Sergi Aliaga

Northeastern University

# Homework 5

## Question 4

*Let's revisit computing the value of pi, but this time we will use a series. For instance, we provide you code for the Leibniz's series, developed by Jose Cintra. Implement this series on the GPU, allowing the user to enter the number of iterations. Make sure to find an efficient computation of this kernel that utilizes the parallelism provided on the GPU. Then modify this code to use single precision math. Show results for at least 10 different number of iterations of the series and discuss how precision plays a role in the rate of convergence.*

The code used for this part of the assignment is included in the files *"Q4.cu"* and *"Q4_single.cu"* for double and single precision, respectively. Again, the version of CUDA used is 10.2 and the version for gcc is 6.4.0.. For both versions, 11 different values of iteration have been tried and measured (N = $2^{10}$, $2^{11}$, …, $2^{20}$). Results are reported in Figure 1. From the Figure, we again observe the scalability properties of GPU, resulting into a fairly constant compute time for the different values of input iterations, which increase exponentially.

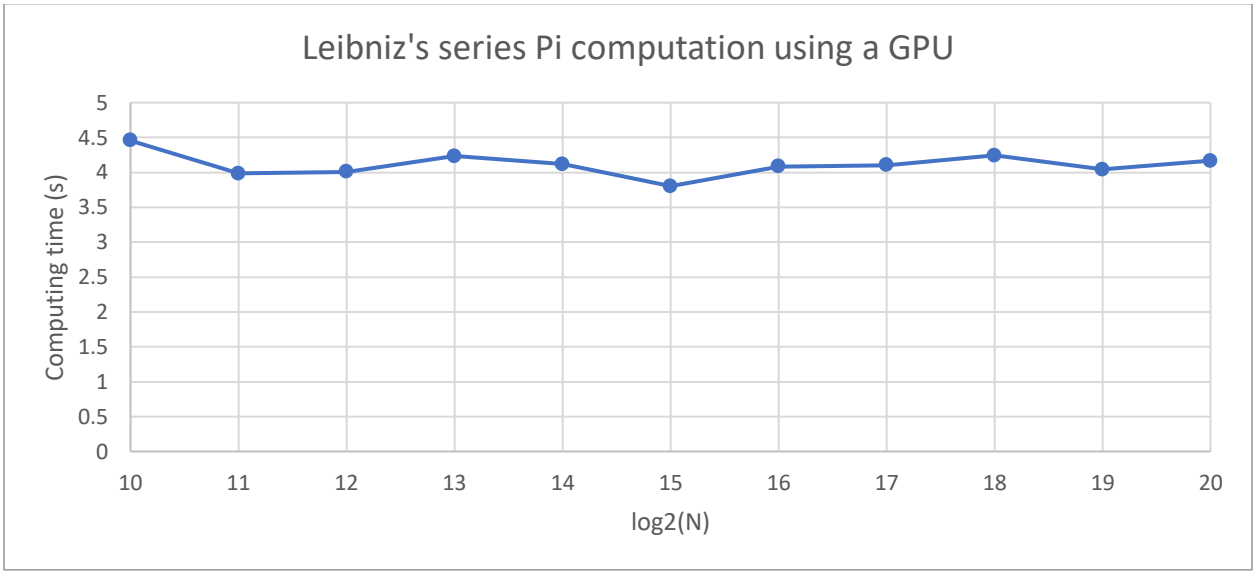


Figure 1: Performance of the single precision Leibniz's series Pi computation using a GPU

The results for double precision have not been included because the proposed code uses the "atomicAdd" function from cuda, which is incompatible with double precision input variables for the architecture of the GPU used. An attempt to provide a double atomic implementation as indicated in here has been tried but did not work either.