Теория параллелизма

Отчёт

Уравнение теплопроводности на CUDA

Выполнил: Лейсле Александр Геннадьевич, группа 21932

05.04.2023

Цели работы:

1. Реализовать решение уравнение теплопроводности (пятиточечный шаблон) в двумерной области на равномерных сетках:
   - 128x128
   - 256x256
   - 512x512
   - 1024x1024

2. Перенести программу на GPU, используя CUDA

3. Операцию редукции (вычисление максимального значения ошибки) на графическом процессоре реализовать через вызовы функций из библиотеки CUB

4. Сравнить скорость работы для разных размеров сеток на GPU текущей реализации и прошлой реализации через OpenACC

5. Произвести профилирование программы с использованием NsightSystems.

Используемый компилятор: nvcc

Используемый профилировщик: nsys

Как производили замер времени работы:

В начале и в конце программы производилась фиксация текущего времени с использованием команд из chrono, разница этого времени выводилась в стандартный поток вывода.

Реализация через OpenAcc
без библиотеки cuBLAS

CPU-multicore

Размер сетки	Время выполнения	Точность	Количество итераций
128x128	0.10c	$9.9 * 10^{-7}$	30080
256x256	0.52c	$9.9 * 10^{-7}$	102912
512x512	5.12c	$9.9 * 10^{-7}$	339712
1024x1024	79.77c	$1.37 * 10^{-6}$	1000000

GPU

Размер сетки	Время выполнения	Точность	Количество итераций
128x128	0.68c	$9.9 * 10^{-7}$	30080
256x256	1.72c	$9.9 * 10^{-7}$	102912
512x512	6.08c	$9.9 * 10^{-7}$	339712
1024x1024	39.35c	$1.37 * 10^{-6}$	1000000

Реализация через OpenAcc
с библиотекой cuBLAS

GPU

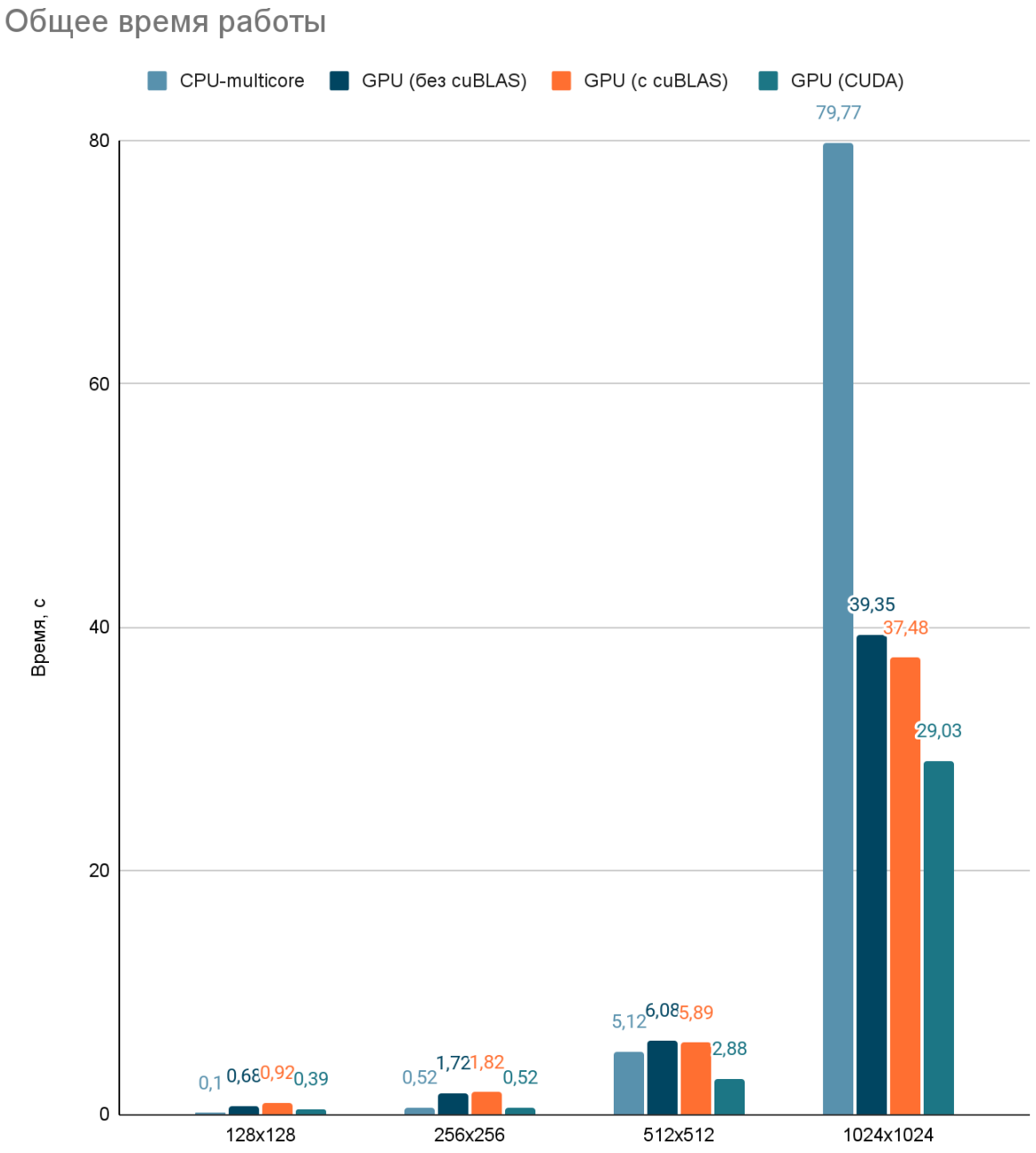
Размер сетки	Время выполнения	Точность	Количество итераций
128x128	0.92c	$9.9 * 10^{-7}$	30080
256x256	1.82c	$9.9 * 10^{-7}$	102912
512x512	5.89c	$9.9 * 10^{-7}$	339712
1024x1024	37.48c	$1.37 * 10^{-6}$	1000000

Реализация через CUDA

GPU

Размер сетки	Время выполнения	Точность	Количество итераций
128x128	0.39c	$9.9 * 10^{-7}$	30080
256x256	0.52c	$9.9 * 10^{-7}$	102912
512x512	2.88c	$9.9 * 10^{-7}$	339712
1024x1024	29.03c	$1.37 * 10^{-6}$	1000448

Диаграмма сравнения времени работы CPU-multicore и GPU



Выполнение на GPU

Этапы оптимизации на сетке 512x512

№	Время выполнения	Точность	Кол-во итераций	Комментарии (что было сделано)
1	0.21c	0.04	256	Реализация только с использованием директив OpenACC; Сетка представляется в виде одномерного массива; Выделение памяти под сетки на текущем и новом шагах происходит на CPU, затем обе сетки отправляются на GPU; Под обе сетки выделяется память, и происходит копирование массивов (которые заполнены нулями) с помощью acc data copyout. По выходу из секции данные массивов копируются обратно на CPU; В CPU происходит выделение памяти под массив из 4 элементов, представляющих собой размеры шагов для линейной интерполяции между углами сетки по границам. Под данный массив происходит выделение памяти на GPU с помощью acc data create; В память GPU копируются значения: общего количества элементов сетки, размера стороны сетки и максимальная ошибка, которая

