

УДК 004.4'242

І. В. Білоконь, аспірант,  
С. Д. Погорілий, д-р техн.наук, проф..  
Київський національний університет ім. Тараса Шевченка, м. Київ, Україна  
[deimos@univ.net.ua](mailto:deimos@univ.net.ua), [sdp77@i.ua](mailto:sdp77@i.ua)

## Застосування генетичних алгоритмів для оптимізації конфігурацій обчислювального кластера з вузлами у вигляді віртуальних машин

*З метою дослідження генетичного алгоритму пошуку оптимальних конфігурацій обчислювального кластера з вузлами у вигляді віртуальних машин показано доцільність практичного використання алгоритму для розв'язання цієї задачі та досліджено можливість явно врахування кількості доступних обчислювальних ресурсів при побудові фітнес-функції.*

**Ключові слова:** обчислювальний кластер, віртуалізація, віртуальна машина, генетичний алгоритм, фітнес-функція.

### Вступ

Задача адаптації кластерних обчислювальних систем до потреб різноманітних програмних реалізацій паралельних алгоритмів на сьогоднішній день не дістала однозначного розв'язку. Широка доступність та функціональні можливості сучасних платформ апаратно-програмної віртуалізації стали поштовхом до пошуку шляхів застосування віртуальних машин (ВМ) для цих цілей. У роботі [1] запропоновано концепцію побудови обчислювального кластера (ОК) з вузлами у вигляді ВМ. Оскільки ВМ фактично є певними програмними абстракціями, реконфігурація ресурсів такого ОК (кількість вузлів, об'єм оперативної пам'яті (ОП) вузлів, взаємне розташування вузлів відносно серверів ВМ (СВМ) тощо) може бути виконана виключно програмними засобами (у межах наявних апаратних ресурсів) [2]. Це означає можливість реалізації великої кількості різноманітних конфігурацій, що, в свою чергу, вимагає розв'язання задачі пошуку найбільш доцільних із них для певних задач, або черги завдань ОК, оскільки від конфігурації обчислювальної системи безпосередньо залежить продуктивність виконання обчислень [3].

У роботі [4] було сформульовано NP-повну задачу оптимізації завантаженості ресурсів ОК з вузлами у вигляді ВМ і запропоновано застосування генетичних алгоритмів (ГА) для її розв'язку. В роботі [5] запропоновано ГА, а також метод побудови фітнес-функції (ФФ), що базується на використанні показників завантаженості центральних процесорів (ЦП) ВМ, отриманих при попередньому дослідженні низки ключових конфігурацій. Обчислення значень ФФ для довільних конфігурацій ОК вимагає певного моделювання показників завантаженості ЦП для цих конфігурацій, що потребує виконання додаткових досліджень [6-8]. Робота присвячена подальшому дослідженню ГА та пошуку шляхів покращення його характеристик.

### Експериментальна установка

Дослідження виконувалися на ОК, побудованому на базі комп'ютерного класу ІТ академії Microsoft в інформаційно-обчислювальному центрі Київського національного університету імені Тараса Шевченка (8 персональних комп'ютерів: Intel Core 2 Quad CPU Q6600 2.4 GHz, 8 GB RAM, Gigabit Ethernet NIC, Windows Server 2008 R2 Enterprise). Вузлами ОК є ВМ на платформі Microsoft Hyper-V [9], із встановленим пакетом Microsoft Windows HPC Server 2008 R2 [10]. Більш детально ОК описано у роботі [1].

Як тестову задачу було використано пакет LINPACK [11] з розмірностями задачі  $N = 4608$ ,  $15936$ ,  $31872$ , та параметром  $NB = 192$ . Збір даних про завантаженість процесора відбувався протягом періоду роботи завдання із моменту запуску до моменту завершення. Для проведення вимірювань було створено автоматичну систему, яка виконує реконфігурацію обчислювального кластера та збір експериментальних даних.

