

РОЗДІЛ V. ІНФОРМАЦІЙНО-КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

УДК 004.94

В.В. Литвинов, д-р техн. наук, професор

І.В. Стеценко, канд. техн. наук, доцент

Інститут проблем математичних машин і систем, м. Київ, Україна

Черкаський державний технологічний університет, м. Черкаси, Україна

МОДЕЛЬ СИСТЕМИ УПРАВЛІННЯ РОЗПОДІЛЕНИМИ ОБЧИСЛЮВАЛЬНИМИ РЕСУРСАМИ

Запропоновано використання Петрі-об'єктної технології моделювання для побудови імітаційної моделі системи управління розподіленими обчислювальними ресурсами. Побудовано модель розподілу обчислювальних ресурсів однорівневої грід-системи та здійснено дослідження ефективності вибраної стратегії управління.

Постановка проблеми. На сьогодні розроблені ефективні програмні та інструментальні засоби створення грід-систем. Найбільш відомими відкритими програмними продуктами, що забезпечують стандартний набір бібліотек для інтеграції користувачів, є Globus Toolkit та Sun Grid Engine. Комерційна версія останнього – Sun Grid Engine, Enterprise Edition, – призначена для управління ресурсами підприємств і спроможна обслуговувати кілька незалежних проектів та груп користувачів.

Основним питанням, що залишається недостатньо дослідженим, є алгоритми розподілу обчислювальних та інформаційних ресурсів, які знаходяться у спільному використанні користувачів грід-системи. Користувачі прагнуть отримувати бажані ресурси з найменшим часом очікування, а проектувальники систем управління забезпечують насамперед несуперечність прав доступу до ресурсів грід-системи, відсутність конфліктних ситуацій, відсутність відмов обслуговування.

Моделі статичного розподілу ресурсів не задовольняють потреби обчислювальних систем, тому є необхідність розробки моделі динамічного розподілу ресурсів.

Аналіз досліджень і публікацій. Огляд публікацій [1], [2], [3], [4] що розглядають розподіл ресурсів грід-систем, підтверджує наявність різноманітних підходів щодо розв'язання проблеми, а також про необхідність розробки ефективних способів моделювання різних стратегій управління ресурсами розподілених систем. Існуючі системи моделювання грід-систем, такі як MicroGrid, OptorSim, GridSim, SimGrid, мають ряд недоліків, пов'язаних з вузькою спеціалізацією, обмеженістю модельованої архітектури грід-систем, недостатньою гнучкістю у визначенні способу управління [5].

Мета дослідження. У цій публікації наведені результати моделювання системи управління розподіленими обчислювальними ресурсами з метою дослідження її ефективності та вибору параметрів управління. Використання Петрі-об'єктного підходу [6] дозволило розробити та дослідити модель динамічного розподілу обчислювальних ресурсів грід-системи.

Модель системи управління з блокуванням захоплення ресурсу. Кластерні системи можуть використовуватись для масиву послідовних програм, або розподілених програм, що використовують спільні дані, або паралельних програм. У роботі розглянуто [7] цей огляд архітектури кластерних систем та наведено аналіз програмних пакетів управління потоками завдань таких систем. Система управління кластера розподіляє обчислювальні ресурси кластера між завданнями, що надходять від користувачів. Система має черги завдань користувачів та виділяє необхідний обчислювальний ресурс для виконання завдання таким чином, щоб максимально задовольнити потреби користувачів.

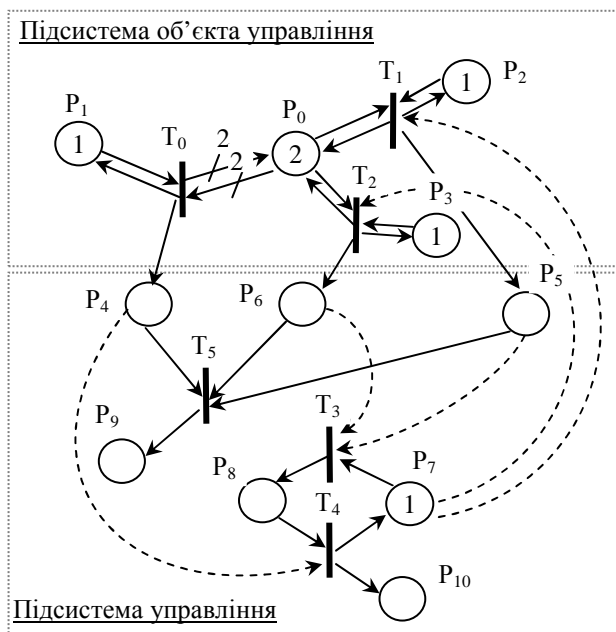


Рис. 1. Модель системи управління розподілом обчислювальних ресурсів, що використовує блокування захоплення ресурсу

