

Провал производительности GPU в алгоритмах с произвольным доступом к памяти

А.В. Кашковский, А.А. Шершнёв

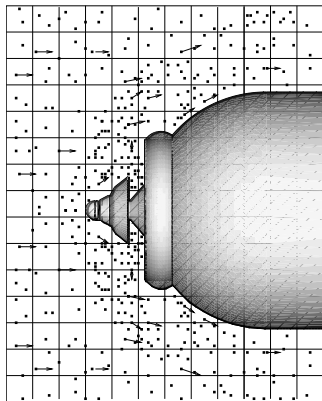


Институт теоретической и прикладной механики
им. С.А Христиановича СО РАН
630090, Институтская 4/1, Новосибирск e-mail: sasa@itam.nsc.ru

16 февраля 2017 г.

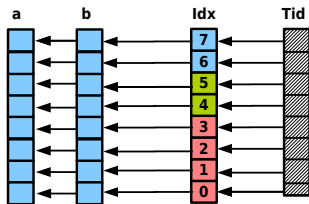
Введение

- Создан код, моделирующий течение разреженного газа методом ПСМ параллельно на нескольких GPU
- Код успешно эксплуатируется на разных платформах, в том числе в ССКЦ, на Tesla M2090 (6 Гб памяти каждая)
- Появилась возможность проводить вычисления на кластере с GPU Tesla K40m (12 Гб памяти)
- Ожидалось, что K40m в ~ 1.5 раза быстрее, чем M2090
- Оказалось, что K40m в $\sim 1.5 - 4$ раза медленнее, чем M2090
- Причина:
использование алгоритма с произвольным доступом к памяти

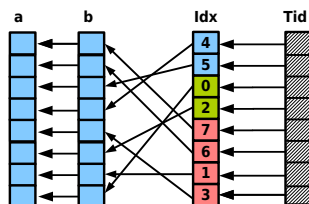


Тест скорости доступа к памяти

```
--global__ void cuSum(  
double *a, // data array 1 [len]  
double *b, // data array 2 [len]  
int *idx, // index array [len]  
int len // length of array  
)  
{  
    register int tid = blockDim.x * blockIdx.x + threadIdx.x;  
    register int nth = blockDim.x * gridDim.x;  
    register int i;  
    for( ;tid < len; tid+=nth )  
    { i = idx[tid];  
      a[i] += b[i];  
    }  
}
```

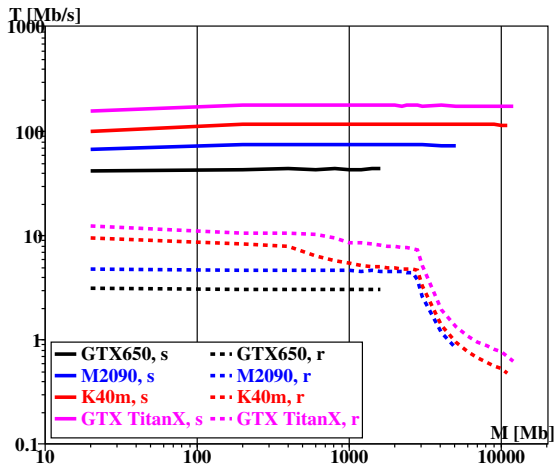


Последовательный доступ



Произвольный доступ

Результаты тестирования



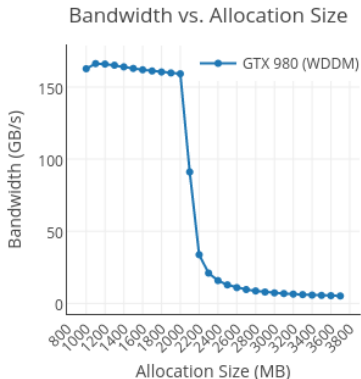
Память: GTX 650 – 2 Гб; M2090 – 6 Гб; K40m, GTX Titan X – 12 Гб.
s - последовательный, r - произвольный

Другие тесты

- Провал после 2-3 Гб наблюдается на многих GPU.
- <https://devtalk.nvidia.com/default/topic/940304/cuda-programming-and-performance/grim-memory-bandwidth-gtx->

Провал после 1 (!) Гб

- <https://gist.github.com/allanmac/f91b67c112bcba98649d>
allanmac, probe_bw
тест:



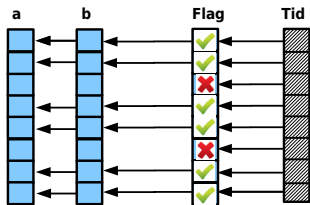
- По всей видимости , связано с архитектурой GPU

Потенциально проблемные задачи

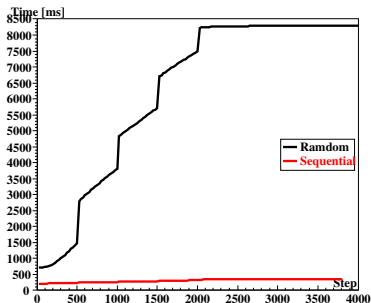
- Методы, использующие частицы и ячейки. Например:
 - ПСМ
 - методы молекулярной динамики
 - PIC-MCC
 - и другие
- Задачи, использующие графы или бинарные деревья
- CFD на неструктурированных и гибридных сетках
- **!!!** в программе NuCFS (решающей уравнения Навье-Стокса на структурированной сетке) провал в производительности не выявлен.

