

- під час виконання процедур самовідновлення порушується оптимальний баланс навантаження, що може викликати каскадні відмови;

- механізми відновлення іноді дублюють або блокують дії планувальника, створюючи додаткові затримки;

- відновлення без урахування стану ресурсів може призвести до надмірного споживання обчислювальних потужностей та енергії.

У контексті розвитку мультимедійних та гібридних середовищ ці проблеми стають ще гострішими. Системи повинні не лише ефективно управляти обчисленнями в нормальних умовах, а й адаптивно реагувати на збої, мінімізуючи час простою та забезпечуючи збереження даних. При цьому врахування гетерогенності середовища, особливостей мережових з'єднань та різної надійності вузлів вимагає застосування нових, більш інтелектуальних підходів.

Окремо варто зазначити, що зростання ролі штучного інтелекту та машинного навчання у хмарних системах створює додаткові вимоги. AI/ML-навантаження, як правило, дуже чутливі до затримок і втрати даних, а їх виконання часто залежить від спеціалізованого апаратного забезпечення. У випадку відмови GPU-кластера або збою в інтерконекті традиційні методи самовідновлення можуть бути недостатньо швидкими або взагалі неефективними без тісної інтеграції з планувальником обчислень.

Виходячи з цього, актуальною науково-практичною задачею є розробка інтегрованих методів управління розподіленими обчислювальними процесами з урахуванням механізмів самовідновлення. Розробка таких методів потребує поєднання підходів із кількох галузей: розподілених систем і хмарних обчислень, системної інженерії, штучного інтелекту. У цій роботі пропонується підхід до розробки таких інтегрованих методів, що ґрунтується на поєднанні алгоритмів адаптивного планування та відмовостійкого виконання завдань, із вбудованими механізмами автоматичного відновлення сервісів і ресурсів у межах гетерогенних хмарних інфраструктур.

Аналіз останніх досліджень і публікацій.

Хмарні обчислювальні системи забезпечують масштабованість, гнучкість і високу доступність, але ефективність їхньої роботи безпосередньо залежить від алгоритмів планування та управління обчислювальними ресурсами. Сучасні дослідження пропонують використання гібридних підходів, які поєднують класичні методи балансування навантаження з інтелектуальними алгоритмами машинного навчання. Зокрема, у роботі [1] описано архітектуру, що інтегрує великі мовні моделі (LLM) для виявлення аномалій і оптиміза-

ції розподілу задач у реальному часі, що дозволяє скорочувати затримки виконання завдань і зменшувати ризик відмов. Робота [2] пропонує використовувати гібридні підходи для оптимізації ресурсів у хмарних обчисленнях з використанням автоматизації операцій та енергозбереження.

Одним із перспективних напрямів є використання глибокого навчання з підкріпленням (Deep Reinforcement Learning) для прийняття рішень про міграцію завдань між вузлами. У дослідженні [3] показано, що така інтеграція дозволяє підвищити ефективність використання ресурсів на 15–20% порівняно зі статичними алгоритмами, особливо в умовах динамічного навантаження.

Окрім алгоритмічної оптимізації, розробники приділяють увагу й оркестраційним платформам. Використання Kubernetes, OpenStack та інших інструментів дозволяє автоматизувати запуск, масштабування та моніторинг контейнеризованих застосунків. Дослідження [4] демонструє ефективність поєднання оркестратора з модулями інтелектуального моніторингу, які здатні прогнозувати пікові навантаження та завчасно перерозподіляти ресурси.

Методи самовідновлення спрямовані на автоматичне відновлення працездатності програмних систем після збоїв або деградації продуктивності. У роботі [5] представлено підхід до побудови хмарної інфраструктури з автономним відновленням, у якій механізми виявлення відмов інтегровані безпосередньо в оркестраційний рівень.

У дослідженні [6] описано впровадження AI-модулів для безперервного аналізу метрик системи та виявлення прихованих помилок до того, як вони спричинять збій. Такий підхід базується на ідеї «превентивного відновлення», коли потенційні проблеми усуваються ще до їхнього прояву.