				высчитывается между итерациями (в начальный момент превышает заданную ошибку на 1); Обращение к индексам массива происходит с помощью макроса ID, который позволяет перевести индексы двумерного массива в индекс одномерного; Заполнение границ вспомогательной и основной сеток происходит на GPU асинхронно при помощи директивы acc async. При этом циклы, в которых происходит заполнение, распараллеливаются с помощью acc parallel loop independent. В конце заполнения происходит синхронизация для обеих сеток; Новое состояние каждой точки сетки рассчитывается через среднее 4х соседних точек. Это происходит в 2х вложенных циклах, которые распараллеливаются с помощью директивы acc parallel loop collapse(2) independent present(grid, grid_new). Для вычисления максимальной ошибки между двумя итерациями используется редукция по функции максимума для переменной max_error с помощью директивы acc reduction(max:max_error) Обновление основной сетки происходит с помощью смены указателей сеток на текущем и новом шагах на CPU, а затем с помощью

				функции acc_attach происходит пересвязывание адресов этих сеток в памяти GPU; Происходит обновление значения максимальной ошибки в памяти CPU, если прошедшее количество итераций кратно половине размера стороны сетки или пройдено заданное количество итераций, с помощью acc update host
2	0.56c	0.04	256	Реализация с использованием директив OpenACC и функций cuBLAS; На CPU создается контекст cuBLAS и инициализируется с помощью функции cublasCreate; Режим обращения к указателям для функций cuBLAS устанавливается в режим обращения на GPU; Если прошедшее количество итераций кратно половине размера стороны сетки или пройдено заданное количество итераций, то вычисляется максимальная ошибка с помощью функций cuBLAS: Текущее состояние сетки сохраняется во временный массив путем копирования сетки с помощью функции cublasDcopy; Из нового состояния сетки вычитается скопированный массив с помощью cublasDaxpy, результат записывается на место скопированного массива;

				В конце с помощью функции cublasIdamax в переменную id записывается адрес самого большого по модулю значения, которое представляет собой значение наибольшей ошибки между двумя итерациями. В переменную max_error записывается значение наибольшей ошибки После обновления текущего состояния сетки происходит обновление максимальной ошибки в памяти CPU.
3	0.32c	0.04	256	Реализация с использованием функций CUDA; В памяти CPU инициализируются сетка на текущем и новом шагах, а также значение максимальной ошибки; Помимо этого объявляются указатели для соответствующих переменных в памяти GPU, а также для временного массива, который будет хранить разницу между сетками на новом и текущим шагами; Границы сеток на текущем и новом шагах заполняются на CPU; Для сеток, максимальной ошибки и временного массива в памяти GPU выделяется память с помощью cudaMalloc;

				Данные сеток и максимальной ошибки переносятся в память GPU с помощью функции cudaMemcpy; Находится необходимое количество байт для нахождения максимума в массиве разниц двух сеток с помощью cub::DeviceReduce::Max, при этом массив для временных вычислений равен NULL; Выделяется память в GPU для массива временных вычислений с помощью cudaMalloc; Происходит объявление графа, а также происходит инициализация размера сетки и размера блока сетки; Количество потоков в блоке устанавливается равным 8x8; Происходит инициализация графа: начинается режим записи в граф с помощью cudaStreamBeginCapture, затем создается цикл, в котором помещается дважды вычисление нового шага сетки. Запись прекращается с помощью cudaStreamEndCapture, затем создается экземпляр графа с помощью cudaGraphInstantiate; В цикле запускается проход по графу, после чего находится абсолютная разница между текущим и предыдущими шагами, находится максимальная ошибка с помощью cub::DeviceReduce::Max и в конце

				происходит обновление максимальной ошибки в памяти CPU; По выходу из цикла данные сетки на текущем шаге копируются в память CPU.

Вывод

В ходе работы было реализовано решение уравнения теплопроводности на сетках размера 128x128, 256x256, 512x512, 1024x1024. Реализация была перенесена на GPU с помощью функций CUDA. Нахождение максимальной ошибки было реализовано с помощью функций библиотеки cub. Было произведено сравнение скорости выполнения реализаций через OpenAcc, cuBLAS и CUDA на разных размерах сеток. Для полученных реализаций была выполнена профилировка с помощью NsightSystem.

Использование функций CUDA, а в частности использование графов, позволяет значительно ускорить выполнение программы, за счет того, что kernels записываются в граф и в дальнейшем не происходит расходов на их вызов, следовательно, между выполнениями отдельных kernels находятся очень маленькие промежутки. Помимо этого, использование CUDA позволяет использовать вычислительные способности GPU по максимуму за счет применения блоков с потоками, что позволяет получить выигрыш во времени как на маленьких, так и на больших сетках.

Приложение

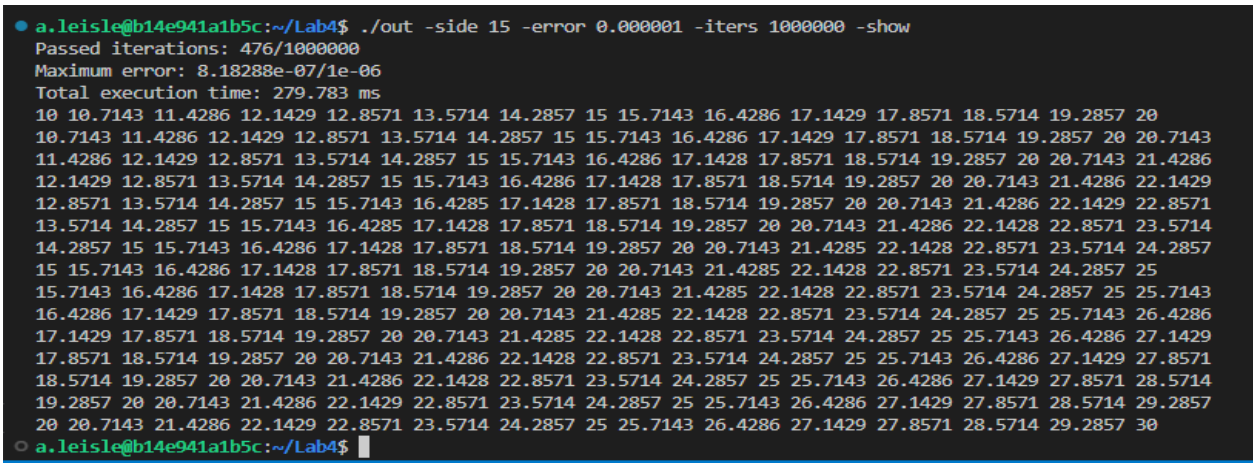
https://github.com/HerrPhoton/Heat_equation_on_CUDA

```
sanya@sanya-desktop:~/Lab3$ ./out -side 15 -iters 1000000 -error 0.000001 -show
Passed iterations: 539/1000000
Maximum error: 9.52953e-07/1e-06
Total execution time: 80.759 ms
10 10.7143 11.4286 12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4286 17.1429 17.8571 18.5714 19.2857 20
10.7143 11.4286 12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143
11.4286 12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286
12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1429
12.8571 13.5714 14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1429 22.8571
13.5714 14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1428 22.8571 23.5714
14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857
15 15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25
15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25 25.7143
16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286
17.1429 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429
17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571
18.5714 19.2857 20 20.7143 21.4286 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571 28.5714
19.2857 20 20.7143 21.4286 22.1429 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571 28.5714 29.2857
20 20.7143 21.4286 22.1429 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571 28.5714 29.2857 30
sanya@sanya-desktop:~/Lab3$
```

