

Generation of Test Sequences for Digital Circuits // International Conference on High-Performance Computing and Networking, Brussels (Belgium), April, Lecture Notes In Computer Science.- Vol.1067.- 1996.- P.454-459.

14. Krishnaswamy D., Hsiao M.S., Saxena V., Rudnick E.M., Patel J.H., Banerjee, P. Parallel genetic algorithms for simulation-based sequential circuit test generation // Proceedings Tenth International Conference on VLSI Design, 1997, 4-7 Jan 1997.- P.475-481.

15. Corno F., Sonza Reorda M., Rebaudengo M. Experiences in the use of evolutionary techniques for testing digital circuits // Aproc. of Conf. Applications and science of neural networks, fuzzy systems, and evolutionary computation, San Diego CA, ETATS-UNIS, 1998, Vol.3455.- P.128-139.

16. Барашко А.С., Скобцов Ю.А., Сперанский Д.В. Моделирование и тестирование дискретных устройств. – К.: Наукова думка, 1992.– 288 с.

17. Brgles F., Bryan D., Kozminski K. Combinational profiles of sequential benchmark circuits // International symposium of circuits and systems, ISCAS-89, 1989.– P.1929-1934.

18. Гергель В.П., Стронгин Р.Г. Основы параллельных вычислений для многопроцессорных вычислительных систем. Учебное пособие.- Нижний Новгород: Изд-во ННГУ им. Н.И. Лобачевского, 2003.- 184 с.

Надійшла 7.9.2009 р.

УДК 004.052.42: 004.725

О.В. ОГНЄВИЙ

Хмельницький національний університет

ПОБУДОВА ПАРАЛЕЛЬНИХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ НА БАЗІ КЛАСТЕРНИХ ТЕХНОЛОГІЙ

В статті розглядається ідея побудови паралельних обчислювальних систем із загальнодоступних комп'ютерів на базі процесорів Intel і недорогих Ethernet-мереж з установкою на ці комп'ютери Linux і об'єднанням їх в кластери за допомогою однієї з безкоштовно розповсюджуваних комунікаційних бібліотек (PVM та MPI).

In article the idea of construction of parallel computing systems from popular computers on the basis of processors Intel and inexpensive Ethernet-networks with installation on these computers Linux and their association in clusters by means of one of free of charge extended communication libraries (PVM and MPI) is considered.

Ключові слова: паралельні обчислювальні системи, кластерні технології.

Вступ

Тема збільшення швидкості обчислень досить актуальна для всіх тих, чия діяльність пов'язана з великим об'ємом обчислювальних робіт. Так, наприклад, розрахунки гравітуючих газових дисків найчастіше вимагають декількох місяців безперервного рахунку на машині класу іРІІ. У середині 80-х перед вченими виникла проблема збільшення швидкодії обчислювальних процесів. Але ні тоді, ні навіть зараз найчастіше немає можливостей для закупівлі потужних комп'ютерів типу nCube, Cray або подібних. Однак з розвитком програмного забезпечення й появою вільно розповсюджуваної операційної системи стало можливим створити обчислювальний комплекс із ефективною швидкодією, порівнянною зі швидкодією суперкомп'ютерів, але з вартістю в десятки разів меншою.

Виявилось, що на багатьох класах завдань і при достатньому числі вузлів такі системи дають продуктивність, порівнянну з тією, що можна одержати, використовуючи дорогі суперкомп'ютери.

Постановка задачі

При реалізації кластерної технології необхідно уважно проаналізувати завдання. Паралельні обчислення – досить специфічна область математики й далеко не завжди паралельні обчислення можуть бути застосовні.

Важливим є також питання принципової можливості "розпаралелити" завдання. Деякі чисельні схеми в силу особливостей алгоритму не піддаються ефективній паралелізації. Перш ніж орієнтуватися на застосування кластера для рішення завдання, необхідно впевнитися в можливості застосування паралельних алгоритмів.

