Теория параллелилзма

Отчёт

Оптимизированные библиотеки

Выполнил Грищенко Александр Михайлович, 21932

03.2023

1 Цели работы

Реализовать решение уравнение теплопроводности (пятиточечный шаблон) в двумерной области на равномерных сетках.
Перенести программу на GPU используя директивы OpenACC.
Операцию редукции на графическом процессоре реализовать через вызовы функций из библиотеки cuBLAS.
Произвести профилирование программы и оптимизацию кода.
Сравнить скорость работы для разных размеров сеток на центральном и графическом процессоре (реализация с библиотекой cuBLAS и реализация без неё).

2 Используемый компилятор

pgc++ с флагом -Mcudalib=cublas

3 Используемый профилировщик

nsys (NVIDIA Nsight Systems) с флагом –trace=cublas,openacc,nvtx.

4 Как проводился замер времени работы

Для замера времени работы использовалась библиотека chrono.
Замер времени производился несколько раз, затем бралось среднее время.

5 Выполнение на CPU

Данные из предыдущего задания.

5.1 CPU-onecore

Размер сетки	Время выполнения, с	Точность	Количество операций
128*128	0.1	9.5e-07	11136
256*256	1.8	9.8e-07	37376
512*512	25	9.8e-07	120832

5.2 CPU-multicore

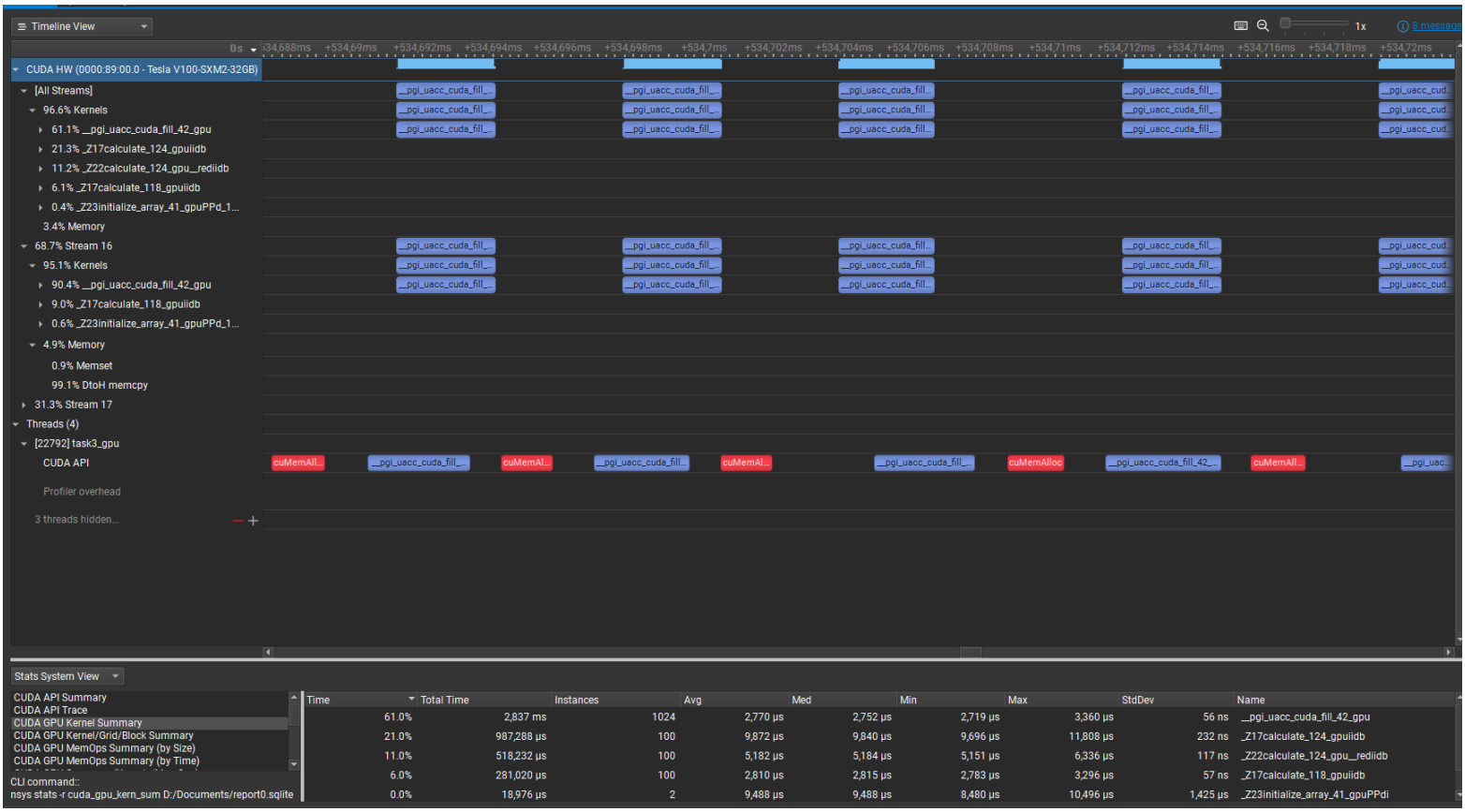
Размер сетки	Время выполнения, с	Точность	Количество операций
128*128	0.5	9.5e-07	11136
256*256	3.5	9.8e-07	37376
512*512	20	9.8e-07	120832
1024*1024	145	9.89e-07	365568