нерівність споживання ресурсів призводить до того, що при відсутності процесів управління завдання з меншою потребою ресурсу використовують ресурс у повній мірі, а завдання з більшою потребою знаходяться у нескінченному очікуванні вивільнення необхідного обсягу ресурсу. Завдання конфліктують за захоплення ресурсу настільки, що, наприклад, при часових затримках у переходах T_0 , T_1 , T_2 в середньому 1, 0,157 і 0,333 відповідно, завдання користувачів B і C , один раз захопивши ресурс і по черзі його вивільняючи, не дають завданням користувача A ніякої можливості зайняти ресурс. Присвоєння пріоритету переходу T_0 призводить до зайняття ресурсу виключно завданнями користувача A , отже не вирішує проблему розподілу ресурсів між завданнями. Присвоєння ймовірності запуску переходу T_0 також не вирішує проблему, оскільки не виникає конфлікт переходу T_0 з іншими переходами.

Введемо динамічне управління розподілом ресурсів, що здійснюється блокуванням можливості захоплення ресурсів на основі інформації про кількість виконаних завдань. Порівняння кількості виконаних завдань здійснюється переходом T_5 так, що в позиціях P_4 , P_5 , P_6 міститься інформація про різницю кількості виконаних завдань користувачів A , B , C у порівнянні з іншими завданнями. При більшій кількості виконаних завдань користувачів B і C приймається рішення про блокування виконання цих завдань (перехід T_3), а при більшій кількості виконаних завдань користувача A – рішення про зняття цього блокування (перехід T_4).

Дослідження імітаційної моделі системи управління дозволяє оцінити ефективність функціонування системи при різних значеннях параметрів управління. Введемо параметри управління, що відповідають умовам, за яких приймається рішення про блокування:

$$a = w_{5,3} = w_{5,5}, \quad b = w_{6,3} = w_{6,5}, \quad c = w_{4,4} = w_{4,5} = 1,$$

де $w_{i,j}$ – вага дуги, що з'єднує позицію P_i і перехід T_j .

Результати експериментів наведені на рис. 2. За результатами виходить, що $a = 1$, $b = 2$ є найкращою комбінацією параметрів з точки зору найбільшої загальної

Розглянемо найпростіший приклад розподілення обчислювальних ресурсів між трьома користувачами (рис.1). Завдання користувача A потребують використання всіх обчислювальних ресурсів, а завдання інших користувачів можуть виконуватись одночасно. Модель системи представимо мережею Петрі з часовими затримками, з конфліктними та багатоканальними переходами, з інформаційними зв'язками (рис.1). Позиція P_0 представляє кількість доступного спільного ресурсу, позиції P_1 , P_2 , P_3 представляють буфер завдань A , B , C відповідно. Припускається, що у буфері завдань завжди є завдання для обчислення. Часові затримки в переходах T_0 , T_1 , T_2 відповідають часу виконання завдань користувачів A , B , C . Нерівномір-

кількості виконаних завдань, $a = 2, b = 1$ є найліпшою комбінацією параметрів з точки зору досягнення найменшої різниці у кількості виконаних завдань різних класів, а комбінацією $a = 2, b = 2$ є найоптимальнішою з точки зору найбільшої кількості виконаних завдань класу C.

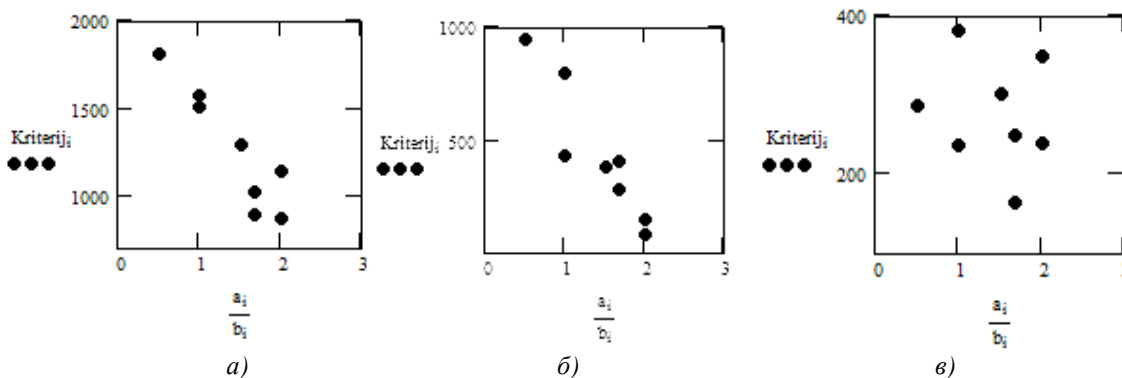


Рис. 2. Результати експериментального дослідження параметрів системи управління розподілом ресурсів при різних критеріях: а) загальна кількість виконаних завдань усіх класів; б) загальна різниця у кількості виконаних завдань усіх класів; в) кількість виконаних завдань класу C