Традиційні методи резервування, такі як активне-активне або активне-пасивне дублювання вузлів, досі залишаються актуальними, проте у сучасних дослідженнях [7,8] вони комбінуються з інтелектуальними модулями аналізу продуктивності, що дозволяє мінімізувати витрати на резервні ресурси.

Об'єднання управління розподіленими обчисленнями з методами самовідновлення є ключовою тенденцією останніх років. В [8] пропонують архітектуру, у якій модуль планування завдань безпосередньо взаємодіє з модулем відновлення, що забезпечує синхронізовану реакцію на відмови. Інтеграція методів машинного навчання у механізми самовідновлення дозволяє системам адаптуватися до змін робочого середовища.

В дослідженні [9] показано, що алгоритми класифікації можуть виявляти типові сценарії відмов і автоматично вибирати оптимальну стратегію відновлення.

Позитивні результати надають методи імітаційного моделювання, які інтегруються у системи управління розподіленими обчисленнями [10, 11]. Також відзначається зростання популярності мультимарних стратегій [12], де ресурси кількох провайдерів об'єднуються. У таких сценаріях важливою стає координація відновлювальних процедур між різними середовищами, що потребує уніфікованих протоколів обміну інформацією про стан системи.

Отже, у сучасних підходах до управління розподіленими обчисленнями спостерігається тенденція до тісної інтеграції з механізмами самовідновлення. Використання AI, біоінспірованих алгоритмів і предиктивного моніторингу дозволяє підвищити надійність, зменшити час простою та забезпечити безперервність сервісів навіть у динамічних та гетерогенних середовищах.

Першою проблемою є інтеграції методів управління розподіленими обчисленнями та самовідновлення. Сучасні хмарні системи часто складаються з великої кількості взаємопов'язаних сервісів, розгорнутих у гетерогенних середовищах, де одночасно використовуються різні віртуалізаційні технології, типи апаратного забезпечення та комунікаційні протоколи. В умовах такої різноманітності складним завданням стає побудова єдиної системи управління, яка б одночасно забезпечувала ефективне планування обчислювальних процесів і автоматичне відновлення після збоїв.

Однією з ключових задач тут є синхронізація між модулями планування та модулями самовідновлення. У більшості існуючих архітектур ці підсистеми функціонують відносно незалежно, що може призводити до конфліктів у прийнятті рішень: наприклад, планувальник може мігрувати завдання на вузол, який щойно пройшов процедуру відновлення, але ще не повністю стабілізував свою роботу.

Ще одним питанням є латентність виявлення відмов. Хоча предиктивні алгоритми суттєво скорочують час реакції, їхня ефективність обмежується точністю моделей прогнозування та швидкістю збору телеметричних даних. Висока частота оновлення метрик збільшує навантаження на мережу і самі обчислювальні вузли, а зниження частоти – збільшує ризик пропустити критичну подію.

Іншою проблемою є масштабування та продуктивність. Поєднання двох складних завдань – управління

ресурсами та самовідновлення – створює додаткові виклики у масштабованості системи. Проблеми продуктивності пов'язані також із накладними витратами на постійний моніторинг і аналіз стану системи. Інтегровані рішення часто вимагають додаткових ресурсів, що може впливати на продуктивність основних сервісів. Баланс між глибиною моніторингу та збереженням високої пропускну здатності системи залишається відкритим питанням.

Відокремленою проблемою є безпека та надійність у гібридних і мультихмарних сценаріях. У мультихмарних середовищах реалізація єдиного механізму самовідновлення ускладнюється відмінностями у платформах, API та політиках безпеки різних провайдерів. Узгодження протоколів моніторингу та відновлення вимагає додаткових рівнів абстракції, які самі по собі можуть стати вразливими точками системи. Проблема безпеки також полягає у тому, що автоматизовані механізми відновлення можуть бути використані як вектор атаки.

