

Теория Параллелизма

Отчет

Лабораторная №6 (OpenACC)

Выполнил,
Бородин Кирилл Андреевич 22931

30.07.2024

Цель работы

Используемый компилятор: pgcc

Используемый профилировщик: NVIDIA Nsight System

Как производили замер времени работы: chrono

Выполнение на CPU

CPU-onecore

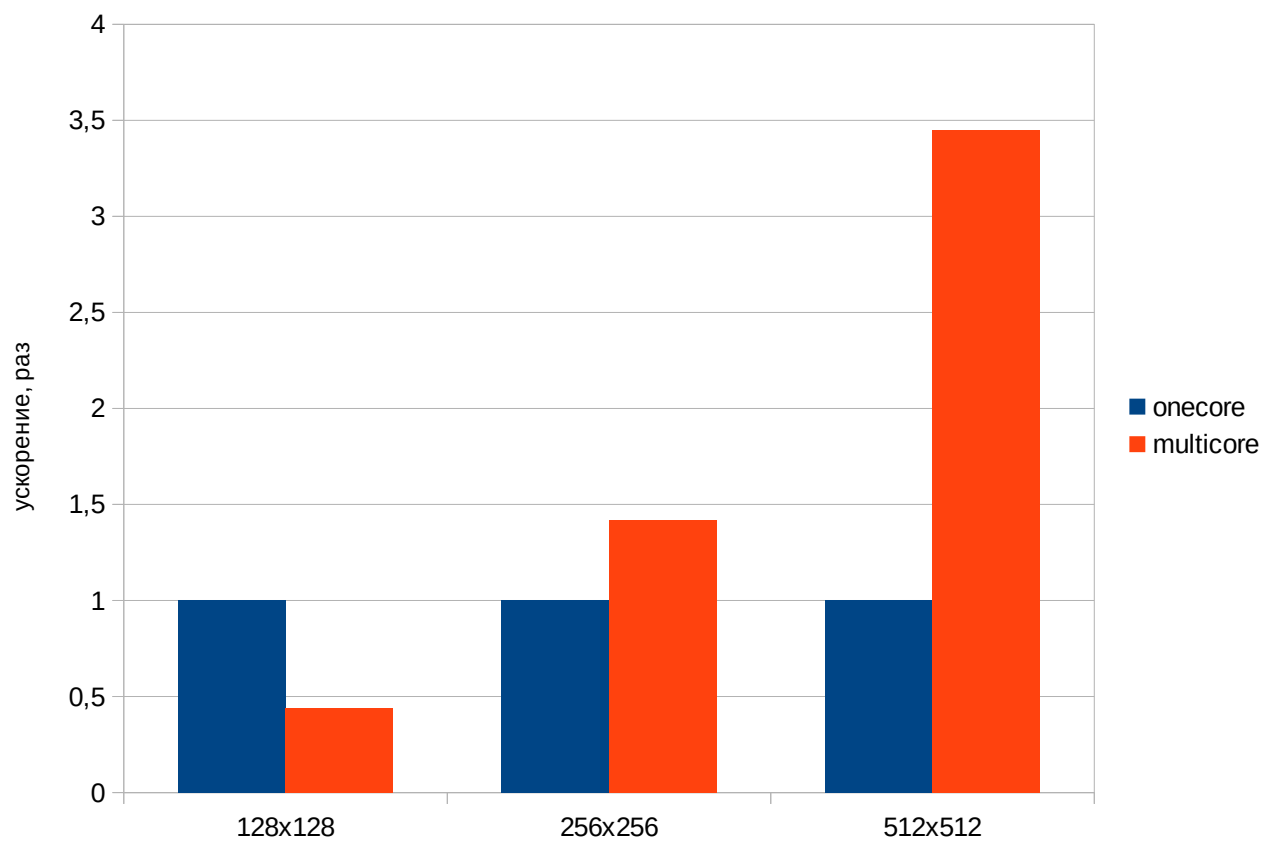
Размер сетки	Время выполнения	Точность	Количество итераций
128*128	0.43	1e-6	30100
256*256	5.68	1e-6	102900
512*512	93.89	1e-6	339600

CPU-multicore

Размер сетки	Время выполнения	Точность	Количество итераций
128*128	0.96	1e-6	30100
256*256	3.98	1e-6	102900
512*512	27.16	1e-6	339600
1024*1024	138.64	1e-6	1000000

