

## ПІДХІД ДО МІГРАЦІЙ У ЗАДАЧІ РОЗМІЩЕННЯ КОНТЕЙНЕРІВ

*Анотація. У сучасному світі інформаційних технологій, хмарні обчислення, та контейнера віртуалізація набувають все більшої популярності. Широкого попиту набуває послуга розміщення контейнерів. Перед хмарними провайдерами постає завдання не тільки ефективного початкового розміщення контейнерів, а й їхньої динамічної зміни розташування у відповідь на змінні вимоги, що вимагає розробки та застосування міграційних алгоритмів. У статті проаналізовано важливість використання міграції контейнерів у контексті задачі розміщення контейнерів у хмарних центрах обробки даних. Метою дослідження є підвищення продуктивності хмарного ЦОД у задачі розміщення контейнерів, шляхом використання алгоритмів міграції. Використання міграції дозволяє уникнути перевантаження або недовантаження серверів та віртуальних машин. В рамках дослідження було розроблено гібридний алгоритм міграції, що включає в себе як міграцію контейнерів, так і віртуальних машин. Описано математичні аспекти задачі розміщення контейнерів та описано алгоритм міграцій. Експерименти показали, що впровадження цього алгоритму дозволяє знизити операційні витрати на 24% порівняно з традиційними методами, які не використовують міграцію, і на 9% порівняно з менш гнучкими міграційними стратегіями. Також продемонстровано зниження часу порушень SLA на 40%, що свідчить про покращення якості обслуговування. Висновки цього дослідження підтверджують, що міграція контейнерів та віртуальних машин є необхідною для ефективного управління динамічними ресурсами у хмарних ЦОД у контексті задачі розміщення контейнерів.*

*Ключові слова: контейнерна віртуалізація, міграція, хмарний провайдер, ЦОД, віртуальна машина, оптимізація, метод управління, алгоритми.*

### Постановка проблеми

У сучасному світі інформаційних технологій хмарні обчислення набирають все більшої популярності, оскільки вони пропонують значні переваги у гнучкості, масштабованості та ефективності. Це технологія дозволяє організаціям зменшити витрати на обладнання, енергію та обслуговування, адже фізичні дані та ресурси зберігаються у віртуальних хмарах, якими можна керувати віддалено. Зі зростанням об'ємів даних і складності обчислювальних завдань, хмарні рішення пропонують значну гнучкість у розподілі ресурсів, дозволяючи користувачам оптимізувати процеси без необхідності інвестицій у дороге обладнання.

Паралельно з ростом популярності хмарних обчислень, технології контейнерної віртуалізації також зазнають значного поширення. Контейнери надають чимало переваг порівняно з традиційними віртуальними машинами, зокрема завдяки їх легкій вазі та швидкодії [1, 2].

Перед хмарними провайдерами постає задача розміщення контейнерів – надати клієнтам можливість за запитом розмістити заданий контейнер. Контейнери, в свою чергу, розміщуються на віртуальних машинах [3].

Тому, хмарний провайдер має необхідність у ефективному вирішенні задачі розміщення: віртуальних машин (VM) на фізичних серверах (ФС), та контейнерів на VM задля зменшення витрат, а саме зменшення витрат на електроенергію та порушення угод про надання послуг (SLA) [4,5].

Задача розміщення розглядається як задача багатовимірного пукування, де необхідно розмістити ресурси так, щоб була використана найбільша наявна кількість ресурсів.

Така задача широко описана науковою спільнотою і вже наявно багато рішень ефективного початкового розміщення контейнерів. Проте, робота хмарного ЦОД не обмежується лише початковим розміщенням контейнерів, адже, у реальному світі надходять запити як на розміщення, так і на видалення контейнерів. В результаті чого, деякі VM або ФС можуть бути перевантажені, або недовантажені, що спричинить неефективне використання ресурсів. У таких випадках необхідно використовувати механізм міграції.

У даній статі розглядаються аспекти міграції контейнерів і VM у задачі розміщення контейнерів. Запропоновано алгоритм, який покращує використання ресурсів.