6 Выполнение на GPU

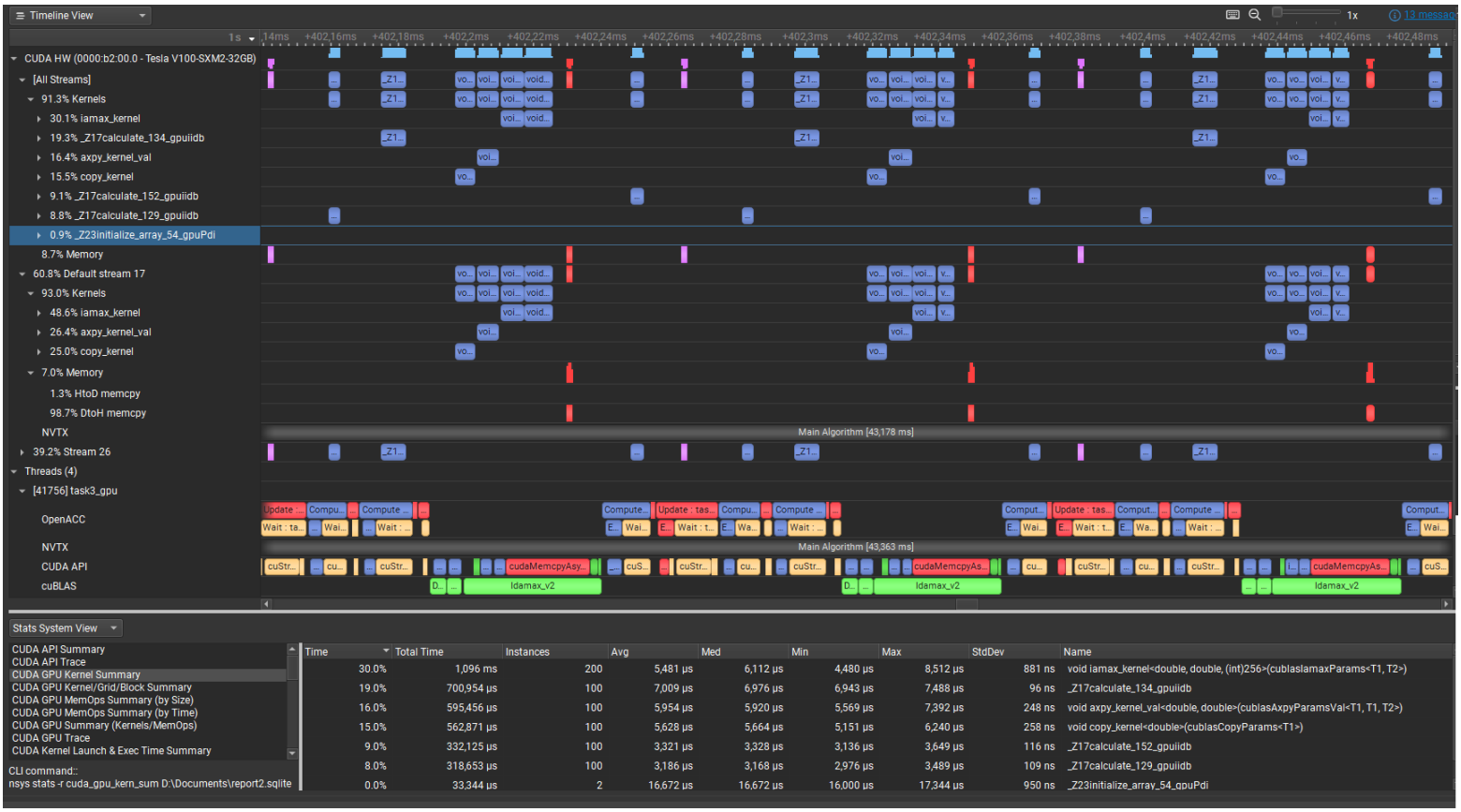
6.1 Этапы оптимизации на сетке 512*512 (количество итераций при профилировании 100)