Диаграмма сравнения время работы CPU-one и CPU-multi

Onecore vs multicore



Выполнение на GPU

Этапы оптимизации на сетке 512*512

Этап №	Время выполнения	Точность	Максимально е количество итераций	Комментари и (что было сделано)
1	23.23	1e-6	1_000_000	Not opt
2	15.21	1e-6	1_000_000	swap
3	138.62	1e-6	1_000_000	Error1per100
и т.д.			1_000_000	

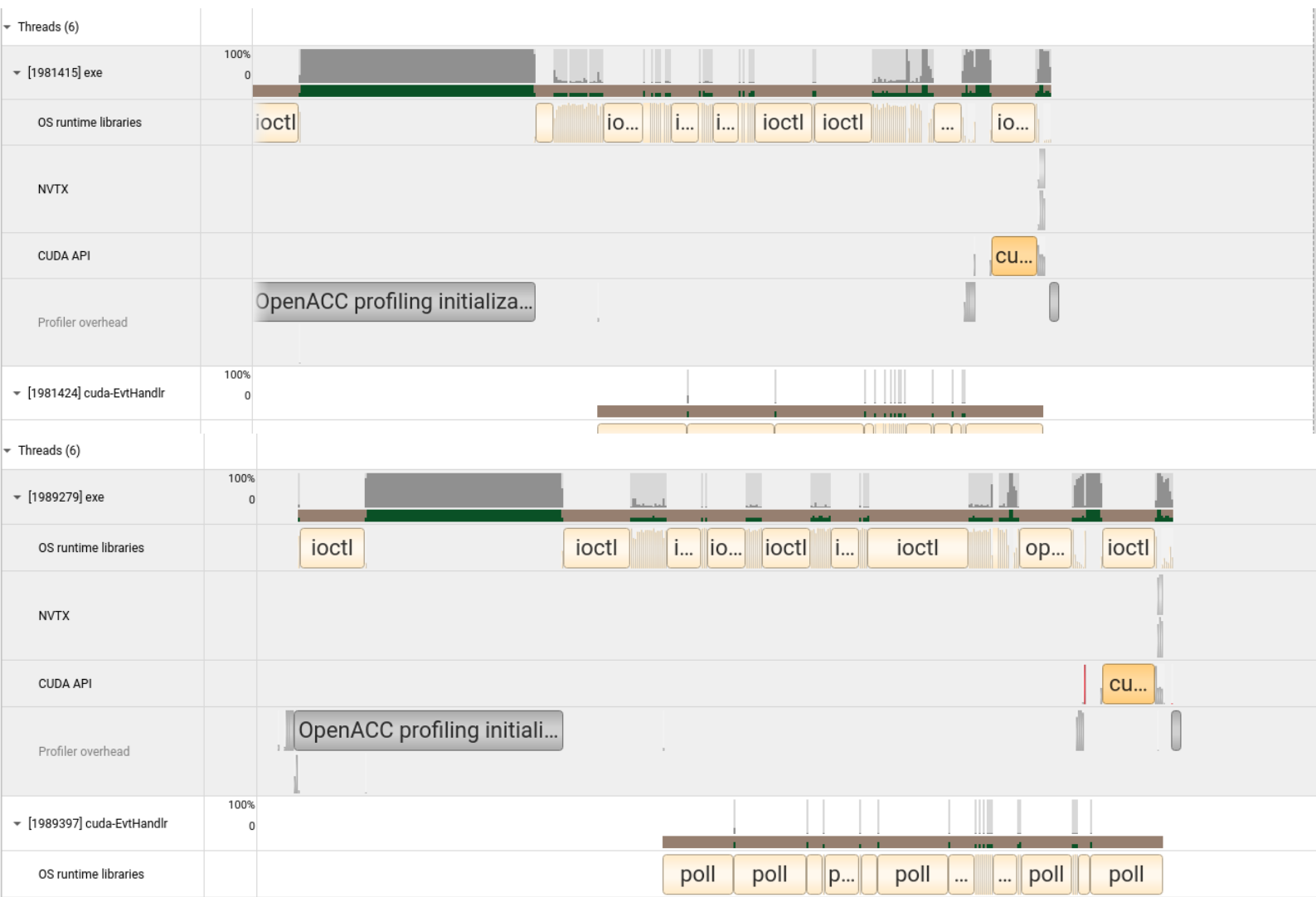


Диаграмма оптимизации

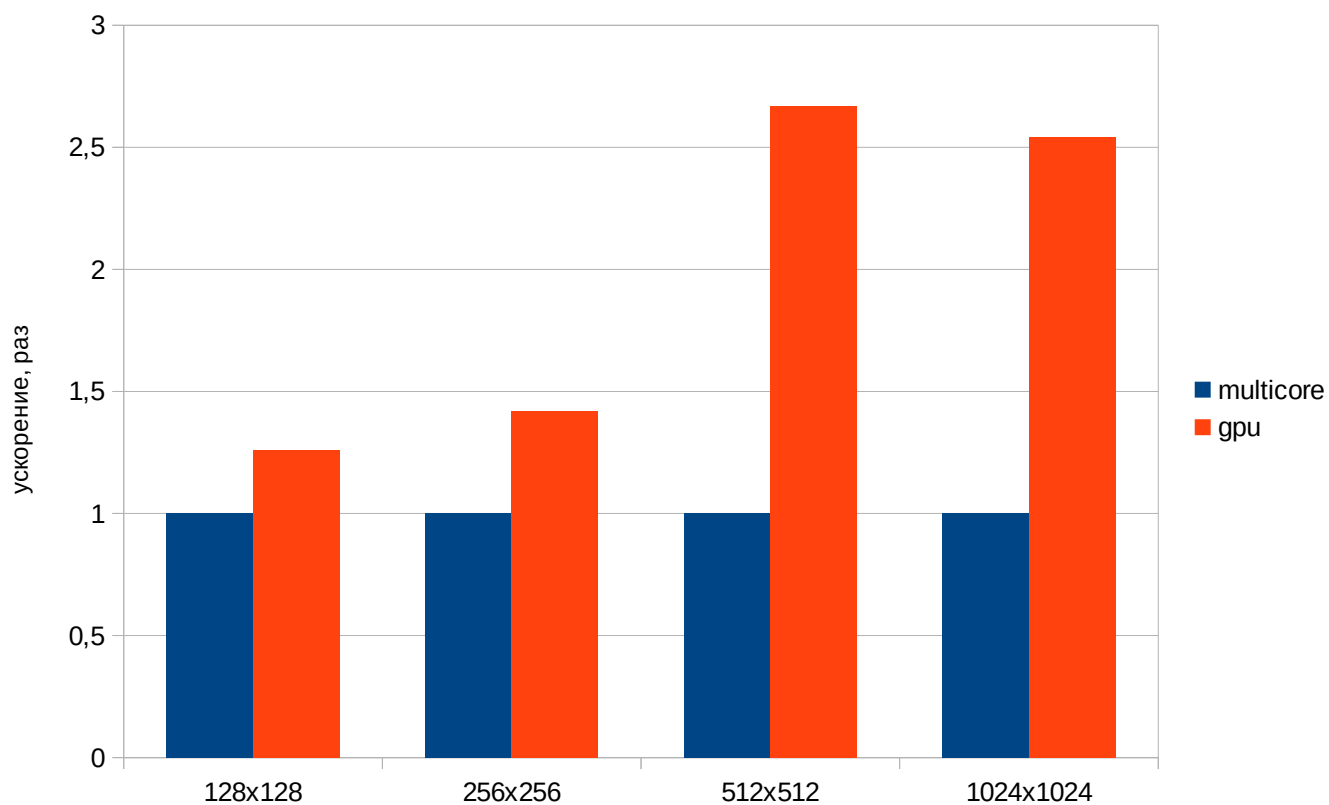
(по горизонтали номер этапа; по вертикали время работы)

GPU – оптимизированный вариант

Размер сетки	Время выполнения	Точность	Количество итераций
128*128	0.76	1e-6	30100
256*256	2.80	1e-6	102900
512*512	10.17	1e-6	339600
1024*1024	54.53	1e-6	1000000

Диаграмма сравнения времени работы CPU-one,
CPU-multi, GPU(оптимизированный вариант) для
разных размеров сеток

GPU vs Multicore



Вывод: все что можно распараллелить на CPU надо переводить на GPU

GPU with CUBLAS

Размер сетки	Время выполнения	Точность	Количество итераций
128*128	0.84	1e-6	30100
256*256	3.06	1e-6	102900
512*512	10.70	1e-6	339600
1024*1024	64.36	1e-6	1000000

Вывод: из-за дополнительной необходимости копирования
скорость падает