Performance Evolution of GPU versus CPU in Iterative algorithms

Follow this and additional works at: https://ecommons.udayton.edu/stander_posters

Recommended Citation
https://ecommons.udayton.edu/stander_posters/1229

This Book is brought to you for free and open access by the Stander Symposium at eCommons. It has been accepted for inclusion in Stander Symposium Posters by an authorized administrator of eCommons. For more information, please contact frice1@udayton.edu, mscllangen1@udayton.edu.
Performance Evolution of GPU versus CPU in Iterative Algorithms
Hassan Alsaad
Advisor: Mohammadjafar Esmaeili, Ph.D.
Department of Engineering Management, Systems, and Technology, University of Dayton

Objectives
- Evaluate Graphics Processing Unit (GPU) and Central Processing Unit (CPU), two of the main technologies used in high performance computing.
- Determine the advantages and disadvantages of GPU and CPU in multi-threaded applications.

Methodology
1. Develop a C++ code for an iterative program using a multi-threaded algorithm and analyze the execution times in both GPU and CPU.
2. Examine the performance of CPU and GPU under different iterative programs.
3. Develop the same testing platform using OpenCL and analyze the GPU and CPU performance using GPU-Intel 4000 HD and CPU Intel Core i7 3570M.

Results
- Matrix multiplication and division adopted for 1 billion, 100 million, 10 million and 1 million iterations.

Conclusions
1. CPU is more efficient in small-scale iterative algorithms.
2. GPU performance is better than CPU in large-scale parallel computing iterations.
3. The combination of a GPU and CPU can deliver much better performance.