

УДК 004.272

Світлана Наконечна
Марина Семенюк

ОЦІНКА ПРОДУКТИВНОСТІ СИСТЕМ ПАРАЛЕЛЬНО-ІЄРАРХІЧНОЇ ПАМ'ЯТІ

При розробці паралельних обчислювальних систем та додатків, дуже важливо мати хороші інструменти щодо прогнозування продуктивності. У статті розглядається новий підхід – PetaSIM, який призначено для швидкого прототипування етапів проектування машин чи додатків. PetaSIM спрямовано на хороші, але не повністю точні результати із зручним інтерфейсом аплета для уточнення та інтерактивних змін параметрів системи.

При разработке паралельных вычислительных систем и приложений, очень важно иметь хорошие инструменты относительно прогнозирования производительности. В статье рассматривается новый подход – PetaSIM, который предназначен для быстрого прототипирования этапов проектирования машин или приложений. PetaSIM направлен на хорошие, но не полностью точные результаты с удобным интерфейсом аплета для уточнения и интерактивных изменений параметров системы.

In both the design of parallel computer systems and the development of applications, it is very important to have good performance prediction tools. This paper describes a new approach - PetaSIM, which is designed for the rapid prototyping stage of machine or application design. PetaSIM aims at good but not fully accurate results with a convenient applet interface for specification and interactive change of system parameters.

Ключові слова: ієрархічна пам'ять, оцінка продуктивності, PetaSIM, опис архітектури.

Вступ

Продуктивність прогнозування для великих завдань, таких як прогнозування погоди, пошук нафти чи фундаментальна фізика є важливими і складними проблемами, особливо, коли це стосується масивно паралельних комп'ютерів, де кожен вузол має складну ієрархію пам'яті. Це не так легко проектувати чи порівнювати нову архітектуру комп'ютера для продуктивності петафлоп, де треба протиставляти різні підходи і швидко вивчати нові поняття.

© Наконечна С.В., Семенюк М.С., 2012

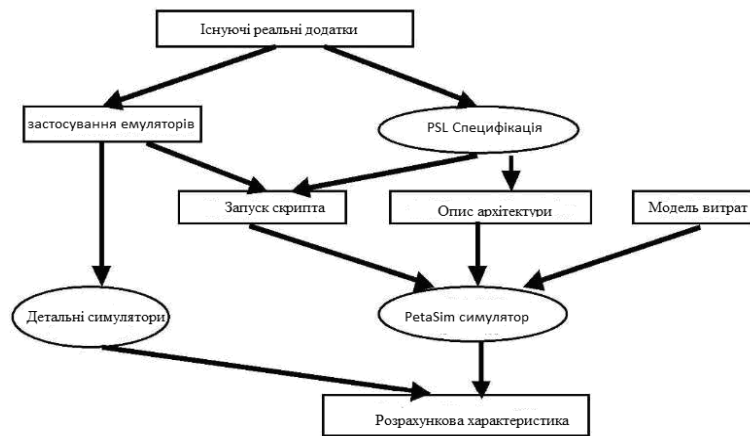


Рис. 1. Продуктивний процес прогнозування

Навіть сьогодні, проектування обчислювальних мереж для підтримки розподіленої спільної техніки не легке, тому що, навіть якщо система концептуально проста, то вона за своєю природою динамічна та гетерогенна і потребує підтримки системного інженера, який приймає точні стратегічні рішення по проектуванню мережі і розміщенню основних баз даних та програмних компонентів. PetaSIM була розроблена для вирішення таких проблем, приділяючи особливу увагу легко інтерактивному порівнянню різних конструкцій системи. Дуже зручно змінювати структуру програми PetaSIM, але потрібно зосередити увагу на тих випадках, коли є відносно фіксований набір додатків і потрібно швидко дослідити діапазон системи конструкцій. Інтерфейс Java-аплет [3,4], використовується для оптимізації інтерактивної оцінки.

Оцінка продуктивності PetaSIM будується навколо процесу виконання прогнозування, який зображено на рис. 1. Відмінною рисою нашого підходу є використання машин і проблем абстракцій, які хоч і менш точні, ніж детальне повне уявлення, але будуть більш надійними і, надалі, цілком підходять при необхідності швидкого прототипування в розробці нової техніки, програмного забезпечення та алгоритмів. Серцем процесу прогнозування продуктивності є дві технології – PSL (Performance Specification Language) (продуктивність специфікації мови) і PetaSIM [1]. Багато досліджень довели, що ефективність прогнозування та оцінка є дуже складною проблемою у зв'язку з різними видами додатків і безліччю різних комп'ютерних систем. [5-7] Важливо створити загальні специфікації мови (PSL) для підтримки оцінки продуктивності.

