

ИССЛЕДОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ РЕАЛИЗАЦИИ ПАРАЛЛЕЛЬНОЙ РЕДУКЦИОННОЙ СХЕМЫ ТЕОРЕТИКО-МНОЖЕСТВЕННЫХ ОПЕРАЦИЙ

Ивашенко В. П.

Кафедра интеллектуальных информационных технологий, Факультет информационных технологий и управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: ivashenko@bsuir.by

В работе приведены результаты исследований для обработки информации на уровне управления знаниями, представленными средствами модели унифицированного семантического представления знаний, особенностью которой является наличие базовой теоретико-множественной семантики представленных знаний.

ВВЕДЕНИЕ

В технических системах, способных к обучению и ориентированных на решение задач, к которым относятся системы, управляемые знаниями (СУЗ) [1], существенным является качество масштабируемости [2] за счёт соблюдения требований по ключевым качествам, включая производительность. С целью обеспечения соблюдения этих требований предлагаются различные модели обработки информации для разных уровней управления СУЗ [1], [3].

I. РЕДУКЦИОННАЯ СХЕМА ДЛЯ РЕШЕНИЯ ЗАДАЧИ

Рассмотрим задачу, удовлетворяющую двум требованиям: 1) входными данными задачи являются линейно упорядоченные множества данных (массивы); 2) её решение сводится к пошагово-последовательному чтению данных последовательно из ячеек этих массивов и возможной записи в качестве выходных данных результата, вычисленного на основе прочитанных на текущем шаге данных и данных, полученных в результате выполнения операции редукции (ОР) – бинарной ассоциативной операции, применённой к прочитанным данным текущего шага и аналогичному результату ОР предыдущего шага. Следовательно, решение задачи может быть разбито на две фазы вычислений: результата ОР и окончательного результата. Очевидно, что задача в фазе 2 является частным случаем исходной задачи для одноэлементных массивов входных данных. С целью параллельной обработки массивы разбиваются на части. Чтобы не хранить результаты ОР для каждого элемента массивов, предлагается хранить результат ОР только для начальных элементов частей массивов, а для всех последующих элементов вычислять результат ОР совместно с вычислением окончательного результата, т. е. решать частный случай исходной задачи целиком. Для первых частей массивов решение задачи может быть начато сразу, без решения задачи фазы 1, так как

не требуется расчёт результата ОР ввиду отсутствия предшествующих элементов, также не требуется расчёт результата ОР – решение задачи фазы 1 – для последних частей массивов, так как после них нет каких-либо частей, которые бы требовали этот результат.

Коэффициент ускорения (КУ) решения задачи на p процессорах можно выразить как $(k + p)/(k + 1)$, при этом исходные массивы разбиваются на $p + 1$ часть, а исходная задача сводится к задачам обработки этих частей; полагая, что отношение времени решения задачи фазы 1 ко времени решения всей задачи составляет k , то, чтобы выравнять время решения всех параллельно решаемых задач на первом этапе, первая часть в k раз короче остальных. Соответствующий график для $k = 1$ представлен на рисунке 1, а редукционная схема (РС) – на рисунке 4 (светло-серым показана фаза 1 решения задачи, а тёмно-серым – решения остальных задач для частей).

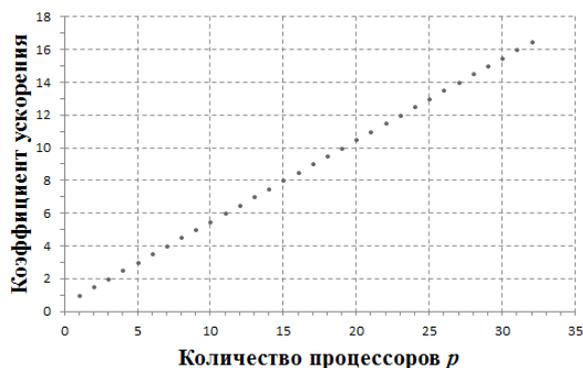


Рис. 1 – Теоретическая зависимость коэффициента ускорения от количества процессоров

II. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Для экспериментов были выбраны задачи вычисления результатов операций пересечения, объединения и симметрической разности множеств, заданных парой массивов с n и m 32-битными элементами ($n \leq m$). Тестирование проводилось на архитектуре AMD Ryzen Threadripper 2950X (8Гиб ОЗУ, Windows 10 x64)

для размеров множеств от 1 до 10^8 . В качестве последовательного решения были взяты алгоритмы, обеспечивающие временные сложности решения задач: $O(n * \ln(m/n) * f(n + m))$ – для пересечения, $O((n+m) * f(n+m))$ – для объединения и симметрической разности, где $f(x)$ – время доступа к элементу массива размера x . Элементами для пар множеств брались натуральные числа случайно из диапазона от 1 до $2 * (n + m)$, а также – комбинации чётных, нечётных и т. п. наборов чисел из этого диапазона, чисел от его начала и с конца, одинаковые наборы и наборы одинаковых чисел. Для разбиения на части применялся подход MergePath [4]. Графики результатов экспериментов приведены на рисунках 2 и 3.