Этап №	Время выполнения, с	Точность	Количество операций	Комментарии (что было сделано)
0	0.23	N/A*	100	Код из предыдущего задания.
0	0.22	0.035	100	Код из предыдущего задания (ошибка считается на каждой итерации, одномерные матрицы).
1	0.6	0.035	100	Вычисление максимальной ошибки через функции из cuBLAS
2	0.6	N/A*	100	Ошибка считается не каждую итерацию, асинхронность

* Из-за того, что период пересчёта ошибки намного меньше размера сетки, нет возможности определить точность на сотой итерации.

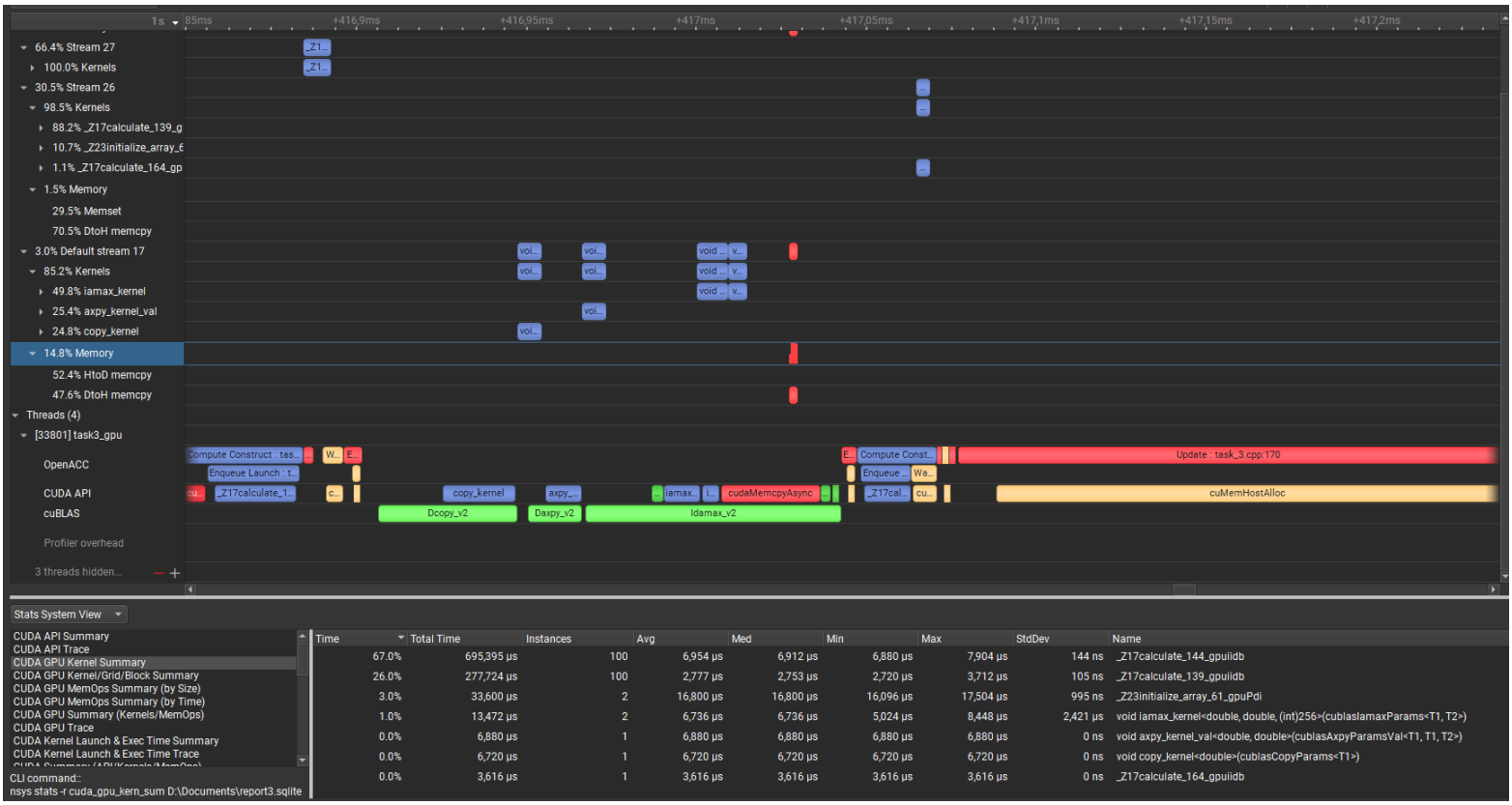


Этап 0 (без оптимизаций)



Time	Total Time	Instances	Avg	Med	Min	Max	StdDev	Name
30.0%	1,096 ms	200	5,481 µs	6,112 µs	4,480 µs	8,512 µs	881 ns	void iamax_kernel<double, double, (int)256>(cublasIamaxParams<T1, T2>)
19.0%	700,954 µs	100	7,009 µs	6,976 µs	6,943 µs	7,488 µs	96 ns	_Z17calculate_134_gpuiidb