#### **Аналіз останніх досліджень і публікацій**

У роботі [6] автори описують гібридний підхід до консолідації віртуальних машини, який спрямований на мінімізацію споживання енергії та порушення SLA шляхом застосування модифікованої евристики зменшення найкращої відповідності до початкового розміщення віртуальної машини (VM) і застосування алгоритму beam search для керування міграціями VM. У даній роботі описано і доведено необхідність застосування міграцій при вирішенні задачі розміщення віртуальних машин, проте, проблема розміщення і міграції контейнерів не розглядається.

Автори [7-10] розглядають задачу міграції віртуальних машин у хмарному середовищі. Автори використовують як статичні, так і динамічні методи міграції. Представлено гібридні алгоритм оптимізації для обробки міграції VM. Запропоновані методи міграції націлені на зменшення споживання енергії, часу обчислень і вартості міграції, а також, на максимальне використання ресурсів. Доведено, що використання міграцій зменшує операційні витрати. Проте, в даних роботах задача розміщення та міграції контейнерів не розглядається. Задача розміщення контейнерів включає в себе задачу розміщення віртуальних машин, тому, використання вже існуючих методів є доцільним.

У роботі [11], автори аналізують та порівнюють існуючі алгоритми розміщення контейнерів, використовуючи інструмент Cloudsim [12]. Основна увага зосереджена на оцінці середнього споживання енергії, кількості активних віртуальних машин та фізичних машин. Виявлено, що алгоритм розміщення контейнерів за методом First Fit Decreasing (FFD) перевершує інші алгоритми за ефективністю використання ресурсів. Розглядається задача лише початкового розміщення контейнерів.

У статті [13], вивчається міграція в реальному часі для обох типів віртуалізації: VM та контейнерів. Результати показують, що використання міграції контейнерів, якщо це можливо, замість міграції VM, знижує споживання енергії, скорочує час міграції, що впливає на QoS та знижує порушення SLA. Проте, використовується лише статичний трешхолд на базі RAM, хоча ресурсна ємність описується, в тому числі і часом процесора, дисковим простором та кількістю операцій вводу-виводу в секунду.

Автори [14] запропонували метод міграції на основі контейнерів і порівняли продуктивність з існуючою схемою, на базі міграцій лише VM. Результати показують, що порівняно з існуючою схемою міграції, запропонована методика міграції контейнерів скорочує час простою та час міграції. Проте, міграція лише контейнерів, без міграції VM не дозволить ефективно управляти ресурсами, адже, можлива ситуація перевантаженості/недовантаженості фізичного сервера.

### Мета дослідження

Метою дослідження є підвищення продуктивності хмарного ЦОД у задачі розміщення контейнерів, шляхом використання алгоритмів міграції контейнерів. Завдяки використанні міграцій зменшити використання електроенергії у хмарному ЦОД, та зменшити час порушення SLA. Така мета зумовлена необхідністю хмарному провайдеру використовувати свої ресурси ефективно задля надання якіснішого сервісу, та утримання конкурентної ціни на свої послуги.

### Викладення основного матеріалу дослідження

Задача розміщення контейнерів можна представити як задачу багатовимірної пакування, де ФС, VM та контейнер мають ресурсну ємність, описану рівняннями 1-3

$$PM_i = (PM_i^{CPU}, PM_i^{RAM}, PM_i^{DISK}, PM_i^{IOPS}), \#(1)$$

$$VM_i = (VM_i^{CPU}, VM_i^{RAM}, VM_i^{DISK}, VM_i^{IOPS}), \#(2)$$

$$C_i = (C_i^{CPU}, C_i^{RAM}, C_i^{DISK}, C_i^{IOPS}), \#(3)$$

На рисунку 1 схематично зображено розміщення VM на ФС та контейнерів на VM, так, що, кожен з компонентів має свою ресурсну ємність.