Також паралельні архітектури повинні створюватися з розрахунком на ефективне використання ресурсів цієї архітектури. Під цим розуміється те, що задача повинна бути розділена на частини, здатні виконуватися паралельно на декількох процесорах, і розділена ефективно, щоб складові, що виконуються окремо, мінімально впливали на виконання інших частин.

Класи завдань, які можна ефективно розпаралелити:

1. Одномірні масиви.
2. Двовимірні масиви.
3. Клітинні автомати.
4. Системи диференціальних рівнянь.

Клас завдань, вирішувати які можна, використовуючи паралельні алгоритми досить широкий і вкрай важливий. Однак варто враховувати, що паралельність завдання визначається не тільки її фізичним змістом, але й обраним чисельним алгоритмом.

Кластери Beowulf

Кластери Beowulf забезпечують гарною платформою для вивчення паралельного програмування й недорогою продуктивною обчислювальною системою.

Beowulf – це мультикомп'ютерна архітектура, що може використовуватися для паралельних обчислень. Ця система складається з одного серверного вузла й одного або більше клієнтських вузлів, з'єднаних за допомогою Ethernet або деякої іншої мережі. Система, побудована з готових промислових компонентів, наприклад ПЕОМ, на яких може працювати ОС Linux, стандартних адаптерів Ethernet і комутаторів (рис. 1). Вона не містить специфічних апаратних компонентів і легко відновлюється. Beowulf також використовує програмні продукти, такі як ОС Linux, середовища програмування Parallel Virtual Machine (PVM) і Message Passing Interface (MPI). Серверний вузол управляє всім кластером і є файлом-сервером для клієнтських вузлів. Він також є консоллю кластера й шлюзом у зовнішню мережу. Більші системи Beowulf можуть мати більше одного серверного вузла, а також можливо спеціалізовані вузли, наприклад, консолі або станції моніторингу. У більшості випадків клієнтські вузли в Beowulf пасивні. Вони конфігуруються й управляються серверними вузлами й виконують тільки те, що запропоновано серверним вузлом. У бездисковій конфігурації клієнтів, клієнтські вузли навіть не мають IP-адрес або імен, поки їх не призначить сервер. Одне з основних відмінностей Beowulf від кластера робочих станцій також у тім, що Beowulf працює як одна машина. У більшості випадків клієнтські вузли не мають клавіатур і моніторів, і можуть бути доступні тільки через віддалене підключення. Вузли Beowulf можуть розглядатися як елементи процесор + пам'ять, які вставляються в кластер так само як процесор або модуль пам'яті вставляються в материнську плату.

Beowulf – це не специфічний пакет програм, нова топологія мережі або новітня модифікація ядра ОС. Beowulf – це технологія кластеризації комп'ютерів, що працюють під керуванням ОС Linux.

Побудова кластерного комп'ютера не самоціль, а засіб досягти більшої ефективності й продуктивності. Є певний клас завдань, що вимагають продуктивності більше високої, ніж ми можемо одержати, використовуючи звичайні комп'ютери. У цих випадках з декількох потужних систем створюють НРС (High Performance Computing) кластер, що дозволяє рознести обчислення не тільки по різних процесорах (якщо використовуються багатопроцесорні SMP-системи), але й по різних комп'ютерах. Для завдань, що дозволяють розпаралелювання і не пред'являють високих вимог по взаємодії паралельних потоків, часто приймають рішення щодо створення НРС кластера з великого числа малопотужних однопроцесорних систем. Найчастіше подібні рішення, при низькій вартості, дозволяють досягти набагато більшої продуктивності, ніж продуктивність суперкомп'ютерів.