16.0%	595,456 µs	100	5,954 µs	5,920 µs	5,569 µs	7,392 µs	248 ns	void axpy_kernel_val<double, double>(cublasAxpyParamsVal<T1, T1, T2>)
15.0%	562,871 µs	100	5,628 µs	5,664 µs	5,151 µs	6,240 µs	258 ns	void copy_kernel<double>(cublasCopyParams<T1>)
9.0%	332,125 µs	100	3,321 µs	3,328 µs	3,136 µs	3,649 µs	116 ns	_Z17calculate_152_gpuiidb
8.0%	318,653 µs	100	3,186 µs	3,168 µs	2,976 µs	3,489 µs	109 ns	_Z17calculate_129_gpuiidb
0.0%	33,344 µs	2	16,672 µs	16,672 µs	16,000 µs	17,344 µs	950 ns	_Z23initialize_array_54_gpuPdi

Этап 1



Stats System View

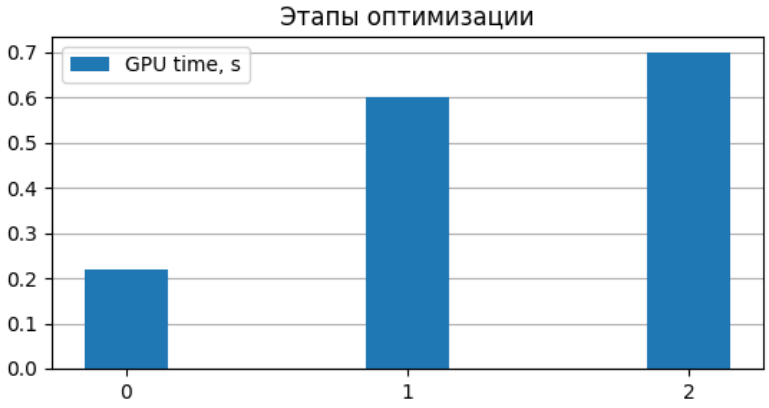
CUDA API Summary
CUDA API Trace
CUDA GPU Kernel Summary
CUDA GPU Kernel/Grid/Block Summary
CUDA GPU MemOps Summary (by Size)
CUDA GPU MemOps Summary (by Time)
CUDA GPU Summary (Kernels/MemOps)
CUDA GPU Trace
CUDA Kernel Launch & Exec Time Summary
CUDA Kernel Launch & Exec Time Trace