Модель системи управління з виділенням ресурсу планувальником завдань

Розглянемо однорівневу грід-систему, що складається з планувальника системи та розподілених обчислювальних ресурсів. Обчислювальні ресурси представлені комп'ютерами користувачів, а планувальник управляє розподілом ресурсів між користувачами системи так, щоб забезпечувати синхронізований та несуперечливий доступ користувачів грід-системи до спільних обчислювальних ресурсів. Планувальник взаємодіє не з апаратними ресурсами, а з грід-сервісами програмної інфраструктури, що встановлені у розподілених обчислювальних вузлах. Грід-сервіси з певним інтервалом часу, що називають тактом управління, надають планувальнику інформацію про поточний стан обчислювального ресурсу. Введення такту управління є необхідною умовою організації управління грід-систем.

Окрім несуперечливого та безвідмовного доступу до ресурсів, планувальнику ставиться також завдання забезпечення найбільш високої вдовolenості потреб користувачів, що вимірюється такими величинами як середній час очікування завдань, інтенсивність відмов обслуговування завдань користувача, інтенсивність споживання обчислювального ресурсу користувачами та іншими.

Рішення про виділення ресурсу приймається планувальником системи управління на початку кожного такту управління відповідно до заданого алгоритму розподілу обчислювальних ресурсів або відповідно до вибору тієї чи іншої політики розподілу. Програмний пакет Sun Grid Engine, Enterprise Edition (SGEEE), наприклад, надає можливість вибору однієї з чотирьох політик розподілу ресурсів між незалежно працюючими користувачами: політика розділених ресурсів, функціональна політика, політика „роботи до строку”, політика явного виділення ресурсів [9]. Поняття „справедливого” управління розподілом ресурсів пов'язують з таким розподілом ресурсів грід-системи, при якому віртуальний обчислювальний ресурс розподіляється між активними користувачами пропорційно їх внеску у загальний віртуальний обчислювальний ресурс [1]. Активні користувачі отримують доступ до віртуального ОР пропорційно своїй квоті:

$$x_i = \frac{p_i}{\sum_{i \in U} p_i}, \quad \sum_i x_i = 1, \quad (1)$$

де p_i – квота i -ого користувача, що розраховується як частка ресурсу, внесеного користувачем у загальний віртуальний обчислювальний ресурс, x_i – частка доступного

віртуального ресурсу з урахуванням тільки активних користувачів, U – множина активних користувачів.

Модель системи управління грид-ресурсами побудована з застосуванням технології Петрі-об'єктного моделювання [6] і складається з Петрі-об'єктів “Планувальник”, “Користувач”, “Завдання”. Динаміка функціонування Петрі-об'єктів описується стохастичною мережею Петрі з часовими затримками, з багатоканальними та конфліктними переходами, з інформаційними зв'язками (рис. 3, 4, 5).

