CS554 Project Ideas

GeMTC:MIC – Supporting MTC Applications on Intel Xeon Phi Many-Core Accelerators

Overview

GeMTC is a CUDA based GPU framework which allows Many-Task Computing workloads to run efficiently on NVIDIA GPUs. NVIDIA is only one manufacturer of hardware accelerators; several other brands/manufacturers include AMD GPUs and the Intel Xeon Phi.

This project aims to provide support for the Intel Xeon Phi. The Intel Xeon Phi is a hardware coprocessor from Intel. It is a PCI device with roughly 60 cores and over 240 hardware threads. To program GPUs you typically need to learn another programming language such as CUDA (NVIDIA) or OpenCL (AMD). However, the Intel Xeon Phi contains traditional x86 cores, which are programmed using traditional programming languages(C, OpenMP, Pthreads) or new emerging languages (e.g. OpenCL and OpenACC).

For this project you will write a module (in any language such as C/PThreads/SCIF, OpenCL, OpenMP, or OpenACC) that plugs into the GeMTC framework enabling tasks called from the GeMTC API to run on the Intel Xeon Phi. Some preliminary work was done in 2013 on enabling MTC on the Xeon Phi accelerator, see GCASR 2013.

Relevant Systems and Reading Material

- GeMTC – http://datasys.cs.iit.edu/projects/GeMTC
- Xeon Phi - http://software.intel.com/en-us/mic-developer
- Swift – http://swift-lang.org
- Scott J. Krieder, Justin M. Wozniak, Timothy Armstrong, Michael Wilde, Daniel S. Katz, Benjamin Grimmer, Ian T. Foster, Ioan Raicu. “Design and Evaluation of the GeMTC Framework for GPU-enabled Many-Task Computing”, ACM HPDC 2014; http://datasys.cs.iit.edu/publications/2014_HPDC14_GeMTC.pdf
- Jeff Johnson, Scott Krieder, Benjamin Grimmer, Justin Wozniak, Michael Wilde, Ioan Raicu. "Understanding the Costs of Many-Task Computing Workloads on Intel Xeon Phi Coprocessors", 2nd Greater Chicago Area System Research Workshop (GCASR), 2013; http://datasys.cs.iit.edu/reports/2013_GCASR13_paper_MIC.pdf

Preferred/Required Skills

- Required: C
- Preferred: OpenMP, OpenACC, OpenCL, Threaded programming
- No CUDA programming skills required

Evaluation

You will be responsible for writing several test applications that utilize this software stack as well as some high level tests other developers can run to assert code is functioning properly, similar to those found in the HPDC 2014 paper. Experiments can be done on a single node, on a single Xeon Phi on the Jarvis cluster.

Project Mentor

Ioan Raicu, iraicu@cs.iit.edu