

МОДЕЛЮВАННЯ АСОЦІАТИВНОЇ ПАМ'ЯТІ ЗАСОБАМИ ПАРКС

На сьогодні функція пошуку інформації в більшості інформаційних систем є однією з головних і найважливіших. Водночас обсяги даних зростають швидкими темпами, оскільки впровадження сучасних потужних систем аналізу даних потребує зберігання навіть надлишкової інформації. Наприклад, нові версії операційної системи Windows збирають інформацію про будь-які дії користувача. На підставі цієї інформації надається контекстна реклама, а отже, є необхідність шукати користувачів за певним критерієм, і в цьому контексті асоціативна пам'ять є найкращим варіантом пошуку. У цій статті розглянуто підхід до організації моделювання асоціативної пам'яті засобами ПАРКС. Наведено схему підходу до організації керуючого простору та пояснення до неї. Також проведено аналіз прискорення та ефективності цього підходу.

Ключові слова: ПАРКС (паралельні асинхронні рекурсивні керуючі системи), моделювання, асоціативна пам'ять.

Асоціативна пам'ять (Content-Addressable Memory – CAM) являє собою пристрій, у якому реалізовані паралельне опитування масиву осередків пам'яті і знаходження слова за його змістом. Асоціативна пам'ять (АП) посідає значне місце у світі сучасної інформатики та інформаційних технологій. Системи на базі АП широко використовуються для пошуку інформації, розв'язання задач штучного інтелекту тощо. Крім того, методи АП знаходять застосування в моделюванні роботи мозку та вивченні психічних процесів живих істот, зокрема людини. Такий підхід застосовують тому, що пам'ять живих істот є асоціативною, на відміну від звичайної комп'ютерної пам'яті.

Існує велика кількість принципів та підходів до апаратної та програмної реалізації АП – від порівняно простих, на кшталт хеш-адресації, до досить складних методів пошуку в базах даних. На сьогодні вже існують апаратні засоби моделювання асоціативної пам'яті, але їхні розробники вимушені балансувати між швидкістю, розміром та вартістю такої пам'яті. Проблема полягає в тому, що, на відміну від чіпа RAM (Random-Access Memory), до кожної окремої комірки пам'яті в повністю паралельній асоціативній пам'яті має бути приєднана власна схема порівняння для виявлення збігу між збереженим і вхідним бітами. До того ж виходить від схем порівнянь кожної комірки в слові даних повинні бути об'єднані. Це потрібно для отримання повного результату при порівнянні даних розміром зі слово. Додаткова схема збільшує фізичний розмір чіпа асоціативної пам'яті,

що, своєю чергою, збільшує вартість виробництва. Додаткова схема також збільшує споживану потужність, оскільки всі схеми порівнянь активні на кожному такті. Як наслідок, асоціативна пам'ять використовується тільки в спеціалізованих програмах, де потрібної швидкості пошуку не можна досягнути іншими доступними способами.

Моделювання

У більшості інформаційних систем однією з основних і найважливіших є функція пошуку потрібної інформації. Оскільки обсяги даних зазвичай дуже великі, інформація постійно розміщується не в оперативній пам'яті, а на зовнішніх запам'ятовуваних пристроях і організована у вигляді файлів, що складаються із записів. Запис є одиницею пошуку та обміну між користувачем і файловою системою.

Методи доступу до файлів можна умовно поділити на два типи: послідовний перебір і прямий доступ. При такій широкій класифікації індексно-послідовний доступ може бути віднесений до виду прямих.

Послідовний перебір може бути ефективний лише для вузького кола задач, наприклад, задач побудови звітів тощо. Тобто в тих випадках, коли за умовою задачі потрібно перебрати всі або майже всі записи файлу.

При прямому доступі є можливість прочитати в оперативну пам'ять і опрацювати деякий запис, не застосовуючи прямий перебір. Однак для цього потрібно вказати ключ запису, який

буде перетворений в адресу запису на зовнішньому пристрої. Тут також можливий послідовний перебір, але серед незначної кількості записів. Тобто, інакше кажучи, прямий доступ ефективний тоді, коли критерій пошуку визначається значенням ключових атрибутів записів.