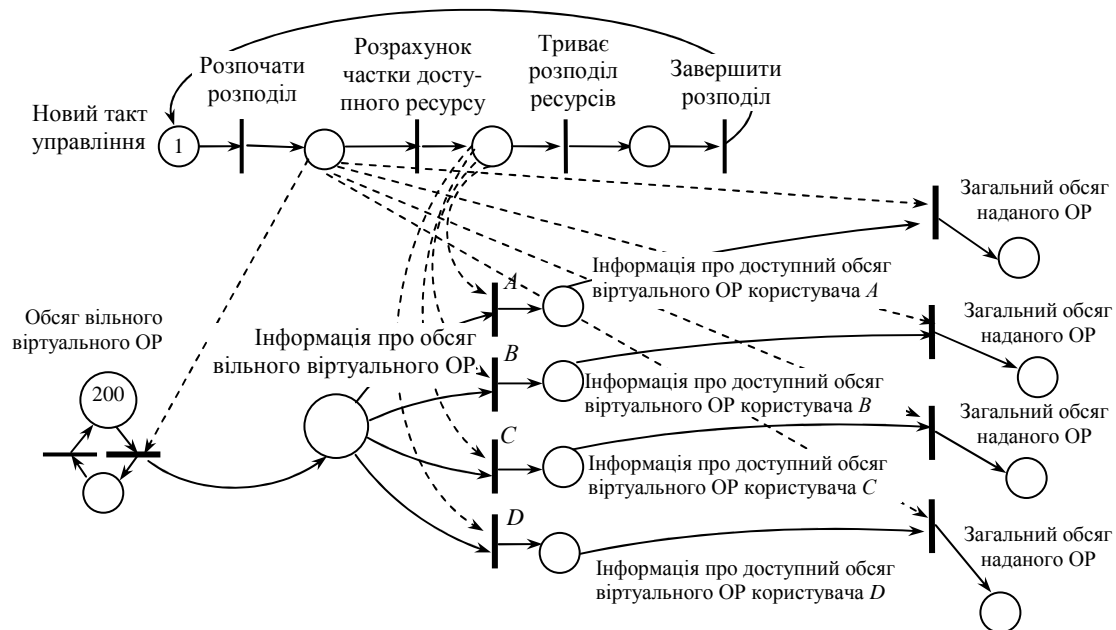


Рис. 3. Мережа Петрі-об'єкта «Планувальник»

Перехід „Розпочати розподіл” здійснюється з часовою затримкою, що дорівнює тривалості такту управління. В результаті запуску переходу „Розрахунок частки доступного ресурсу” Петрі-об'єкта “Планувальник” встановлюються відповідні значення ймовірностей запуску переходів „A”, „B”, „C”, „D”. Вільний віртуальний обчислювальний ресурс, обсяг якого вимірюється в умовних процесорах чи в одиницях MIPS (Million Instruction Per Second), розподіляється випадковим чином між користувачами відповідно до заданих імовірностей.

Буфер завдань користувача може містити одне або кілька завдань. Виконання кожного завдання, що зберігається в буфері завдань, моделюється відповідним Петрі-об'єктом „Завдання”. Якщо наданого планувальником обсягу віртуального обчислювального ресурсу достатньо для виконання завдання, то завдання виконується. В іншому випадку завдання отримує відмову через відсутність достатньої кількості ресурсу в цьому такті управління.

Користувач може знаходитись у пасивному та активному стані, якщо користувач активний, то із заданою інтенсивністю він генерує завдання та потребу завдання в обчислювальному ресурсі. Кожне сформоване завдання відправляється в буфер завдань користувача або отримує відмову, якщо вільного місця в буфері завдань немає. Петрі-об'єкт “Користувач” містить інформацію про поточний стан користувача, кількість місць у буфері завдань, кількість відмов та обсяг невдоволеної потреби користувача.

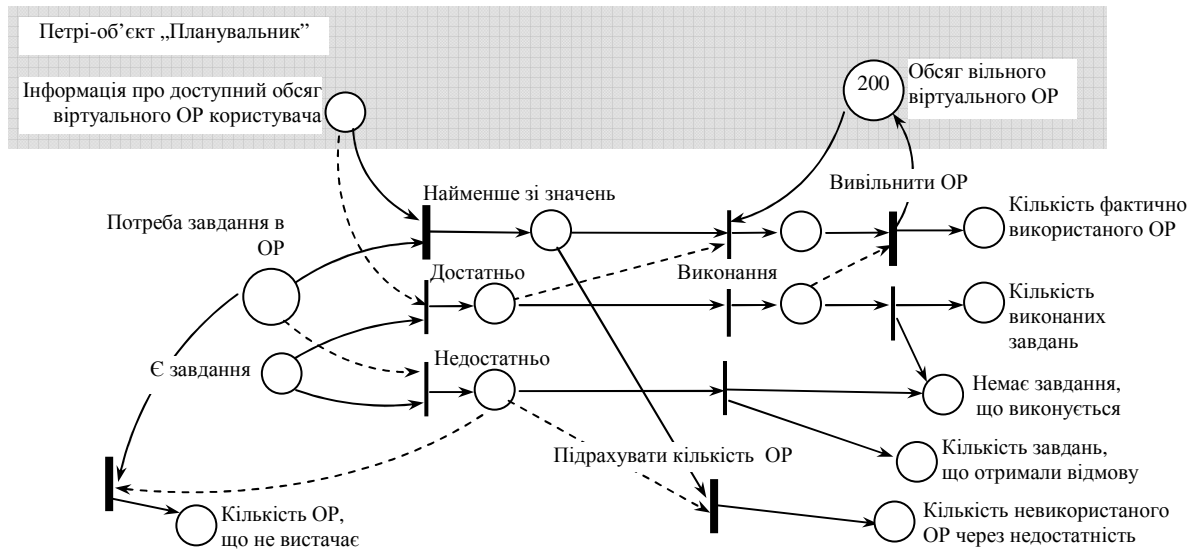


Рис. 4. Мережа Петрі-об'єкта «Завдання»

