

# Теория парралелизма

Отчёт

Уравнение теплопроводности на Cuda  
с несколькими видеокартами  
05.26.2023

Гольцев Никита Сергеевич, 21932

## 1. Цели работы

Реализовать решение уравнения теплопроводности, переписав программу на Cuda, при том добавить реализацию вычислений для нескольких видеокарт с использованием MPI.

## 2. Компиляция

Компилятор используемый при работе:  
mpic

Так же в процессе, для замера времени, использовалась библиотека chrono.

3. Используемый профилировщик  
nsys (NVIDIA Nsight Systems)

## 4. Этапы оптимизации

В данной работе в качестве дополнительного способараспараллеливания вычислений при помощи видеокарт используется интерфейс MPI. При помощи него мы можем задействовать сразу несколько устройств и при помощи функции MPI\_Sendrecv передавать данные между устройствами. Для дополнительного распараллеливания вычислений интерполяция границ и внутренних составляющих матрицы происходят в разных асинхронных потоках. Чтобы обезопасить код, все вычисления помещены в блок try..catch, при помощи которого отлавливаются ошибки и выводятся причины их возникновения.

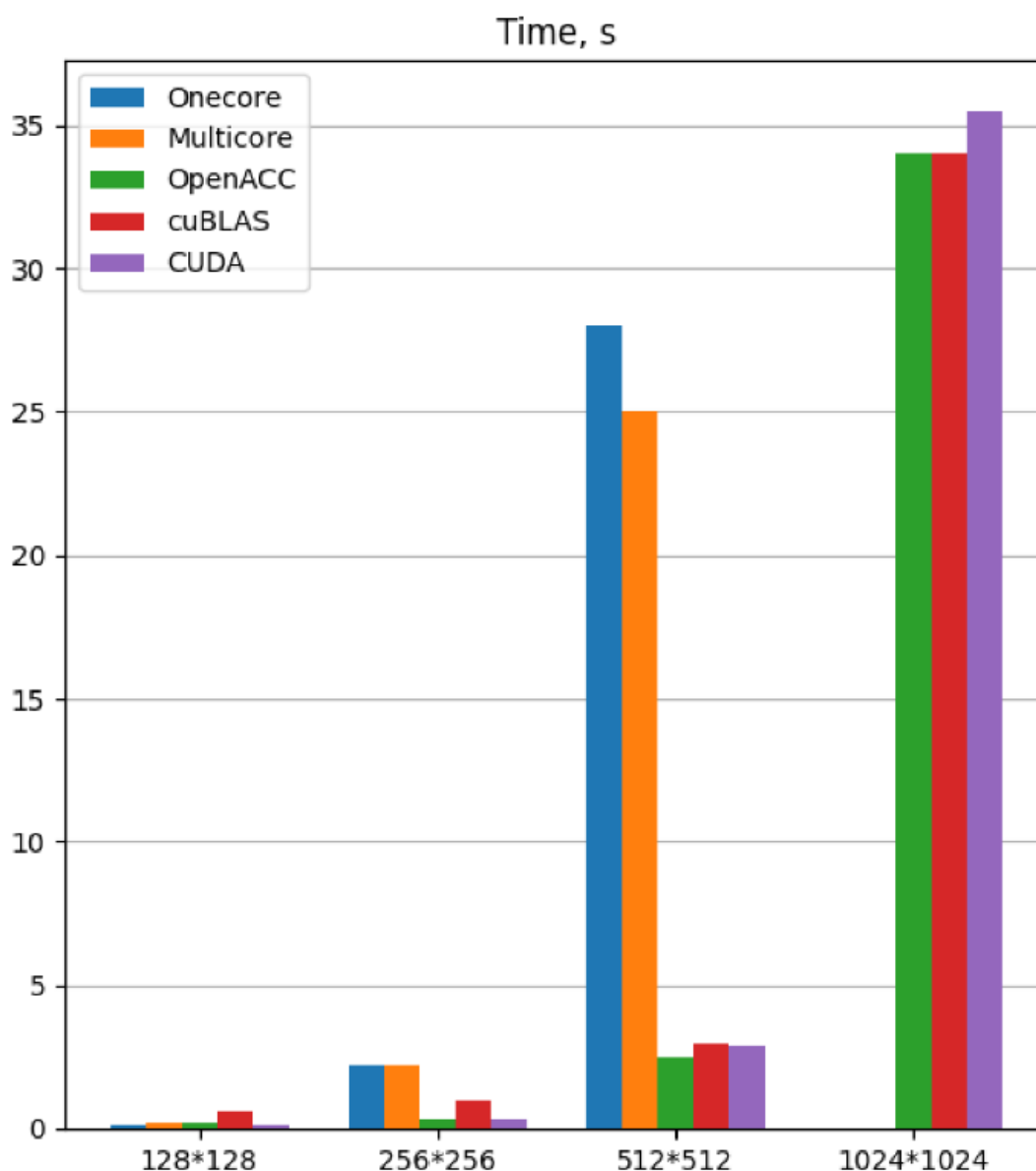
## 5. Точность вычислений

При выводе после создания и по завершению вычислений матрица совпадает с аналогичными проверками прошлого задания.

--- Borders ---												
10.00	10.83	11.67	12.50	13.33	14.17	15.00	15.83	16.67	17.50	18.33	19.17	20.00
10.83	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	20.83
11.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	21.67
12.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	22.50
13.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	23.33
14.17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	24.17
