



Рисунок 2 – Використання MDCS на комп'ютерах кластера

4.Тестування роботи кластера. Тестування кластера включає перевірку з'єднання між вузлами (програма ring) та наявності доступу до загального каталогу. На наступному етапі були скомпільовані та запущені на виконання тестові програми – спершу локально на консольному комп'ютері, а згодом – на всіх наявних робочих станціях одночасно. Аналогічно, кластер протестовано на спроможність виконання програм в середовищі MATLAB. Цим було засвідчено готовність розробленого кластера до роботи.

Таким чином, в результаті виконання вищеописаних дій, отримано 8-вузлову кластерну систему. Створену кластерну систему призначено для розпаралелення задач ідентифікації інтервальних моделей, реалізацію яких виконано в середовищі MATLAB. Проте, встановлення додаткового програмного забезпечення дає можливість виконання інших паралельних програм.

#### Список використаних джерел

1. Шокин Ю.И. Интервальный анализ. – Новосибирск: Наука, 1981.
2. Дивак М. П. Метод формування допускової еліпсоїдної оцінки параметрів інтервальних моделей на основі виділення із інтервальної системи лінійних алгебричних рівнянь основних активних обмежень / Дивак М. П., Козак О. Л. // Реєстрація, зберігання і обробка даних. – 2009. – Т. 11, № 2. – С.25-36.
3. Спортак М., Франк Ч., Паппас Ч. и др. Высокопроизводительные сети. Энциклопедия пользователя. — К.: ДиаСофт, 1998. — 432 с.
4. Сбитнев Ю. Linux. Кластер. Практическое руководство по параллельным вычислениям. [Електронний ресурс] / Інтернет-ресурс. — Режим доступу: <http://cluster.linux-ekb.info>

УДК 004.75

## ОСОБЛИВОСТІ ПОБУДОВИ КЛАСТЕРА ДЛЯ РЕАЛІЗАЦІЇ ПАРАЛЕЛЬНИХ ОБЧИСЛЮВАЛЬНИХ СХЕМ

Олійник І.С.

Тернопільський національний економічний університет

### І. Постановка проблеми

Із розвитком сучасних інформаційних технологій все гостріше постає питання збільшення швидкодії розв'язування задач, отримання результату яких неможливе без проведення великої кількості складних обчислювальних операцій. На практиці подібні обчислення реалізуються за допомогою суперкомп'ютерів, але головним їхнім недоліком є висока вартість обладнання. Тому все більшої актуальності набуває практика розпаралелення обчислювальних схем, поставлених перед користувачем задач, за допомогою кластерних систем, які будуються на основі наявної комп'ютерної мережі і не потребують великих фінансових затрат.

### II. Мета роботи

Метою дослідження є визначення особливостей побудови низьковартісної кластерної системи для реалізації паралельних обчислювальних схем.

### III. Особливості створення кластерних систем для паралельних обчислень

Існують такі задачі, для розв'язання яких недостатньо можливостей звичайного персонального комп'ютера, адже вони вимагають вищої продуктивності. В такому випадку з декількох потужних

систем створюють НРС (High Performance Computing) кластер, що дозволяє здійснити розподіл обчислювальних процедур не лише між різними процесорами, а й між різними комп'ютерами. Якщо досліджувана задача не вимагає високої взаємодії паралельних потоків, то НРС кластер може складатися з багатьох малопотужних однопроцесорних систем. Доведено, що часто такі технічні рішення забезпечують більшу продуктивність при нижчій вартості у порівнянні із продуктивністю суперкомп'ютерів [1].

Реалізація обчислювальної процедури розв'язання задачі на кластері здійснюється за посередництвом програмно створених додатків. Під паралельним додатком кластерної системи розуміють декілька процесів, що взаємодіють між собою в мережі. Якщо користувач правильно і ефективно розподілить своє завдання між процесорами на вузлах кластера, то отримає суттєвий вигаш у швидкості обчислень, пропорційно числу процесорів [3].

Відзначимо, що основою для кластера є комунікаційне середовище (MPI (Message Passing Interface), PVM (Parallel Virtual Machine)), а не операційна система. Саме це середовище забезпечує ефективну взаємодію окремих частин однієї паралельної програми, що виконуються на різних комп'ютерах [1].

Для організації локальної мережі між обчислювальними вузлами кластера використовується доступна та широко розповсюджена технологія Gigabit Ethernet, адже вона як за пропускну здатністю, так і за параметрами затримок задовольняє вимоги до побудови мережі обміну по протоколу MPI [2].

#### **IV. Висновки**

Дослідження особливостей побудови кластерів показує, що при проектуванні такої системи важливим є саме вибір комунікаційного середовища, а не операційної системи. Очевидно, що для розгортання кластера, достатньо і декількох однопроцесорних систем, об'єднаних гігабітною мережею, що є значно дешевшим рішенням, ніж застосування суперкомп'ютерів. І наскільки б точним і сучасним не було обладнання, лише від правильного розподілу користувачем завдань між процесорами кластера залежатиме швидкість реалізації обчислювальної схеми.

#### **Список використаних джерел**

1. Огневий О. В. Побудова паралельних обчислювальних систем на базі кластерних технологій. // Вісник Хмельницького національного університету. – 2009. – № 4. С. 99-102.
2. Погорілий С.Д., Бойко Ю.В., Грязнов Д.Б., Ломакін О.Д., Мар'яновський В.А. Концепція створення гнучких гомогенних архітектур кластерних систем. // Проблеми програмування. – 2008. - №2-3. Спеціальний випуск. – С. 84-90.
3. <http://www.intuit.ru/department/supercomputing/tbucs/class/free/status/>

УДК 519.688

## **ДОСЛІДЖЕННЯ РОЗПАРАЛЕЛЕННЯ АЛГОРИТМУ ПОШУКУ МІНІМАЛЬНОГО ЗНАЧЕННЯ НА ГРАФІЧНОМУ ПРОЦЕСОРІ**

**Струбицька І.П.**

Тернопільський національний економічний університет

Проблема ефективності обробки великих об'ємів даних є однією з актуальних на сьогоднішній день. Багатоядерні процесори можуть виконувати одночасно лише декілька потоків, при цьому більш висока продуктивність має іноді дуже велику ціну.

Технологія NVIDIA CUDA — це фундаментально нова архітектура обчислень на графічних процесорах, яка призначена для вирішення комплексу обчислювальних задач споживачів, бізнесу і технічної індустрії. На поточний момент обчислення на графічних процесорах з технологією CUDA — це інноваційне поєднання обчислювальних особливостей нового покоління графічних процесорів NVIDIA, які обробляють відразу тисячі потоків з високим рівнем інформаційного завантаження, які доступні через стандартну мову програмування C.

Використання GPU для обчислень загального призначення є перспективним при розв'язку задач. Наприклад, для задачі оптимізації параметрів дискретної динамічної моделі,