

# Исследование эффективности применения сжатия при передаче данных из основной памяти в память графического ускорителя\*

С.О. Приказчиков, П.С. Костенецкий

Южно-Уральский государственный университет

Одной из основных проблем при работе с графическими ускорителями (ГПУ) является низкая скорость передачи данных из основной памяти в память ГПУ. В рамках данной работы была исследована эффективность применения сжатия для ускорения обмена данными с ГПУ при обработке баз данных. Рассмотрены два метода сжатия: RLE и Null Suppression. Проведенные вычислительные эксперименты показали, что сжатие может быть эффективно использовано для ускорения обработки баз данных на ГПУ.

В настоящее время существует большое количество задач, требующих параллельной обработки сверхбольших баз данных [5, 10, 12]. Использование при обработке баз данных многоядерных сопроцессоров и ГПУ может дать существенный прирост производительности [1, 11], однако узким местом становится шина передачи данных PCI Express [3, 8]. В работе [2] было показано, что эффективным при обработке баз данных с использованием многоядерных и графических сопроцессоров является подход, при котором данные передаются на сопроцессор и обратно в сжатом виде. Данный подход был исследован при помощи программного эмулятора [3, 1]. Целью данной работы является оценка эффективности применения сжатия при передаче данных из основной памяти в память графического ускорителя.

Для исследования эффективности использования сжатия в СУБД, использующей ГПУ, были выбраны два алгоритма: *RLE* и *Null Suppression*. Данные алгоритмы позволяют выполнять многие реляционные операции над сжатыми данными без предварительной распаковки. Алгоритмы, требующие предварительную распаковку данных перед обработкой (например, LZSS) не рассматривались, так как это менее эффективно при обработке базы данных на многоядерных сопроцессорах [1].

В Лаборатории суперкомпьютерного моделирования ЮУрГУ установлен суперкомпьютер «Торнадо ЮУрГУ» с 384 сопроцессорами Intel Xeon Phi, на котором уже производились исследования в области параллельной обработки баз данных [4]. В декабре 2014 года, для расширения спектра исследуемых технологий, введен в эксплуатацию небольшой вычислительный комплекс на базе трех серверов с двенадцатью ГПУ NVIDIA Tesla K40m. Характеристики сервера комплекса, использовавшегося для экспериментов, приведены в таблице 1.

Таблица 1 Характеристики оборудования для вычислительных экспериментов

| Оборудование           | Характеристики                 |                   |
|------------------------|--------------------------------|-------------------|
| Процессор              | Intel Xeon E5-2687W v2 – 2 шт. |                   |
| ОЗУ                    | 128 Гб DDR3 ECC Registered     |                   |
| Системная шина         | PCI Express 3.0                |                   |
| Графический ускоритель | Модель                         | NVIDIA Tesla K40m |
|                        | Ядер CUDA                      | 2880              |
|                        | Тактовая частота ядра          | 745 МГц           |
|                        | Тактовая частота памяти        | 6000 МГц          |
|                        | Пропускная способность памяти  | 288 Гб/с          |
|                        | Разрядность памяти             | 384-bit           |

\* Работа выполнена при финансовой поддержке Минобрнауки РФ в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2014–2020 годы» (Соглашение № 14.574.21.0035).

Для выполнения вычислительных экспериментов был разработан программный эмулятор, выполняющий передачу сжатого отношения базы данных на ГПУ и выполняющий его обработку. Для реализации выбранных методов использовался язык программирования C++ и библиотека CUDA Toolkit v 6.5. Вычислительные эксперименты производились на базе данных объемом 1.5 ГБ. Для каждого метода приводится зависимость коэффициента сжатия от выбранного параметра сжимаемых данных, который характеризует эффективность этого метода.

Степень сжатия методом RLE зависит от размера серий повторяющихся элементов. Поэтому в качестве параметра сжимаемых данных для данного метода было выбрано отношение количества серий к общему количеству элементов данных. Зависимость степени сжатия от данного параметра показана на рис. 1. На рис. 2 показана зависимость времени обработки данных, сжатых методом RLE от выбранного параметра. На графике видно, что применение метода сжатия RLE при обработке данных позволяет получить существенный прирост производительности, пропорциональный степени сжатия.

