Supplementary Material

1 LSU History with GPUs

The researchers in LSU's CCT and HPC group, members of Cactus, Pluto, STE| |AR, and LA-SiGMA projects, and the investigators in the proposal are well qualified to meet these goals. The CCT has administered a (now five years old) GPU cluster, Spider, with 32 NVIDIA Tesla GPUs since 2005. The LSU HPC group administers 16 CPU clusters totaling over 110 TF including the TeraGrid Queen Bee resource, in addition to a 2-node with 6 Fermi M2070 GPUs, a 4-node and a 8-node with 8 and 16 Fermi M2090 GPUs. The CCT events team includes 4 full time staff who host a variety of computational sciences workshops. Last summer, the CCT hosted 12 workshops each lasting from several days to a week. The Cactus project provides a general parallel framework for a variety of scientific applications with short ranged correlations. This request will enable Cactus to add elements to the framework which enable GPU acceleration, aiding all who use Cactus to quickly derive performance benefits.

2 Community Leadership

2.1 Recent Awards

- First Prize at 2nd IEEE International Scalable Computing Challenge (SCALE 2009), Shanghai May 2009, for Large Scale Problem Solving Using Automatic Code Generation and Distributed Visualization.

2.2 Funding Grants with GPU hardware included

- *Louisiana Alliance for Simulation-Guided Materials Applications (LA-SIGMA): Leveraging Next Generation Supercomputing for the Study of Complex Multiscale Phenomena in Materials*; Agency: NSF and Louisiana Board of Regents; Inclusive Dates: 9/01/10 - 9/30/15; Annual Funding: $1,824K
- *STCI: An Environment for Portable High Productivity High Performance Computing on GPUs/Accelerators*; Agency: NSF; Inclusive Dates: 9/01/09 - 8/31/12; Annual Funding: $88K
- *MRI: Development of Melete: an interaction-oriented, software-rich compute cluster with tangible interface support for collaborative research and the classroom*; Agency: NSF; Inclusive Dates: 9/01/11 - 8/31/15; Annual Funding: $225K
- *II-NEW: Research Software Infrastructure for Tensor Computations*; Agency: NSF; Inclusive Dates: 6/01/11 - 5/31/14; Annual Funding: $100K
- *II-NEW: Shelob - A Heterogeneous Computing Platform to Enable Transformation of Computational Research and Education in the State of Louisiana*; Agency: NSF; Inclusive Dates: 7/01/12 - 6/30/15; Annual Funding: $180K

2.3 Software and Libraries Released

- *CaCUDA (http://code.google.com/p/cacuda/)* We present a massive data parallel computational framework that can be used to develop large scale scientific applications on such petascale/exascale hybrid systems. The framework is built upon the highly scalable Cactus computational framework that has been used by scientists and engineers from various fields to solve a wide spectrum of real world scientific problems. Within Cactus, we further explore the performance and scalability of hybrid systems by making use of the computing power of the accelerating devices, in particular GPU's connected to many-core CPU's.
- *PLUTO (http://pluto-compiler.sourceforge.net/)* is an automatic parallelization tool based on the polyhedral model. The polyhedral model for compiler optimization provides an abstraction to perform high-level transformations such as loop-nest optimization and parallelization on affine loop nests. Pluto transforms C programs from source to source for coarse-grained parallelism and data locality

simultaneously. The core transformation framework mainly works by finding affine transformations for efficient tiling and fusion, but not limited to those.

2.4 Selected Publications

- Marek Blazewicz, Steven R. Brandt, Peter Diener, David M. Koppelman, Krzysztof Kurowski, Frank Löffler, Erik Schnetter, and Jian Tao. A massive data parallel computational framework on petascale/exascale hybrid computer systems. In *International Conference on Parallel Computing*, Ghent, Belgium, 2011.

- Jian Tao, Marek Blazewicz, and Steven R. Brandt. GPGPU Stencil Computation Kernel Abstraction and Implementation for the Development of Large Scale Scientific Applications on Hybrid Systems, Poster Paper. In *17th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, 25-29 February 2012, New Orleans, Louisiana USA, 2011.

- L.-N. Pouchet, U. Bondhugula, C. Bastoul, A. Cohen, J. Ramanujam, P. Sadayappan and N. Vasilache. Loop Transformations: Convexity, Pruning and Optimization. In *Proc. Symposium on Principles of Programming Languages (POPL),* pp. 549–561, Austin, TX, January 2011.

