

## ІНТЕЛЕКТУАЛЬНА СИСТЕМИ УПРАВЛІННЯ ЕЛАСТИЧНІСТЮ ОБЧИСЛЮВАЛЬНОЇ ХМАРИ

Вінницький національний технічний університет

### Анотація

Запропонована та досліджена на сучасній кластерній системі Amazon EC2 інтелектуальна система управління еластичністю обчислювальної хмари, алгоритм функціонування якої використовує метод балансування навантаження.

**Ключові слова:** кластерна система, балансування навантаження, хмарний сервіс, віртуальна машина.

### Abstract

Intelligent control system of cloud computing flexibility, functional algorithm, based on load balancing method was proposed and tested on modern Amazon EC2 cluster system.

**Keywords:** cluster system, load balancing, cloud service, virtual machine.

### Вступ

Обчислювальні хмари є сучасним потужним інструментом для швидкого розгортання доданків у мережі, що надають їх власникам гнучкі можливості керування різноманітними програмно-апаратними ресурсами. Однією з ключових характеристик обчислювальної хмари є еластичність, що означає можливість миттєвого надання нових чи розширення/звуження існуючих послуг та ресурсів в автоматичному режимі. Існуючі методи управління еластичністю [1] досить прості і не враховують профілі навантаження доданків, що вони обслуговують. Це призводить до відмов у обслуговуванні в процесі стрибкоподібного зростання навантаження чи до простоювання високовартісних обчислювальних ресурсів при його точкових змінах. Метою даної роботи є розробка системи управління еластичністю обчислювальної хмари, що враховує зазначені недоліки.

### Аналіз задачі

Обчислювальна хмара це інтегрований ресурс, який включає в себе множину незалежних серверів, мережеве обладнання, системи зберігання даних та мережу, що пов'язує їх. Архітектура кластерного обчислювального середовища, що наведена на рисунку 1, включає такі компоненти:



Рисунок 1 – Архітектура кластерної системи

- сервіс хмарних обчислень – система, яка забезпечує спільне використання ресурсів хмари та надає засоби їх керування;
- віртуальна машина – програмний комплекс, що реалізує емуляції апаратного обладнання для операційної системи, що функціонує як внутрішньо системний процес;

- підсистема автоматичного масштабування – реалізує властивості еластичності шляхом зміни кількості віртуальних машин в залежності від навантаження;
- підсистема балансування навантаження – розподіляє запити користувачів поміж наявних віртуальних машин за одним із алгоритмів round robin, least connections, ip hash;
- гіпервізор – система, яка забезпечує одночасне, паралельне функціонування декількох віртуальних машин на одному фізичному сервері.

Хмарні системи інтегрують зазначені компоненти в єдину керовану інфраструктуру, що дозволяють власникам сервісів забезпечувати виконання угоди про рівень послуг (SLA). Зазвичай до SLA входять показники як проектний час відгуку, пропускна спроможність системи, рівень відмов та час безперебійної роботи системи.

Основними причинами, що призводять до порушення зазначених показників є перевантаження процесору, пропускної здатності мережі, диску або оперативної пам'яті, результатом чого є збільшення часу на обробку запиту або відмовлення у обслуговуванні. Забезпечення характеристик SLA реалізується за допомогою еластичності хмарної системи. Механізми, що реалізують еластичність в хмарних системах є досить простими, оскільки реалізуються правилами ЯКЩО-ТО, які визначають користувачі, на основі яких підсистема автоматичного масштабування здійснить додавання/видалення віртуальних машин до групи на яку балансується навантаження. Недоліком у таких системах є те, що зазначені правила не є гнучкими, не беруть до уваги історичні дані щодо навантаження, тобто ігнорують звичний профіль навантаження системи та особливості її функціонування. Це зазвичай призводить до відмов у період стрибкоподібного росту навантаження та до більших витрат на обслуговування при точкових його сплесках.

### Результати досліджень

В роботі запропоновано інтелектуальну систему управління еластичністю, в якій пропонується використання власної системи балансування навантаження та програмних агентів, що встановлюються на віртуальні машини з метою отримання інформації про їх параметри діяльності серед яких інтенсивність вхідних запитів, час відгуку, завантаження процесора, використання оперативної пам'яті, довжина черги вхідних запитів, iowait, iops.

Алгоритм функціонування інтелектуальної системи можна описати наступним чином: агент аналізує та накопичує вхідні запити користувачів. На основі аналізу миттєвих та історичних даних за попередні проміжки часу прогнозується ймовірність збільшення або зменшення навантаження на систему за допомогою метода прогнозування запропонованого в [2]. Метод побудований на використанні логістичної регресії (передбачення ймовірності виникнення деякої події шляхом підгонки даних до логістичної кривої) та методів машинного навчання з використанням бібліотеки Apache Mahout, яка надає пакет для логістичної регресії: org.apache.mahout.classifier.sgd.TrainLogistic. На виході отримуємо одне із двох можливих значень 0 – система стабільна, необхідне уточнення для визначення чи необхідно видалення вузла, 1 – система перевантажена, потрібно додати додатковий обчислювальний ресурс. Базуючись на отриманому результаті здійснюється розгортання Базуючись на отриманому результаті прогнозування, підсистема планування прийматиме рішення про необхідність розгортання додаткових апаратних ресурсів (віртуальних машин) в кластерній системі або вилучення ресурсів кластера. У випадку якщо ймовірність стрибкоподібної зміни навантаження менше 50%, необхідне додаткове уточнення необхідне для прийняття рішення про навантаження. Для цього в роботі розроблено нечітку базу знань надану експертами, що використовує та поєднує наступні характеристики навантаження на робочі сервери для здійснення прогнозування, а саме: навантаження на процесор, завантаження оперативної пам'яті, завантаження мережі. В процесі прийняття рішення використовується нечітка логіка, яка інтегрована в бібліотеку Fuzzy Framework library для мови програмування високого рівня C#.