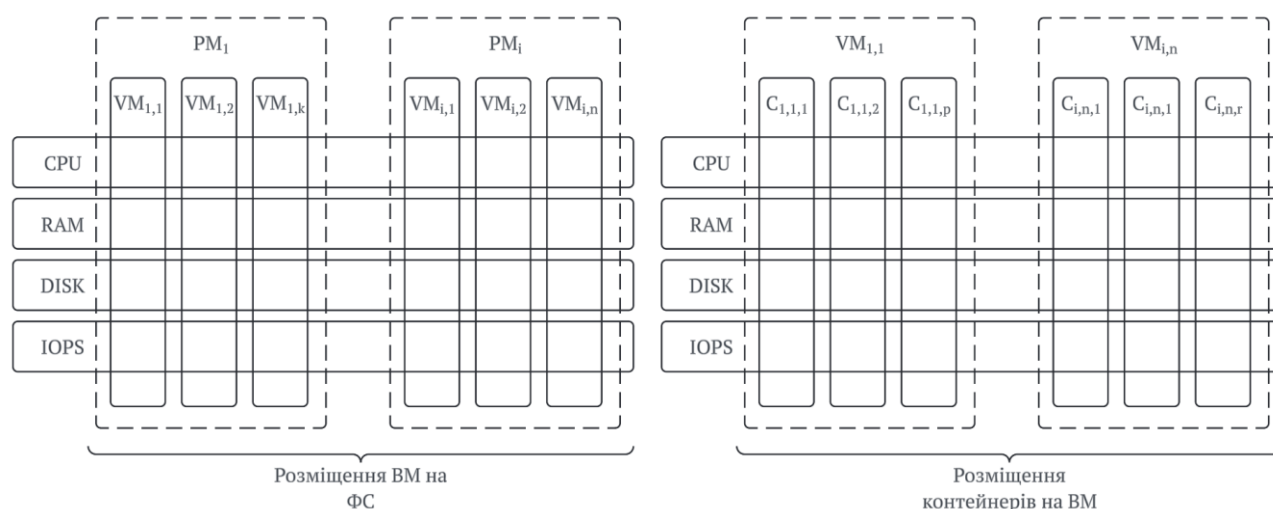


Рисунок 1 – Схематичне зображення розміщення VM на ФС та контейнерів на VM

Для кожного кроку управління ЦОД, існують матриці РМА (4), елементами якої є цілочисельні  $pma_{ij}(t) \in \{0,1\}$ , які вказують, чи  $j$ -та ВМ з множини  $VM$  розташована на  $i$ -му ФС, де  $PM = \{PM_i | i \in \{1, 2, \dots, n\}\}$  – множина фізичних серверів та  $VM = \{VM_i | i \in \{1, 2, \dots, m\}\}$  – множина віртуальних машин.

$$PMA(t) = \begin{pmatrix} pma_{1,1}(t) & \dots & pma_{1,n}(t) \\ \vdots & \ddots & \vdots \\ pma_{m,1}(t) & \dots & pma_{n,m}(t) \end{pmatrix} \quad \#(4)$$

Також, для віртуальних машин з контейнерами справедлива матриця (5), де  $vma_{ij}(t) \in \{0,1\}$ , які вказують, чи  $j$ -тий контейнер розташована на  $i$ -й ВМ

$$VMA(t) = \begin{pmatrix} vma_{1,1}(t) & \dots & vma_{1,n}(t) \\ \vdots & \ddots & \vdots \\ vma_{m,1}(t) & \dots & vma_{n,m}(t) \end{pmatrix} \quad \#(5)$$

Віртуальна машина може знаходитися в одному зі станів: недовантажена, нормальний стан, перевантажена.  $VMO_i(t) \in [0,1]$  – якщо в момент часу  $t$ ,  $i$ -та віртуальна машина недовантажена, то  $VMO_i(t) \in [0,0.2)$ , якщо перевантажена, то  $VMO_i(t) \in [0.8, 1]$ , інакше –  $VMO_i(t) \in [0.2, 0.8)$ .

Аналогічно справедливо і для фізичних серверів:  $PMO_i(t) \in [0,1]$ : якщо  $i$ -й фізичний сервер недовантажений, то  $PMO_i(t) \in [0,0.2)$ , якщо перевантажений, то  $PMO_i(t) \in [0.8, 1]$ , інакше –  $PMO_i(t) \in [0.2, 0.8)$ .