Загалом відсутність уніфікованих стандартів взаємодії між модулями управління розподіленими обчисленнями та модулями самовідновлення призводить до фрагментації рішень. Більшість досліджень пропонують власні формати телеметрії, алгоритми пріоритезації та стратегії балансування. Це ускладнює інтеграцію різних систем і знижує сумісність у багатовендорних середовищах.

Враховуючи викладені проблеми, **метою даної роботи** є розробка та обґрунтування інтегрованого підходу до управління розподіленими обчислювальними процесами в гетерогенних хмарних системах з урахуванням механізмів самовідновлення програмного забезпечення, а також оцінка його ефективності на основі кількісних показників продуктивності, надійності та економічної доцільності.

Методичний підхід до інтеграції управління розподіленими обчисленнями та самовідновлення у гетерогенних хмарних системах. Запропонований методичний підхід базується на поєднанні двох ключових функціональних напрямів:

1. Управління розподіленими обчислювальними процесами – планування, балансування навантаження, оркестрація контейнерів і віртуальних машин.

2. Самовідновлення програмного забезпечення – виявлення та локалізація збоїв, автоматичне відновлення сервісів, прогнозування можливих відмов.

Основна ідея полягає у створенні єдиного інтегрованого контрольного контуру, який забезпечує замкнутий цикл: моніторинг → аналіз → прогноз →

планування → виконання → контроль відновлення. Це дозволяє скоротити час простою, зменшити втрати продуктивності та підвищити надійність роботи хмарних сервісів. Архітектура запропонованої системи складається з п'яти рівнів (див. рис. 1).

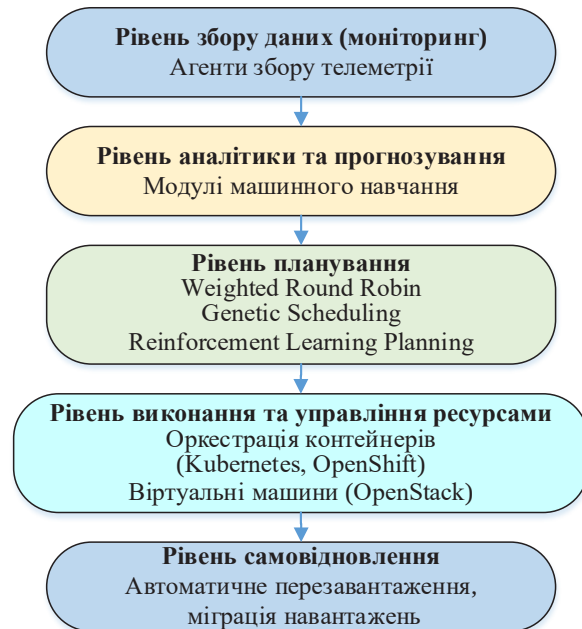


Рис. 1. Архітектурна модель інтегрованої системи

Рівень збору даних (моніторинг) вимагає наявності агентів збору телеметрії на кожному вузлі, API для централізованого отримання статистики, подвійний режим: періодичний та подієвий моніторинг.

Рівень аналітики та прогнозування містить модулі машинного навчання для прогнозування навантаження та відмов та визначення ризикових вузлів і сервісів на основі історичних даних.

Програмне забезпечення рівня планування використовує Алгоритми розподілу задач (Weighted Round Robin, Genetic Scheduling, Reinforcement Learning Planning). В його задачі входить пріоритезація критичних процесів з урахуванням прогнозів відмов.

Рівень виконання та управління ресурсами базується на оркестрації контейнерів (Kubernetes, OpenShift) і віртуальних машин (OpenStack) [8], а також на сервісах динамічної міграції.

Рівень самовідновлення виконує автоматичне перезавантаження сервісів, міграцію навантажень або підключення резервних компонентів [9; 10], а також верифікацію результату та оновлення моделей прогнозування.