Вывод сетки 15x15 на реализации через OpenAcc

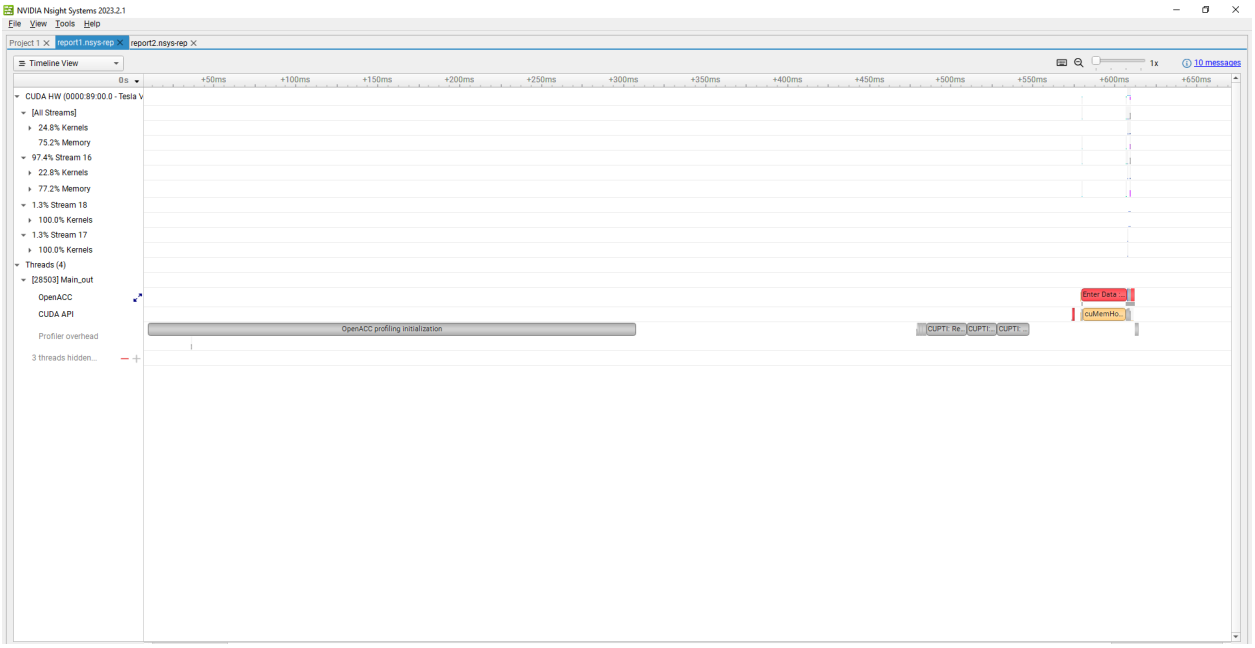
```
sanya@sanya-desktop:~/Lab3$ ./out -side 15 -iters 1000000 -error 0.000001 -show
Passed iterations: 539/1000000
Maximum error: 9.52953e-07/1e-06
Total execution time: 754.439 ms
10 10.7143 11.4286 12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4286 17.1429 17.8571 18.5714 19.2857 20
10.7143 11.4286 12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143
11.4286 12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286
12.1429 12.8571 13.5714 14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1429
12.8571 13.5714 14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1429 22.8571
13.5714 14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1428 22.8571 23.5714
14.2857 15 15.7143 16.4285 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857
15 15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25
15.7143 16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25 25.7143
16.4286 17.1428 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286
17.1429 17.8571 18.5714 19.2857 20 20.7143 21.4285 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429
17.8571 18.5714 19.2857 20 20.7143 21.4286 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571
18.5714 19.2857 20 20.7143 21.4286 22.1428 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571 28.5714
19.2857 20 20.7143 21.4286 22.1429 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571 28.5714 29.2857
20 20.7143 21.4286 22.1429 22.8571 23.5714 24.2857 25 25.7143 26.4286 27.1429 27.8571 28.5714 29.2857 30
sanya@sanya-desktop:~/Lab3$
```

