Exercise 9

360.252 - Computational Science on Many-Core Architectures
WS 2021/22

January 12, 2022

The following tasks are due by 23:59pm on Tuesday, January 18, 2022. Please document your answers (please add code listings in the appendix) in a PDF document and email the PDF (including your student ID to get due credit) to karl.rupp@tuwien.ac.at.

You are free to discuss ideas with your peers. Keep in mind that you learn most if you come up with your own solutions. In any case, each student needs to write and hand in their own report. Please refrain from plagiarism!

> "The borrowed thoughts just like the borrowed money,
> only showcase the poverty bestowed on the borrower.'"
> — Lady Marguerite Blessington

There is a dedicated environment set up for this exercise:

https://gtx1080.360252.org/2021/ex9/.

To have a common reference, please run all benchmarks for the report on this machine.

HIP (3 Points)

In order to extend your previous implementation of conjugate gradients to AMD GPUs, please convert the CUDA implementation of the Conjugate Gradient solver from previous exercises to HIP (2 Points). Compare the performance with the CUDA implementation and discuss any performance differences you may observe (1 Point).

COVID simulator (4 Points + 2 Bonus)

The exercise environment provides a simple CPU-based COVID pandemic simulator. In a loop over each day of the year, each individual is simulated whether or not it transmits the virus. Further details can be found right in the code.

Please work on the following:

- The simulator includes a random number generator based on `rand()`. Implement a mechanism to retrieve random numbers inside GPU threads based on either pseudo-random numbers

generated on the GPU[1] or a pre-generated pool of random numbers copied over from the CPU to the GPU. Describe how your implementation works and justify your choice. (1 Point)

- Port the simulator to the GPU using either CUDA, HIP, or OpenCL. Port the initialization phase (1 Point) as well as the simulation phase (1 Point). For each day simulated only the current number of infections should be communicated back from the GPU, all other data should remain on the GPU.
- Develop a simple performance model and compare it to the observed execution times. (1 Point)
- **Bonus**: Suggest a mechanism to speed up the CPU implementation significantly (i.e. more than 2x). Is the optimization easily applicable to the GPU port? (1 Point)
- **Bonus**: Implement a non-trivial[2] refinement of your choice in the COVID simulator (CPU-only suffices, but GPU-implementations preferred). (1 Point)

[1] see for example https://developer.nvidia.com/gpugems/gpugems3/part-vi-gpu-computing/chapter-37-efficient-random-number-generation-and-application

[2] Simply changing existing parameters is considered trivial. It should be at least a few lines of code. Good ideas with little code are better than a lot of code implementing a bad idea. ;-)