CLI command::
nsys stats -r cuda_gpu_kern_sum D:\Documents\report3.sqlite

Time	Total Time	Instances	Avg	Med	Min	Max	StdDev	Name
67.0%	695,395 µs	100	6,954 µs	6,912 µs	6,880 µs	7,904 µs	144 ns	_Z17calculate_144_gpuiidb
26.0%	277,724 µs	100	2,777 µs	2,753 µs	2,720 µs	3,712 µs	105 ns	_Z17calculate_139_gpuiidb
3.0%	33,600 µs	2	16,800 µs	16,800 µs	16,096 µs	17,504 µs	995 ns	_Z23initialize_array_61_gpuPdi
1.0%	13,472 µs	2	6,736 µs	6,736 µs	5,024 µs	8,448 µs	2,421 µs	void iamax_kernel<double, double, (int)256>(cublaslamaxParams<T1, T2>)
0.0%	6,880 µs	1	6,880 µs	6,880 µs	6,880 µs	6,880 µs	0 ns	void axpy_kernel_val<double, double>(cublasAxpyParamsVal<T1, T1, T2>)
0.0%	6,720 µs	1	6,720 µs	6,720 µs	6,720 µs	6,720 µs	0 ns	void copy_kernel<double>(cublasCopyParams<T1>)
0.0%	3,616 µs	1	3,616 µs	3,616 µs	3,616 µs	3,616 µs	0 ns	_Z17calculate_164_gpuiidb

Этап 2

6.2 Диаграмма оптимизации (по горизонтали номер этапа; по вертикали время работы)



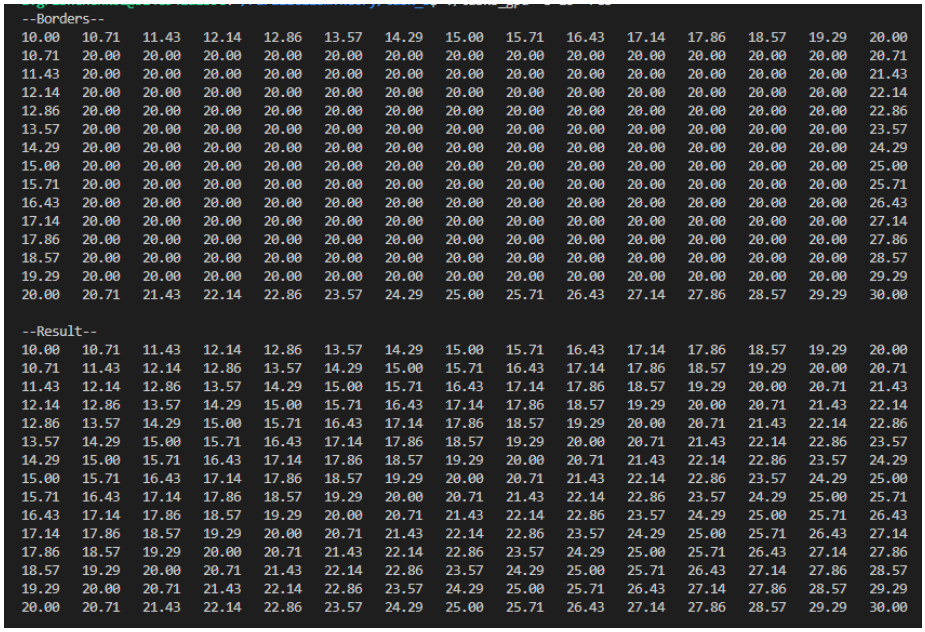
6.3 GPU – оптимизированный вариант (без cuBLAS)

Размер сетки	Время выполнения, с	Точность	Количество опреаций
128*128	0.3	9.5e-07	11136
256*256	0.5	9.8e-07	37376
512*512	1.5	9.8e-07	120832
1024*1024	16.7	9.9e-07	365568

