Blakley Parker

3/15/19

CSCI-473-SP

Dr. Jones

**Monte Carlo Integration**

Utilizing the math behind integrals, this command line program finds the area underneath a given function with bounds on an XY plane. This may seem complex, but with the power of Monte Carlo simulations this problem can be solved with a small amount of error. It works by spawning a user-specified number of points and using the ratio of points underneath/above the function to find the calculated bounded integral. The calculated area will not be perfectly correct as the generation of points is technically random, but as the number of points generated increases, the error trends downward. After creating the serial version, I progressed to a parallel version where we can monitor various statistics with a varying number of processes and n. Both versions were created in C and can be compiled easily at the command line with either,

Serial,

**gcc –Wall mc-integration-serial.c –o mc-integration**

Parallel (with MPI),

**mpiicc –Wall mc-integration-parallel.c –o mc-integration**

and can be ran with,

**./mc-integration <a> <b> <n>**

**a = first x bound**

**b = second x bound**

**n = number of points generated**

To test the performance of the program the same function and bounds will be used while modifying ‘n’ to find the optimal speed-up, efficiency percent, and execution time. The bounded integral used for this experiment is...

A = 333.3333333

**Data gathered from n = 100000000**

Average error after 10 runs: 0.000306667

Note: I noticed an irregularity while running the program multiple times with this number of generated points. After 8 processes, the execution time seemingly goes up. This causes the speed-up and efficiency percent to deviate from what would be expected.

**Data gathered from n = 1000000000**

Average error after 10 runs: 0.0000446667

Note: Execution and speed-up acted as expected with this number of points generated