Нехай,  $r$  функція, яка визначає залишок ресурсу на хості. Тоді,  

$$VMO_i(t) = \frac{\frac{r(VM_i^{CPU})}{VM_i^{CPU}} + \frac{r(VM_i^{RAM})}{VM_i^{RAM}} + \frac{r(VM_i^{DISK})}{VM_i^{DISK}} + \frac{r(VM_i^{IOPS})}{VM_i^{IOPS}}}{4}$$
. Аналогічно і для  $PMO_i(t)$ .

Величини  $VMO$  та  $PMO$  приймають діапазон значень, що познає ступінь перевантаженості, чи недовантаженості задля можливості пошуку ФС та ВМ, з яких потрібно мігрувати.

Для кожного кроку управління ЦОД виконуються послідовні дії: розмістити контейнер та запустити процес міграції. Розміщення контейнера полягає в виборі ВМ, вільна ресурсна ємність якої більша за ресурсну ємність контейнера та розміщення контейнера на цій ВМ. За умови, що доступної ВМ не існує – створити нову та розмістити її на ФС. Якщо включеного ФС у ЦОД немає, то включити ФС, і розмістити на ньому ВМ. За умови, якщо ФС не знайдено – відмовити у наданні послуги.

Гібридний алгоритм міграції полягає у виконанні наступних кроків:

- Крок-1: Визначити  $VMO_i(t)$  для кожної  $VM_i$  із множини  $VM$ . Сортування за зростанням  $VMO_i(t)$ .

- Крок-2: Визначити всі  $VMO_i(t) < 0.2$ , як множину  $GU_{VM}$

- Крок-3: В кожній VM з  $GU_{VM}$ , для кожного контейнера в віртуальній машині – мігрувати контейнер на нову віртуальну машину так, щоб після міграції  $VMO_i(t) < 0.8$ , де  $i$  – номер віртуальної машини, на яку мігровано контейнер.
- Крок-4: видалити всі VM з  $GU_{VM}$
- Крок-5: Визначити всі  $VMO_i(t) \geq 0.8$ , як множину  $GO_{VM}$
- Крок-6: Визначити множину контейнерів  $GO_{VM}$ , змігрувавши які, залишковий  $VMO_i(t) < 0.8$ . Мігрувати  $GO_{VM}$  так, щоб після міграції  $VMO_i(t) < 0.8$ , де  $i$  – номер віртуальної машини, на яку мігровано контейнер.

Після того, як виконано всі міграції контейнерів, виконати аналогічні кроки (Крок-1 – Крок-6) по міграції віртуальних машин між фізичними серверами. Таким чином, після міграцій не буде залишатись перенавантажених, або недовантажених VM або ФС.

Таким чином, використовуючи гібридний алгоритм, зменшені операційні витрати на підтримку ЦОД, адже, в кінцевому виді, буде використано менше ресурсів, ніж якби міграцій не було.

Задля підтвердження ефективності запропонованого гібридного алгоритму, проведено серію із 10 експериментів, використовуючи розроблений інструментарій для симуляції розміщення контейнерів у хмарному ЦОД. Експерименти проведено у гетерогенній хмарі з двома типами фізичних серверів із ресурсною ємністю: CPU (3, 24) ядер, RAM (16384, 131072) мБ, DISK (256000, 1024000) б, IOPS (40000, 60000) та трьома типами віртуальних машин: CPU (1.5, 8, 24) ядер, RAM (8192, 43690, 65536) мБ, DISK (128000, 341333, 512000) б та IOPS (15000, 20000, 30000). Для початкового розміщення використовується алгоритм First Fit. Визначені ціни на ресурсну ємність (в умовних одиницях): CPU – 0.01 (за 1 секунду одного ядра), RAM - 0.0001, DISK - 0.00001, IOPS - 0.000001 (за 1 мБ); ціна порушення SLA – 0.01 за секунду.

На рисунку 2 зображено графіки залежності кумулятивної ціни експерименту та часу порушення SLA, в залежності від алгоритму.