Вывод сетки 15x15 на реализации через cuBLAS

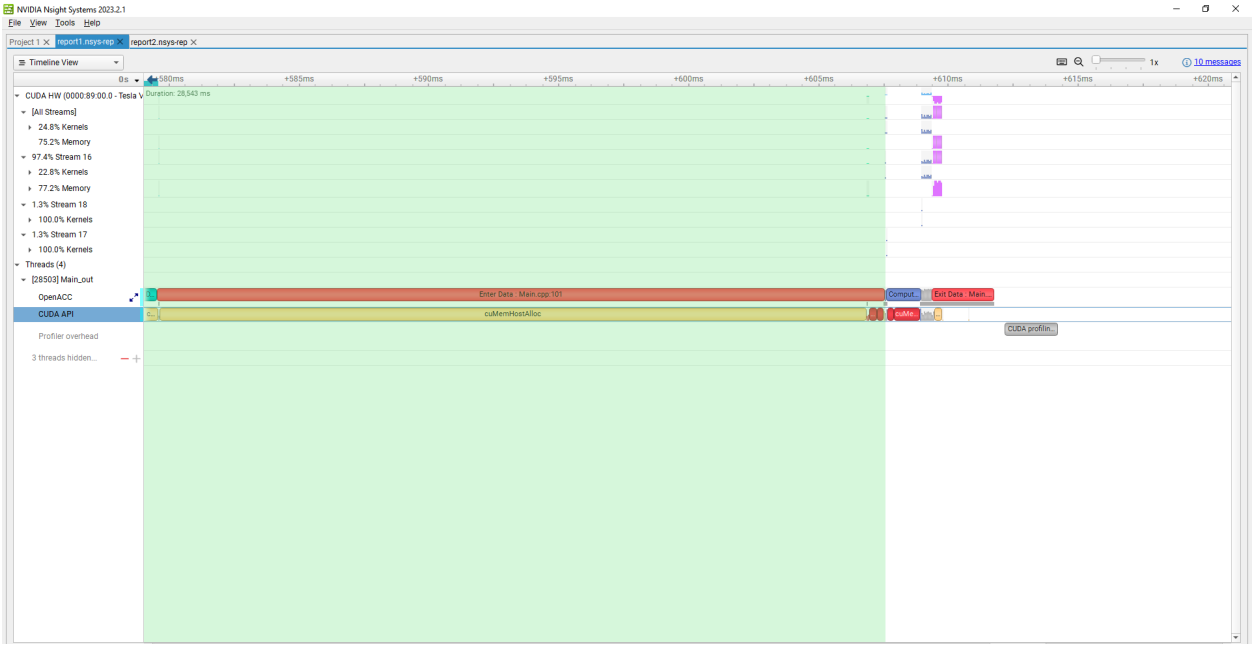


Вывод сетки 15x15 на реализации через CUDA

Профилирование реализации через OpenAcc

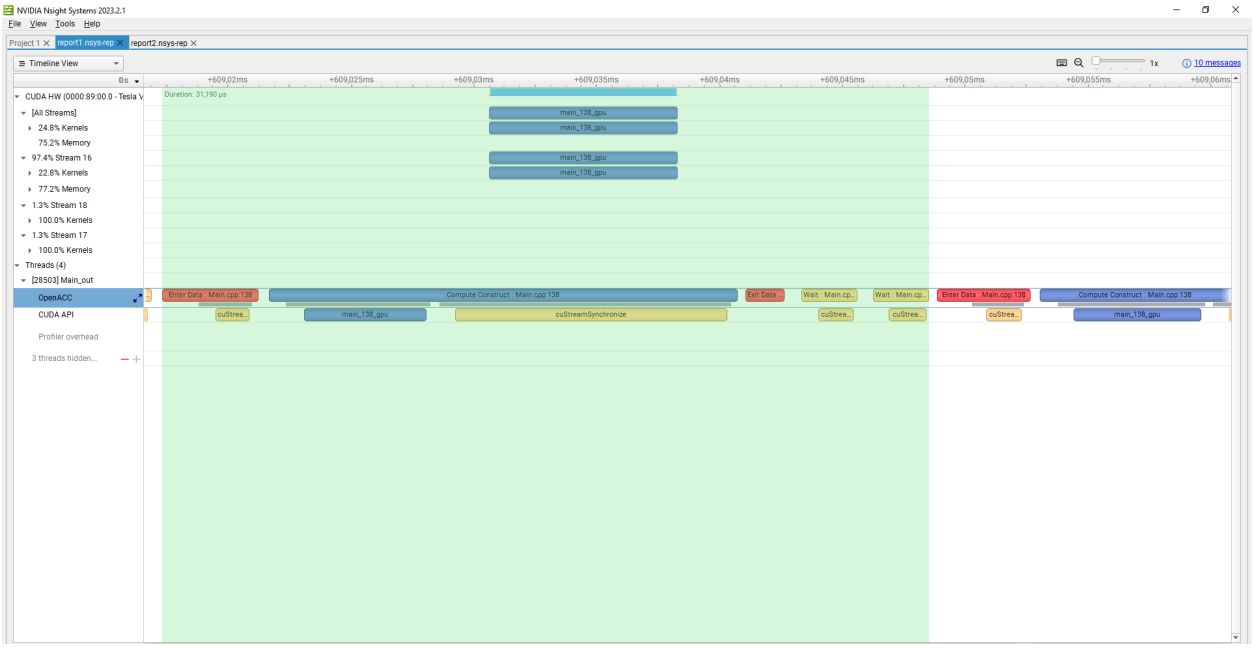


Общий вид профилирования

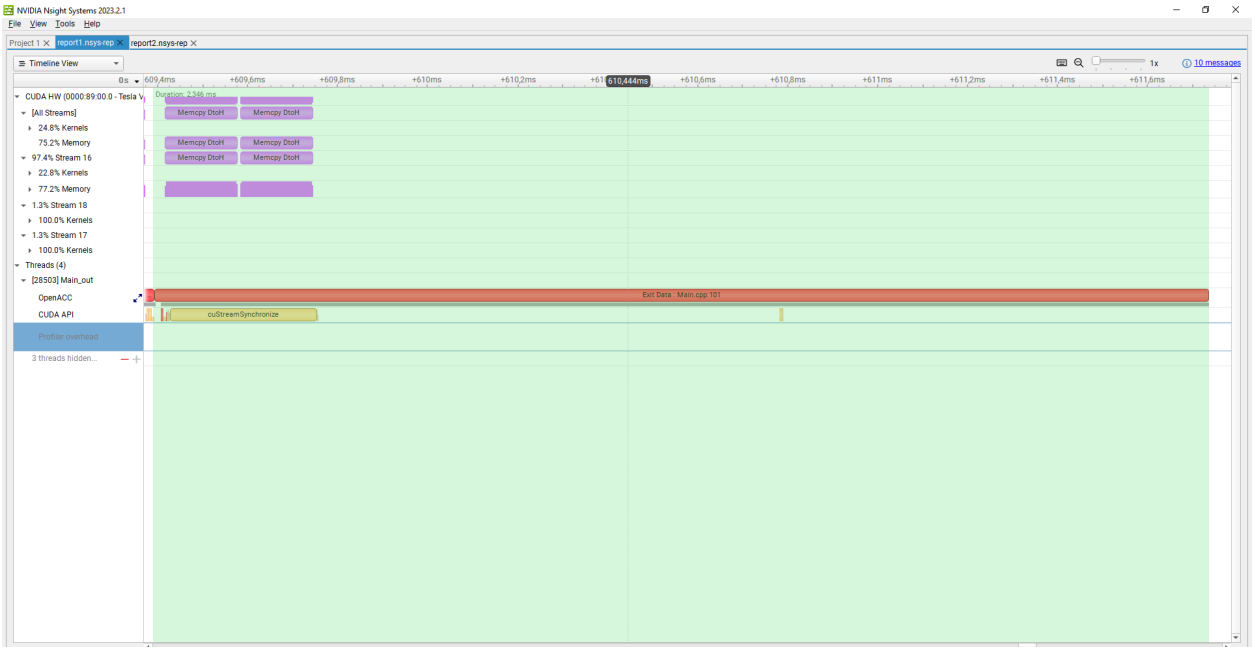


Перенос данных на GPU + выделение памяти

Общий вид процесса расчета сетки 512x512 с 10ю итерациями



Одна итерация расчета сетки