Інший підхід потрібен у тих випадках, коли критерій залежить від значень будь-яких атрибутів запису, а результатом пошуку є множина записів. Очевидно, що прямий доступ у чистому вигляді тут неможливий. Послідовний же перебір підходить за умови суттєвого обмеження кількості записів, що перебираються. У сучасних СУБД використовують різні методи, що забезпечують перебір лише частини записів файлу. У найзагальнішому вигляді цю задачу можна сформулювати в контексті теорії реляційних моделей баз даних, де потрібен симетричний пошук стосовно всіх атрибутів, що входять у відношення. Часто такий доступ називають асоціативним, а апаратні засоби та програмне забезпечення, що його підтримують, – асоціативною пам'яттю.

Нехай записи файлу нормалізовані за аналогією з нормалізованими відношеннями реляційної моделі, тобто представлені у вигляді рядка таблиці з атомарними атрибутами. Як буде видно далі, припущення не є принциповим, а введено лише для зручності розгляду. Формула доступу або елементарного запиту має вигляд: $P(k)OV$, де P – ім'я атрибута запису, V – значення, k – ключ, або адреса запису (або сам запис), а O – один із символів $\{=, \neq, >, <, \leq, \geq\}$. Тоді елементарний запит до асоціативної пам'яті символічно можна позначити: $P(?)OV$. Тобто P, O, V – відомі, а k потрібно знайти. Зауважимо, що ми поки що обмежені пошуком в одному файлі.

Для ускладнення задачі будемо розглядати запити, що представляють диз'юнктивну нормальну форму (ДНФ) від елементарних запитів. Крім того, на місці другого операнду в елементарному запиті ми будемо допускати не тільки константу, але й ім'я другого атрибута того ж запису, тобто допустимий вираз: $P1(?)OP2(?)$.

На рисунку показано структуру КП для моделювання АП. Спочатку алгоритмічний модуль (АМ) вузла С завантажує у вузли виду АНДИЗ (аналіз диз'юнкції) кортежі реляції; це ініціальне завантаження даних; кількість вузлів АНДИЗ відповідає кількості кортежів реляції. Потім АМ вузла АНЗАП (аналіз запиту) приймає черговий запит у вигляді ДНФ і розсилає його по вузлах виду АНДИЗ. Своєю чергою АМ вузла АНДИЗ аналізує ДНФ і розсилає окремі диз'юнкції (разом із фрагментами кортежу) по вузлах виду АНКОН (аналіз кон'юнкції), а той далі аналізує диз'юнкції

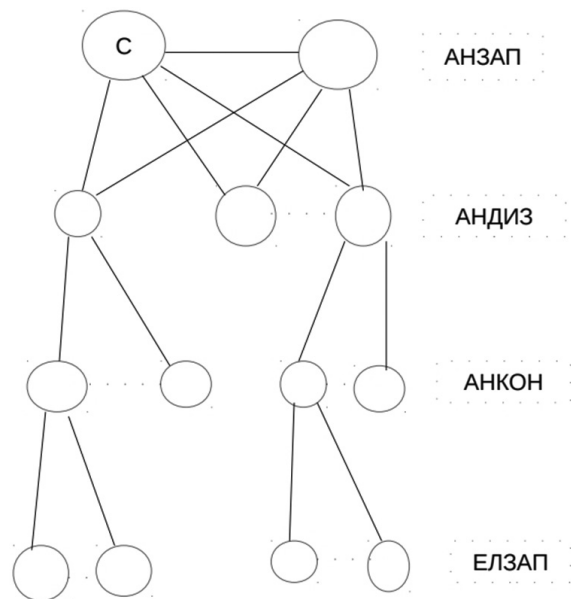


Рисунок. Моделювання асоціативної пам'яті

і розсилає окремі елементарні порівняння та потрібні фрагменти кортежу по вузлах виду ЕЛЗАП (елементарний запит). АМ ЕЛЗАПу виконує порівняння і передає нагору його результат TRUE чи FALSE. АМ АНКОНу збирає результати знизу, здійснює їх кон'юнкцію і передає логічний результат у відповідний вузол АНДИЗ. Алгоритмічний модуль АНДИЗу збирає результати знизу, виконує їх диз'юнкцію і передає АНЗАПу кортеж, якщо результат був TRUE, у разі FALSE передається спеціальний сигнал. Уся ієрархічна структура спочатку динамічно розгортається, а потім згортається, причому АНКОН завершує кон'юнкцію за першим же сигналом FALSE, передаючи його вище, а АНДИЗ завершує диз'юнкцію за першим же сигналом TRUE.

Прискорення та ефективність алгоритму моделювання асоціативної пам'яті

Прискорення: $\alpha = T_1 / T_p$, де T_1 – час, витрачений одним процесором, а T_p – час, витрачений p процесорами. Ефективність: $\beta = \alpha / p$.