Мотивація та характеристика PetaSIM

Оцінка продуктивності PetaSIM спрямована на підтримку (концептуального і детального) етапів проектування паралельних алгоритмів, системи програмного забезпечення та апаратної архітектури. Спочатку PetaSIM була розроблена в результаті дослідів на двох семінарах – PAWS and PetaSoft. Найбільш складне пета ядро [1] було звичайною проблемою кінцевих різниць, що вирішилася завдяки простій ітерації Якобі. Однак, існуючі інструменти навіть не оцінили продуктивність цих простих ядер. Ці семінари підкреслили необхідність в інструментах, щоб описати складні ієрархії пам'яті і відображення проблем таким

ІНФОРМАЦІЙНІ, ТЕЛЕКОМУНІКАЦІЙНІ ТА ЕНЕРГОЗБЕРІГАЮЧІ ТЕХНОЛОГІЇ

чином, що дозволяє надійно (якщо високий рівень) оцінювати продуктивність на початку проектування і швидко прототипувати стадії.

PetaSIM спрямована на золоту середину - на півдорозі між детальним моделюванням на рівні машинних інструкцій і простого «зворотнього боку» оцінки продуктивності. Він піклується про складність – ієрархії пам'яті, затримки, адаптивності, які роблять важким високий рівень оцінки продуктивності. Він використовує важливе спрощення – розглядає дані природних блоків запропонованих пам'яттю систем, які прискорюють продуктивність моделювання і в багатьох випадках призводять до більш повного уявлення щодо покращення продуктивності. Мотивуємо та ілюструємо дизайн PetaSIM за відомою формулою для паралельного виконання простих звичайних додатків на вузлах без істотної ієрархічної пам'яті [2]: Прискорення = Кількість вузлів / (1 + Накладна).

Накладні витрати пропорційні $(\text{Grain_Size})^{-g} (t_{\text{comm}}/t_{\text{calc}})$, в цьому разі розмір блоку природних даних є розміром волокна або кількістю точок даних у кожному вузлі. Потужність g вимірює ефекти над краєм області та є $1/d$ для системи геометричних вимірів d ($t_{\text{comm}}/t_{\text{calc}}$) являє собою відношення зв'язку для розрахунку продуктивності обладнання.

Таку формула показує важливість виявлення природних блоків даних, а також, як високий рівень аналізу дозволяє зрозуміти співвідношення продуктивності розміру па'яті та обчислити можливості апаратного забезпечення. PetaSIM включає в себе не тільки примітивні повідомлення продуктивності, але також колективні (наприклад, груповий) механізми, які присутні у більшості додатків, але проігноровані в багатьох простих аналізах. Проста оцінка продуктивності є дійсно вищою на машинах з простою дворівневою розподіленою ієрархічною пам'яттю - саме пам'ять є ввімкненою або вимкненою на процесорі - яка, по суті є моделлю, побудованою на нинішнє покоління паралельних систем програмування, що уособлюють HPF або MPI. Таким чином, можна вважати, що наш досвід роботи з PetaSIM буде дуже цінним, допомагаючи розробці нового покоління в середовищі паралельного програмування, необхідного для підвищення комплексу високої продуктивності систем в оперативному режимі. Відзначимо ще одну важливу характеристику конструкції, яка може розглядатися як функція або помилка. Коли хтось один розробляє додатки і апаратні засоби, було б корисно мати можливість точно знати, як рухаються дані. У майбутніх версіях PetaSIM ця функція може бути доповнена можливістю автоматизації за допомогою конкретних стратегій управління кешем.

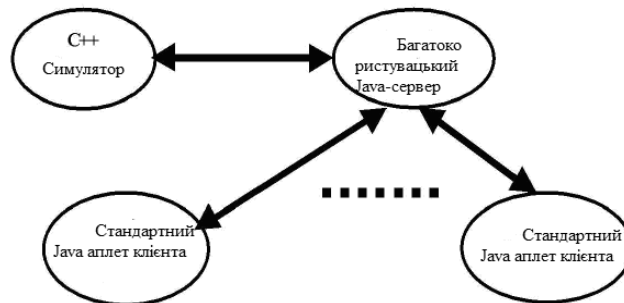


Рис. 2. Архітектура PetaSIM

Загальна Архітектура PetaSIM

Однією з найважливіших мотивацій PetaSIM є забезпечення виконання оцінки продуктивності інструменту для нового дизайну комп'ютерної архітектури, щоб отримати швидкий і точний прогноз продуктивності протягом проектування концептуальної архітектури. PetaSIM повинно бути простим в експлуатації і гнучким до зміни особливостей архітектури комп'ютера [3,4] Серцем PetaSIM є модель C++, яка приймає опис архітектури комп'ютера і опис програми, щоб дати оцінку продуктивності. Багатокористувацький Java Server надає послуги різним клієнтам Java-апплет з глобальної мережі Інтернет. Як показано на рис.3 PetaSIM підтримує обидва входи з емуляторів і рукописні коди.

