

Рисунок 5 – Залежність часу виконання від кількості потоків

Висновки

Як видно з рисунку 4 отримані, в результаті розв'язку запропонованим методом задачі синтезу допусків на параметри РЕК, допуски входять в область заданих нами на початку допусків. При використанні паралелізму часова складність зменшується по мірі збільшення вхідних даних. І для задовільнення критерію максимізації допускової оцінки області параметрів апроксимація многогранної області представлення оцінок параметрів у вигляді еліпсоїда значно краще, адже, за умов особливої витягнутості області оцінок параметрів, еліпсоїд здатен покривати більшу область параметрів.

Список використаних джерел

1. Козак О.Л. Метод ітеративного формування допускової еліпсоїдної оцінки в задачах параметричної ідентифікації інтервальних моделей. Вип.2 (2008), 254-261.
2. Дивак М.П. Еліпсоїдне оцінювання допусків параметрів радіоелектронних кіл/ Дивак М.П., Козак О.Л.// Реєстрація, зберігання і обробка даних. – 2009. – Том 11, №1. – С.93-104.
3. Лычак М.М. Множественная модель неопределенного процесса и ее использование для обработки результатов измерений/ Лычак М.М.// Проблемы управления и информатики. – 1996. - №1 – 2. С.184-192.
4. Дивак М.П., Манжула В.І. Активна ідентифікація параметрів інтервальних моделей методом локалізації з вибором насиченого блоку експерименту. ЛПНУ, Радіоелектроніка і комунікації, 440 (2002), 241-246.

УДК 004.75

ОРГАНІЗАЦІЯ КЛАСТЕРНИХ ОБЧИСЛЕНЬ ДЛЯ РОЗВ'ЯЗУВАННЯ ЗАДАЧ ПАРАМЕТРИЧНОЇ ІДЕНТИФІКАЦІЇ ІНТЕРВАЛЬНИХ МОДЕЛЕЙ

Новіцький О.П., Пукас А.В.

Тернопільський національний економічний університет

I. Постановка проблеми

Для розв'язування задач параметричної ідентифікації інтервальних моделей як статичних, так і динамічних систем, на сьогоднішній день розроблено значну кількість методів [1, 2]. Ефективність їх реалізацій залежить від розмірності та точності моделі, а також, від розміру вхідної вибірки даних. Зважаючи на те, що вказані методи базуються на ітераційному розв'язуванні інтервальних систем лінійних (або нелінійних) алгебричних рівнянь (ІСЛАР), вони характеризуються високою обчислювальною складністю. З іншого боку, наявність операцій з матрицями забезпечує можливість розпаралелення обчислень.

Одним із найефективніших засобів розпаралелення є кластерні системи, що характеризуються відносно невисокою вартістю та зручністю в організації і експлуатації, зокрема в комп'ютерних навчальних лабораторіях.

II. Мета роботи

Метою дослідження є організація кластерної системи для підвищення ефективності обчислювальних задач, зокрема, задачі параметричної ідентифікації інтервальних моделей.

III. Особливості реалізації кластерної системи

При побудові обчислювального кластера весь процес був розподілений на декілька етапів, відповідно до використаних технологій та протоколів.

1. Підготовка обладнання. На даному етапі визначено кількість необхідних комп'ютерів (обчислювальних вузлів) та технологію передачі даних. Вибір кількості вузлів зроблено виходячи з умови, що чим більше комп'ютерів використовується в кластері, тим більше навантаження припадає на мережеве з'єднання [3]. При використанні мережевого обладнання стандарту Ethernet навантаження стає відчутним вже при використанні більше, ніж чотирьох вузлів. Тому вибрано технологію Fast Ethernet. Для мінімізації кількості мережевих з'єднань використано 8-портовий комутатор, відповідно до можливостей якого і вибрано кількість обчислювальних вузлів. Особливу увагу приділено характеристикам продуктивності робочих станцій, характеристики яких наступні: CPU – AMD Athlon II X2 260 3,2 GHz, HDD – 250 Gb, RAM – 2 Gb.

