ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

NURE
Харківський національний університет радіоелектроніки

Магістерська кваліфікаційна робота

Методи підвищення ефективності використання ресурсів в системах хмарних обчислень

Студентка гр. КСМзм-22-1 Якименко А.С.
Керівник доц. каф. ЕОМ Саранча С.М.

Харків, 2024

Мета та задачі роботи

Метою роботи Метою роботи є підвищення ефективності виконання складних наукових застосувань в системах хмарних обчислень шляхом зменшення енергоспоживання та кількісті обчислювальних ресурсів

Для реалізації мети робити необхідно вирішити наступні задачі:

1. Запропонувати ресурсоефективну структуру на основі планування робочого процесу для хмарних обчислень.
2. Розробити безпечну політику планування робочого процесу шляхом генерації хеш-кодів.
3. Зменшити тривалість створення та накладні витрати на систему за рахунок об'єднання подібних завдань в пули або кластери.
4. Виконувати аналіз використання ресурс з динамічним перерозподілом віртуальних машин.
5. Вибирати найбільш відповідні віртуальні машини для виконання динамічного робочого процесу.

Об'єктом досліджень є процес управління ресурсами в хмарних системах.

Предмет досліджень: Методи підвищення ефективності використання ресурсів в системах хмарних обчислень

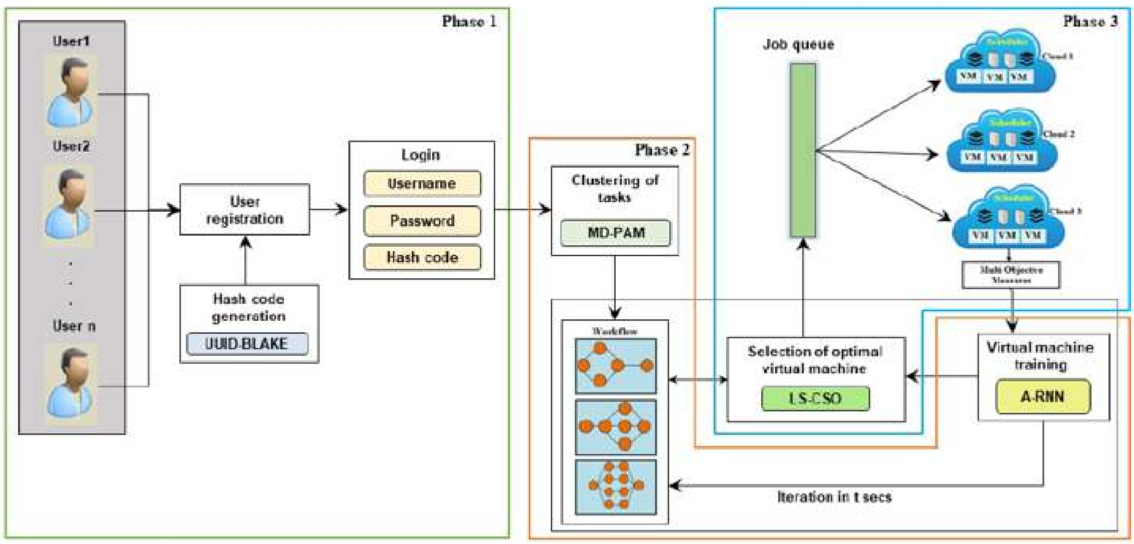
3

Аналіз предметної області

Algorithm	Type of Workflow used						Scheduling Objectives							
	Montage	SIPHT	LIGO	Cybershake	Epigenomics	Random workflow	Makespan	Resource Utilization	Deadline	Cost	Reliability	Security	Energy Consumption	Budget
BTS	√	√	√	√	√	√	-	-	√	√	-	-	-	-
EMO WFS in cloud	√	√	√	√	√	-	√	-	-	√	-	-	-	-
NMMWS	√		-	-	-	-	√	√	-	-	-	-	-	-
HMOPSO for WFS	√	√	√	√	√	-	-	-	-	√	-	-	√	-
RSO	-	-	-	-	√	-	√	√	-	√	-	-	-	-
HMO Workflow Scheduling in IaaS Cloud	√	√	√	√	-	-	√	-	-	-	-	-	-	-
GRP-HEFT	√	√	-	√	-	-	√	√	-	√	-	-	-	√
Task Replication	√	√	√	√	-	-	-	-	-	√	√	-	-	-