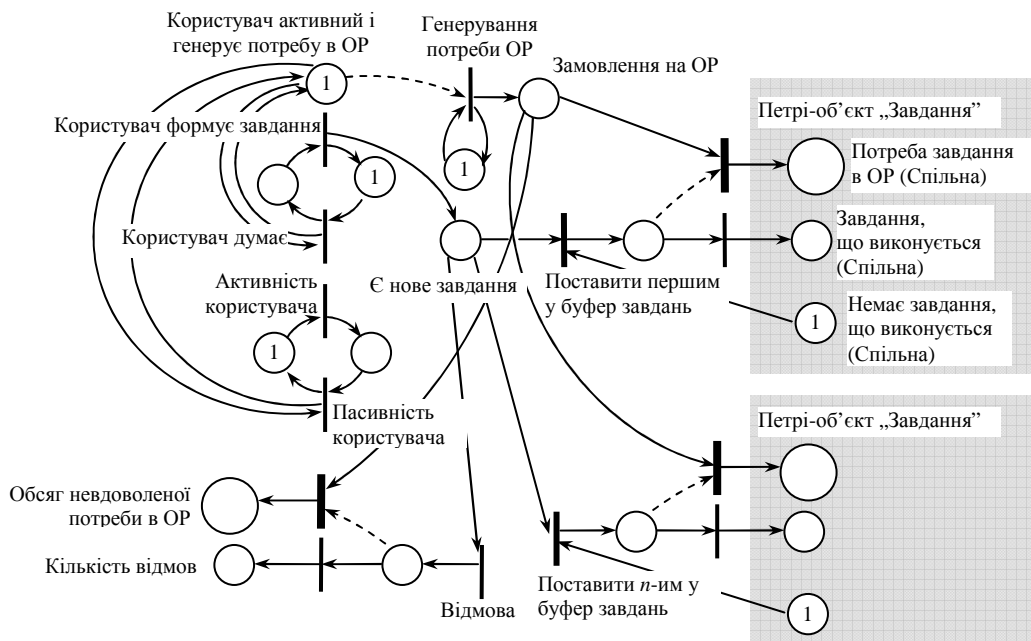


Рис. 5. Мережа Петрі-об'єкта «Користувач»

Результати експериментального дослідження Петрі-об'єктної моделі системи управління грид-ресурсами представлені на рис. 6-9. Серії експериментів проводились з виділенням ресурсу з урахуванням інформації про активних користувачів (формула (1)) та без урахування інформації про активних користувачів ($x_i = p_i = const$), тобто динамічне та статичне управління розподілом ресурсів. Завдання користувачів B, C, D в усіх експериментах надходять з інтенсивністю 1 завдання/4 одиниці часу і потребують у середньому 20 MIPS обчислювального ресурсу, а завдання користувача A надходять з інтенсивністю 1 завдання/8 одиниць часу і потребують у середньому 100 MIPS. Для завдань користувачів B, C, D передбачений буфер завдань місткістю одне завдання, а для завдань користувачів A – місткістю два завдання.

З результатів виходить, що використання динамічного управління є доцільним тільки при достатньо великій кількості ресурсу. Так, для досліджуваної системи управління

поліпшення характеристик функціонування забезпечується при кількості ресурсу користувача *A* більше 50 MIPS та більше.

Значний вплив на характеристики функціонування системи спричиняє тривалість циклу управління. Використання динамічного управління, у порівнянні зі статичним, дозволяє зменшити інтенсивність відмов. Збільшення обсягу ресурсу користувача *A* при різних значеннях тривалості циклу управління сильно впливає на зростання продуктивності системи при незначних обсягах ресурсів інших користувачів. Тривалість циклу управління сильно впливає на зменшення інтенсивності відмов при зростанні обсягу користувача *A* та великих обсягах ресурсів користувачів *B, C, D*.