В роботі було проведено тестування поведінки сучасних систем під час зміни навантаження на систему, що дозволило зробити висновки про ефективність запропонованої інтелектуальної системи. За основу було обрано хмарну систему Amazon EC2, яка має ряд переваг серед, яких гнучкість, простота використання та налагодження, економічність та здатність до швидкого масштабування. Основним показником було обрано час відгуку системи при надходженні запитів та кількість системних одиниць віртуальних машин, що було створено при збільшенні навантаження. За допомогою засобу Amazon

Elastic Beanstalk було розгорнуто тестову веб-систему, призначену для перевірки стійкості системи при подачі навантаження. Інтенсивність запитів імітувалася з використанням утиліти apache-jmeter. При подачі різної кількості запитів кластерна система Amazon розгортала допоміжні ресурси. Результати дослідження наведено в таблиці 1 та рисунках 2,3.

Таблиця 1 – Дослідження характеристики системи Amazon EC2

#	Інтенсивність запитів	Кількість віртуальних машини	Час відгуку
1	6000	1	2
2	12000	3	5
3	24000	6	9
4	56000	13	15
5	100000	25	30

На рисунках 2,3 показано, що при подачі однакового навантаження на кластерну систему, модуль прогнозування, що міститься в Amazon створює більшу кількість вузлів кластера ніж запропонована при цьому кластерна система, що надає web-сервіси та забезпечує час обслуговування запитів відповідно до поставлених обмежень. Тому можна стверджувати, що запропонована інтелектуальна система управління навантаженням кластера побудована на основі методу прогнозування [2] є більш ефективною, так як вимагає менше хмарних ресурсів для безперебійного функціонування веб-системи.



Рисунок 2 – Результати аналізу розмірності кластеру

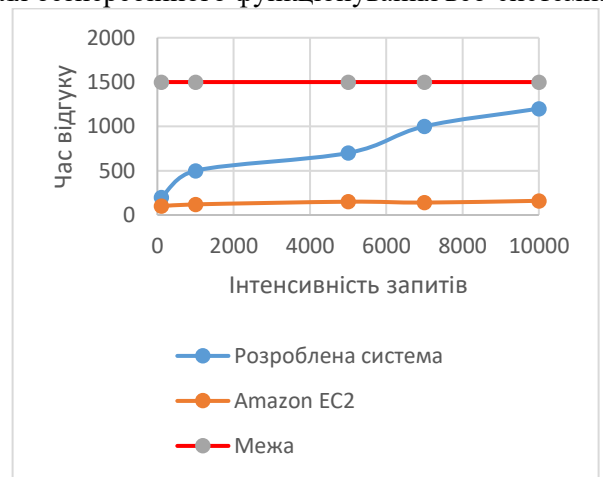


Рисунок 3 – Результати аналізу інтенсивності запитів

### Висновки

В роботі запропонована інтелектуальна система управління еластичністю обчислювальної хмари, алгоритм функціонування якої оснований на методі балансування навантаження з використанням машинного навчання та нечіткої логіки. Проведено дослідження ефективності функціонування запропонованої системи на сучасній кластерній системі Amazon EC2, результати якого показали її високу економічність, що дозволяє застосовувати її в сучасних веб-системах.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Larsen K.R.T. A Cost and Performance Model for Web Service Investment / K. R. T. Larsen and P. A. Bloniarz. – ACM, Feb. 2000, pp. 109-116.
2. Menasce D. A. Capacity Planning and Performance Modeling: From Mainframes to Client-Server Systems / D. A. Menasce, V. A. F. Almeida, and L. W. Dowdy. – Prentice Hall, Upper Saddle River, New Jersey, 1994.
3. Москвін О.М. Метод прогнозування навантаження на кластерну систему / О.М. Москвін, А.І. Деркач. - [Електронний ресурс]: // vntu– Режим доступу до ресурсу:<http://ies.vntu.edu.ua/reports/Proceedings/PROCEEDING-IES-2016.pdf>.

**Анна Ігорівна Деркач** — студент групи 2АКІТ-16м, факультет комп'ютерних систем і автоматики, Вінницький національний технічний університет, Вінниця, e-mail: [derkachanika@gmail.com](mailto:derkachanika@gmail.com);

**Москвіна Світлана Михайлівна** – к.т.н., професор кафедри комп'ютерних систем управління, Вінницький національний технічний університет, Вінниця, email : [moskvina@ukr.net](mailto:moskvina@ukr.net);

**Москвін Олексій Михайлович** — к.т.н., ст. викладач кафедри комп'ютерних систем управління, Вінницький національний технічний університет, Вінниця, e-mail: [moskvin.aleksey@gmail.com](mailto:moskvin.aleksey@gmail.com);

**Derkach Anna I.** — Department of Computer Systems and Automation, Vinnytsia National Technical University, Vinnytsia, email : [derkachanika@gmail.com](mailto:derkachanika@gmail.com);

**Oleksiy M. Moskvın** – Ph.D., lecturer in Computer Control Systems, Vinnytsia National Technical University, Vinnitsa, e-mail: [moskvin.aleksey@gmail.com](mailto:moskvin.aleksey@gmail.com);

**Svetlana M. Moskvina** – Ph.D, Professor of the Chair of Computer Control Systems, Vinnytsia National Technical University, Vinnytsia, email : [moskvina@ukr.net](mailto:moskvina@ukr.net)