Нехай n – число записів у файлі; t – час обробки одного запиту; m – число запитів; r – час на читання запиту чи кортежу (запису); p – число процесорів; $n + 2 = p$. Тоді:

$$T_1 = (n * (r + t) + r) * m;$$

$$T_p = n * r + m * (r + t);$$

$$\alpha = \frac{(n * (r + t) + r) * m}{n * r + m * (r + t)} = \frac{\left(n * \left(1 + \frac{t}{r} \right) + 1 \right) * m}{n + m * \left(1 + \frac{t}{r} \right)} \approx \frac{(n + 1) * m}{n + m}.$$

При $r \gg t$.

1) Нехай $n = m$, тоді:

$$\alpha = \frac{n+1}{2};$$

$$\beta = \frac{\alpha}{p} = \frac{n+1}{2*(n+2)} = 0.5 * \left(1 - \frac{1}{n+2}\right) \rightarrow 0.5, \text{ при } n \rightarrow \infty.$$

2) Нехай $m = k * n$, тоді:

$$\alpha \approx \frac{(n+1)*k*n}{n*(k+1)} = \frac{(n+1)*k}{k+1};$$

$$\beta \approx \frac{(n+1)*k}{(k+1)*(n+2)} = \left(\frac{k}{k+1}\right) * \left(\frac{n+1}{n+2}\right) \rightarrow \frac{k}{k+1}, \text{ при } n \rightarrow \infty.$$

Висновки

Враховуючи оцінки прискорення та ефективності, можемо зробити висновок, що моделювання асоціативної пам'яті може бути ефективним методом пошуку даних. Окремо слід зауважити, що пришвидшення залежатиме від реалізації алгоритмів ПАРКС. Враховуючи, що до керуючого простору ПАРКС можуть входити різноманітні апаратні засоби, зокрема апаратні

засоби моделювання асоціативної пам'яті, серверні комплекси та звичайні комп'ютери (незалежно від встановленої операційної системи), потрібно за можливості забезпечити ефективний розподіл даних (більша потужність процесора – більше даних).

Виділимо основні особливості змодельованої асоціативної пам'яті:

1) операції в пам'яті виконуються не над певним словом, а відразу над групою чи всіма словами;

2) час пошуку може не залежати від розміру одиниці даних (слова);

3) швидкість пошуку більше залежить від наявних фізичних ресурсів, ніж від обсягу даних.

Наведені вище особливості майже повністю збігаються з особливостями класичного розуміння «асоціативної пам'яті», а отже, системи ПАРКС можна використовувати для моделювання асоціативної пам'яті. Крім того, оскільки вони мають особливості асоціативної пам'яті, їх можна використовувати для побудови високопродуктивних систем пошуку, для яких недостатньо ресурсів лише однієї машини або сервера.

Список літератури

1. Анисимов А. В. Система ПАРУС-JAVA для паралельних вычислений на компьютерных сетях / А. В. Анисимов, А. В. Дерев'янченко // Кибернетика и системный анализ. – 2005. – № 1. – С. 25–36.
2. Анисимов А. В. Особенности ПАРУС-технологии / А. В. Анисимов, П. П. Кулябко // Кибернетика и системный анализ. – 1993. – № 3. – С. 128–137.
3. Дерев'янченко О. В. Моделювання паралельних програм за допомогою системи ПАРКС-JAVA / О. В. Дерев'янченко // Наукові записки НаУКМА. – 2005. – Т. 36 : Комп'ютерні науки. – С. 32–38.

O. Fedorus

MODELING ASSOCIATIVE MEMORY BY USING PARCS

Nowadays, information retrieval is one of the main and most important features in most information systems. At the same time, rapid data volumes grow as a result of usage of modern, powerful data analysis systems, which require the storage of superfluous information. For example, new versions of the Windows operating system collect information about any user actions (audio, keyboard input, etc.). Contextual advertising is provided on the basis of this information, and therefore there is a need in search for users according to some criterion, and in this context associative memory is the best search option. This article considers the approach to organization of associative memory modeling by means of PARCS. The article gives a scheme of the approach to organization of the control space and explanation to it. An analysis of the acceleration and effectiveness of this approach is also conducted.

Keywords: PARCS (Parallel Asynchronous Recursive Control System), modeling, associative memory.

Матеріал надійшов 19.10.2017