BDCWS	√	-	√	-	√	-	-	-	√	√	-	-	-	√
MOWS	√	-	-	√	-	-	√	-	-	-	-	√	-	-
TOPSIS	√	-	√	-	√	-	-	√	√	√	-	-	-	-
ALO-PSO	√	-	-	-	-	-	√	-	-	√	√	√	-	-
GMPSO	√	√	-	√	√	-	√	√	-	-	-	-	-	-
REEWS	-	-	-	-	-	√	-	-	-	-	√	-	√	-
CPCS	√	√	-	√	-	√	-	-	√	-	√	√	-	-

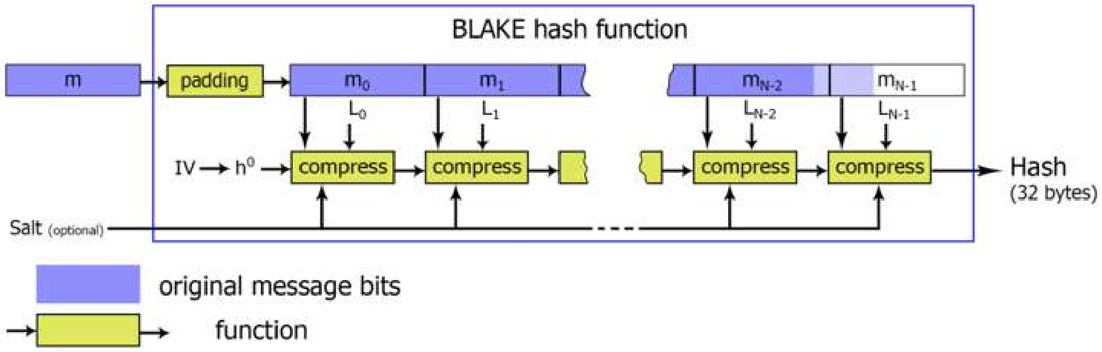
2

Мультіоб'єктний фрейворк



4

Автентифікація користувачів на основі алгоритму хешування UUID-BLAKE



5

Метод кластерізації програмних завдань

Етап1. На початковому етапі розподілу серед усіх завдань *v* обираються завдання з мінімальною відстанню до інших завдань з формуванням множини *O*. Під об'єктами розуміються програмні завдання та зв'язок між ними для обміну даними. Відстані між об'єктами *i* та *j* відповідно *Ai* та *Bj* обчислюються за допомогою формули відстані, яка задана як $MD = \sum^{n} |A_i - B_j|$.

Етап 2. Наступне завдання розглядається як кандидат, який додається на виконання в конкретний кластер, якщо він має найбільший приріст параметра ефективності $TG_i = \sum Max\{H_i - \vartheta(j,i), 0\}$, де *O* – множина завдань для розподілення, при цьому, якщо $H_i > \vartheta(j,i)$, то якість кластерізації покращується. $\vartheta(j,i)$ - функція обчислення ефективності *Hi* – середнє значення параметру ефективності для кластеру.

Етап 3. Після обчислення загального приросту всіх об'єктів в *O*, об'єкт *G*, який має максимум TG_g, вибирається, що математично сформулюється як $k = k \cup \{G\}$, $O = O - \{G\}$. Ці кроки повторюються, доки всі об'єкти *v* не будуть обрані.

Етап 4. На даному етапі можна покращити якість кластеризації шляхом оптимізації набору вибраних об'єктів, які закінчуються, враховуючи всі пари обміну. Наприклад, якщо є два кластера *v* і *k*, то ефект обміну можна обчислити за виразом $\xi_{vk} = \sum \{v_{tvk} | t \in O\}$, де v_{tvk} позначає внесок кожного об'єкта t в множину *O* для заміни *v* і *O*. Заміна можлива, якщо $\vartheta(t,v) > H_t$ або $\vartheta(t,v) = H_t$.