### Дослідження актуальності практичного використання ГА

Для ілюстрації актуальності дослідження генетичного підходу в роботі [5] було виконано порівняння швидкості роботи ГА із швидкістю пошуку повним перебором. Для дослідження актуальності практичного застосування ГА, в цій роботі виконано вимірювання швидкості пошуку за допомогою ГА та пошуку випадковим вибором. Було досліджено пошук оптимальних конфігурацій для черги із 3 завдань кластера (LINPACK  $N = 4608$ ,  $15936$ ,  $31872$ ). Параметри ГА для цього дослідження [5]: розмір покоління – 256 особин, імовірність мутації 0.1, селекція методом рулетки. Оскільки застосовувалася нормована ФФ, максимальне значення ФФ для цієї черги рівне 3.

На рисунок 1 наведено усереднені результати 24 експериментів із відповідними похибками безпосередніх вимірів (довірча імовірність 0.9). Згідно отриманих результатів, швидкість пошуку за допомогою ГА значно вища швидкості пошуку

випадковим вибором, що підтверджує актуальність практичного застосування запропонованого у роботі [5] ГА для пошуку оптимальних конфігурацій ОК з вузлами у вигляді ВМ.

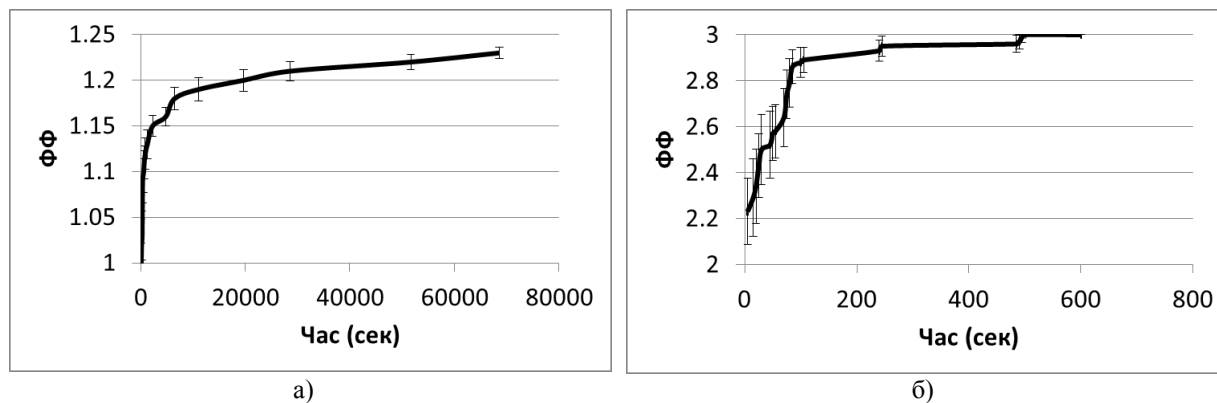


Рисунок 1 – Залежність нормованої ФФ найкращої знайденої конфігурації від часу пошуку при випадковому виборі (а) та пошуку за допомогою ГА (б).

### Дослідження впливу доступного об'єму ОП на завантаженість ЦП

Запропонований у роботі [5] метод обчислення значень ФФ базується на алгоритмі прогнозування значень ФФ на основі попередньо вимірних показників завантаженості ЦП. З метою пошуку способів покращення точності прогнозування шляхом врахування залежності завантаженості ЦП від кількості доступних обчислювальних ресурсів (ОП, швидкість і затримки мережевого з'єднання) було виконано дослідження залежності завантаженості ЦП від

доступного для вузла ОК об'єму ОП. Як тестові завдання було використано LINPACK з  $N=15936$  та  $N=31872$ .

Дослідження виконувалося для такої конфігурації ОК: 2 вузли (ВМ) по 4 віртуальні ЦП, 1 ВМ на 1 СВМ. При кожному дослідженні об'єм ОП ВМ зменшувався від попередньо виміряного для цієї конфігурації максимального значення показника виділеної ОП. Зменшення відбувалося з кроком 32 МБ. Результати експерименту наведено на рис. 2.