Степень сжатия методом Null Suppression зависит от количества нулевых байт, которые можно удалить из каждого элемента. Поэтому в качестве параметра сжимаемых данных для этого метода выбран оставшийся после сжатия объем элементов. Зависимости степени сжатия и времени обработки от объема элемента в байтах показаны на рис. 3. и рис. 4. На графиках видно, что при применении метода сжатия Null Suppression, наблюдается прирост производительности. Неравномерность графика объясняется тем, что для хранения элементов сжатых данных использовались стандартные типы данных, занимающие 1, 2, 4 и 8 байт. От использования нестандартных типов данных и побитовых операций в данном методе пришлось отказаться, так как это значительно снижало производительность алгоритмов обработки данных на сопроцессоре.

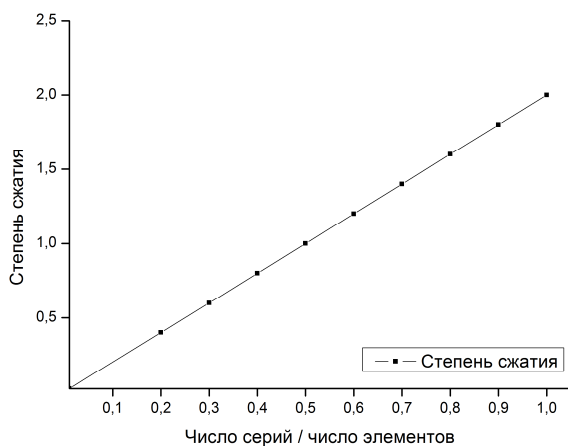


Рис. 1. Степень сжатия методом RLE

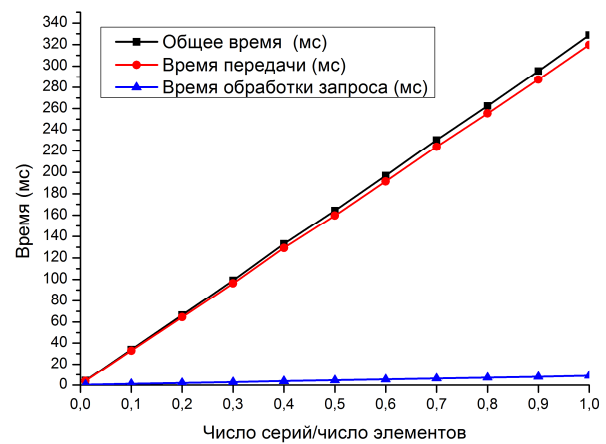


Рис. 2. Время обработки данных, сжатых методом RLE

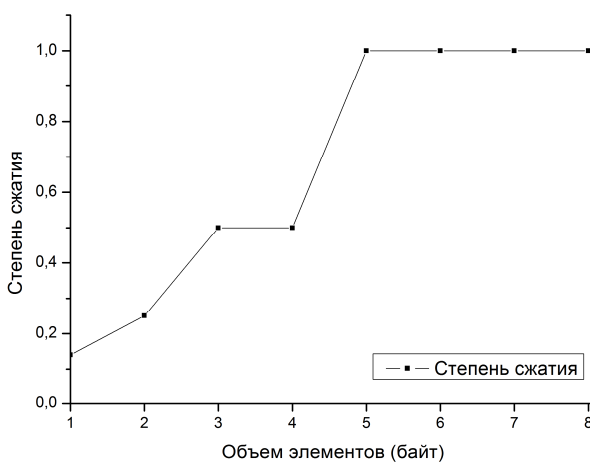


Рис. 3. Степень сжатия методом Null Suppression

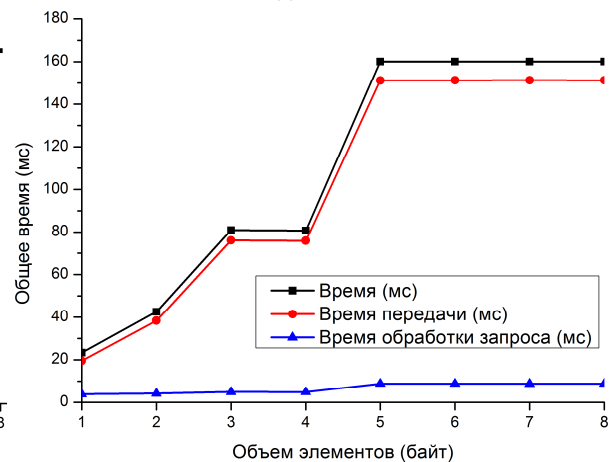


Рис. 4. Время обработки данных методом Null Suppression

Проведенные эксперименты показывают эффективность применения сжатия при обработке баз данных на ГПУ.