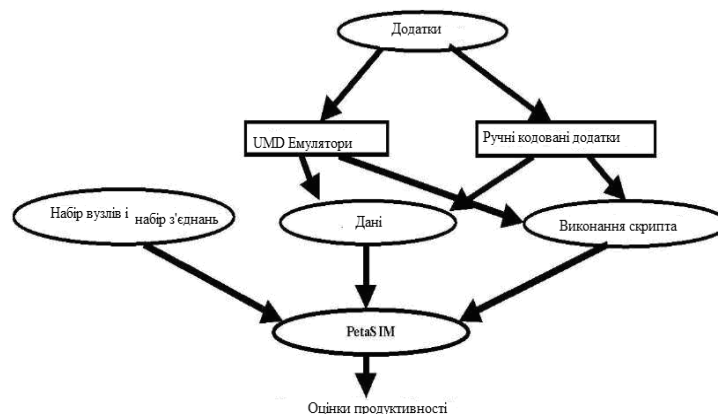


Рис. 3. PetaSIM: емулятори і застосування ручного кодування

Є дві частини опису в PetaSIM оцінки продуктивності системи, архітектурний опис та опис програми. Найзагальніша комп'ютерна архітектура може бути задана за допомогою вузлів PetaSIM та об'єктів з'єднань, тоді як додатки можуть бути визначені за допомогою набору даних і розподілу об'єктів.

Вузли являють собою сукупність осіб з поточних типів, що допускаються в якості:

- кеш пам'яті як особливого випадку
- диску, який по суті ж, як пам'ять.
- процесора, де можуть бути обчислені результати
- шляхи, такі як шини, перемикач або мережевий кабель, який концентрує дані

Набір з'єднань з'єднує разом вузли різними способами. Розподіли вказують на горизонтальне (геометричне) підключення вузлів і набори з'єднань. Зазвичай вони розташовані в природі за замовчуванням для класичних однорідних архітектур. Відображення за замовчуванням виводиться з розмірів вузлів і зроблено в стилі простого одномірного блоку. Вертикальне (потік інформації) підключення в архітектурі, визначено у вигляді виконання сценарію за замовчуванням, мається на увазі в специфікації архітектури. Застосування задається об'єктами даних, реалізація яких перебуває під контролем розподілу об'єкта, який визначає класичний стиль HPF геометричного розкладання через пам'ять і об'єкти процесора. Обчислення задається виконанням скрипта, який визначає переміщення даних.

ІНФОРМАЦІЙНІ, ТЕЛЕКОМУНІКАЦІЙНІ ТА ЕНЕРГОЗБЕРІГАЮЧІ ТЕХНОЛОГІЇ

Набір вузлів, набір з'єднань та набір даних мають відповідно такі властивості: назва, тип, розмір точок, пропускна здатність, обчислення тощо; назва, тип, топологія, кількість, затримка, пропускна здатність, розподіл, список членів набору вузлів; назва, тип, оновлення, передача.

Прототип PetaSIM для оцінки продуктивності системи був реалізований в NPAC. Він заснований на архітектурному описі та описі застосування (клас даних та розподіл класу, а також виконання сценаріїв для опису операцій над даними). Ці два види опису відповідають специфікації цільової машини, високому рівню представництва додатків і запуску скрипта в розробці PSL (продуктивність конкретної мови), який визначає вхідні дані для оцінки продуктивності PetaSIM.

В даний час у виконанні сценарію PetaSIM, підтримується чотири різних види заяв, передач, переміщення, використання розподілу, обчислення і синхронізації команд, яких вистачить, щоб висловити основні особливості переміщення даних і обчислювальних операцій в додатках. Однак ця проста абстракція може зробити виконання сценаріїв додатків дуже довгими, особливо для великих реалістичних програм. Таким чином, ми маємо намір побудувати бібліотеку виконання скрипта, яка міститиме агрегати переміщення даних і розрахунку за один виклик функції.

У PetaSIM продуктивність оцінки додатків, що працюють на конкретній машині ґрунтується залежно від особливостей цільової машини, даних з додатків, а також виконання сценаріїв застосування.