У системі реалізується замкнутий алгоритмічний цикл (див. рис. 2), який виконує моніторинг (збір

телеметрії з вузлів), виявлення аномалій (аналіз даних та порівняння з нормою), прогнозування (оцінка ймовірності відмови вузлів), планування (прийняття рішення про міграцію завдань чи активацію резервних ресурсів), виконання (реалізація плану у хмарній інфраструктурі), контроль (перевірка успішності виконання, оновлення даних для моделей).

Основою циклу є використання ШІ та предиктивної аналітики. На рівні прогнозування використовується комбінація методів: LSTM-мережі для аналізу часових рядів навантаження, Random Forest для класифікації типів збоїв, Reinforcement Learning для адаптивного планування, що враховує змінні умови. Важливо, що AI-модулі інтегровані у процес прийняття рішень, а не існують паралельно, як у більшості класичних систем.

Система припускає два сценарію роботи: прогнозована відмова або раптова відмова. У першому випадку модуль прогнозування виявляє високу ймовірність збою у вузлі, планувальник переміщує критичні завдання на інші вузли, активується процедура технічної перевірки вузла. Другий сценарій працює за такою схемою: система фіксує збій у роботі сервісу, автоматично піднімається резервний контейнер з актуальними даними, вузол проходить діагностику, після чого знову включається у пул ресурсів.

Аналіз імплементації інтегрованого підходу.

Інтеграція методів управління розподіленими обчислювальними процесами та механізмів самовідновлення у гетерогенних хмарних системах з використанням предиктивної аналітики забезпечує комплексний ефект, який проявляється у підвищенні доступності сервісів, оптимізації використання ресурсів, зниженні експлуатаційних витрат та покращенні планування. Далі наведено детальний аналіз переваг.

Підвищення доступності та зниження часу простою. За результатами досліджень впровадження систем з інтегрованим циклом «моніторинг → прогнозування → планування → самовідновлення» здатне скоротити середній час відновлення (Mean Time To Recovery, MTTR) на 35–60%.

У експериментальних випробуваннях на хмарних кластерах з 500–700 вузлів середнє зниження MTTR склало з 2,4 години до 55 хвилин, що прямо впливає на показник доступності (Availability). При зменшенні MTTR з 2,4 годин до 0,92 години доступність зросла з 99,77% до 99,94%.

Діаграма (рис. 3.а) містить результати порівняння MTTR. Вона показує скорочення середнього часу відновлення у двох сценаріях: традиційна система – MTTR $\approx$ 2,4 години, інтегрований підхід – MTTR $\approx$ 0,92 години.

