

СИСТЕМА ПАРКС-JAVA ЯК ЗАСІБ ВИРІШЕННЯ ПАРАЛЕЛЬНИХ АЛГОРИТМІВ НА КОМП'ЮТЕРНІЙ МЕРЕЖІ

Анісімов А.В., Дерев'янченко О.В.

Київський національний університет ім. Тараса Шевченка
факультет кібернетики
03127, м.Київ, просп. Академіка Глушкова 2, кор. 6, тел. (044)2590427
e-mail: ava@mi.unicyb.kiev.ua
der@mi.unicyb.kiev.ua

Система ПАРКС-JAVA забезпечує програмні конструкції та можливість зміни комутаційної структури при вирішенні задач паралельних обчислень на комп'ютерній мережі. Вона дозволяє загрузати, зберігати та передавати інформацію алгоритмічним модулям. ПАРКС-технологія застосовується для побудови паралельних програм, базованих на статичному та динамічному паралелізмі.

PARCS-JAVA system provide software construction and modification of logical and communication structure of a solved problem, storage and transmission of information, load by algorithmic modules. PARCS-technologies is used for supporting of parallel programming on the basis of static and dynamic parallelism.

Вступ

Метою даної роботи є створення системи підтримки виконання паралельних алгоритмів на комп'ютерній мережі. Реалізація даної системи може застосовуватися для рішення специфічних задач, що характеризуються високим ступенем регулярності. Клас таких задач досить широкий, він містить у собі такі задачі, як, наприклад, обробка зображень, рішення диференціальних рівнянь, моделювання тривалих у часі процесів, у тому числі й у реальному часі, багато інших наукових і інженерних задач, а також інші види потокової обробки даних. Значна частина з цих задач не може бути вирішена за реальний час на найбільш поширених на сьогодні персональних комп'ютерах. На сьогодні запропоновано досить багато програмних засобів різного рівня для опису паралельно функціонуючих та взаємодіючих систем.

Паралельні системи (ПС) характеризуються паралельним функціонуванням і взаємодією складових об'єктів. У ПС з динамічно змінюваною структурою об'єкти можуть динамічно створюватися й знищуватися, а зв'язки між ними - динамічно перекомутуватися.

Для таких систем необхідні програмні засоби адекватного опису динамічного паралелізму. Цікавий набір подібних засобів надає мова паралельного програмування Ада. У мовах паралельного програмування, реалізованих на трансп'ютерних комплексах (ОССАМ, Parallel C, Parallel Fortran і т.д.), таких засобів практично немає. Основний упор у них зроблений на розв'язання задач із статичним паралелізмом. Найпростіші засоби динамічного паралелізму в цих мовах підтримуються апаратом паралельних галузей (Parallel Execution Threads) і процесорних ферм (Processor Farms).

Система програмування ПАРКС (Паралельні Асинхронні Рекурсивно Керовані Системи), як і система паралельного програмування на трансп'ютерах, це сукупність програмних засобів, що підтримують процес розробки та реалізації алгоритмів паралельної обробки інформації за рахунок розширення базових алгоритмічних мов (JAVA, C, Pascal, Fortran, Modula-2). Відмінність системи програмування ПАРКС полягає в тому, що запропоновані нею програмні засоби сполучають найбільш могутні алгоритмічні концепції - рекурсію і паралельність та призначені для опису взаємодій зі змінною комутацією зв'язків і рекурсивно-паралельного розвитку процесів, які описуються на основі конструкцій базової мови. Акцент робиться на програмні засоби опису динамічного паралелізму. ПАРКС засоби забезпечують динамічне створення та знищення паралельних

процесів (із використанням рекурсії над даними та рекурсії над керуванням), динамічну перекомутацію зв'язків між процесами.

У роботі розглядається проектування однієї з таких систем, яка базується на технології ПАРКС [1-6]. Досліджуються практичні сторони використання цієї потужної технології. Представлена система ПАРКС-JAVA - розширення мови програмування JAVA засобами технології ПАРКС, надання розробникам паралельних алгоритмів зручних та надійних засобів взаємодії та управління процесами.

ПАРКС-технологія програмування

ПАРКС-технологія програмування є певною надбудовою над базовою платформою. Це може бути мова програмування, команди операційної системи, тощо. По суті ця надбудова є набором об'єктів – структур даних та класів, опис яких робиться на базовій мові в нашому випадку це JAVA, та дозволяє створювати керуючий простір (КП), що використовуються для паралельного (або псевдопаралельного) виконання процесів. Важливою особливістю ПАРКС-технології є можливість динамічно (у ході роботи програми) модифікувати структуру КП та керувати його діями. Така множина об'єктів називається ПАРКС-розширенням базової мови програмування.

Основними термінами ПАРКС-технології програмування є:

- точка;
- програмний канал (ПК);
- алгоритмічний модуль (АМ).