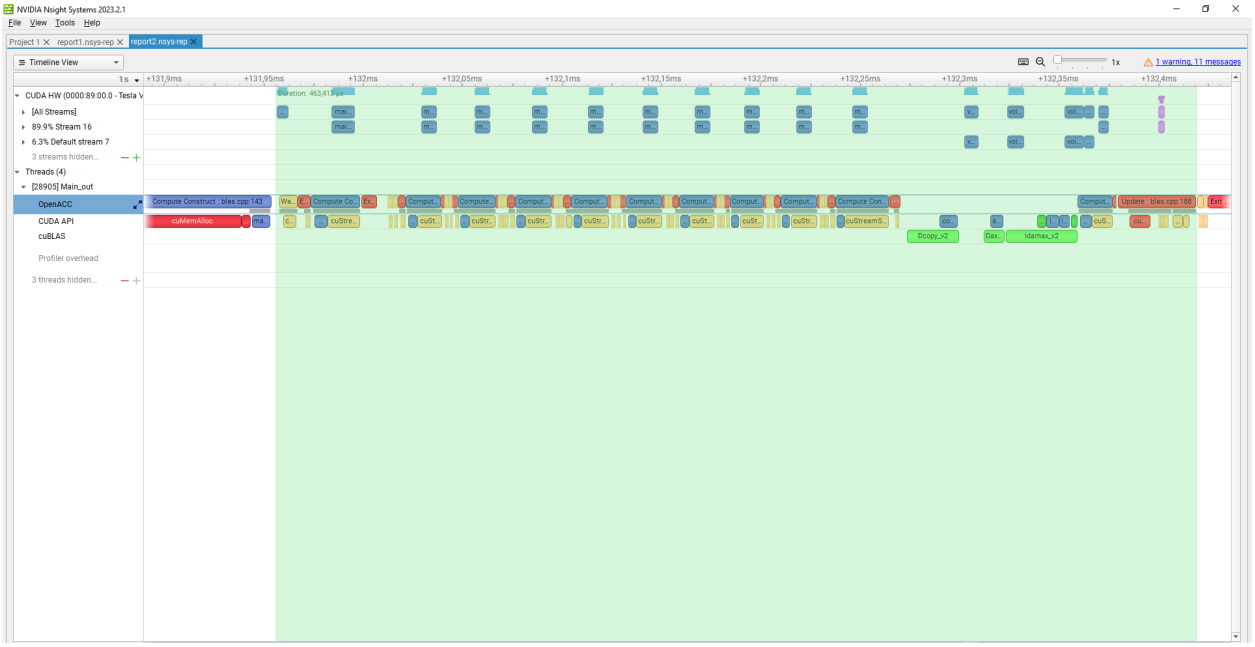
Копирование данных с памяти GPU

Профилирование реализации через cuBLAS

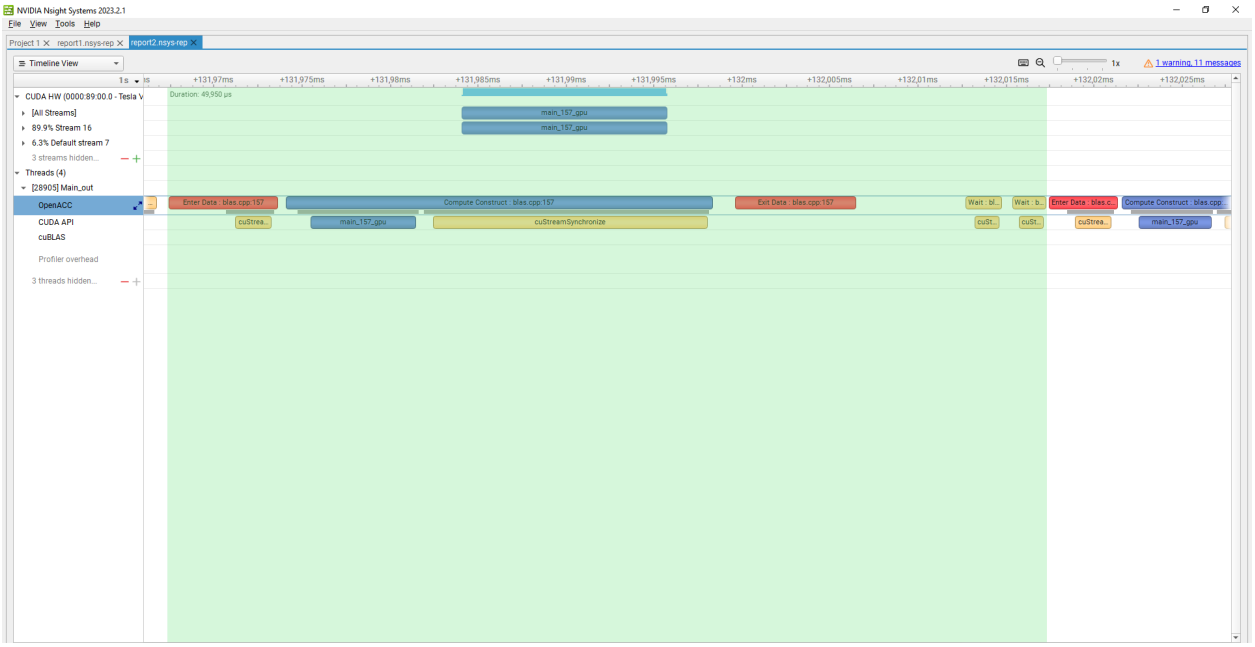
Общий вид профилирования

Инициализация контекста cuBLAS

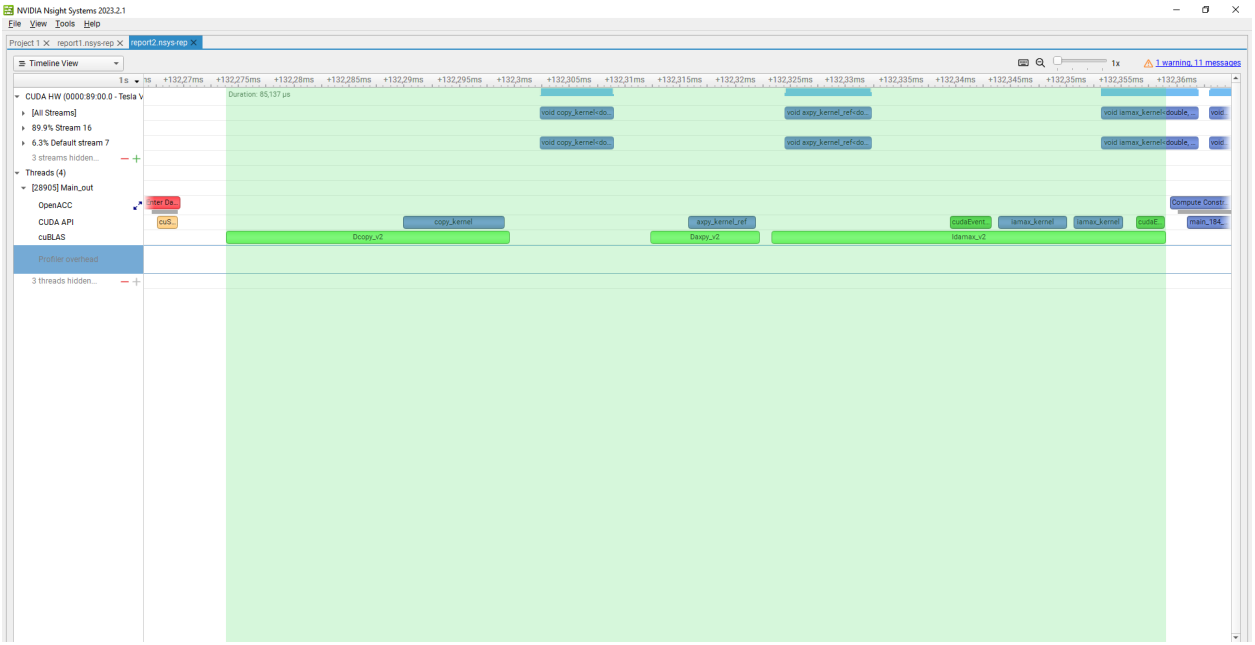
Перенос данных на GPU + выделение памяти



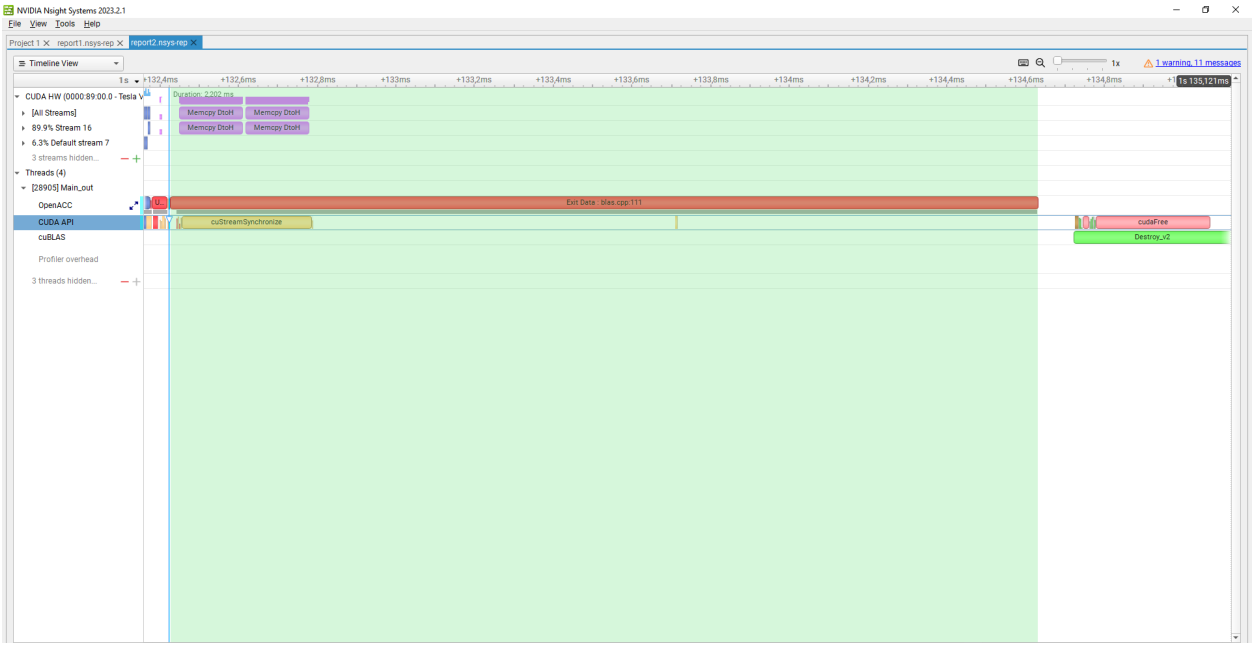
Общий вид процесса расчета сетки 512x512 с 10ю итерациями