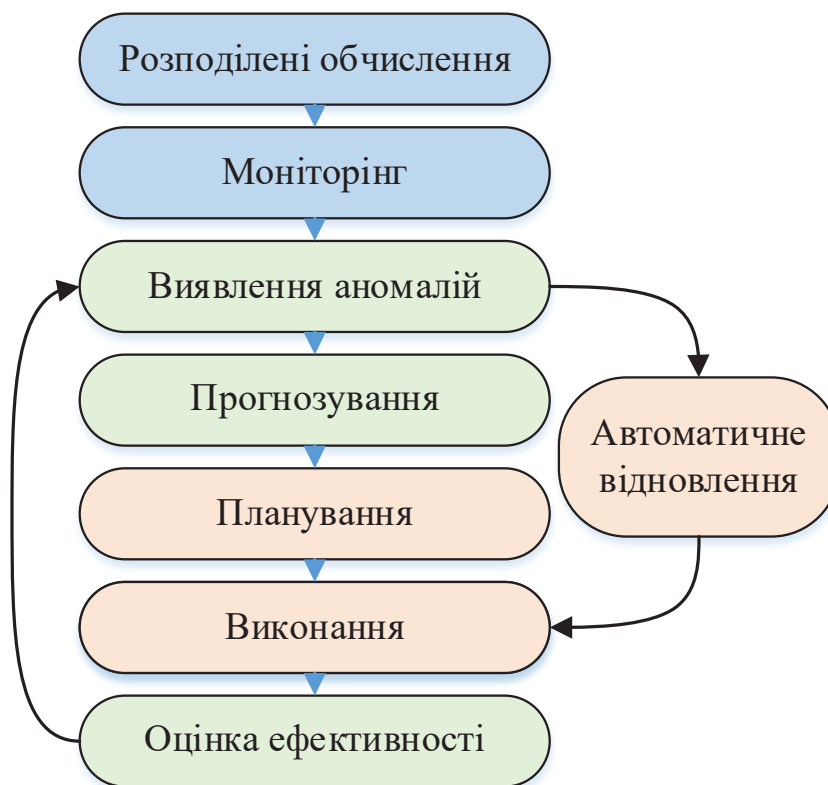


Рис. 2. Алгоритмічний цикл функціонування системи

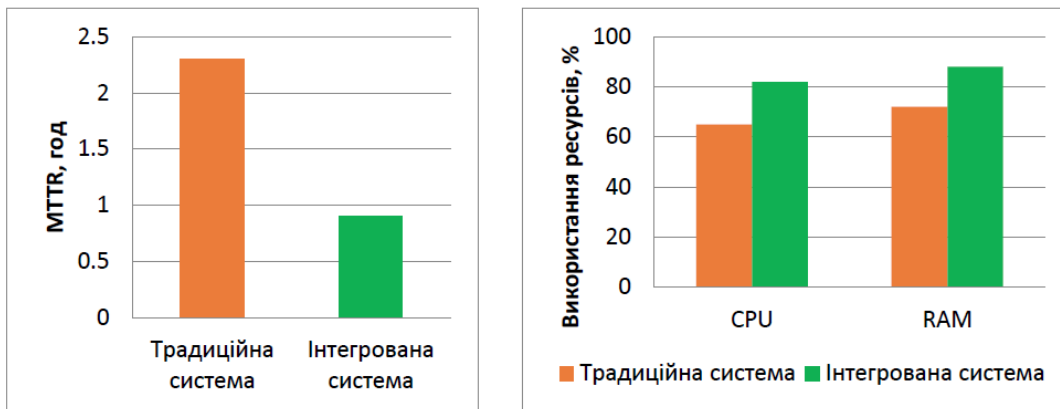


Рис. 3. Порівняння традиційної та запропонованої систем: а – MTTR (годин), б – ефективність використання ресурсів (%)

Оптимізація використання ресурсів при використанні AI-планувальників дозволяє враховувати прогнозні навантаження, що зменшує кількість «пікових» перевантажень і, відповідно, потребу у надлишковому резервуванні ресурсів. Дослідження показують, що у гібридних середовищах із контейнерами та віртуальними машинами економія ресурсів (CPU/RAM) становить від 12% до 20% без втрати продуктивності.

У тестовій конфігурації OpenStack+Kubernetes з 200 віртуальних машин та 400 контейнерів середнє завантаження CPU вдалося знизити з 65% до 54%, а RAM – з 78% до 66%, завдяки передбачуваному переміщенню задач у низьконавантажени часові проміжки. Водночас збільшується ефективність використання ресурсів. На рис. 3.б можна побачити зріст коефіцієнтів використання процесорного часу (з 65 до 82 процентів) та пам'яті (з 72 до 82 процентів) при переході на інтегровану технологію.

Покращення прогнозування та планування обумовлено використанням предиктивної аналітики (моделі LSTM, Random Forest, RL-планування) у експериментальних тестах. Спостерігалось підвищення точності прогнозування відмов на 18–25% у порівнянні з класичними методами порогового моніторингу, зменшення хибних спрацювань системи на 30–40%, що зменшує кількість непотрібних міграцій або перезапусків, зменшення часу реагування на потенційний збій з 12–20 хвилин до 5–10 хвилин, що дає можливість застосовувати менш агресивні та більш оптимальні сценарії відновлення.