Взаємодія PetaSIM з PSL

Зараз PetaSIM займається описом архітектури і введенням опису програми з файлів. Маємо п'ять файлів, що відповідають безліччю вузлів, набору з'єднань, набору даних, розподілу опису та відповідно виконанню скриптів. PetaSIM доступні ці файли, використовуючи стандартне з'єднання розетки. В архітектурному описі машини, у нас є дві частини, набір вузлів і набір з'єднань, які докладно описують ієрархію пам'яті та зв'язок відносин між кожним рівнем ієрархії. Набір вузлів опису файлу містить всю інформацію про кожен вузол з набору, і всіх його членів з набору вузлів та зв'язки сусідів з кожним членом набору вузлів. Файл набору з'єднань містить інформацію про особливості кожного з'єднання, і членів набору з'єднань, які пов'язують разом члени двох наборів вузлів.

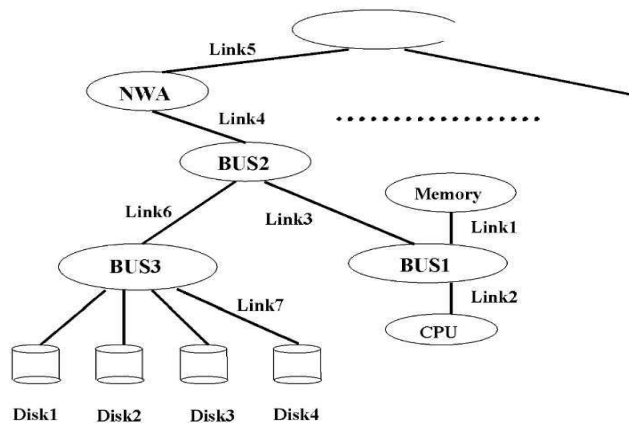


Рис. 4. Компоненти набору вузлів і набору з'єднань

ІНФОРМАЦІЙНІ, ТЕЛЕКОМУНІКАЦІЙНІ ТА ЕНЕРГОЗБЕРІГАЮЧІ ТЕХНОЛОГІЇ

Висновки

PetaSIM обґрунтовує свою оцінку продуктивності на кілька входів, а саме: опис архітектури комп'ютера, набір вузлів та набір з'єднань, опис програми, набір даних і розподілу, а також опис операції з даними, виконання сценарію при застосуванні. У порівнянні з деякими іншими оцінками продуктивності, PetaSIM має деякі особливості. PetaSIM використовує виконання сценаріїв для програми, зазначеної в текстовому форматі, який відповідає грубозернистому опису програми. Підхід PetaSIM з'явився, щоб впровадити більш інтуїтивно зрозумілий інтерфейс для додатків та опису ресурсів, які природно підтримують швидке прототипування досліджень в широкому діапазоні комп'ютерних архітектур. Можливості PetaSIM включають паралельне виконання оцінки та створення можливостей щодо вирішення нових типів завдань, намічених у вище зазначеному матеріалі.

ЛІТЕРАТУРА

1. "The Petaflops Systems Workshops", Proceedings of the 1996 Petaflops Architecture Workshop (PAWS), April 21-25, 1996 and Proceedings of the 1996 Petaflops System Software Summer Study (PetaSoft), June 17-21, 1996, edited by Michael J. MacDonald (Performance Issues are described in Chapter 7).
2. Geoffrey C. Fox, Roy D. Williams, and Paul C. Messina, Morgan Kaufmann, "Parallel Computing Works!", 1994.
3. Kivanc Dincer and Geoffrey C. Fox, "Using Java in the Virtual Programming Laboratory: A web-Based Parallel Programming Environment", to be published in special issue of Concurrency: Practice and Experience on Java for Science and Engineering Computation.
4. Geoffrey C. Fox and Wojtek Furmanski, Computing on the Web -- New Approaches to Parallel Processing-- Petaop and Exaop Performance in the Year 2007", submitted to IEEE Internet Computing, <http://www.npac.syr.edu/users/gcf/petastuff/petaweb/>
5. M. Rosenblum, S. Herrod, E. Witchel, A. Gupta, "Complete Computer System Simulation: The SimOS Approach", IEEE Parallel and Distributed Technology: Systems and Applications, 3(4), winter, 1995, pp 34-43.
6. E. Brewer, A. Colbrook, C. Dellarocas, W. Weihl, "Proteus: A High-Performance Parallel Architecture Simulator", Performance Evaluation Review, 20(1) Jun 1992, pp 247-8.
7. D. Park, R. Saavedra, "Trojan: A High-Performance Simulator for Shared Memory Architectures", Proceedings of the 29th Annual Simulation Symposium, April 1996, pp 44-53.