15.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	25.00
15.83	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	25.83
16.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	26.67
17.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	27.50
18.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	28.33
19.17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.17
20.00	20.83	21.67	22.50	23.33	24.17	25.00	25.83	26.67	27.50	28.33	29.17	30.00
--- Result ---												
10.00	10.83	11.67	12.50	13.33	14.17	15.00	15.83	16.67	17.50	18.33	19.17	20.00
10.83	11.67	12.50	13.33	14.17	15.00	15.83	16.67	17.50	18.33	19.17	20.00	20.83
11.67	12.50	13.33	14.17	15.00	15.83	16.67	17.50	18.33	19.17	20.00	20.83	21.67
12.50	13.33	14.17	15.00	15.83	16.67	17.50	18.33	19.17	20.00	20.83	21.67	22.50
13.33	14.17	15.00	15.83	16.67	17.50	18.33	19.17	20.00	20.83	21.67	22.50	23.33
14.17	15.00	15.83	16.67	17.50	18.33	19.17	20.00	20.83	21.67	22.50	23.33	24.17
15.00	15.83	16.67	17.50	18.33	19.17	20.00	20.83	21.67	22.50	23.33	24.17	25.00
15.83	16.67	17.50	18.33	19.17	20.00	20.83	21.67	22.50	23.33	24.17	25.00	25.83
16.67	17.50	18.33	19.17	20.00	20.83	21.67	22.50	23.33	24.17	25.00	25.83	26.67
17.50	18.33	19.17	20.00	20.83	21.67	22.50	23.33	24.17	25.00	25.83	26.67	27.50
18.33	19.17	20.00	20.83	21.67	22.50	23.33	24.17	25.00	25.83	26.67	27.50	28.33
19.17	20.00	20.83	21.67	22.50	23.33	24.17	25.00	25.83	26.67	27.50	28.33	29.17
20.00	20.83	21.67	22.50	23.33	24.17	25.00	25.83	26.67	27.50	28.33	29.17	30.00

## 6. Сравнение времени работы, как с предыдущими версиями, так и при вычислении на для разного количества видеокарт

Во всех случаях количество операций совпадало, завися от того, какой был размер сетки.