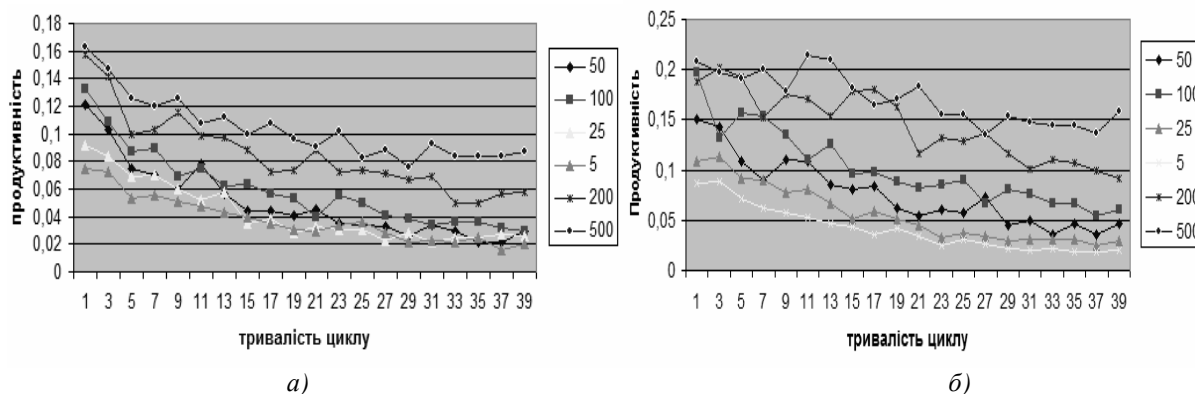


Рис. 6. Продуктивність при змінюванні тривалості циклу управління для різних значень обсягу ресурсу користувача *A* (обсяг ресурсу інших користувачів 5 MIPS): а) статичне управління; б) динамічне управління

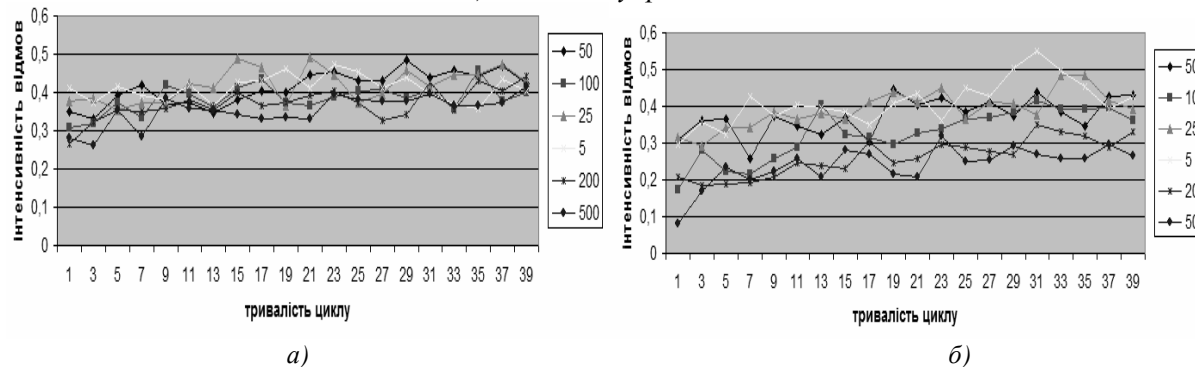


Рис. 7. Інтенсивність відмов при змінюванні тривалості циклу управління для різних значень обсягу ресурсу користувача *A* (обсяг ресурсу інших користувачів 5 MIPS): а) статичне управління; б) динамічне управління

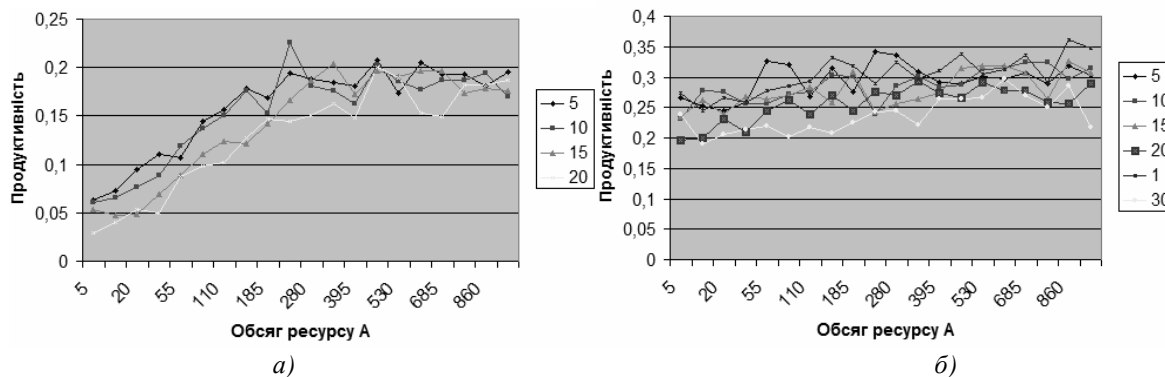


Рис. 8. Продуктивність при змінюванні обсягу ресурсу користувача *A* для різних значень циклу (динамічного) управління: а) обсяг ресурсу інших користувачів 5 MIPS; б) обсяг ресурсу інших користувачів 100 MIPS