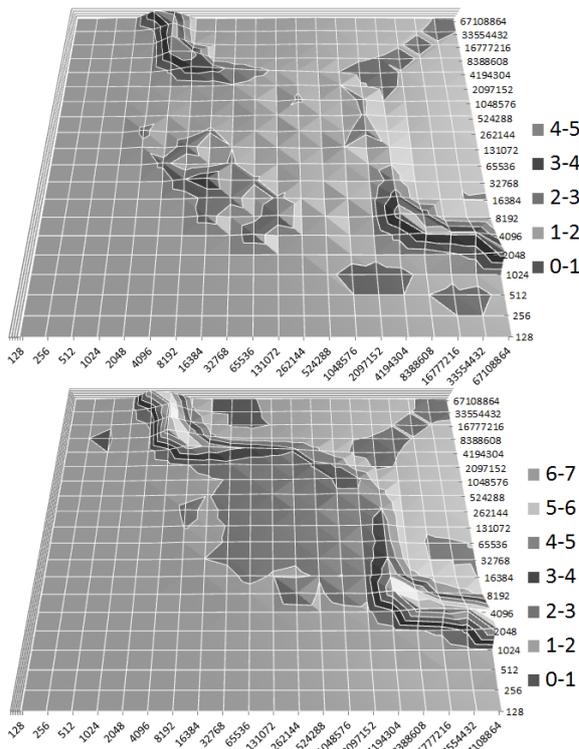


Рис. 2 – КУ пересечения мультимножеств для 4х (сверху) и 8и (снизу) потоков

Аномально высокие значения КУ для операции пересечения можно объяснить особенностью работы кэш-памяти и спецификой задачи пересечения мультимножеств. В остальном полученные значения КУ соответствуют теоретическим. Влияние архитектуры существенно: выигрыш от применения параллельной РС отсутствует или незначителен для множеств, ёмкость которых не превышает 32КиБ (размер кэша первого уровня) либо превышает 16МиБ (размер кэ-

ша третьего уровня на пару ССХ). Увеличение количества потоков более 8и не даёт существенного прироста в силу особенностей (NUMA) архитектуры AMD Ryzen Threadripper 2950X.

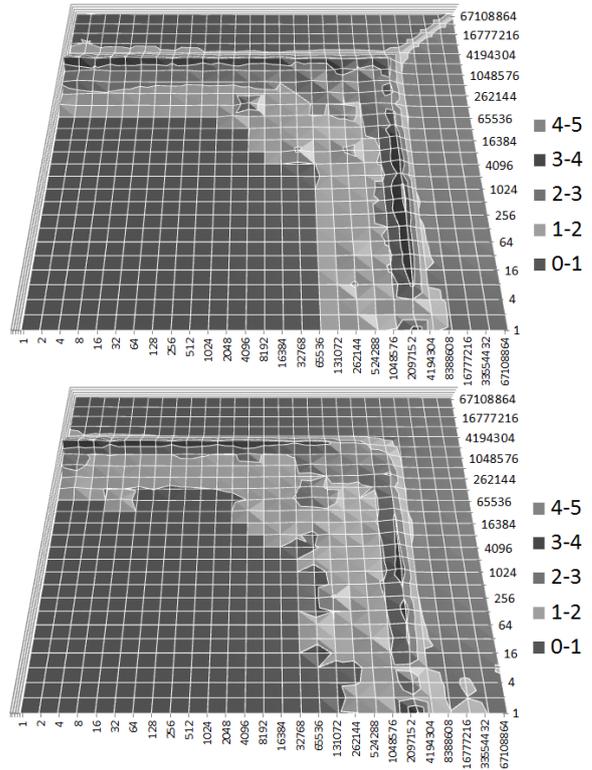


Рис. 3 – КУ симметрической разности (сверху) и объединения (снизу) мультимножеств (8 потоков)

ЗАКЛЮЧЕНИЕ

Эксперименты показали работоспособность предложенной РС, её согласованность с теоретическими оценками и возможность её применения для повышения производительности операций обработки знаний в СУЗ.

1. Ивашенко, В. П. Операции управления массивами данных в линейно адресуемой памяти / В. П. Ивашенко, С. В. Синцов // Доклады БГУИР. – Vol. 100, №6, – Минск, 2016. – С. 86–93.
2. Hill, M. D. What is scalability? / M. D. Hill // ACM SIGARCH Computer Architecture News. – Vol. 18, №4. – 1990. – pp. 18–21.
3. Ivashenko, V. P. String processing model for knowledge-driven systems / V. P. Ivashenko // Doklady BGUIR. – Vol. 18, №6, – Минск, 2020. – pp. 33–40.
4. Green, O. GPU merge path: A GPU merging algorithm / O. Green, R. McColl, D. A. Bader // Proceedings of the 26th ACM International Conference on Supercomputing, ser. ICS '12. – NY, USA: ACM, 2012. –pp. 331–340.



Рис. 4 – Редукционная схема решения задачи