Етап 5. Пара *(v, k)* з мінімумом ξ_{vk} вибирається, щоб визначити, чи виконано заміну чи ні. Якщо $\xi_{vk} < 0$, відбувається процес обміну і повертаємося до початкової фази обміну. В іншому випадку процес обміну вважається завершеним.

Етап 6. На наступному етапі об'єкти, які є найближчими до медоїдів з мінімальними відмінностями між об'єктами, групуються в кластери, і це визначається як $C_P = \{C_1, C_2, C_3, \ldots, C_n\}$, де *n* –кількість кластерів.

6

Алгоритм вибору віртуальних машин

Algorithm 1 Pseudocode for LS-CSO algorithm

Input:Number of virtual machines
Output: Selection of optimal virtual machines
Begin
    Initialize the positions $\mathrm{VM} = \{\mathrm{vm}_1, \mathrm{vm}_2, \mathrm{vm}_3, \ldots\ldots \mathrm{vm}_n\}$ randomly.
    Evaluate the memory of each vm by the initial position.
    Compute the fitness value for each vm.
    Update the position of vm according to the random another vm.
    Improve the awareness probability by using,
        $\mathrm{g(h)} = \frac{\mathrm{s(x)} - \min \mathrm{s(x)}}{\max \mathrm{s(x)} - \min \mathrm{s(x)}}$
    if $\mathrm{R_j} \geq \mathrm{AP_{j,Itr}}$
        $\mathrm{x_{i,Itr}} = \mathrm{x_{i,Itr}} + \mathrm{R_i} \times \mathrm{Fl_{i,Itr}} \times (\mathrm{Mem_{j,Itr}} - \mathrm{x_{i,Itr}})$
    else
        Relocate the VM randomly
    endif
    Check the feasibility of new position
    Update the VMs memory
    The process continue until the best solutions obtained
End

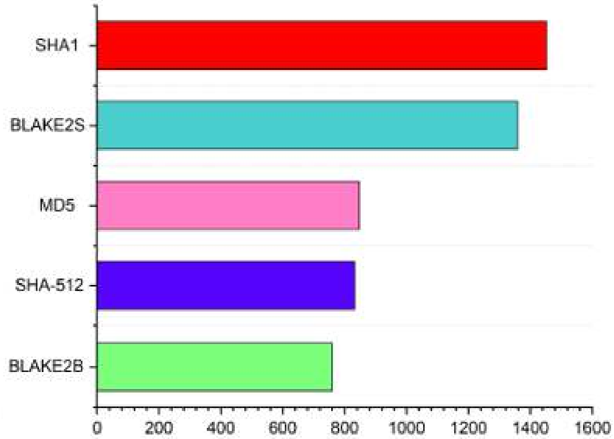
7

Конфігурація експериментального кластеру

Хмарна платформа	Тип	Кількість комп'ютерів	Ядра	Процесор	Пам'ять	Тип пам'яті	Віртуалізація	Мережа
Azure	A10	8	8	Intel Core i7-11800H	64 GB	DDR3	Hyper-V	10 gigabit Ethernet