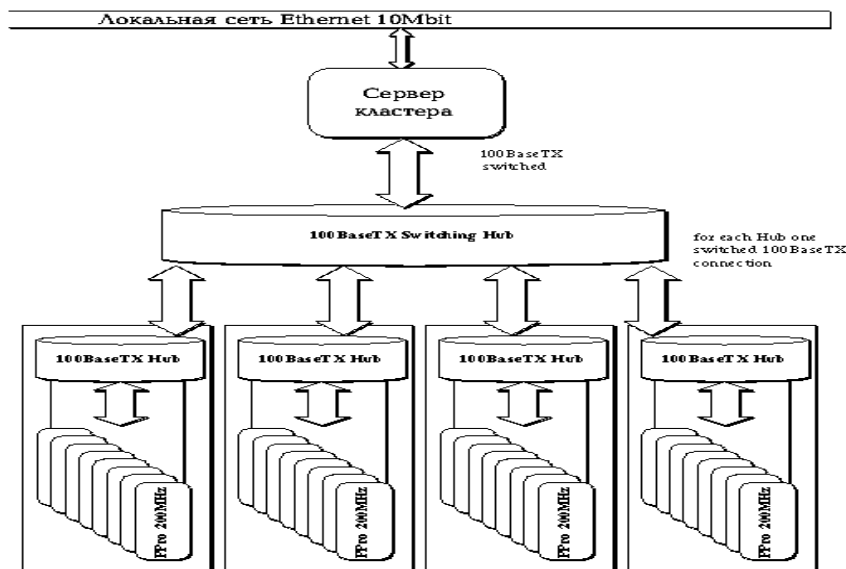


Рис. 1. Об'єднання ЕОМ в кластері

Кластер Beowulf – гетерогенна структура. У нього можуть входити найрізноманітніші за параметрами комп'ютери, побудовані на різних апаратних платформах, наприклад Intel Pentium різних версій, Alpha, RISC-процесори, Transmeta, 32-х і 64-х бітові процесори. Більше того, на комп'ютерах у кластері можуть бути встановлені всілякі системи: Linux, Windows, OS/2 WARP.

Що стосується розходження в параметрах (швидкодія, пам'ять, ...) у комп'ютерів, що входять у кластер, то це припустимо. Але в цьому випадку доведеться враховувати ці розходження при написанні паралельних програм, розподіляючи об'єм залежно від можливості кожного окремого комп'ютера. У

протилежному випадку кластер буде працювати як система, що складається з машин з мінімальними робочими параметрами.

Оскільки кластер Beowulf – масштабована система, то питання кількості вузлів не є життєво важливим. У міру необхідності можливо доволіно додавати кількість вузлів у будь-який час. Однак захоплюватися кількістю вузлів не завжди доцільно, тому що найбільш вузьке місце у кластері – це середовище передачі даних між вузлами, тобто пропускна здатність використовуваної мережі. Рекомендується при початковій побудові кластера обмежитися чотирма вузлами (одна консоль і три slave-ноди).

Департаментом комп'ютерних наук університета Північної Айови (США) був реалізований проект BCCD по створенню завантажувального компакт диска з варіантом, призначеним для побудови кластера. У результаті ми маємо можливість використати кластер взагалі без установки на вінчестер операційної системи й кластерного ПО. Для установки кластера необхідно два й більше мережних комп'ютери, що мають можливість завантажувати ОС безпосередньо з CD-ROM.

Організація кластерної мережі

Мережа – це модульна й адаптуєма комутаційна система, яку можна настроїти відповідно до всіляких вимог. Її модульність полегшує додавання нових компонентів або переміщення існуючих, а адаптивність спрощує внесення змін і вдосконалень. Мережа кластера Beowulf нічим принципово не відрізняється від мережі робочих станцій, тому в найпростішому випадку для побудови кластера необхідні звичайні мережні карти й хаби /комутатори, які використалися б при облаштуванні якого-небудь комп'ютерного класу. Однак, у випадку кластера є одна особливість. Мережа кластера, в першу чергу, призначена не для зв'язку машин, а для зв'язку обчислювальних процесів. Тому чим вище буде пропускна здатність вашої мережі, тим швидше будуть виконуватися паралельні завдання, запущені на кластері, отже робочі характеристики мережі мають першорядне значення.

Використовуване мережне устаткування характеризується двома параметрами:

Пропускна здатність – швидкість передачі даних між двома вузлами після того, як зв'язок установлений.

