

## ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ NVIDIA CUDA ДЛЯ РАСЧЁТА ГИСТОГРАММ БОЛЬШОГО ОБЪЁМА.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Кулак В. Н.

Демидчук А. И. – ассистент кафедры ЭВМ

В работе представлены два эффективных алгоритма расчёта гистограмм с использованием архитектуры параллельных вычислений NVIDIA Cuda. Данные алгоритмы могут использоваться для параллельного вычисления гистограмм большого объёма данных и для тысячи ячеек (bins). Традиционно вычисление гистограмм на GPU было трудным и неэффективным. Зачастую алгоритмы, основанные на GPU, требуют вычисления гистограмм, как часть своих вычислений, которые нуждаются в переносе данных между GPU и хост-памятью, что является узким местом. Данные алгоритмы устраняют необходимость в такой дорогостоящей передаче данных, позволяя эффективно рассчитывать гистограммы на GPU.

Исследование проводилось с использованием наиболее популярной технологии в научных вычислениях – Nvidia CUDA.

CUDA (англ. Compute Unified Device Architecture) — программно-аппаратная архитектура, позволяющая производить вычисления с использованием графических процессоров Nvidia, поддерживающих технологию GPGPU.

Гистограмма представляет собой фундаментальный статический инструмент, который широко используется в различных областях, таких как обработка изображений, машинного обучения и интеллектуального анализа данных. Взвешенная гистограмма часто служит ключевым компонентом обработки их массивных наборов данных.

Первый метод основан на моделировании мьютекса путём пометок ячеек памяти и продолжением обновления памяти до тех пор, пока данные не будут успешно записаны. После этого тег сохраняется. Второй метод поддерживает гистограммы матрицы  $V \times N$  размер, где  $V$  является количеством ячеек (bins), а  $N$  является количеством нитей. Это предоставляет возможность свободно от коллизий обновлять память каждой нитью. В конечном счёте, выполняется параллельная редукция на матрице. Данные счетчиков вдоль строк комбинируются для получения конечной гистограммы.

В работе рассмотрены узкие места для параллельной реализации места данных алгоритмов, для них разработаны решения в рамках технологии CUDA.

Рис. 1 показывает, что первый метод значительно превосходит по производительности CPU. Второй метод, также превосходит CPU производительности, но, как кажется, на первый взгляд уступает первому методу. Оба метода имеют свои недостатки. Недостатком первого метода является то, что пропускная способность зависит от распределения данных. Случай, когда на вход поступает случайный массив с равномерным распределением, близок к идеалу, вероятность возникновения коллизий при расчёте минимальна. Случай, когда на вход поступает массив с вырожденное распределение, является слабым местом данного алгоритма. Это обусловлено тем, что все нити пытаются обновить одну и ту же ячейку (bin), что приводит к возникновению коллизий. Производительность расчёта гистограммы на CPU является почти постоянной относительно количества ячеек (bins) и не зависит от распределения входных данных. Второй метод представляет собой альтернативное решение проблемы связанной с зависимостью производительности от распределения входного массива. Преимущество в том, что учитывая размер глобальной памяти, практически для любого числа ячеек (bins), алгоритм требует только одну итерацию для завершения. Кроме того, не будет одновременного обновления одних ячеек памяти несколькими нитями, это в свою очередь значит, что производительность метода не зависит от распределения входных данных.

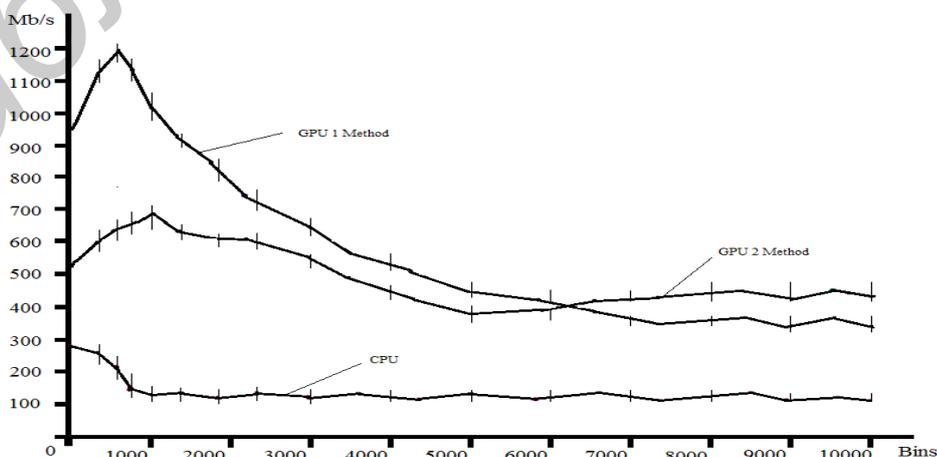


Рис. 1 Производительность расчёта гистограмм с использованием рассматриваемых методов в зависимости от количества ячеек (bins)