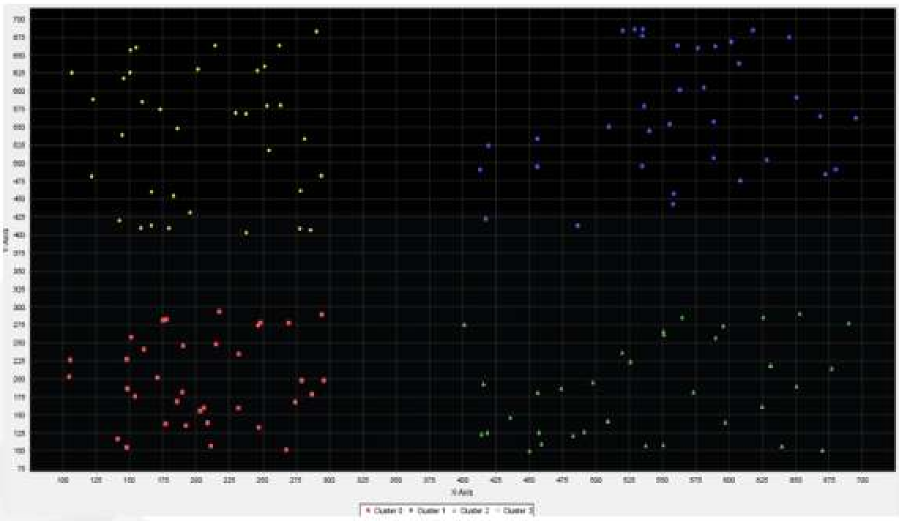
8

Порівняння часу виконання різних алгоритмів хешування з розробленою модифікацією



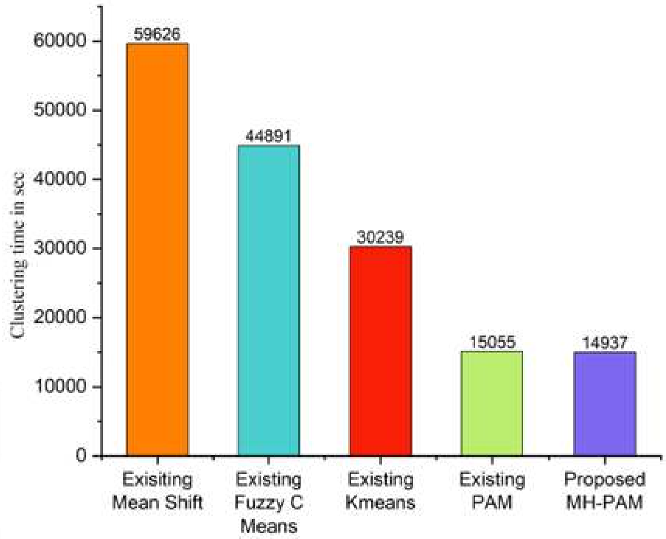
9

Кластерізація завдань з використанням розробленого методу



10

Порівняння часу виконання різних алгоритмів планування



11

Висновки

Результати виконання роботи забезпечили підвищення ефективності виконання складних наукових застосувань в системах хмарних обчислень шляхом зменшення енергоспоживання та кількості обчислювальних ресурсів

Було вирішено наступні задачі:

1. Запропонована ресурсоефективну структуру на основі планування робочого процесу для хмарних обчислень із використанням UUID-BLAKE, MH-PAM, A-RNN і LS-CSO.
2. Розроблено безпечну політику планування робочого процесу шляхом генерації хеш-кодів. Хеш-код генерується за допомогою алгоритму UUID-BLAKE для підвищення автентичності хмарного сервера, дозволяючи законному користувачеві отримати до нього доступ.
3. Зменшено тривалість створення та накладні витрати на систему за рахунок об'єднання подібних завдань в пули або кластери. Процес кластеризації виконується за допомогою MH-PAM.
4. Віртуальні машини відстежуються за допомогою A-RNN, який передбачає аналіз використання ресурсів. На основі наявності ресурсів створюється динамічний робочий процес.
5. За допомогою алгоритму LS-CSO обираються найбільш відповідні віртуальні машини для виконання динамічного робочого процесу. Можливості кожної віртуальної машини оцінюються, а завдання призначаються на найбільш підходящий ресурс. Завдяки цьому споживання енергії зменшується, а ресурси використовуються ефективно.

Апробація:

Саранча С.М., Якименко А.С., Баляба Ю.В., Сєрих О.О. Методи розподілення віртуальних машин за хмарними ресурсами. Проблеми інформатизації: Матеріали одинадцятої міжнародної науково-технічної конференції. –Баку – Харків – Бельсько-Бяла , 16 – 17 листопада 2023 року, с.50

12