Структура КП - це граф, вершини якого - точки КП, а ребра - ПК, що їх з'єднують. При цьому одні й ті ж точки можуть бути з'єднані за допомогою кількох ПК різного типу. До кожної точки КП приписано АМ, що є процедурою ПАРКС - розширення базової мови (відповідний клас на JAVA). Виконання АМ може призвести до зміни у структурі КП: вилучення або створення точок, ПК, блокування виконання деяких процесів, тощо. АМ виконуються паралельно, однак модуль може бути затримано до надходження деякої події (умовно) або на деякий час (абсолютно). Точки КП можуть містити в собі керуючі підпростори будь-якої глибини, тобто КП може мати будь-який рівень рекурсії. Точки КП, з'єднані за допомогою ПК, можуть надсилати та отримувати повідомлення. Канали можуть бути типізованими, що дозволяє розподіляти повідомлення на класи, не декодуючи їх. Тип каналу задається цілим невід'ємним числом. Повідомлення - це рядок символів, а його значення визначається надісланим та прийнятим АМ.

ПАРКС - технологія дозволяє:

- підтримувати структурне паралельне програмування на основі як статичного, так і динамічного паралелізму;
- використовувати розвинуті засоби опису рекурсії за даними та рекурсії по керуванню, розглядати, у залежності від ситуації, керування як дані, а дані — як керування (з урахуванням можливої ефективної реалізації), забезпечувати за допомогою взаємодій не тільки синхронізацію й комунікацію, але й видозміну керування (на основі засобів базової мови);
- на логічному рівні в явному виді описувати розподіл ресурсів, схеми комутації та перекомутації зв'язків (з урахуванням рекурсивного розгортання алгоритмів) чи одержувати логічну структуру системи, розподіл ресурсів і схему перекомутації неявно як результат обробки й побудови КП;
- будувати систему віртуальних ПАРКС - машин, що забезпечують можливість паралельної обробки інформації на різних рівнях деталізації; ефективна реалізація отриманої системи віртуальних ПАРКС - машин можлива на широкому класі паралельних і спеціалізованих ЕОМ.

Система ПАРКС-JAVA

Система ПАРКС-JAVA реалізується (рис.1) на основі локальної чи глобальної комп'ютерної мережі обробки інформації, надаючи можливість користувачам малопотужних комп'ютерів використовувати ресурси мультипроцесорних комплексів, або ресурси комп'ютерної мережі. В обох випадках програмне забезпечення підтримує процеси взаємодії різних вузлів мережі між собою на основі обміну повідомленнями по мережі.

Розглянемо множину операторів паралельного програмування, яка реалізує ПАРКС-технологію програмування та надає можливість розширити базову мову JAVA до паралельної. Система має кілька модулів.

Головний модуль системи – `parcs`. Він містить реалізацію функцій системи ПАРКС-JAVA, що приймають участь в обчисленні. Основні компоненти:

клас `parcs.point` - об'єкт точка

`public point()` - створення нової точки

`public channel createChannel()` - створення каналу, що з'єднує поточну точку (у якій виконується код) із даною точкою, функція повертає посилання на канал

`public channel chan = null;` - посилання на канал зв'язку з точкою

`public HostInfo host = null;` - інформація про машину, до якої приписана точка

`public void setp(String AMname)` - приписати точці клас, що зберігається у файлі `AMname`, клас пересилається потрібній точці і запускається на виконання його метод `run`. Клас, ім'я якого передається як параметр функції, повинен реалізовувати інтерфейс `AM`, що містить єдиний метод `public void run(channel parent)`, з якого і починається виконання алгоритмічного модуля (`parent` – посилання на канал, що зв'язаний з батьківською точкою).

клас `parcs.channel` - об'єкт канал

`public void sendp(byte x)`

`public void sendp(int x)`

`public void sendp(long x)`

`public void sendp(double x)` - відіслати в канал змінну `x` заданого типу

`public void sendp(Serializable obj)` - відіслати об'єкт, який реалізує інтерфейс `Serializable`, тобто розроблений з можливістю запису в канал (більшість стандартних класів мови Java 2, що зберігають дані, реалізують цей інтерфейс)

`public void sendp(byte[] buf)` - відіслати в канал масив байтів

`public byte getp_b()` - отримати з каналу число типу `byte`

`public int getp_i()` - отримати з каналу число типу `int`

`public long getp_l()` - отримати з каналу число типу `long`

`public double getp_d()` - отримати з каналу число типу `double`

`public Object getp_o()` - отримати з каналу об'єкт

`public void getp(byte[] buf)` – отримати з каналу масив байтів та записати його.