- M. Baskaran, J. Ramanujam, and P. Sadayappan. Automatic C-to-CUDA Code Generation for Affine Programs. In *Proc. CC 2010 - International Conference on Compiler Construction,* (R. Gupta Ed.), Lecture Notes in Computer Science, Vol. 6011, pp. 244–263, Springer-Verlag, 2010.

- M. Baskaran, U. Bondhugula, S. Krishnamoorthy, J. Ramanujam, A. Rountev, and P. Sadayappan. A Compiler Framework for Optimization of Affine Loop Nests for General Purpose Computations on GPUs. In *Proc. 22nd ACM International Conference on Supercomputing,* pp. 225–234, Island of Kos, Greece, June 2008.

2.5 Conference Tutorials

One of the co-PIs, Jagannathan Ramanujam, has taught (with Prof. P. Sadayappan) a number of tutorials on compiler optimizations and programming models for GPUs at recent conferences and these are listed below:

- *Multicore Processors and GPUs: Programming Models and Compiler Optimizations*:
  - half-day tutorial at the International Symposium on Code Generation and Optimization (CGO 2012), San Jose, CA, April 2012;
  - half-day tutorial at the 20th International Conference on Parallel Architectures and Compilation Techniques (PACT), Galveston, TX, October 2011.

- *GPUs and General-Purpose Multicores: Programming Models, Compiler Optimization and Tuning,* half-day tutorial at the 25th International Conference on Supercomputing (ICS 2011), Tucson, AZ, June 2011.

- *GPU Programming Models, Optimizations and Tuning,* half-day tutorial at the International Symposium on Code Generation and Optimization (CGO 2011), Chamonix, France, April 2011.

- *Programming Models and Compiler Optimizations for GPUs and Multicore Processors*:
  - full-day tutorial at the 18th International Conference on Parallel Architectures and Compilation Techniques (PACT 2009) Raleigh, NC, September 2009;
  - half-day tutorial at the 23rd International Conference on Supercomputing (ICS 2009), New York, NY, June 2009;
  - half-day tutorial at the 14th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP 2009), Raleigh, NC, February 2009.

2.6 Distance learning graduate level classes

Last year, nine graduate courses were offered, in many cases involving faculty from different locations:

- *GPU Programming* by David Koppelman (LSU): Programming GPUs for graphics and non-graphical computations using OpenGL, OpenGL Shader Language, and CUDA.
- *GPU Microarchitecture* by David Koppelman (LSU): Low-level design and programming of GPUs, including organization, memory hierarchy, and native instruction sets, with examples.
- *Performance Optimization of Applications on GPUs* by Ramanujam (LSU): Optimization of science applications, GPUs, heterogeneous systems, CUDA, OpenCL
- *Program Parallelization* by Ramanujam (LSU): Compiler optimizations, source-to-source transformations. Programming paradigms and support for HPC architectures.
- *Introduction to Scientific Computing* by Gabrielle Allen (LSU, NSF), and Frank Löffler (LSU).
- *Computational Physics*: Computing for Petascale Systems by Karen Tomko (Ohio Supercomputer Center) and Juana Moreno (LSU).
- *Simulations of Quantum Many-Body Systems* by Matthias Troyer (ETH Zurich) and faculty at the Ludwig-Maximilians-Universität, Munich; École normale supérieure, Paris; Max Planck Institute for the Physics of Complex systems, Dresden; University of Massachusetts, Amherst; University of Wyoming; and Louisiana State University.
- *Advanced Solid State Physics with Computation* by Mark Jarrell (LSU).
- *Computational Solid State Physics* by Rongying Jin (LSU) and Cyrill Slezak (Hillsdale College, Michigan).

References

[1] Tom Goodale, Gabrielle Allen, Gerd Lanfermann, Joan Massó, Thomas Radke, Edward Seidel, and John Shalf. The Cactus framework and toolkit: Design and applications. In *High Performance Computing for Computational Science - VECPAR 2002, 5th International Conference, Porto, Portugal, June 26-28, 2002*, pages 197–227, Berlin, 2003. Springer.

[2] http://www.cactuscode.org/. Cactus Computational Toolkit home page.

[3] PLUTO. http://pluto-compiler.sourceforge.net/.

[4] Hartmut Kaiser, Maciej Brodowicz, and Thomas Sterling. ParalleX: An advanced parallel execution model for scaling-impaired applications. In *Parallel Processing Workshops*, pages 394–401, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

[5] The HPX project. http://stellar.cct.lsu.edu/downloads/.

[6] The LA-SiGMA project. http://www.institute.loni.org/lasigma/.