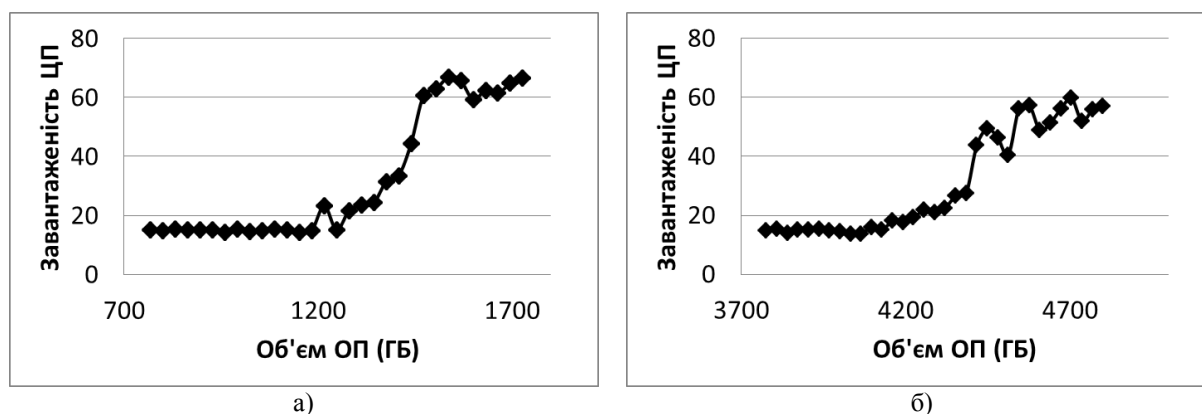


Рисунок 2 – Завантаженість ЦП при різних об'ємах ОП. LINPACK  $N=15936$  (а),  $N=31872$  (б).

При зменшенні об'єму ОП до значень менших за необхідні для роботи програми, операційна система активніше використовує файл підкачки сторінок. Це призводить до збільшення часу запису/читання даних із ОП, а отже і збільшення часу очікування ЦП і, як результат, падіння завантаженості ЦП. Відсутність монотонності графіків на рис. 2 можна пояснити особливостями реалізації у ОС доступу до ресурсів ОП,

організації віртуальної пам'яті, логікою роботи файлу підкачки сторінок.

Аналізуючи наведені на рис. 2 залежності можна зробити наступні висновки.

1. Графік завантаженості ЦП залежить не лише від програмної реалізації завдання кластера, а й від параметрів запуску завдання. Це унеможливає однозначне формування (теоретичне чи експериментальне) такої залежності як для різних реалізацій так і в межах

однієї і тієї самої програмної реалізації і означає необхідність попереднього дослідження цієї залежності для конкретних параметрів запуску завдання перед можливістю використання її при побудові ФФ.

2. Відсутність монотонності залежності потребує виконання значної кількості експериментів для її побудови. Це фактично означає вагоме збільшення часу попереднього дослідження конфігурацій і, таким чином, неможливість реалізації такого підходу на практиці. Тому практичніше робити інженерні оцінки зменшення завантаженості ЦП в залежності від об'єму доступної ОП, обмежуючись апроксимацією залежності по меншій кількості точок (наприклад, по 3).

### Вплив топології міжпроцесних зв'язків

При прогнозуванні завантаженості ЦП варто враховувати ще один фактор – мережеве з'єднання. У роботі [3] досліджено залежність характеристик мережевого з'єднання від взаємного розташування ВМ відносно СВМ і показано вплив взаємного розташування ВМ на продуктивність роботи завдань ОК. Для досліджуваної апаратно-програмної конфігурації комп'ютерного класу затримки мережевого з'єднання між ВМ в межах одного і того самого СВМ у 2 рази менші, ніж

затримки між ВМ, розташованих на різних СВМ. Для конфігурацій ОК, що передбачають розташування процесів завдання на різних СВМ топологія міжпроцесних зв'язків може суттєво впливати на якість прогнозування показників завантаженості ЦП. Для дослідження ступеня можливого впливу було виконано наступний експеримент. При 8 процесах завдання із початковою конфігурацією «1 процес на 1 СВМ» (позначимо «11111111») досліджуємо показники завантаженості ЦП, поступово збільшуючи кількість процесів завдання на одному із СВМ (конфігурації «11111111», «11111120»,..., «80000000»). Як тестове завдання було взято LINPACK з  $N=31872$ .