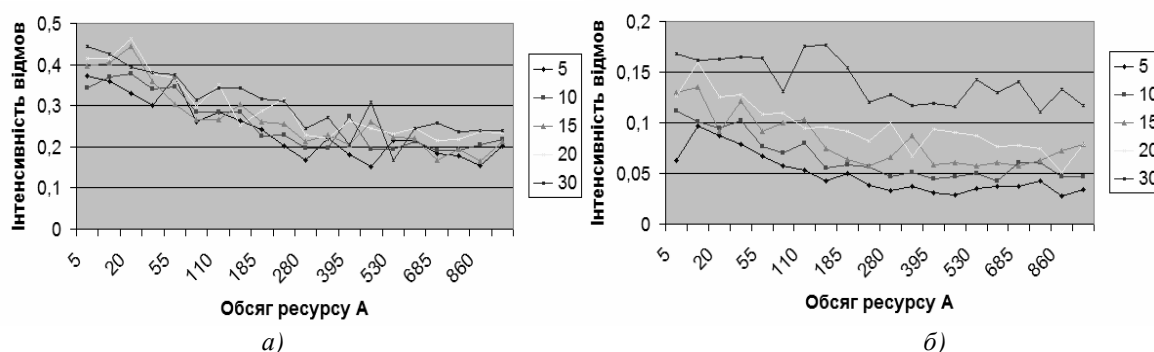


Рис. 9. Інтенсивність відмов при змінюванні обсягу ресурсу користувача А для різних значень циклу (динамічного) управління: а) обсяг ресурсу інших користувачів 5 MIPS; б) обсяг ресурсу інших користувачів 100 MIPS

Таким чином, вибір стратегії управління і параметрів управління спричиняє значний вплив на характеристики функціонування системи. Відтворення системи управління розподіленими обчислювальними ресурсами з використанням стохастичних мереж Петрі з часовими затримками, з багатоканальними та конфліктними переходами дозволяє враховувати найдрібніші деталі процесу функціонування системи управління та об'єкту управління, такі як: динамічне захоплення обчислювального ресурсу, виділення доступного ресурсу користувача в кожному такті управління відповідно до його квоти, індивідуальні властивості користувачів грид-системи.

Висновки. У результаті наукового дослідження розроблені підходи щодо побудови імітаційної моделі системи управління розподіленими обчислювальними ресурсами, що дозволяє враховувати стратегію та параметри управління, індивідуальні якості користувачів, динамічне захоплення віртуального обчислювального ресурсу користувачами системи залежно від обраної стратегії управління. Дослідження імітаційної моделі управління розподіленими обчислювальними ресурсами дозволяє здійснити вибір стратегії управління та параметрів управління. Використання Петрі-об'єктної технології моделювання систем надає можливість конструювання складної грид-системи з довільною кількістю користувачів та завдань, що зберігаються в буфері завдань.

Список використаних джерел

1. Сайт Інститута проблем математики ім. М. В. Келдыша Російської Академії наук Коваленко В. Н., Корягин Д. А. Организация ресурсов грид // Preprint, Inst. Appl. Math., the Russian Academy of Science. – Москва, 2004 [Електронний ресурс]. – Режим доступу: http://www.keldysh.ru/papers/2004/prep63/prep2004_63.html#_Тoc85204258.
2. Шелестов А. Ю. Имитационная модель взаимодействия GRID-узлов с очередью доступа к общей памяти / А. Ю. Шелестов // Автоматика. Автоматизация. Электротехнические комплексы и системы. – Херсон: ХНТУ, 2008. – №1 (21). – С. 32-40.
3. Эвристики распределения задач для брокера ресурсов Grid / А. И. Аветисян, С. С. Гайсарян, Д. А. Грушин [и др.] // Труды Института системного программирования РАН. Том 5. – М., 2004. – С. 269-280.
4. Петренко А. І. Комп'ютерне моделювання грид-систем / А. І. Петренко // Електроніка і зв'язь 5' Тематический выпуск «Електроніка і нанотехнології». – № 5. – К.: Політехніка, 2010. – С. 40-48.
5. Sulistio A., Cibej U., Venugopal S., Robic B., Buyya R. A toolkit for modeling and simulating Data Grids: an extension to GridSim / Concurrency and Computation: Practice and Experience - Concurrency. – 0123 John Wiley & Sons, Ltd, 2008. – Vol.20, No 13. – P.1591-1609.
6. Стеценко І. В. Формальне описання систем засобами Петрі-об'єктних моделей / І. В. Стеценко // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: зб. наук. пр. – № 53. – К.: ВЕК+, 2011. – С.74-81.

