ME759

High Performance Computing for Engineering Applications

Default Midterm Project Option 1: Solving a Large, Dense Banded Linear System

Date Due: November 15, 2013 – 23:59 PM

Intermediate Report Due: October 31– 23:59 PM

You will have to write CUDA code that solves on a GTX480 and Kepler K20X a linear system $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{n \times n}$ is a banded matrix [1], and $\mathbf{b} \in \mathbb{R}^{n \times m}$. The easy way out is to take $m = 1$. Kudos to you if your program handles the nontrivial case $m > 1$; i.e., when the linear system has multiple right hand sides.

This project is only vaguely specified in terms of the size and structure of the dense matrix $\mathbf{A}$. It will be up to you to push the limit on the value of n and the value of the bandwidth k. The goal is to solve systems as large as possible, as fast as possible. Note that between solving a system with $n = 10^7$ and a small value of k such as $k = 20$, I prefer the scenario where the matrix has a smaller dimension but a larger bandwidth. That is, I am more interested in cases where the values of n and k are relatively close, say $k \approx 0.5n$. However, it is ok if you prefer the former scenario. In terms of input, generate your own $\mathbf{A}$ and $\mathbf{b}$ inputs. To keep things simple, have $\mathbf{A}$ be diagonally dominant, set $\mathbf{x} = [1, 1, \ldots, 1]^T$, and choose $\mathbf{b} = \mathbf{A} \cdot \mathbf{x}$ (in other words, you know what the solution should be).

In your report, you will have to touch on the following:

- The mathematical algorithm embraced to solve this problem
- The format in which the code expects the inputs $\mathbf{A}$ and $\mathbf{b}$ to be provided.
- Your software design solution. Comment on
  a) your use of shared memory, if any
  b) the type of global memory access (coalesced vs. non-coalesced)
  c) use of synchronization barriers
  d) any other CUDA features relevant to your design
- Run a **cuda-memcheck** on the final version of your code from within **cuda-gdb** and provide a printout of the report produced by **cuda-memcheck**. Comment on any unusual output you notice in that report.
- Profile your code using **nvvp** and interpret/comment on the profiling results. Include pictures if helpful.
- Run a scaling analysis on GTX480 and K20X. To this end, consider a variety of dimensions n and a variety of bandwidths k. Understand how the new Kepler architecture is impacting the run time.
- Compare your linear solver against Intel's MKL banded solver over a spectrum of dimensions n and bandwidths b. The MKL banded solver is available on Euler.

REMARKS:

a) If you write code that systematically beats the MKL banded solver over a reasonable spectrum of dimensions n and bandwidths k you will earn an automatic A grade in ME759.
b) I would be very happy to meet with you and discuss algorithm design ideas. This can happen during or outside office hours.
c) You can work alone or team up with one other ME759 colleague to work on this project.
d) An intermediate report that documents your progress towards finishing this project is due on October 31.

REFERENCES:

[1] Band Matrix: Wikipedia http://en.wikipedia.org/wiki/Band_matrix (accessed October 11, 2013)