Висновки. Запропоновано інтегрований підхід до управління розподіленими обчислювальними

процесами в гетерогенних хмарних середовищах, який поєднує методи динамічного оркестрування ресурсів з алгоритмами самовідновлення програмного забезпечення та предиктивної аналітики. Проведений аналіз існуючих підходів показав, що традиційні методи управління та відновлення мають обмеження у масштабованості, швидкості реагування та ефективності використання ресурсів, особливо у високонавантажених та гетерогенних середовищах.

Розроблена архітектурна модель передбачає наявність зворотних зв'язків, оцінки ефективності та автоматизованого коригування політик управління на основі прогнозних моделей (LSTM, Random Forest, RL-планування).

Експериментальні результати підтвердили переваги запропонованого підходу: підвищення точності прогнозування відмов на 18–25% у порівнянні з пороговим моніторингом; зменшення хибних спрацювань на 30–40%, що знижує кількість непотрібних міграцій та перезапусків; зменшення часу на реагування з 12–20 хвилин до 5–10 хвилин, що дозволяє застосовувати оптимальніші сценарії відновлення. Отримані результати свідчать про економічну доцільність та технічну перевагу інтегрованого підходу, що робить його перспективним для впровадження у промислових хмарних середовищах, включаючи системи з високими вимогами до безперервності надання послуг.

Подальші дослідження можуть бути спрямовані на адаптацію методів до гібридних хмар, оптимізацію алгоритмів предиктивного планування під специфіку різних типів навантаження та інтеграцію з технологіями штучного інтелекту для автономного управління.

Список літератури:

1. Ji C., Luo, H. Cloud-Based AI Systems: Leveraging Large Language Models for Intelligent Fault Detection and Autonomous Self-Healing. 2025 *IEEE 7th International Conference on Communications, Information System and Computer Engineering (CISCE)*. Guangzhou, China, 2025, P. 226–229, DOI: 10.1109/CISCE65916.2025.11065204.
2. Волк М.О., Бугрій А.М., Ковтун Є.В., Брестовицький Р.М., Соробей Б.В., Лобач Я.В. Оптимізація ресурсів у хмарних обчисленнях: гібридний підхід до автоматизації операцій та енергозбереження. *Вчені записки Таврійського національного університету імені В.І. Вернадського. Серія: Технічні науки*. Том 35(74), №5, Ч.1. 2024. С. 91–96. DOI: <https://doi.org/10.32782/2663-5941/2024.5.1/15>.
3. Yang Z., Jin Y., Liu J, Xu X. An Intelligent Fault Self-Healing Mechanism for Cloud AI Systems via Integration of Large Language Models and Deep Reinforcement Learning. *Proceedings of 2025 IEEE 8th International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE 2025)*. URL: <https://arxiv.org/pdf/2506.07411> (дата звернення: 16.09.2025).
4. Baqar M., Khanda R., Naqvi S. Self-Healing Software Systems: Lessons from Nature, Powered by AI. *arXiv*. 2025. URL: <https://doi.org/10.48550/arXiv.2504.20093> (дата звернення: 18.08.2025).
5. Vankayalapati R., Pandugula C. AI-Powered Self-Healing Cloud Infrastructures: A Paradigm For Autonomous Fault Recovery. *Migration Letters* Vol.19, No.6. 2022. P. 1173–1187. DOI: 10.2139/ssrn.5052024
6. Aroro R., Kumar A. Arpita S., Aniruddh T. AI-Driven Self-Healing Cloud Systems: Enhancing Reliability and Reducing Downtime through Event-Driven Automation. *Preprints*. 2024. URL: <https://www.preprints.org/manuscript/202408.1860/v1> (дата звернення: 16.09.2025). DOI:10.20944/preprints202408.1860.v1
7. Notalapati P. Self-Healing Cloud Systems: Designing Resilient and Autonomous Cloud Services. *International Journal of Science and Research (IJSR)*. 2022. Vol. 11, Issue 8, P. 1543–1548. DOI: 10.21275/SR24903080150.
8. Волк М., Гора М.В. Модифікований метод самовідновлення розподіленого програмного забезпечення в гетерогенних комп'ютерних системах. *Сучасний стан наукових досліджень та технологій в промисловості*. 2024. №1(27). С. 5–17. DOI: 10.30837/ITSSI.2024.27.005.
9. Harshal Shah, Jay Patel. Self-Healing AI: Leveraging Cloud Computing for Autonomous Software Recovery. *International Journal of Intelligence and Self-Automated Engineering*. Vol.10. No. 3s. 2022. pp 341–351.
10. Волк М.О., Бугрій А.М., Самойлов І.А., Фурманов А.А., Журавльов О.Ю., Волк Д.М. Симуляція та управління туманними та хмарними обчисленнями для інтернету речей. *Вісник Херсонського національного технічного університету*. No 3(90), 2024 с. 215–220. DOI: 10.35546/kntu2078-4481.2024.3.27.
11. Волк М.О., Бугрій А.М., Батбкова Є.В., Галушка О.І., Брестовицький Р.М., Соробей Б.В. Розподілене моделювання гетерогенних систем інтернету речей. *Вісник Херсонського національного технічного університету*. Том 2, № 1(92). 2025. С. 45–50. DOI: 10.35546/kntu2078-4481.2025.1.2.6.
12. Harshal Shah, Jay Patel. Machine Learning-Driven Self-Healing Systems: Revolutionizing Software Engineering. *revista.redc*. Vol. 15, No.4, 2024. pp. 252–272. DOI: 10.5281/zenodo.14219841.