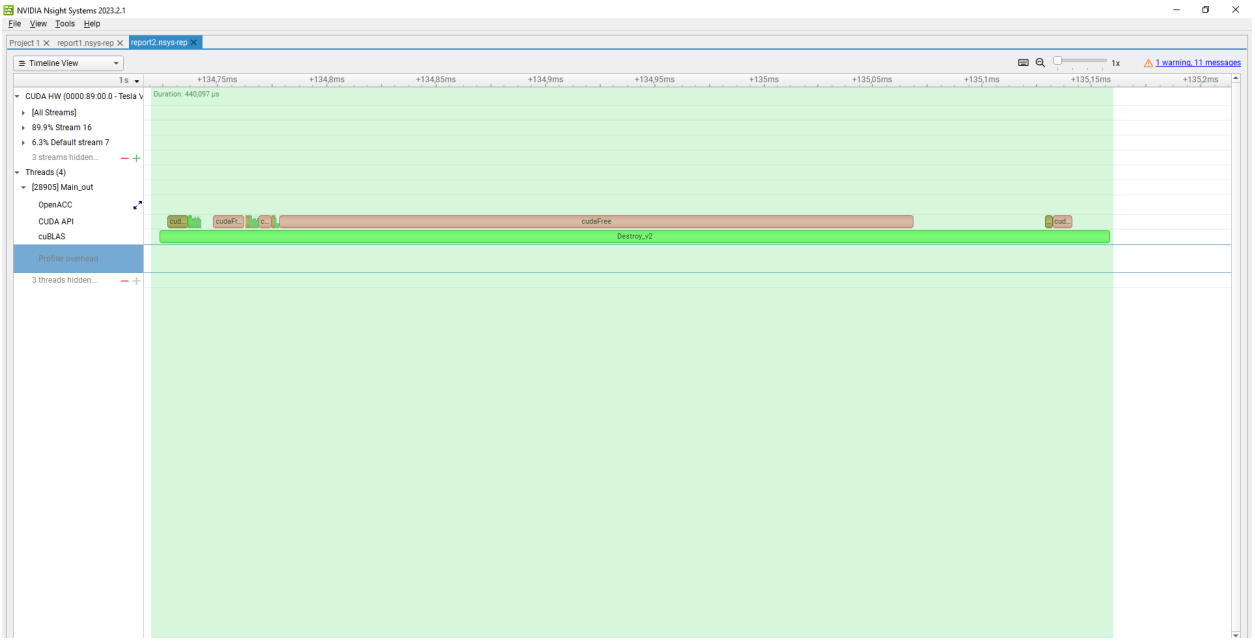
Одна итерация расчета сетки



Вычисление индекса максимальной ошибки

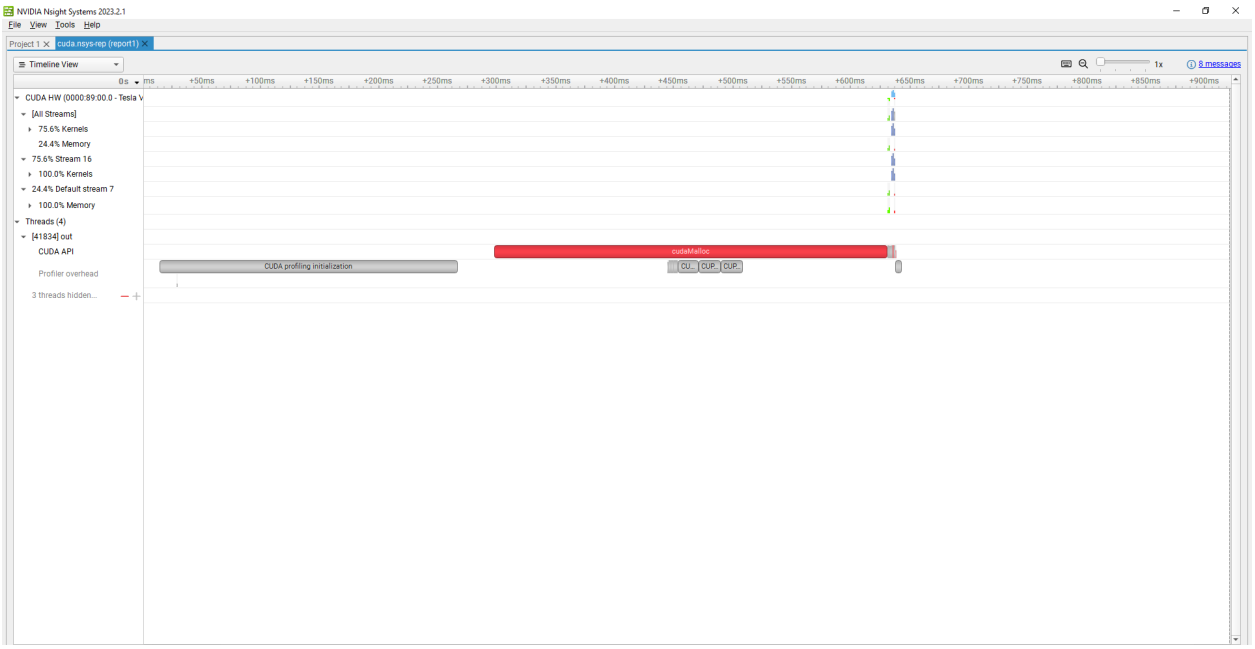


Копирование данных с памяти GPU

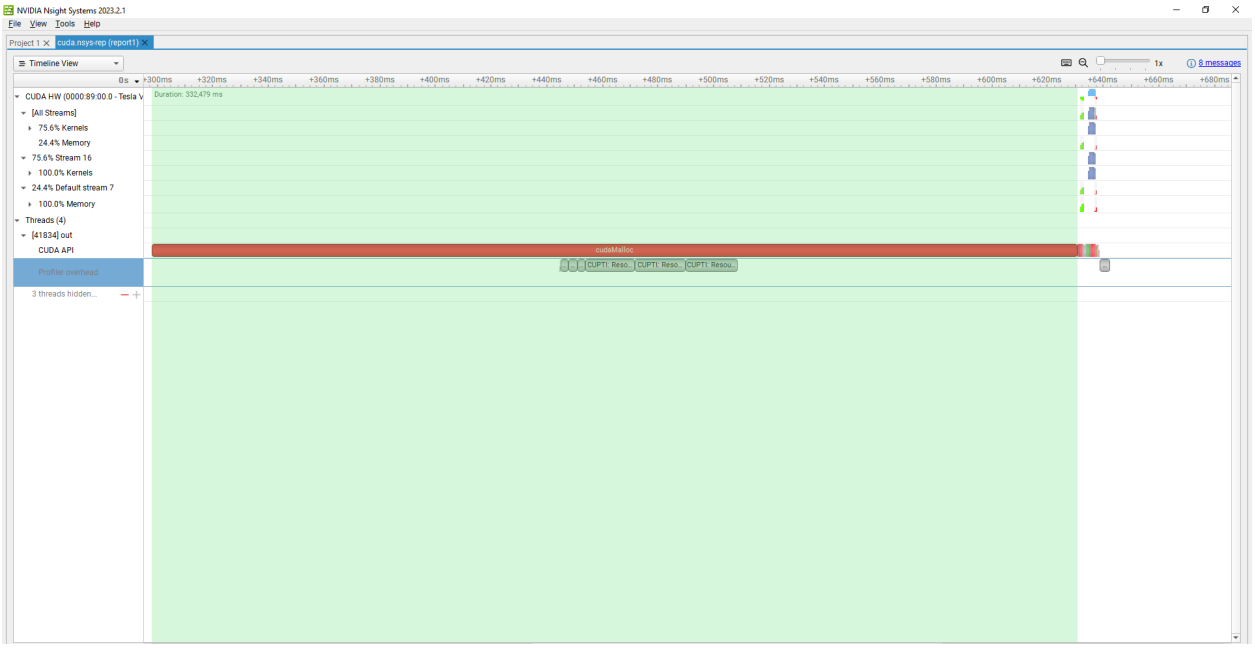


Освобождение памяти контекста cuBLAS

