

 eecse4750 / e4750_2021Fall_students_repo Public[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Settings](#)

Introduction to CUDA Profiling using Nvidia Nsight Compute

[Jump to bottom](#)

Arjun Ahuja edited this page 5 days ago · 2 revisions

Introduction to CUDA Profiling

Relevant Links:

1. https://developer.nvidia.com/nsight-compute-2019_5
2. <https://docs.nvidia.com/nsight-compute/NsightComputeCli/index.html>

Note with the version of CUDA we are using with ubuntu most likely you will be using NSIGHT 2019.x. CLI commands are a bit different post 2020.x. Be sure to refer to relevant CLI commands.

What is profiling?

When you are in the business of making efficient, fast code, you need tools that help you iron out inefficiencies and bottlenecks that might be slowing down your programs.

Code profiling will give you answers to questions like:

- How long does the execution of a portion of my code take?
- How much memory is this process occupying at runtime?
- Is there some unseen overhead causing the code to be slower than expected?
- Core usage, either on CPU or GPU

Tools that allow you to collate and interpret such details are called code profilers.

CUDA Profiling allows you to find answers to such questions and is therefore an indispensable tool. Until 2020 this tool used to be called Nvidia Visual Profiler (NVVP and NVProf). From 2020 Nvidia has moved to a new profiler Nvidia Nsight which has three components NSight Systems, Nsight Compute, Nsight Graphics. For this course we will be using only Nsight Compute you are welcome to try out other tools too.

Prerequisite for running on GCP VM

Either you can run nsight as administrator/root or execute the following commands to allow non administrators to run profiling.

- Create and open .conf file (e.g. profile.conf) in folder /etc/modprobe.d

```
vim /etc/modprobe.d/profile.conf
```

- Add below line in profile.conf

```
options nvidia "NVreg_RestrictProfilingToAdminUsers=0"
```

- Close file /etc/modprobe.d/profile.conf
- run the following command:

```
sudo update-initramfs -u
```

- Reboot the system using:

```
sudo reboot
```

Profiling with Nsight

Run a command in the same vein as example below.

```
(cudaEnv)$ nv-nsight-cu-cli -o metrics python test.py > output.txt
```

This command will profile the code in test.py and pipe the output to output.txt. Also a report will be generated with file name: "metrics.nsight-cuprof-report"

Make sure to add the following line at the top of your python file.

```
#!/usr/bin/env python
```

```
"""
```

```
.  
.   
.
```

Python Code

```
.  
.   
.   
"""
```

Also, make sure your code is executable by running:

```
$ chmod u+x my_pycuda_code.py
```

Viewing the profiling result with Nsight Compute GUI

Note: For this you must first setup a GUI using VNC Viewer, please refer to https://github.com/eecse4750/e4750_2021Fall_students_repo/wiki/VM-GUI-Setup for instructions.

Make sure you run the following command on GUI to view the profiling output:

```
nv-nsight-cu metrics.nsisht-cuprof-report
```

Please ignore the output on the CLI. (should be a bunch of warnings/errors).

EECS E4750: Heterogeneous Computing for Signal and Data Processing (Fall 2021)

► **Pages** 18

E4750 Course Wiki Home

1. [Home Page](#)

2. Tutorials

- Google Cloud
 - [Google Cloud VM Setup](#)
 - [GUI Installation](#)
- Code
 - [Python](#)
 - [PyCUDA Tutorial](#)
 - [Indexing in CUDA & OpenCL](#)

3. Concepts and Additional How-Tos

- [CUDA Profiling](#)
- [CUDA Cores v. Threads](#)

- [Data Types](#)
- [Timing execution in PyOpenCL](#)

4. Assignments (distributed from the Code section)

1. Assignment 1
2. Assignment 2

Clone this wiki locally

https://github.com/eecse4750/e4750_2021Fall_students_repo.wiki.git