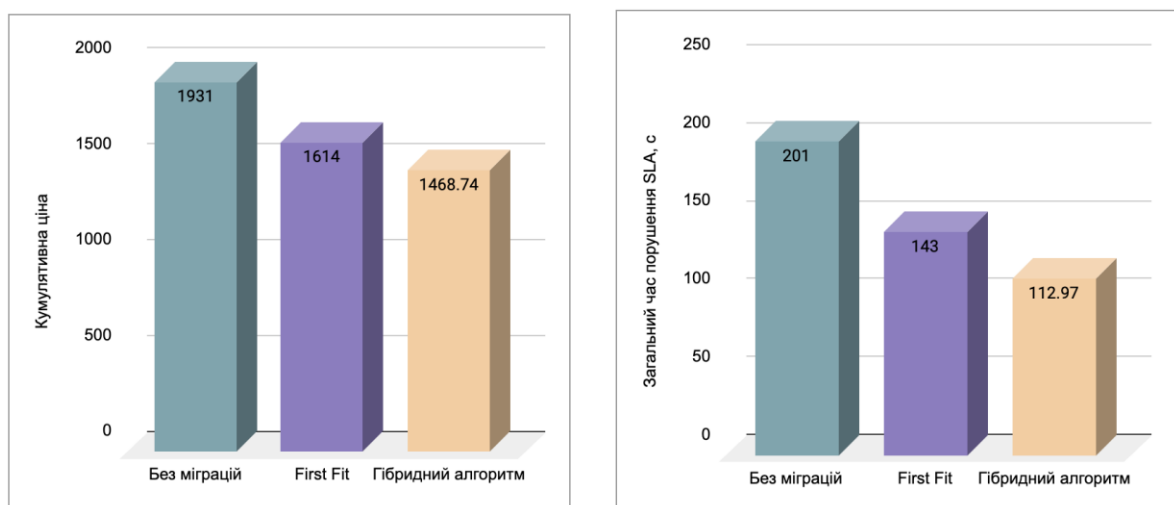


Рисунок 2 – Графіки залежності кумулятивної ціни експерименту та часу порушення SLA, в залежності від алгоритму

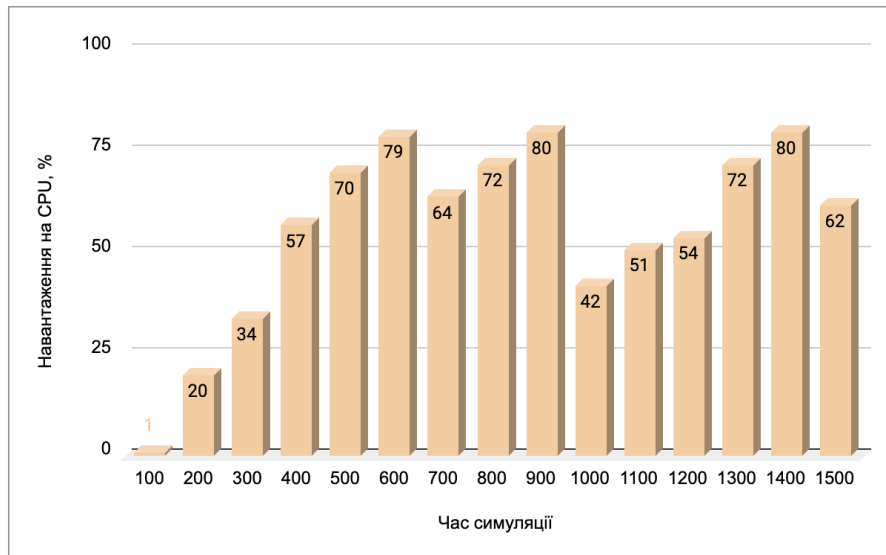


Рисунок 3 – Навантаження на CPU однієї з VM, в залежності від часу (гібридний алгоритм)

Видно, що запропонований гібридний алгоритм дозволяє зменшити кумулятивну ціну хмарного ЦОД на 24%, порівняно із алгоритмом, де міграції не використовуються, та на 9%, порівняно із міграціями за алгоритмом First Fit із статичним трешхолдом у 80% завантаженості з міграціями тільки контейнерів. Також, час порушення SLA зменшився на 40% та 21% відповідно.

