ME 571 Homework 4

Gurpal Singh

March 22, 2017

Abstract

The purpose of this assignment was to compare the computation times for 2 different tasks on the CPU vs. the GPU. The first task was to time simple vector addition on the CPU and GPU. The second task was to time the 1-d Stencil computation on the CPU and GPU. From the findings, it is evident that GPU computations in parallel were quite faster

1 Introduction

In this assignment, parallel computing was explored. CUDA, a parallel computing platform and application programing interface, was used to explore parallel computing. Two simple tasks were completed to see the difference in the amount of time that it take the CPU versus the GPU to do the calculations. Three different vector sizes of n = 1,000,000, n = 10,000,000, and n = 100,000,000 were used for comparison. The GPU is able to divide up the work into blocks and threads and complete the workload faster. The results in the following sections show the usefulness of computation in parallelism.

2 Task 2: Vector Addition

The time for the GPU computation is significantly faster than the CPU computation. Two vectors of the same size with n elements were added together to produce a third vector. In the CPU computation, the vectors are added using a for-loop. This means the for-loop must go through n elements in order. In the GPU computation the blocks are able to work on different parts of the addition. The vector addition is done with each block meaning the computation should be quicker than the CPU method. The results are presented below in table format. The source code file is included with this report and is named "task2.cu." The output results file is also attached and is named "Task2_Result.txt."

Table 1: Vector Addition Results

Vector Length n	CPU Time (ms)	GPU Time (ms)
1,000,000	13.72	9.02
10,000,000	136.68	88.76
100,000,000	1387.02	887.41

3   Task 3: 1-d Stencil

The add vector example was modified to include a CPU function and GPU function to do the stencil computation. The objective of the stencil 1-d computation was essentially to take a moving average of 3 elements to the left and 3 elements to the right and store the result in a new vector. Again, the GPU computation was significantly faster due to the fact that the workload can be divided up to different blocks. The results for an input vector of various sizes is presented in table 2. The source code file is included with this report and is named "task3.cu." The output results file is also attached and is named "Task3_Result.txt."

Table 2: Stencil 1-d Results

Vector Length n	CPU Time (ms)	GPU Time (ms)
1,000,000	21.16	6.00
10,000,000	212.56	59.55
100,000,000	2119.36	594.06