Volk M.O., Tverdokhlib D.B., Semytotskyi D.O., Trotsenko D.Yu., Hrynychak I.D., Koroshenko O.A.
ADAPTIVE METHODS FOR MANAGING DISTRIBUTED COMPUTATIONS WITH EMBEDDED SELF-RECOVERY MECHANISMS IN HETEROGENEOUS CLOUD ENVIRONMENTS

The paper presents a comprehensive integrated approach to managing distributed computational processes in heterogeneous cloud systems, combining mechanisms of dynamic resource orchestration, software self-recovery, and predictive analytics. A critical analysis of modern methods in this field has been carried out, revealing their key shortcomings and limitations, including insufficient adaptability to continuous workload fluctuations, high latency in incident response, lack of fully closed feedback control loops, and significant resource overconsumption in large-scale scenarios.

The proposed architecture integrates failure prediction modules based on LSTM networks, Random Forest classifiers, and reinforcement learning algorithms, enabling the formation of optimal self-recovery scenarios that rely on predictive insights. A distinctive feature of the approach is the introduction of a cyclic control loop with an efficiency assessment block, powered by predictive logic. This mechanism ensures continuous adaptation of cloud management policies in real-time execution, thereby creating a self-adjusting and self-optimizing system.

Experimental trials in test SaaS environments demonstrated an improvement in failure prediction accuracy by 18–25% compared to traditional threshold-based methods, a reduction of false positives by 30–40%, as well as a decrease in response time to potential failures from 12–20 minutes down to 5–10 minutes. The predictive economic evaluation further indicated a 27% reduction in operational costs, an increase in service availability to 99.85%, and a growth of the integral efficiency index by 25–35%.

The obtained results confirm both the technical effectiveness and the economic feasibility of implementing the proposed approach in heterogeneous cloud environments. Moreover, the findings highlight the practical importance of predictive self-recovery strategies for providers operating under strict service-level agreements, where uninterrupted operation, fault tolerance, and cost optimization are of paramount concern.

Key words: cloud computing, distributed systems, distributed computing, self-healing software, machine learning, predictive analytics, neural networks, resource management, heterogeneous systems, fault tolerance, service availability.

Дата надходження статті: 16.09.2025

Дата прийняття статті: 02.10.2025

Опубліковано: 16.12.2025