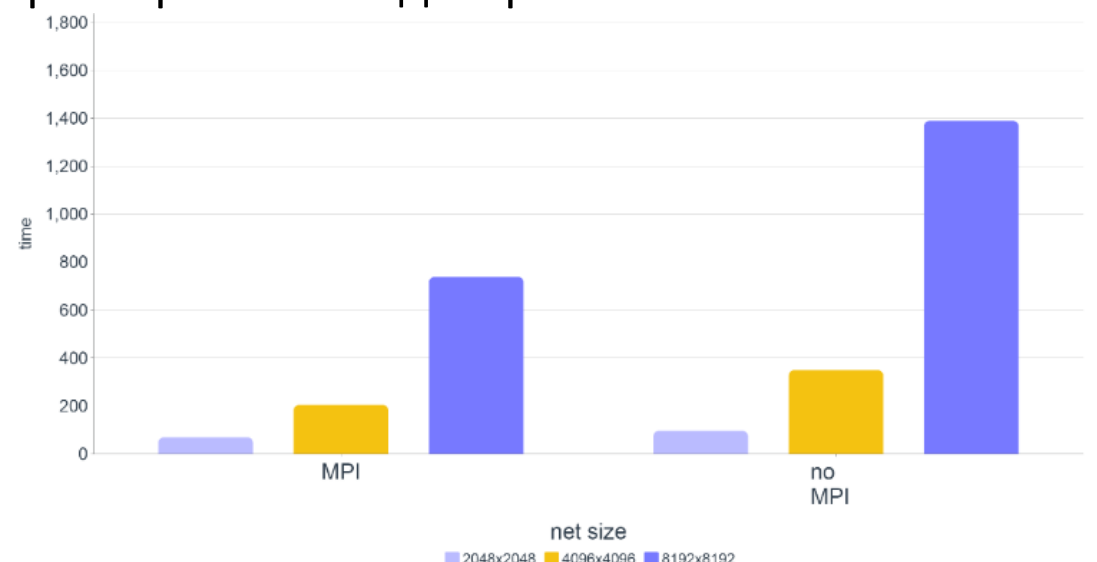
Выше представленно сравнение вычисления на одной видеокарте cuda и прошлыми версиями программ.

Размер стеки	Время, мс	Точность	Итерации
128*128	160 мс	9.9998-e7	31081
256*256	340 мс	9.9998-e7	37913
512*512	2298 мс	9.9998-e7	120657
1024*1024	27890 мс	1.373-e6	367890

Для начала проведём сравнения между одной и двумя видеокартами(MPI с 2мя видеокартами).

Явно видно преимущество по скорости вычисления для варианта с использованием mpi.

Теперь, для сравнения, добавим время работы с тремя видеокартами и наглядно покажем время работы на диаграмме.



Из диаграммы выше можно сделать вывод, что от количества используемых ресурсов явно зависит скорость выполнения задачи.

Ниже представлена полная таблица для 1, 2 и 4 видео карт.

Размер	1gpu	2gpu	4gpu	Время, с	Коэф. Эффективности Для 4 gpu	Точность	Кол-во итераций
128*128	0.16	0.9	1.69	-	0.03	9.9e-07	31081
256*256	0.75	3.2	5.3	-	0.035	9.9e-07	37913
512*512	7.1	12.9	18.5	-	0.096	9.9e-07	120657
1024*1024	64.7	217	54.8	-	0.29	1.4e-06	367890
2048*2048	246	134.0	96.7	-	0.65	1.4e-06	1000000
4096*4096	1056	538	277	-	0.95	9.8e-06	1000000

## 7. Вывод

Наглядно видно, что `mpi` очень полезен при сложных вычислениях(т.к. чем больше размер сетки, тем более явно видна его эффективность), а так же, что с увеличением количества видеокарт, на которых проводится вычисления, увеличивается и скорость работы программы. Но более расширенные тесты показали, что на маленьких размерах сетки он проигрывает. Можно сделать вывод, что вычисление на нескольких видеокартах оказывает

значительный, положительный эффект на скорость работы программы, если у данной задачи достаточно высокая сложность.