На рисунку 3 зображено навантаження на CPU однієї з VM, в залежності від часу симуляції. З графіку видно, що коли навантаження наближалось до граничного, відбувались міграції, і поточне навантаження знижувалось. Завдяки цьому ресурси кластеру використовувались більш ефективно, що призвело до зниження кумулятивної ціни та часу порушення SLA.

### Висновки

У даній статті розглянуто важливість використання міграцій у процесах розміщення контейнерів у хмарних ЦОД. Дослідження підкреслює, що через динамічність обчислювальних потреб і зміни в робочих навантаженнях, систематичні міграції є необхідними для оптимального розподілу ресурсів і забезпечення високої доступності сервісів. Міграції допомагають уникнути перевантажень та недовантажень серверів, знижуючи таким чином ризики порушення угод про рівень обслуговування (SLA) та забезпечуючи більш ефективне використання енергії.

Було розроблено гібридний алгоритм міграцій з метою зниження витрат та підвищення продуктивності хмарних систем. Завдяки впровадженню цього алгоритму, вдалося значно знизити операційні витрати — на 24% порівняно з методами, які не використовують міграцію, і на 9% порівняно з підходом, коли в міграції беруть участь лише контейнери.

Крім того, гібридний алгоритм сприяв значному зниженню часу порушення SLA (на 40%), що демонструє його ефективність у забезпеченні більш стабільної та надійної роботи хмарних сервісів. Ці результати підкреслюють значущість міграції як критичної складової сучасних хмарних обчислень, забезпечуючи адаптацію до змінюваних вимог і оптимальне використання ресурсів.

Загалом, результати дослідження вказують на значний потенціал міграційних алгоритмів у підвищенні ефективності хмарних обчислень, дозволяючи ефективніше реагувати на змінні вимоги до ресурсів та оптимізувати використання доступних потужностей.

#### ЛІТЕРАТУРА

1. Katal A., Choudhury T., Dahiya S. Comparison and Analysis of Container Placement Algorithms in Cloud Data Center. *Emerging Trends in Expert Applications and Security*. Singapore, 2023. С. 239–252. URL: [https://doi.org/10.1007/978-981-99-1946-8\\_22](https://doi.org/10.1007/978-981-99-1946-8_22).
2. Sturm R., Pollard C., Craig J. Managing Containerized Applications. *Application Performance Management (APM) in the Digital Enterprise*. 2017. С. 177–185. URL: <https://doi.org/10.1016/b978-0-12-804018-8.00013-9>.
3. Silva V. G. d., Kirikova M., Alksnis G. Containers for Virtualization: An Overview. *Applied Computer Systems*. 2018. Т. 23, № 1. С. 21–27. URL: <https://doi.org/10.2478/acss-2018-0003>.
4. Resource scheduling for infrastructure as a service (IaaS) in cloud computing: Challenges and opportunities / S. H. H. Madni та ін. *Journal of Network and Computer Applications*. 2016. Т. 68. С. 173–200. URL: <https://doi.org/10.1016/j.jnca.2016.04.016>.
5. Energy efficient scheduling of virtual machines in cloud with deadline constraint / Y. Ding та ін. *Future Generation Computer Systems*. 2015. Т. 50. С. 62–74. URL: <https://doi.org/10.1016/j.future.2015.02.001>.
6. Cloud Resource Management with a Hybrid Virtual Machine Consolidation Approach / E. Zharikov та ін. 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT), м. Kyiv, Ukraine, 18–20 груд. 2019 р. 2019. URL: <https://doi.org/10.1109/atit49449.2019.9030459>.
7. Khan M. S. A., Santhosh R. Hybrid Optimization Algorithm for VM Migration in Cloud Computing. *Computers and Electrical Engineering*. 2022. Т. 102. С. 108152. URL: <https://doi.org/10.1016/j.compeleceng.2022.108152>.
8. Migration-Based Load Balance of Virtual Machine Servers in Cloud Computing by Load Prediction Using Genetic-Based Methods / L.-H. Hung та ін. *IEEE Access*. 2021. Т. 9. С. 49760–49773. URL: <https://doi.org/10.1109/access.2021.3065170>.
9. Najm M., Tamarapalli V. Towards cost-aware VM migration to maximize the profit in federated clouds. *Future Generation Computer Systems*. 2022. Т. 134. С. 53–65. URL: <https://doi.org/10.1016/j.future.2022.03.020>.
10. Rani J. K., Lakshmi M. S. Cloud Computing Challenges and Concerts in VM Migration. *International Conference on Mobile Computing and Sustainable Informatics*. Cham, 2020. С. 135–142. URL: [https://doi.org/10.1007/978-3-030-49795-8\\_12](https://doi.org/10.1007/978-3-030-49795-8_12).
11. Katal A., Choudhury T., Dahiya S. Comparison and Analysis of Container Placement Algorithms in Cloud Data Center. *Emerging Trends in Expert Applications and Security*. Singapore, 2023. С. 239–252. URL: [https://doi.org/10.1007/978-981-99-1946-8\\_22](https://doi.org/10.1007/978-981-99-1946-8_22).
12. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms / R. N. Calheiros та ін. *Software: Practice and Experience*. 2010. Т. 41, № 1. С. 23–50. URL: <https://doi.org/10.1002/spe.995>.