Латентність – середній час між викликом функції передачі даних і самою передачею. Час затрачається на адресацію інформації, спрацьовування проміжних мережних пристроїв, інші накладні витрати, що виникають при передачі даних.

Фактично пропускна здатність і латентність не тільки характеризують кластер, але й обмежують клас завдань, які можуть ефективно зважуватися на ньому.

В якості мережних адаптерів можна використати будь-які наявні в продажі карти, що підтримують роботу в стандартах 100BaseTx і Gigabit Ethernet. Найбільш перспективними на сьогоднішній день є мережні карти фірми "3Com", так як вони мають деякі переваги, що помітно впливають на продуктивність мережних комунікацій. Основні із них:

1) Розвантаження процесора при обчисленні контрольних сум TCP/UDP/IP. Звільняє центральний процесор від інтенсивних обчислень контрольних сум, виконуючи їх у самій мережній платі. Тим самим підвищується продуктивність системи й час життя процесора.

2) Звільнення ЦП при відновленні сегментованих пакетів TCP. Знижує навантаження на центральний процесор, підвищуючи продуктивність системи.

3) Об'єднання переривань. Дозволяє групувати кілька отриманих пакетів. Оптимізує обчислювальну ефективність хост – комп'ютера, скорочуючи число переривань і максимально звільняючи процесорні ресурси для роботи додатків.

4) Режим Bus mastering DMA. Забезпечує більше ефективний обмін даними для зниження завантаження центрального процесора.

Для організації зв'язаного каналу (channel bonding) найкраще вибрати мережні карти Intel EtherExpress PRO/100, 3Com FastEthernet (наприклад 3c905B, 3c905C) або карти GigabitEthernet від 3Com або Intel. Також можливе використання спеціалізованих серверних мережних карт, у яких є більше одного Ethernet-порту. Прикладами таких адаптерів можуть бути Intel EtherExpress PRO/1000 MF Dual Port або 3Com Fast EtherLink Server Dual Port 3c982C-TXM. Використання таких карт дозволить займати в комп'ютері вдвічі менше PCI-слотів і, відповідно, установлювати вдвічі більше мережних карт для об'єднання їх у зв'язаний канал.

Комутатори.

Другим важливим елементом мережі кластера є пристрої комутації мережних каналів. Комутатори й інші елементи мережної структури використовуються для забезпечення комунікацій між процесорами, для підтримки паралельного програмування й різних функцій керування. Вважається, що при збільшенні числа підключених вузлів загальна ширина смуги пропускання, як у всіх комутаторів з дійсною масштабованістю, росте пропорційно, а латентність залишається постійною. Іншими словами, смуга на кожному зі шляхів однакова, а число шляхів (напрямків) залежить від кількості вузлів, при цьому кожний вузол має зв'язок з усіма іншими вузлами незалежно від розміру кластера.

Вибір комутатора здійснюється, насамперед, на основі його характеристик. У найпростішому випадку для побудови мережі кластера можна використати прості хаби. Це рішення, найбільш вигідне за ціною, являється самим невдалим у технологічному змісті. При використанні хабів не відбувається

маршрутизації пакетів переданих даних. Будь-який пакет, переданий у мережу, направляється абсолютно всім учасникам мережі. Кожна машина "чує" всі пакети, що передаються в мережі, незалежно від того, чи призначений конкретний пакет для неї чи ні. При активному між процесорному обміні це може приводити до перевантаження мережі, збільшенні числа колізій і, як наслідок, до зниження ефективної швидкодії паралельної машини. Наприклад, якщо дві пари вузлів кластера одночасно обмінюються даними за допомогою 100 Мбіт хаба, то швидкість їхнього обміну падає вдвічі.

Прикладами комутаторів, що дозволяють настроїти Link Aggregation, можна вважати Cisco Catalyst 2900 series, Cisco Catalyst 3500 series, Cisco Catalyst 5000 series, 3Com SuperStack 3 4950, 4400 і ін. Слід зазначити, що наявність або відсутність функцій Link Aggregation залежить не тільки від моделі комутатора, але й від версії його програмного забезпечення.