Профилирование реализации через CUDA

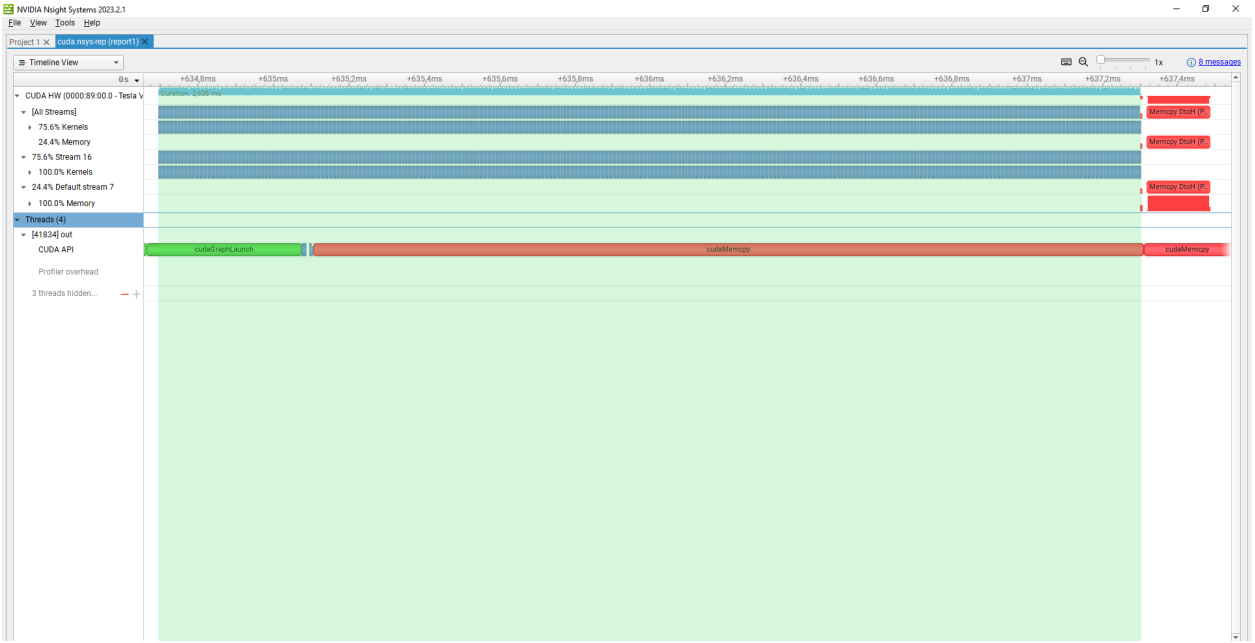


Общий вид профилирования

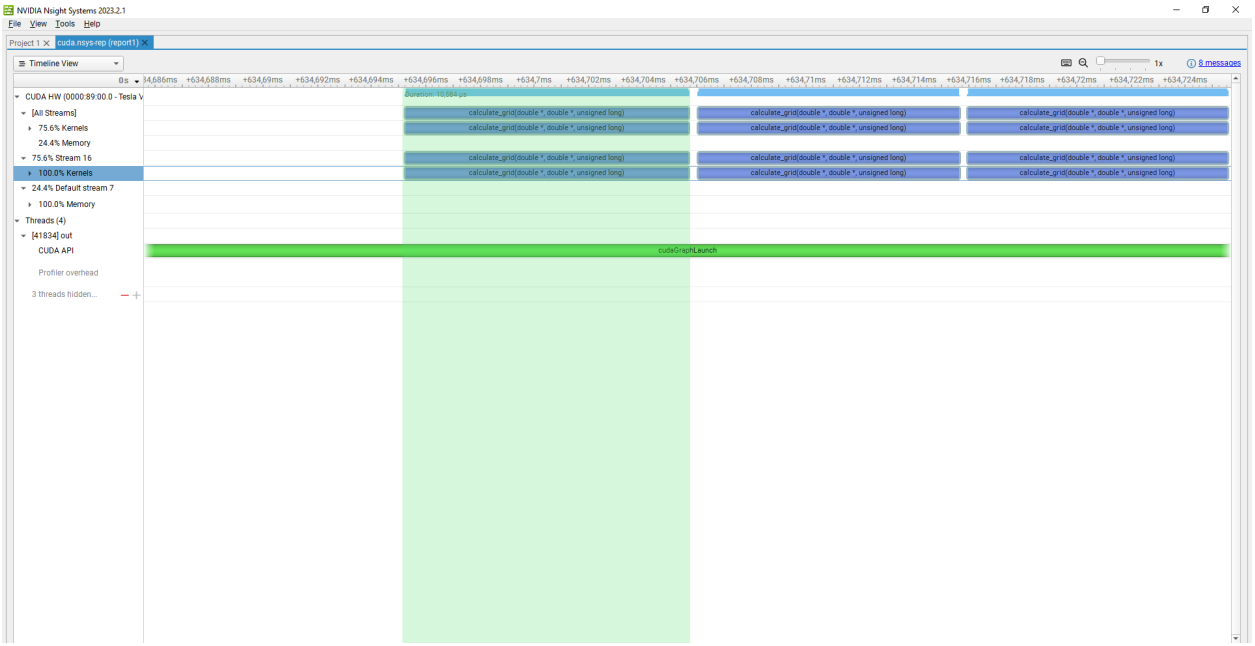


Перенос данных на GPU + выделение памяти

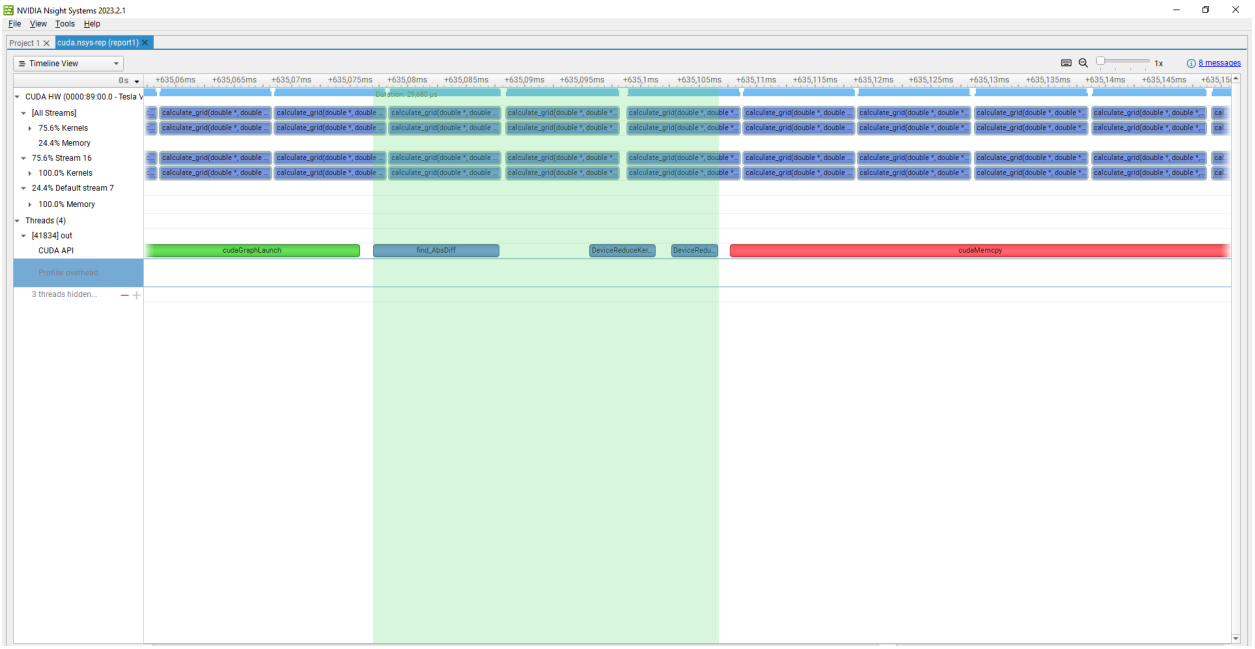
Инициализация графа



