Surviving the End of Scaling of Traditional Microprocessors in HPC

This is a summary of a presentation at IEEE HOT CHIPS 22, Stanford, USA, August 2010.

About the Authors

Olav Lindtjorn is at Schlumberger and Stanford University.
Bob Clapp is with the Center for Computational Earth & Environmental Science (CEES) at Stanford University.
Oliver Pell is VP of Engineering, Oskar Mencer is CEO and Mike Flynn is Chairman at Maxeler Technologies.

Summary

Microprocessors have been hitting the memory and power walls for the past few years, resulting in the current multi-core processor solutions provided by the major microprocessor vendors. In addition to the memory wall and the power wall, since the number of pins per chip does not scale and multiple cores are harder to program than single cores, we are starting to run into real difficulties in scaling and mapping large geophysical applications on the latest microprocessor based machines.

Geophysical applications form a significant component of HPC activity which is driving the high end of the computing industry and from our perspective at Schlumberger, multi-core scaling has recently been disappointing.

Most recently there has been significant effort on acceleration demonstrated last year by nVidia GPGPU technology and in our case, Maxeler with FPGA based accelerators. In our opinion directly mapping a CPU software implementation to an FPGA is doomed to failure. Only by understanding the underlying hardwares strength and weaknesses, and adjusting the algorithmic approach accordingly, can meaningful speedup be achieved.

The role of HPC in oil and gas exploration is both to aid identification of resources, for example by processing seismic data to generate images of the subsurface and identify potential reservoir locations; and also to help access resources, for example by modelling fluid flow in a reservoir during resource extraction. The computational complexity is driven by the large volumes of data involved and the complex physics that we wish to model. For example, Figure 1 shows the relative computational and storage costs of Reverse Time Migration, an imaging algorithm, as the complexity of the physics model is increased.

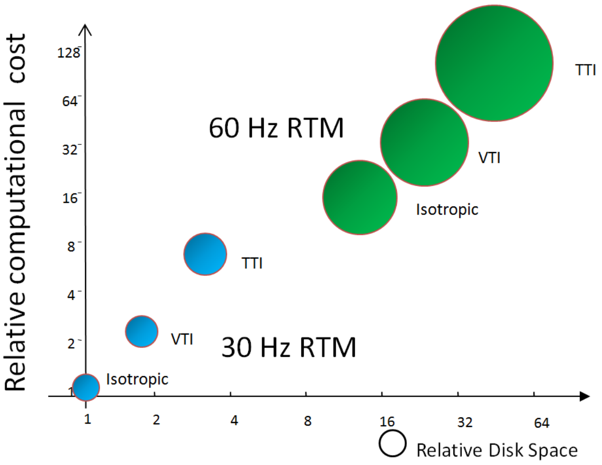

Figure 1: Computation and storage costs for different physics models for Reverse Time Migration.

Many oil and gas applications rely on a relatively small number of computational kernels, making them suitable for acceleration. We target implementation of convolution kernels (for FD modeling and Reverse Time Migration) and sparse matrix solver on an FPGA system using the Maxeler MaxCompiler programming toolset.

Sparse matrix solvers generally suffer poor performance on multi-core systems due to the small number of arithmetic operations per data point. For example, running the Eclipse reservoir modelling tool on 12-cores runs only 3.5x faster than a single core. On FPGA, we can exploit the flexible nature of the accelerator to optimize the data representation and encoding to mitigate the memory bandwidth bottleneck, achieving speedups of 20-40x compared to a non-accelerated compute node.

3D convolution can also suffer from a poor FLOPbyte ratio, and the sparse structure of traditional 'star' shaped convolution operators requires large amounts of buffering which can be mapped efficiently to FPGA on-chip memory. For convolution performance, we can obtain up to a 70x speedup comparing a 1U accelerated to 1U non-accelerated compute node.

Since the FPGA systems run at a low power (e.g. 50-75W per accelerator card), we can obtain about 30x savings in space and similar savings in power consumption, reducing operational costs.

Original Publication Date: August 2010