Центральну роль у такій реалізації відіграє процес, що відпрацьовує основні функції КП: створює точки та канали, приписує `AM` точкам, підтримує взаємодію між точками у ході роботи. Умовна його назва – *демон* (клас `Daemon`). Спілкування з ним проходить або через локальний файл, або через потік вводу-виводу `TCP/IP`. Обидва способи спілкування нічим не відрізняються між собою з точки зору обробки їх, демоном. Але при бажанні, звичайно, можна виставити деякі пріоритети, які будуть враховуватись при обробці. Він повинен бути запущений на кожному комп'ютері мережі, що будуть приймати участь в обчисленнях. Демон постійно

знаходиться в пам'яті і приймає з'єднання з мережі на TCP-порт. При надходженні запиту на виконання алгоритмічного модулю вона породжує новий Java-потік, в якому запускає модуль на виконання.

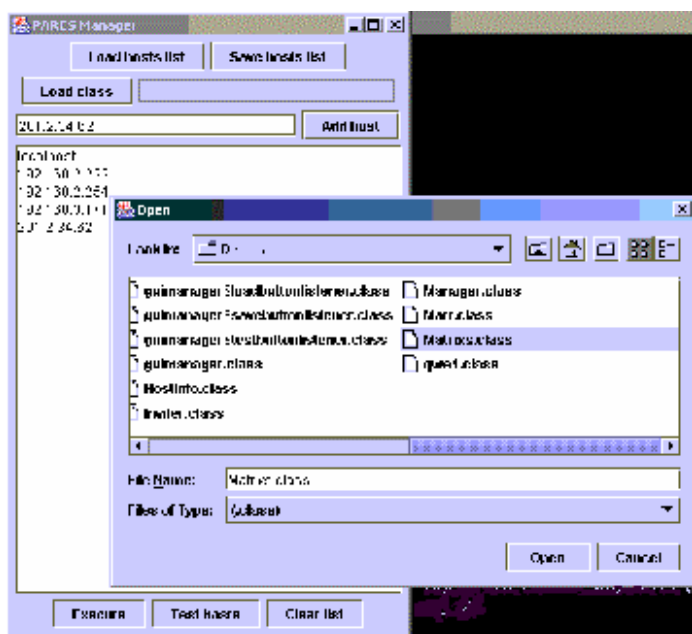


рис.1 Графічна оболонка системи ПАРКС-JAVA

Висновки

В результаті роботи отриманий програмний продукт, що може виконувати реальні обчислювальні задачі, дозволяє будувати динамічну чи статичну структуру керуючого простору та різні його конфігурації (дерево, список, кільце), тобто є достатньо гнучким. Виникає проблема оптимального керування паралельними обчислювальними процесами, яка є однією з найскладніших областей паралельних обчислень. Ефективність роботи паралельної обчислювальної системи ПАРКС-JAVA залежить не тільки від її конкретної структури, а і від того, як реалізується її виконання на конкретній обчислювальній системі, і від багатьох зовнішніх факторів.

Перспективним напрямком розвитку системи ПАРКС-JAVA, яка має платформенну незалежність, є можливість створити сервер у мережі Інтернет, що буде роздавати обчислювальні підзадачі машинам, власники яких бажають прийняти участь в обчислювальних проектах, і при цьому такий сервер зможе працювати з будь-якими клієнтами, незалежно від архітектури їх, машин, необхідна тільки підтримка клієнтом Java, яка сьогодні знаходить все ширшу підтримку в операційних системах.

Більш детально ознайомитися з системою можна на сайті: <http://www.parc.unicyb.kiev.ua>

Література

1. Анисимов А.В., Глушков В.М. Управляющие пространства в асинхронных параллельных вычислениях. // Кибернетика. --1980. --№5. -с.1-9.
2. Анисимов А.В., Кулябко П.П. Программирование параллельных процессов в управляющих пространствах. // Кибернетика. --1984. -- №3. --с.79-88.
3. Анисимов А.В., Кулябко П.П. Особенности ПАРУС-технологии. // Кибернетика и системный анализ.--1993.--№3.--с.128-137.
4. Анисимов А.В., Кулябко П.П. Моделирование сети Петри с помощью ПАРУС-средств. // Проблемы программирования. --1997. --вып. 2. с. 45-56.
5. Анисимов А.В., Дробот В.В., Кулябко П.П., Хабаева М.О. Мережі Петрі та засоби ПАРКС. // Вісник Київського університету. серія Фізико-математичні науки. -- 1997. --вип. 2. -- с. 113 – 134.
6. Анисимов А.В., Деревянченко А.В. Построение виртуального параллельного пространства с использованием технологии ПАРУС-JAVA. //Материалы Международной научно-технической конференции ИНТЕЛЛЕКТУАЛЬНЫЕ И МНОГОПРОЦЕССОРНЫЕ СИСТЕМЫ - 2003, Т.2 с. 18-19.