How can we survive in post-exascale era?

Group B

AICS Youth Workshop 18

Members

- Application Side:
  - Muhammad Redo Ramadhan
    - Condensed Matter Physics
  - Benoit Assi
    - Quantum Physics
- CS Side:
  - W.K.Umayanganie Klassen
    - FPGA Programming Model
  - Antoniette Mondigo
    - Inter-FPGA Communication
  - Yaohung Tsai
    - Autotuning Tools

Motivations

- Application Side:
  - Neural network models.
  - Condensed matter systems.
- CS Side:
  - Power limits are prevalent factors to consider.
  - Silicon transistor density limit has been reached.
  - Traditional CPU and GPU architecture provides insufficient efficiency.
  - Unified programming environment for various platforms.

Hardware Solutions

- GPU:
  - Widely accessible - reasonably priced.
  - Libraries available, for example BLAS and FFT.
  - Power and peak performance: 25 GFlops/Watt in DP.
  - Large community for support.
- FPGA:
  - Highly customized but still programmable.
  - Power and peak performance: 80 GFlops/Watt in SP, 3x higher fixed point performance vs. floating point performance.
  - Partial reconfiguration ability can further reduce power consumption.
  - Integrated into system on chips in some cases.
  - Can be a standalone device.

Challenges

- Programming models:
  - Uniform model is hard to create across platforms.
  - Steep learning curve in writing applications.
- GPU:
  - PCIe connection to MOBO with high latency cost.
  - Specific languages, such as CUDA, differ from standard taught languages such as C/C++.
- FPGA:
  - No well established libraries exist.
  - No standard programming languages for HPC.
  - Hard to generate efficient hardware for complex programs.
  - Different software stack for development.

Contributions to challenges

- Unified programming (Uma):
  - Easily programmable for domain scientists with directive based code such as OpenACC with OpenARC compiler
  - OpenARC supports both FPGAs and GPUs.
  - More compiler optimizations for FPGAs.
  - Improve community support using FPGAs with OpenARC.
- Autotuning (Yaohung Mike Tsai):
  - Optimizing the performance of computational kernels.
  - Targeting CPU, GPU, Accelerators.
  - Portable cross architectures.
  - Statistical model to reduce tuning time.
  - Explore possibility to tune on FPGA.

Contributions to challenges

- Inter-FPGA Communication (Antoniette):
  - Generate an optimized hardware for specific streaming applications
  - Exploit FPGA advantages to FPGA clusters (1D ring, torus, etc.)
  - Improve the communication framework to support more network topologies
- Application side (Benoit and Redo):
  - Sparse matrix computation run time not reduced by FPGA as studies have shown.
  - Supporting the mathematical libraries for FPGA is critical.
  - GPU seems to be up to 5x faster for some condensed matter systems.

Conclusion

- We have discussed the likely challenges we will be facing in the post-exa era.
- The solutions we came up with had challenges that accompanied them.
- Group members provided their contributions to specific areas.
- Collaborations between different fields are necessary to combat these issues.