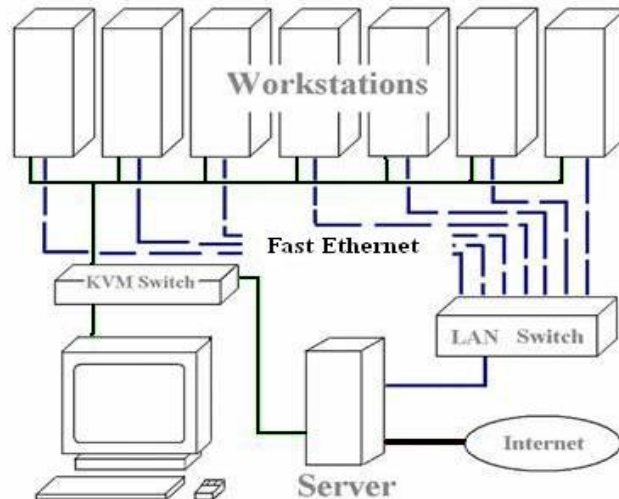


Рисунок 1 – Архітектура кластера

2. Встановлення програмного забезпечення. На всіх комп'ютерах встановлено операційну систему Linux. На один комп'ютер встановлена графічна підсистема, оскільки він виконує роль консолі для всього кластера. Тобто, на цьому комп'ютері компілюються і запускаються на виконання паралельні програми. Зважаючи на це, дана робоча станція має компілятори для мов програмування C і fortran з необхідними для них бібліотеками, а також середовищем розробки програм. Для організації спільного доступу до файлів на комп'ютері встановлено NFS-сервер та NFS-клієнт, а для передачі службової інформації – ssh-клієнт. Для створення та дослідження роботи математичних моделей встановлено пакет MATLAB включно з модулем MDCS (MATLAB Distributed Computing Server). На всі решту обчислювальні вузли кластера встановлено базову операційну систему (без графічного інтерфейсу), бібліотеки, необхідні для виконання паралельних програм, пакет OpenMPI, NFS-client, ssh-client та модуль паралельних обчислень MDCS [4].

3. Налаштування взаємодії між комп'ютерами. На комп'ютері, що є консольним, створено каталог для файлів виконуваних програм і забезпечено загальний доступ до нього інших комп'ютерів з правами зчитування, запису та виконання. На обчислювальних вузлах кластера налаштоване автоматичне монтування спільного каталогу при їх включенні. Для того, щоб з консольного комп'ютера можна було без додаткових запитів запускати частини паралельної програми на обчислювальних вузлах кластера організовано безпарольний доступ по протоколу SSH з консолі кластера на всі обчислювальні вузли. Налаштування OpenMPI збережено у текстовому файлі mpi.hosts, у який внесено список всіх вузлів кластера, як обчислювальних, так і консолі. Налаштування MATLAB для виконання паралельних обчислень на обчислювальних вузлах виконано шляхом перерахунку їх імен в налаштуваннях сервера MDCS.

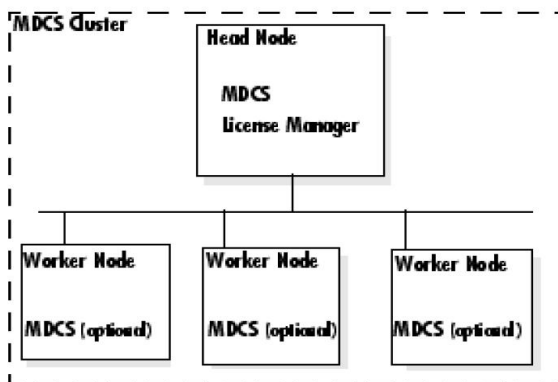


Рисунок 2 – Використання MDCS на комп'ютерах кластера

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