Зважаючи на апаратно-програмну конфігурацію експериментальної установки, реалізацію конфігурацій ОК було здійснено наступним чином. Якщо кількість процесів  $n$  на 1 СВМ не перевищує 4, вони розподіляються на ВМ цього СВМ із кількістю віртуальних ЦП рівною кількості процесів. При кількості 5 – 8 процеси розподіляються на 2 ВМ відповідного СВМ із кількостями ЦП відповідно 4 та  $n-4$ . Результати експерименту наведено на рис. 3.

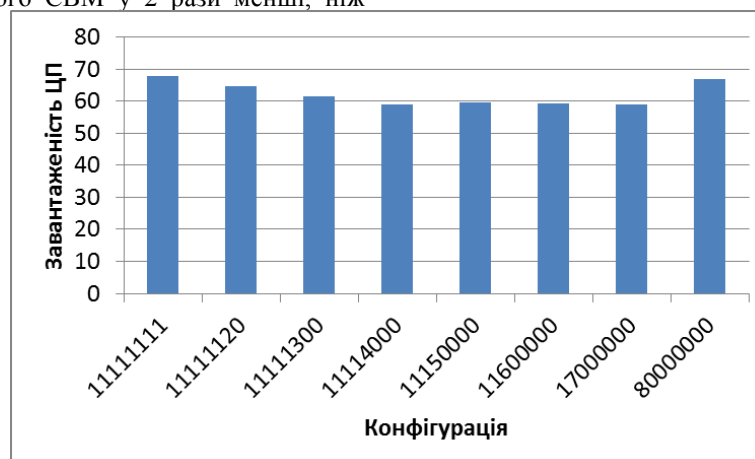


Рисунок 3 – Ілюстрація впливу топології міжпроцесних зв'язків на залежність завантаженості ЦП від конфігурації ОК.

Проаналізувавши залежність на рис. 2 і беручи до уваги апаратну конфігурацію експериментальної установки можна оцінити, що збільшення кількості процесів на 1 СВМ до 6 призводить до боротьби за ресурси ОП. Але на рис. 3 значного падіння завантаженості ЦП не спостерігається і, навіть, можна побачити її збільшення для 8 процесів на 1 СВМ. Це пояснюється тим, що збільшення швидкості міжпроцесного зв'язку компенсує затримки звернення до ОП для досліджуваної програмної реалізації. Це означає, що використання залежності, наведеної на рис. 2 для збільшення точності прогнозування значень ФФ для довільної конфігурації ОК в загальному випадку доцільно виконувати лише тоді, коли опорна та цільова

конфігурації реалізують одне і те саме взаємне розташування процесів завдання. Інакше необхідно додатково враховувати вплив топології міжпроцесних зв'язків.

### Висновки

1. За результатами порівняння швидкості пошуку оптимальних конфігурацій ОК з вузлами у вигляді ВМ шляхом застосування запропонованого ГА та пошуку випадковим вибором показано актуальність практичного застосування ГА для такої задачі.
2. Отримані результати експериментального дослідження впливу доступного об'єму ОП та взаєморозташування процесів завдання ОК на завантаженість ЦП вузлів дозволяють зробити

наступні висновки щодо можливості покращення точності прогнозування завантаженості ЦП при побудові ФФ для ГА.

3. При врахуванні об'єму доступної ОП доцільно обмежитися інженерними оцінками за декількома

вимірними точками для досліджуваної програмної реалізації та її параметрів запуску.

4. Врахування об'єму доступної ОП доцільно виконувати лише у поєднанні із врахуванням впливу топології міжпроцесних зв'язків досліджуваної програмної реалізації.