6.4 GPU – оптимизированный вариант (cuBLAS)

Размер сетки	Время выполнения, с	Точность	Количество опреаций
128*128	0.7	9.5e-07	11136
256*256	0.8	9.8e-07	37376
512*512	1.3	9.8e-07	120832
1024*1024	10	9.9e-07	365568

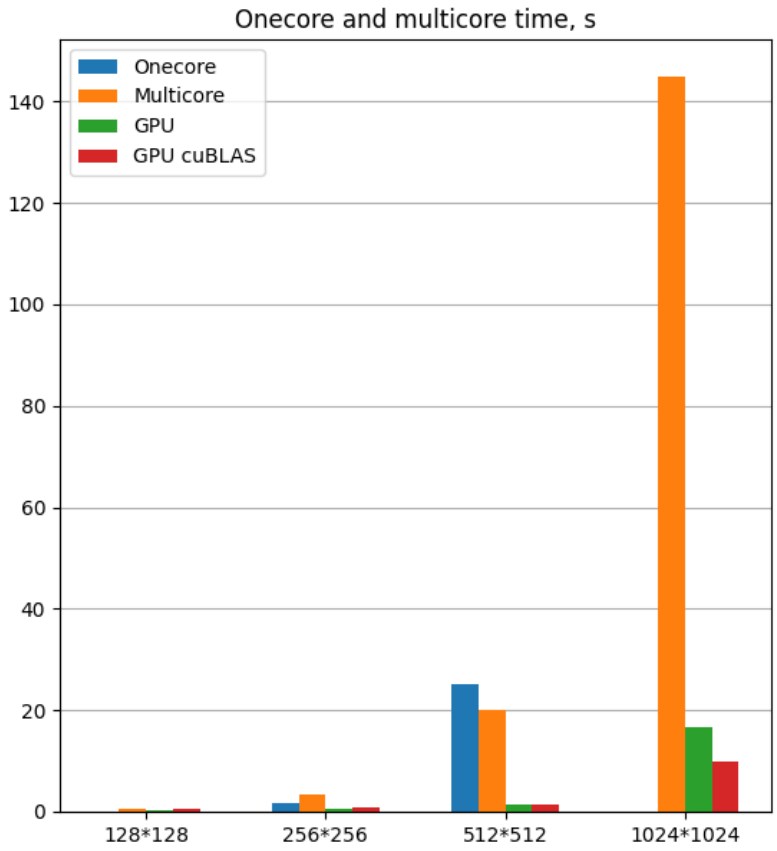
6.5 Скриншот массива 15*15 после заполнения границ и после работы всей программы



```
--Borders--
10.00   10.71   11.43   12.14   12.86   13.57   14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00
10.71   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.71
11.43   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   21.43
12.14   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   22.14
12.86   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   22.86
13.57   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   23.57
14.29   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   24.29
15.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   25.00
15.71   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   25.71
16.43   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   26.43
17.14   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   27.14
17.86   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   27.86
18.57   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   28.57
19.29   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   20.00   29.29
20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71   26.43   27.14   27.86   28.57   29.29   30.00

--Result--
10.00   10.71   11.43   12.14   12.86   13.57   14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00
10.71   11.43   12.14   12.86   13.57   14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71
11.43   12.14   12.86   13.57   14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43
12.14   12.86   13.57   14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14
12.86   13.57   14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86
13.57   14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57
14.29   15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29
15.00   15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00
15.71   16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71
16.43   17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71   26.43
17.14   17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71   26.43   27.14
17.86   18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71   26.43   27.14   27.86
18.57   19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71   26.43   27.14   27.86   28.57
19.29   20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71   26.43   27.14   27.86   28.57   29.29
20.00   20.71   21.43   22.14   22.86   23.57   24.29   25.00   25.71   26.43   27.14   27.86   28.57   29.29   30.00
```

7 Диаграмма сравнения времени работы CPU-one, CPU-multi, GPU, GPU cuBLAS для разных размеров сеток



8 Вывод

Используя библиотеку cuBLAS можно достичь прироста производительности на больших сетках. Для небольших сеток нет смысла использовать библиотеку, поскольку создание handle и работа с памятью занимают существенное время.

9 Приложение

9.1 Ссылка на GitHub

https://github.com/busyhedg03/ParallelismTheory/tree/master/task_3