Способы улучшения производительности

- Опции компилятора ! сомнительно !
- Переход на новую архитектуру Pascal ???
Tesla P100 с широкополосной памятью HBM2 (4096 бит) <—> Tesla K40 с памятью GDDR5 (384 бита)
- Переделка алгоритма из произвольного доступа в последовательный



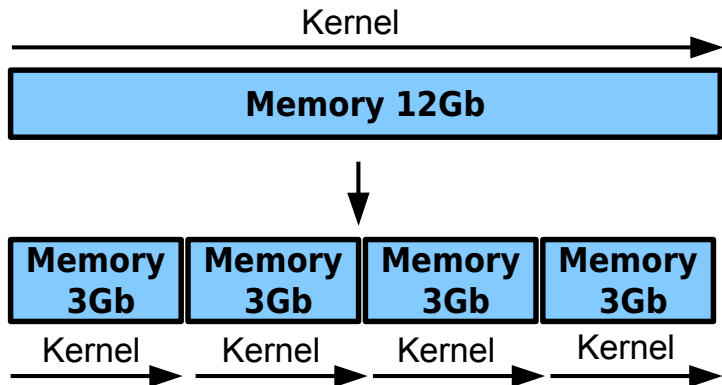
Алгоритм с флагом



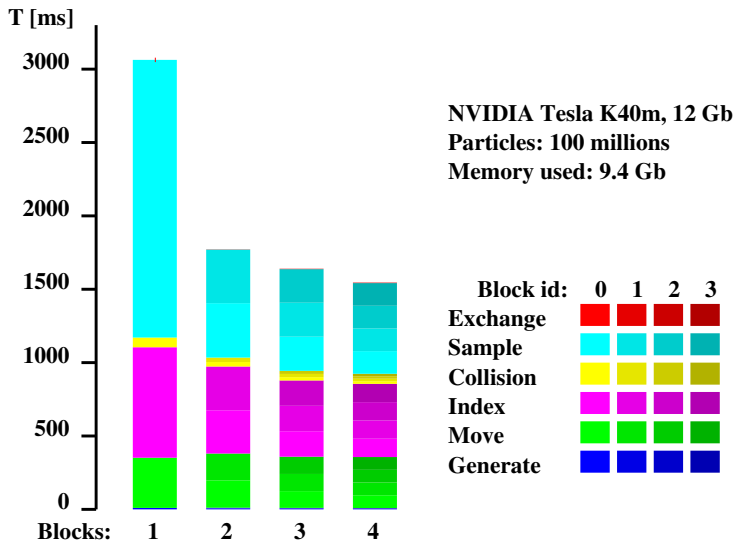
Время выполнения операции Move
(Равномерный поток)

Способы улучшения производительности

- Использовать domain decomposition для разделения всей памяти на блоки, которые выполняются на одном GPU последовательно.



Равномерный поток

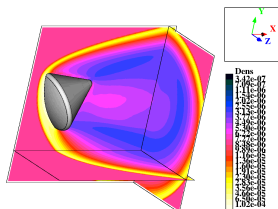


Итого, время вычислений уменьшилось с 11.1 до 1.5 с/шаг (> 7 раз)

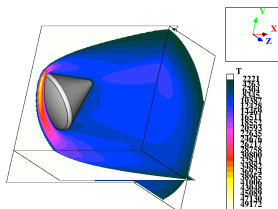
Обтекание КА Аполлон

$V = 9600 \text{ м/с}$, $\alpha = -25^\circ$, $T_w = 1598 \text{ К}$

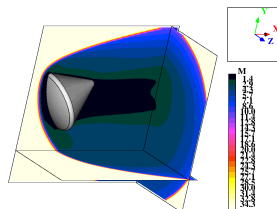
$H = 85 \text{ км}$, $\rho = 7.955\text{e-}06 \text{ кг/м}^3$, $T = 181 \text{ К}$, $Kn = 0.0024$



Плотность



Температура

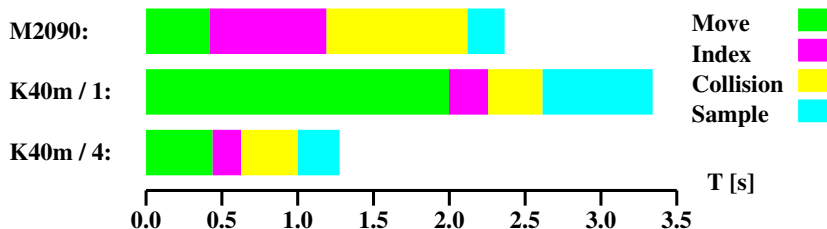


Число Маха

300 млн. частиц, 0.5 млн. ячеек (47 млн. столкновительных),
20 000 шагов (из них 10 000 со сбором статистики)

Время вычислений

| GPU | Шт. | Память [ГБ] | Время [с] | Время [ч:м:с] |
|-------|-----|----------------|--------------|------------------|
| M2090 | 12 | 45 | 40 000 | 11:06:40 |
| K40 | 4/1 | 40 | 148 680 | 41:18:00 |
| K40 | 4/4 | 46 | 58 700 | 16:18:20 |



Суммарное (по всем GPU) время выполнения одного шага

$$\frac{K40/1}{M2090} = 1.41; \quad \frac{K40/1}{K40/4} = 2.61; \quad \frac{M2090}{K40/4} = 1.85;$$

Выводы

- При использовании GPU с большой (≥ 3 Gb) памятью возможны провалы производительности из-за произвольного доступа к памяти.
- Желательно тестировать свои программы на возможность такого провала
- Желательно переделать алгоритмы для последовательного доступа к памяти.
- Разделение памяти на блоки с последовательной обработкой каждого блока позволяет существенно уменьшить проблемы, связанные с произвольным доступом к памяти.
- При выборе оборудования (GPU) требования к размеру памяти должны учитывать класс задач, которые будут решаться на этом GPU.
- Возможно, приобретение нескольких GPU с умеренным размером памяти, будет экономически и по производительности выгоднее одного GPU с большим объемом памяти.

Спасибо за Внимание!