Дальнейшим направлением исследования будет математическое моделирование обработки запросов в параллельной системе баз данных, использующей многоядерные сопроцессоры и графические ускорители [5, 6], а также моделирование обработки данных на сопроцессорах в сжатом виде [2, 7, 9].

## Литература

1. Костенецкий П.С., Беседин К.Ю. Исследование эффективности различных методов сжатия при передаче данных из основной памяти в память сопроцессора Intel Xeon Phi // Вычислительные методы и программирование. 2014. Т. 15 № 4 С. 593–601.
2. Беседин К.Ю., Костенецкий П.С. Моделирование обработки запросов на гибридных вычислительных системах с многоядерными сопроцессорами и графическими ускорителями // Программные системы: теория и приложения: электрон. научн. журн. Института программных систем им. А.К. Айламазяна РАН. 2014. Т. 5, № 1(19). С. 91–110.
3. Беседин К.Ю., Костенецкий П.С. Применение многоядерных сопроцессоров в параллельных системах баз данных // Параллельные вычислительные технологии (ПаВТ'2013): труды международной научной конференции (1–5 апреля 2013 г., г. Челябинск). Челябинск: Издательский центр ЮУрГУ, 2013. С. 583.
4. Костенецкий П.С., Лепихов А.В., Соколинский Л.Б. Технологии параллельных систем баз данных для иерархических многопроцессорных сред // Автоматика и телемеханика. 2007. № 5. С. 112–125.
5. Костенецкий П.С., Соколинский Л.Б. Моделирование иерархических многопроцессорных систем баз данных // Программирование. Москва: МАИК «Наука/Интерпериодика». Т. 39, № 1. 2013. С. 3–22.
6. Костенецкий П.С., Соколинский Л.Б. Моделирование параллельных систем баз данных: учебное пособие. Челябинск: Фотохудожник, 2012. 78 с.
7. Осипова А.М., Костенецкий П.С. Моделирование мультипроцессоров систем баз данных с многоядерными ускорителями // Научный сервис в сети Интернет: все грани параллелизма: Труды Международной суперкомпьютерной конференции (23-28 сентября 2013 г., Новороссийск). М.: Изд-во МГУ, 2013. С. 194.
8. Приказчиков С.О., Костенецкий П.С. Применение графических ускорителей для обработки запросов над сжатыми данными в параллельных системах баз данных // Научный сервис в сети Интернет: многообразие суперкомпьютерных миров: Труды Международной суперкомпьютерной конференции (22–27 сентября 2014 г., Новороссийск). М.: Издательство МГУ, 2014. С. 277–279.
9. Сафина Ю.Н., Костенецкий П.С. Моделирование аппаратной архитектуры и коммуникационных сетей вычислительных кластеров с гибридными узлами для параллельных систем баз данных // Параллельные вычислительные технологии (ПаВТ'2012): Труды международной научной конференции (г. Новосибирск, 26–30 марта 2012 г.) Челябинск: Издательский центр ЮУрГУ, 2012. С. 741.
10. Пан К.С., Цымблер М.Л. Разработка параллельной СУБД на основе последовательной СУБД PostgreSQL с открытым исходным кодом // Вестник ЮУрГУ. Серия «Математическое моделирование и программирование». 2012. № 18(277). Вып. 12. С. 112–120.
11. Иванова Е.В., Соколинский Л.Б. Использование распределенных колоночных индексов для выполнения запросов к сверхбольшим базам данных // Параллельные вычислительные технологии (ПаВТ'2014): труды международной научной конференции (1–3 апреля 2014 г., г. Ростов-на-Дону). Челябинск: Издательский центр ЮУрГУ, 2014. С. 270–275.
12. Соколинский Л.Б. Параллельные машины баз данных // Природа. 2001. № 8. С. 10-17.