13. Performance Evaluation of Virtual Machine and Container-Based Migration Technique / A. Bhardwaj та ін. *Proceedings of Data Analytics and Management*. Singapore, 2024. С. 551–558. URL: [https://doi.org/10.1007/978-981-99-6544-1\\_41](https://doi.org/10.1007/978-981-99-6544-1_41).
14. Smimite O., Afdel K. Containers Placement and Migration on Cloud System. *International Journal of Computer Applications*. 2020. T. 176, № 35. С. 9–18. URL: <https://doi.org/10.5120/ijca2020920493>.

#### REFERENCES

1. Katal A., Choudhury T., Dahiya S. Comparison and Analysis of Container Placement Algorithms in Cloud Data Center. *Emerging Trends in Expert Applications and Security*. Singapore, 2023. P. 239–252. URL: [https://doi.org/10.1007/978-981-99-1946-8\\_22](https://doi.org/10.1007/978-981-99-1946-8_22).
2. Sturm R., Pollard C., Craig J. Managing Containerized Applications. *Application Performance Management (APM) in the Digital Enterprise*. 2017. P. 177–185. URL: <https://doi.org/10.1016/b978-0-12-804018-8.00013-9>.
3. Silva V. G. d., Kirikova M., Alksnis G. Containers for Virtualization: An Overview. *Applied Computer Systems*. 2018. Vol. 23, no. 1. P. 21–27. URL: <https://doi.org/10.2478/acss-2018-0003>.
4. Resource scheduling for infrastructure as a service (IaaS) in cloud computing: Challenges and opportunities / S. H. H. Madni et al. *Journal of Network and Computer Applications*. 2016. Vol. 68. P. 173–200. URL: <https://doi.org/10.1016/j.jnca.2016.04.016>.
5. Energy efficient scheduling of virtual machines in cloud with deadline constraint / Y. Ding et al. *Future Generation Computer Systems*. 2015. Vol. 50. P. 62–74. URL: <https://doi.org/10.1016/j.future.2015.02.001>.
6. Cloud Resource Management with a Hybrid Virtual Machine Consolidation Approach / E. Zharikov et al. *2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT)*, Kyiv, Ukraine, 18–20 December 2019. 2019. URL: <https://doi.org/10.1109/atit49449.2019.9030459>.
7. Khan M. S. A., Santhosh R. Hybrid Optimization Algorithm for VM Migration in Cloud Computing. *Computers and Electrical Engineering*. 2022. Vol. 102. P. 108152. URL: <https://doi.org/10.1016/j.compeleceng.2022.108152>.
8. Migration-Based Load Balance of Virtual Machine Servers in Cloud Computing by Load Prediction Using Genetic-Based Methods / L.-H. Hung et al. *IEEE Access*. 2021. Vol. 9. P. 49760–49773. URL: <https://doi.org/10.1109/access.2021.3065170>.
9. Najm M., Tamarapalli V. Towards cost-aware VM migration to maximize the profit in federated clouds. *Future Generation Computer Systems*. 2022. Vol. 134. P. 53–65. URL: <https://doi.org/10.1016/j.future.2022.03.020>.
10. Rani J. K., Lakshmi M. S. Cloud Computing Challenges and Concerts in VM Migration. *International Conference on Mobile Computing and Sustainable Informatics*. Cham, 2020. P. 135–142. URL: [https://doi.org/10.1007/978-3-030-49795-8\\_12](https://doi.org/10.1007/978-3-030-49795-8_12).
11. Katal A., Choudhury T., Dahiya S. Comparison and Analysis of Container Placement Algorithms in Cloud Data Center. *Emerging Trends in Expert Applications and Security*. Singapore, 2023. P. 239–252. URL: [https://doi.org/10.1007/978-981-99-1946-8\\_22](https://doi.org/10.1007/978-981-99-1946-8_22).



12. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms / R. N. Calheiros et al. *Software: Practice and Experience*. 2010. Vol. 41, no. 1. P. 23–50. URL: <https://doi.org/10.1002/spe.995>

13. Smimite O., Afdel K. Containers Placement and Migration on Cloud System. *International Journal of Computer Applications*. 2020. Vol. 176, no. 35. P. 9–18. URL: <https://doi.org/10.5120/ijca2020920493>

Performance Evaluation of Virtual Machine and Container-Based Migration Technique / A. Bhardwaj et al. *Proceedings of Data Analytics and Management*. Singapore, 2024. P. 551–558. URL: [https://doi.org/10.1007/978-981-99-6544-1\\_41](https://doi.org/10.1007/978-981-99-6544-1_41)

Received 23.06.2024.  
Accepted 27.06.2024.

### ***An approach to migrations in the container placement task***

*The article examines the importance of container migration within the context of container placement tasks in cloud data centers. Cloud computing offers organizations significant benefits such as flexibility, scalability, and reduced costs for equipment, energy, and maintenance. Cloud providers face the challenge of not only efficiently placing containers initially but also dynamically reallocating them, along with virtual machines, in response to changing demands and workloads. This necessitates the development and application of migration algorithms. The study analyzes how migration is a critical component for ensuring optimal resource use, maintaining high service availability, and reducing service level agreement (SLA) violations. The research aims to enhance the productivity of cloud data centers in the task of container placement by utilizing container migration algorithms. This objective is driven by the need for cloud providers to use their resources effectively to provide higher quality service and maintain competitive pricing. The use of migrations prevents overloading or underloading of servers and virtual machines, which is especially important in the highly dynamic environment of cloud computing. The research developed a hybrid migration algorithm that includes the migration of both containers and virtual machines. Mathematical aspects of the container placement task are described, and the migration algorithm is detailed. Experiments showed that implementing this algorithm reduces operational costs by 24% compared to traditional methods that do not use migration, and by 9% compared to less flexible migration strategies. There was also a demonstrated reduction in SLA violation times by 40%, indicating improvements in service quality and reliability of cloud services. The conclusions of this study confirm that container and virtual machine migration is necessary for effective management of dynamic resources in cloud data centers in the context of container placement tasks.*

*Keywords: container virtualization, migration, cloud provider, data center, virtual machine, optimization, control method, algorithm.*

**Сопов Олексій Олександрович** – аспірант кафедри інформатики та програмної інженерії, асистент, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"

**Жаріков Едуард В'ячеславович** – завідувач кафедри інформатики та програмної інженерії, д.т.н, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського"

**Oleksii Sopov** – PhD student of the Department of Computer Science and Software Engineering, assistant, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

**Zharikov Eduard** – Head of the Department of Computer Science and Software Engineering, Doctor of Technical Sciences, Professor, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"