University of BRISTOL

ACRC Case Study:

Accelerating Drug Discovery

James Price, Amaurys Avila Ibarra, Richard Sessions & Simon McIntosh-Smith

Aims

BUDE (Bristol University Docking Engine) helps researchers in the School of Biochemistry to find new drugs to combat, for example, Prion Disease. The aim of this project was to accelerate BUDE by exploiting new parallel compute engines, while still maintaining the portability of the code.

Codes such as BUDE exist to enable science, and thus it is important for it to be able to run on any available computer equipment. Researchers may not have control over what computers are available to run their codes, with shared university or national resources being a case in point. Teams who can run their code on any equipment will always be able to use the fastest and most cost-effective hardware available, and will thus be at an advantage and able to demonstrate best value to funders.

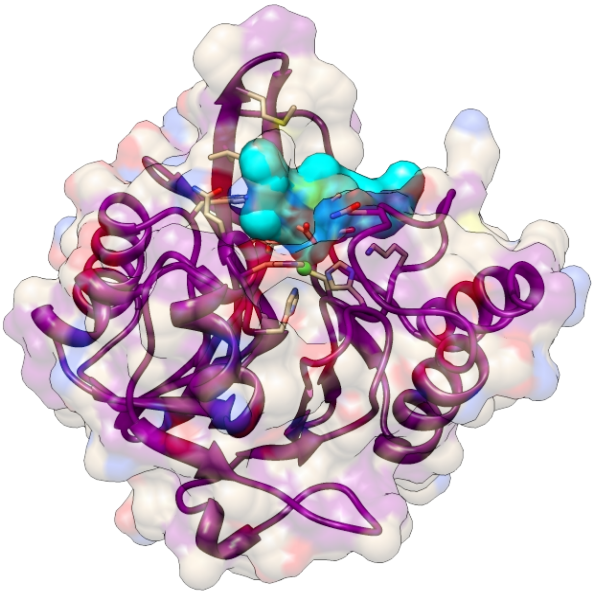

Figure 1: A BUDE docking result: the small molecule in blue binds to the active site of a drug target enzyme blocking its harmful action

Technical Highlights

- The accelerated version of BUDE has been able to simulate some 160 million small molecule (ligand) dockings from a database of 8 million drug candidates in just 2 ½ days on a cluster of commodity GPU accelerators.
- GPUs accelerators are composed of many, simple processors operating in parallel. They are computationally powerful while also being both energy and cost efficient.
- The OpenCL version of BUDE is highly portable and can run on many different platforms: Standard x86 processors, GPUs, Intel Xeon Phi and ARM accelerators.

Challenges

Writing high performance code can be at odds with writing code which is easily maintained. Thus the challenge in this project was to capture the lion's share of the available performance gains, while also ensuring that programmers and researchers to follow would be able to easily understand and adapt the code to their own needs.

Outcomes

By working with programmers in the Department of Computer Science, the original Fortran code was re-written using a combination of the C++ and OpenCL languages. This allows BUDE to be run on a wide variety of hardware platforms. Significant performance gains can be seen (Figure 2) when accelerators are employed. Very good performance can be seen for low-cost, commodity GPU hardware.

University of Bristol, Advanced Computing Research Centre (ACRC), 2014.

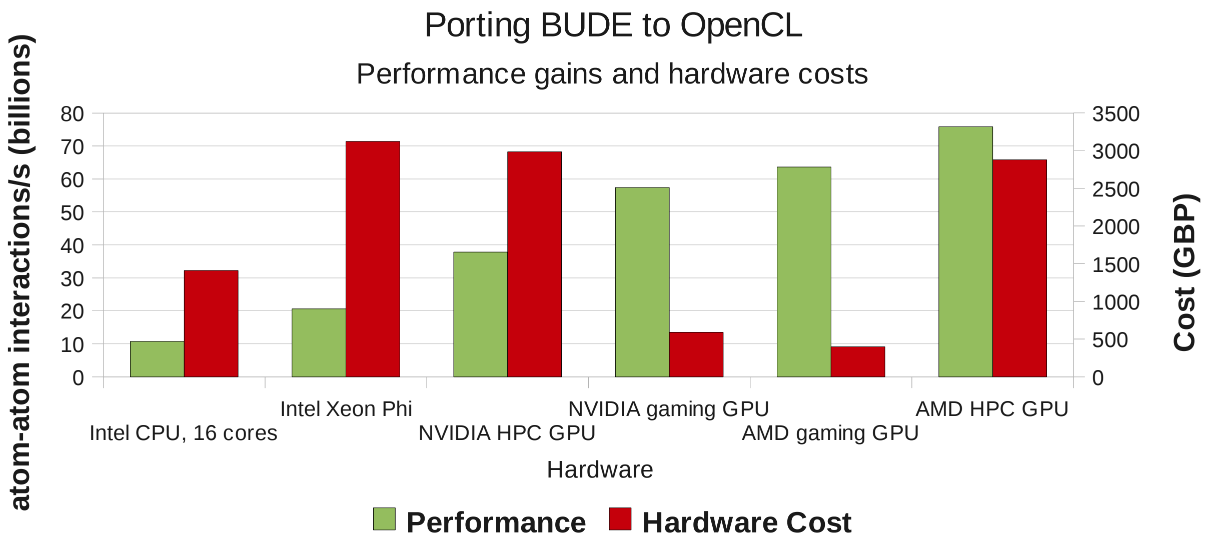

Figure 2: Performance and capital cost comparisons across different computational hardware.
*Intel CPU, 16 cores – **2x Xeon E5-2687W**, Intel Xeon Phi – **SE10P***
*NVIDIA HPC GPU – **Tesla K20c**, NVIDIA gaming GPU – **GTX 780 Ti***
*AMD gaming GPU – **Radeon R9 290X**, AMD HPC GPU – **FirePro S10000***

For a more detailed description of this work see, "High performance in silico virtual drug screening on many-core processors", Simon McIntosh-Smith, James Price, Richard B Sessions and Amaurys A Ibarra, International Journal of High Performance Computing Applications (IJHPCA), published online 9 April 2014. DOI: 10.1177/1094342014528252.

Contact: Dr Gethin Williams
School of Geographical Sciences,
University Road, Bristol. BS8 1SS.
(0117) 928 8301
Gethin.Williams@bristol.ac.uk

Richard Sessions said “An accelerated version of BUDE allows us to search through millions of candidate drug molecules against a given protein drug-target per day. We can do better science faster and cheaper.”

Simon McIntosh-Smith said “Our experience with the BUDE project shows that when domain scientists and computer scientists collaborate, it is often possible to achieve significant speedups in HPC codes, leading to breakthroughs in the science that these applications support. In this case we've succeeded in delivering big speedups even on regular CPUs, but in addition, accelerators such as Intel Xeon Phi, or GPUs from Nvidia and AMD, give additional speedups of up to 7X over even the very latest fast CPUs alone. Accelerators such as Xeon Phi and GPUs are going to become mainstream for HPC in the near future, so it's important that our scientists are aware of this and are investing to prepare their software applications in readiness to exploit these new parallel architectures.”