OpenUH – An Open Source OpenACC Compiler

Xiaonan Tian, Rengan Xu, Yonghong Yan, Zhifeng Yun, Sunita Chandrasekaran, Barbara Chapman
Department of Computer Science, University of Houston
Email: {xtian2, nxu6, yyan3, zyun, schandrasekaran, bchapman}@uh.edu  http://web.cs.uh.edu/~openuh

Introduction

- OpenACC is an emerging directive-based programming model for programming accelerators that typically enable non-expert programmers to achieve portable and productive performance of their applications.
- We constructed a prototype open-source OpenACC compiler OpenUH which is based on a branch of main stream Open64 compiler. The experiences could be applicable to other compiler implementation efforts.
- We provide multiple loop mapping strategies in the compiler on how to efficiently distribute parallel loops to the GPGPU accelerators. Our findings provide guidance for users to adopt suitable loop mappings depending on their application characteristics.
- OpenUH compiler adopts a source-to-source approach and generates readable CUDA source code for GPGPUs. This gives users opportunities to understand how the loop mapping mechanism are applied and to further optimize the code manually. It also allows us to leverage the advanced optimization features in the backend compilation step by the CUDA compiler.

OpenACC Implementation in OpenUH

- An open-source OpenACC compiler is created using OpenUH compiler framework
- Loop mapping mechanisms are designed to translate single loop, double loop and triple nested loop
- Competitive performance compared to a commercial OpenACC compiler
- Explore advanced compiler analysis and transformation techniques to further improve the performance in the future

Results

- Fig. 5: Performance of Double Nested Loop Mapping
- Fig. 6: Performance of Triple Nested Loop Mapping
- Fig. 7: Performance Comparison between PGI and OpenUH

Conclusion

- An open-source OpenACC compiler is created using OpenUH compiler framework
- Loop mapping mechanisms are designed to translate single loop, double loop and triple nested loop
- Competitive performance compared to a commercial OpenACC compiler
- Explore advanced compiler analysis and transformation techniques to further improve the performance in the future

References

Rengan Xu, Xiaonan Tian, Yonghong Yan, Sunita Chandrasekaran, and Barbara Chapman. Reduction Operations in Parallel Loops for GPGPUs, in PIMACOM 2014, Feb., 2014, Orlando, Florida, USA
Xiaonan Tian, Rengan Xu, Yonghong Yan, Zhifeng Yun, Sunita Chandrasekaran, Barbara Chapman. Compiling a High-Level Directive-Based Programming Model for GPGPUs, In LCPC2013, Sep. 2013, San Jose, CA, USA

Acknowledgment

This research was supported by NVIDIA and Department of Energy under Award Agreement No. DE-FG02-12ER20698. We also thank PGI for providing the compiler and support for the evaluation