7. Аветисян А. И. Системы управления кластерами / А. И. Аветисян, Д. А. Грушин, А. Г. Рыжов // Труды Института системного программирования РАН. Том 3. – М., 2002. – С. 39-62.

8. Литвинов В. В. Распределенная система имитационного моделирования на основе архитектуры CORBA / В. В. Литвинов, В. В. Казимир, И. Б. Гавсиевич // Математичні машини і системи. – № 2, 3. – К., 2000. – С. 111-114.

9. Анни П. Этот Grid – неспроста... / П. Анни // Открытые системы. – №1. – М., 2003 [Электронный ресурс]. – Режим доступа: <http://citforum.ru/nets/articles/egridn.shtml>.

УДК 004.932.2(045)

П.О. Приставка, д-р техн. наук

М.О. Рябий, аспірант

Національний авіаційний університет, м. Київ, Україна

ПІДВИЩЕННЯ РІВНЯ СТИСНЕННЯ З ВТРАТАМИ ЦИФРОВИХ ЗОБРАЖЕНЬ З ВИКОРИСТАННЯМ НИЗЬКОЧАСТОТНИХ ФІЛЬТРІВ

Проведено експериментальне дослідження використання лінійних фільтрів (низькочастотних та псевдозворотних до них контрастних) для підвищення відсотка стиснення цифрованих зображень з втратами. Статистично доведено переваги запропонованого підходу.

Проведено экспериментальное исследование использования линейных фильтров (низкочастотных и псевдообратных к ним контрастных) для повышения процента сжатия цифровых изображений с потерями. Статистически доказано преимущества предложенного подхода.

An experimental study using of linear filters (low-frequency and pseudo inverse contrast tothem) is made to increase the percentage of image compression tsyfrovanyh losses. Statistically is proved benefits of the proposed approach.

Вступ. Значні капіталовкладення в комп'ютерну техніку, здатну ефективно працювати з такими графічними середовищами як Microsoft Windows або X / Motif, створили цілий клас комп'ютерів, здатних відтворювати на екрані складні графічні образи з якістю, що наближається до телевізійного або поліграфічного. Програми, що працюють з графічними образами, відносяться до таких різних категорій, як комп'ютерні ігри, освіта, настільні видавничі системи, а також багатьох інших. Спільна особливість у цих програм одна: графічні файли, з якими вони працюють, займають велику кількість дискового простору.

У кінці 70-х – початку 80-х років технології стиснення графічних образів розвивалися в напрямку застосування звичайних алгоритмів стиснення без втрат до графічних даних [1]. Широко поширені у світі персональних комп'ютерів формати зберігання графічних образів, такі як РСХ, ВМР, використовуючи ті чи інші методи стиснення без втрат, дозволяють зменшувати обсяг файлу 20 відсотків [2]. Однак зі зростанням обсягу збереженої графічної інформації такий підхід перестав себе виправдовувати.

Постановка проблеми. Тому проблематику цієї роботи визначено потребою у стисненні цифрованих зображень (ЦЗ) відомими методами з додатковою модифікацією, що за можливості має сприяти більшому ступеню компресії, з втратами та без них.

Аналіз досліджень та постановка задачі. Стиснення ЦЗ поділяються на дві групи: стиснення з втратами та стиснення без втрат [3]. Методи стиснення без втрат дають більш низький коефіцієнт стиску, проте, зберігають точне значення пікселів вихідного зображення. Методи стиснення з втратами дають високі коефіцієнти компресії, але не дозволяють відтворити вихідне зображення з точністю до всіх значень кольорових складових пікселя. Загальновідомо, що під час погляду на фотореалістичне ЦЗ, людське око не сприймає всі відтінки кольорів, тому деякі деталі можуть бути проігноровані без суттєвого спотворення зображення. Найбільш відомими та широко розповсюдженими методами стиснення ЦЗ із втратами та без них є алгоритм JPEG та архіватор WinRAR відповідно [4;5].

Вищезазначені методи стиснення дають досить високий коефіцієнт компресії даних, з незначними втратами або за відсутності останніх. У роботі [6] було запропоновано