Операційна система

При виборі операційної системи варто орієнтуватися насамперед на рекомендації розроблювачів програмного забезпечення. Однак, якщо є вибір, то за інших рівних умов варто віддати перевагу Linux.

Під Linux доступна величезна кількість серверного ПО, компіляторів, бібліотек, засобів відлагодження та ін. Велика кількість програмного забезпечення є у вільному доступі, для багатьох програм є вихідні коди й документація.

Плюсом Linux є "прозорість" для користувача й системного адміністратора, що дозволяє швидше й простіше розв'язувати всі проблеми, що виникають.

Основою кластера є не операційна система, а комунікаційне середовище (PVM, MPI), яке дозволяє, частинам паралельної програми, що виконуються на різних комп'ютерах, ефективно взаємодіяти між собою.

Однак, варто пам'ятати, що переважна більшість кластерів працює все-таки в середовищі UNIX. Сучасний рівень розвитку Linux дозволяє почувати себе користувачеві не менш комфортно, ніж в Windows.

У наш час основною операційною системою є операційна система Windows. При всіх достоїнствах системи їй властиві деякі недоліки, що істотно затрудняють її використання. До таких недоліків можна віднести:

- малу захищеність системи від некваліфікованих дій користувачів (студентів);
- схильність системи різного роду "зломом" при мережному використанні й схильність піддаватись вірусам;
- нестійкість роботи системи, що проявляється в зависаннях і втраті інформації;
- більша вартість ліцензій на використання;
- закритість операційної системи;
- більші вимоги до можливостей комп'ютера (пам'ять, швидкодія);
- часта зміна версій ОС (приблизно кожні два роки).

ОС Linux цілком може співіснувати з Windows на тому самому комп'ютері й завантажуватися тільки в міру необхідності. Використання Linux у якості базової операційної системи, крім можливості побудови кластера з наявних комп'ютерів, дозволить:

- більш ефективно використати наявні обчислювальні засоби;
- знизити витрати на обслуговування всієї системи (завдяки можливостям до гнучкого настроювання й чіткого відстеження прав доступу різних користувачів);
- вирішити проблеми з необхідністю придбання ліцензій на використовуване ПО;
- зробити роботу комп'ютерів у мережі й роботу всього класу більше надійною й стійкою.

Висновок

Таким чином, проаналізувавши класи завдань, які можна ефективно розпаралелити, мережне устаткування, недоліки та переваги операційних Windows та Linux ми бачимо, що створення кластерного суперкомп'ютера технічно досить проста процедура. Цілком працездатний кластер може бути створений досить швидко. Потужності сучасних процесорів цілком достатньо для рішення елементарних кроків більшості завдань, а об'єднання декількох десятків таких процесорів дозволяє швидко й ефективно вирішувати багато поставлених завдань, не вдаючись до допомоги мейнфреймів і супер комп'ютерів.

Зараз у наших наукових організаціях і університетах, як правило, є ентузіасти безкоштовного розповсюдженого ПО й фахівці з ОС Linux. У той же час парк більш-менш сучасних персональних комп'ютерів у цих організаціях так само є. Закономірно з'явилася ідея створення паралельних обчислювальних систем із загальнодоступних комп'ютерів на базі процесорів Intel і недорогих Ethernet-мереж, устанавив на ці комп'ютери Linux і, об'єднавши за допомогою однієї з безкоштовно розповсюджуваних комунікаційних бібліотек (PVM або MPI) ці комп'ютери в кластер. Виявилося, що на багатьох класах завдань і при достатнім числі вузлів такі системи дають продуктивність, порівнянну з тою, яку можна одержати, використовуючи дорогі суперкомп'ютери.

Література

1. Юрий Сбитнев Линукс. Параллельные вычисления. <http://linux-cluster.org.ru/download>.
2. Корнеев В.В. Параллельные вычислительные системы. – М.: «Нолидж», 1999. – 320 с., ил.
3. <http://twcad.ifmo.ru/?rub=cluster>

Надійшла 21.9.2009 р.