### Список літератури

1. Білоконь І. Побудова динамічно реконфігурованої обчислювальної архітектури з використанням технологій віртуалізації / Білоконь І., Грязнов Д., Погорілий С. // Вісник Київського національного університету імені Тараса Шевченка. Серія: радіофізика та електроніка. — 2010. — Вип. 14. — С. 4 – 6.
2. Погорілий С. Технологія віртуалізації. Динамічна реконфігурація ресурсів обчислювального кластера / С. Погорілий, І. Білоконь, Ю. Бойко // Математичні машини і системи. – 2012. – №3. – С. 3 – 18.
3. Білоконь І. Дослідження впливу конфігурації обчислювальної системи на продуктивність програмних реалізацій паралельних алгоритмів / І. Білоконь, Д. Грязнов, С. Погорілий // Праці VII Міжнар. конф. «Електроніка та прикладна фізика» (19-22 жовтня 2011 р. Україна, Київ). — К., 2011.
4. Погорілий С.Д. До задачі оптимізації завантаженості ресурсів обчислювального кластера з вузлами у вигляді віртуальних машин / С.Д. Погорілий, І.В. Білоконь // Матеріали 8 Міжнар. науково-практичної конф. з програмування «УкрПРОГ–2012» Проблеми програмування –№ 2-3 (22-24 травня 2012 р, Україна, Київ). – К., 2012.
5. Bilokon I. Research of Genetic Algorithm for searching optimal configurations of computing cluster with virtual machine nodes / I. Bilokon, S. Pogorilyy // Theoretical and Applied Aspects of cybernetics. Proceedings of the 2nd International Scientific Conference of students and Young Scientists. — Kyiv: Bukrek, 2012.
6. Погорілий С.Д. Дослідження віртуальних кластерних архітектур на прикладі алгоритму Белмана-Форда / С.Д. Погорілий, І.В. Білоконь, О.В. Вареник // Наукові праці ДонНТУ. Серія: "Інформатика, кібернетика та обчислювальна техніка". — 2012. — Вип. 15(203). — С. 40 – 47.
7. Погорілий С.Д. Формування та дослідження паралельних схем алгоритму Данцига / С.Д. Погорілий, І.В. Білоконь, Т.Ю. Проців // Вісник Київського національного університету імені Тараса Шевченка. Серія фізико-математичні науки. — 2012. — Вип. 1. 2012. — С. 207 – 216.
8. Погорілий С. Д. Створення та дослідження паралельних схем алгоритму Пріма / С. Д. Погорілий, Білоконь І. В., Шевченко А. В. // Праці дев'ятої міжнародної конференції «Теоретичні та прикладні аспекти побудови програмних систем» (ТААПСД'2012 3-7 грудня 2012 р.). — К., 2012.
9. Microsoft Hyper-V [Електронний ресурс]. – Режим доступу: <http://technet.microsoft.com/en-us/library/cc753637%28WS.10%29.aspx>.
10. Windows HPC Server 2008 R2 Technical Library. [Електронний ресурс]. – Режим доступу: <http://technet.microsoft.com/en-us/library/ee783547%28WS.10%29.aspx>.
11. LINPACK. [Електронний ресурс]. – Режим доступу: <http://www.netlib.org/benchmark/hpl>.

Надійшла до редакції 01.08.2013

### И. В. БЕЛОКОНЬ, С. Д. ПОГОРЕЛЫЙ

Киевский национальный университет им. Т. Шевченко

### ИСПОЛЬЗОВАНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ ОПТИМИЗАЦИИ КОНФИГУРАЦИЙ ВЫЧИСЛИТЕЛЬНОГО КЛАСТЕРА С УЗЛАМИ В ВИДЕ ВИРТУАЛЬНЫХ МАШИН

С целью исследования генетического алгоритма поиска оптимальных конфигураций вычислительного кластера с узлами в виде виртуальных машин показана целесообразность практического использования алгоритма для решения этой задачи и исследована возможность явного учета количества доступных вычислительных ресурсов при построении фитнес-функции.

**Ключевые слова:** вычислительный кластер, виртуализация, виртуальная машина, генетический алгоритм, фитнес-функция.

### I.V. BILOKON, S.D. POGORILYI

Taras Shevchenko National University of Kyiv

### USING GENETIC ALGORITHMS FOR OPTIMIZATION OF COMPUTING CLUSTER CONFIGURATION WITH VIRTUAL MACHINE NODES

In order to investigate the Genetic Algorithm for searching of the optimal configuration of a computing cluster with virtual machine nodes we have shown the expediency of practical usage of the algorithm and considered the possibility of accounting the available resources during fitness function calculation.

**Key words:** computing cluster, virtualization, virtual machine, genetic algorithm, fitness function.