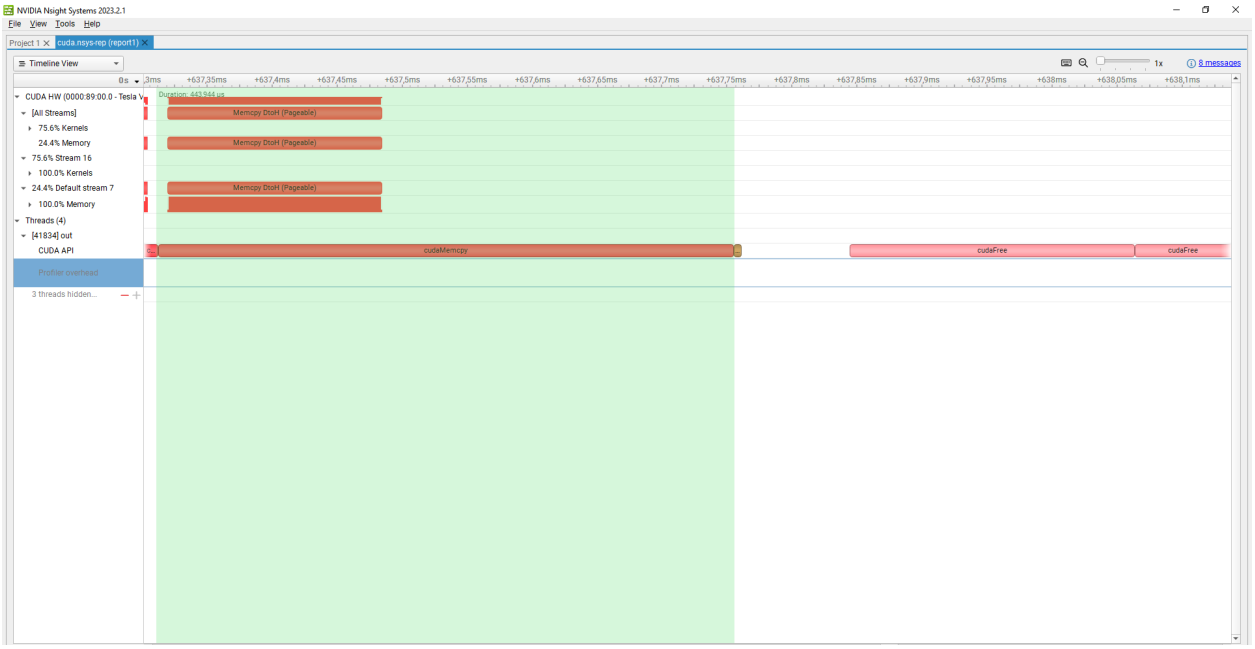
Общий вид процесса расчета сетки 512x512 с 256ю итерациями



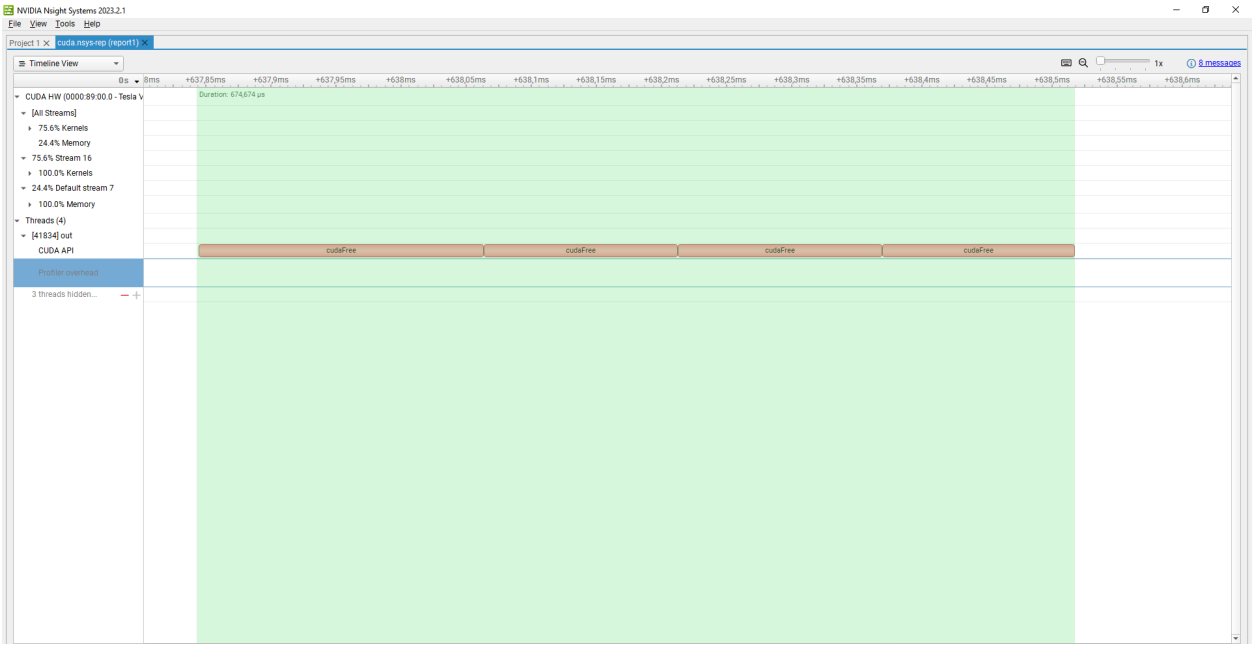
Одна итерация расчета сетки



Нахождение максимальной ошибки



Копирование данных с памяти GPU



Освобождение памяти на GPU