

Building more intelligent machines: Can 'co-design' help?

S. Gildert November 2010

There are many challenges that we face as we consider the future of computer architectures, and as the type of problem that people require such architectures to solve changes in scale and complexity. A recent article written for HPCwire [1] on 'co-design' highlights some of these issues, and demonstrates that the High Performance Computing community is very interested in new visions of breakthrough system architectures. Simply scaling up the number of cores of current technologies seems to be getting more difficult, more expensive, and more energy-hungry. One might imagine that in the face of such diminishing returns, there could be innovations in architectures that are vastly different from anything currently in existence. It seems clear that people are becoming more open to the idea that something revolutionary in this area may be required to make the leap to 'exascale' machines and beyond. The desire for larger and more powerful machines is driving people to try to create more 'clever' ways of solving problems (algorithmic and software development), rather than just increasing the speed and sheer number of transistors doing the processing. Co-design is one example of a buzzword that is sneakily spreading these memes which hint at 'clever' computing into the HPC community.

Generalization and specialization

I will explain the idea of co-design by using a colorful biological analogy. Imagine trying to design a *general purpose animal*: Our beast can fly, run, swim, dig tunnels and climb trees. It can survive in many different environments. However, anyone trying to design such an animal would soon discover that the large wings prevented it from digging tunnels effectively; that the thick fur coat to survive the extreme cold was not helpful in achieving a streamlined, fast swimmer. Any animal that was even slightly more specialized in one of these areas would quickly out-compete our general design. Indeed, for this very reason, natural selection causes specialization and therefore great diversity amongst the species that we see around us. Particular species are very good at surviving in particular environments.

How does this tie in with computer processing?

The problems that processors are designed to solve today are mostly all very similar. One can view this as being a bit like the 'environmental landscape' that our general purpose creatures live in. If the problems that they encounter around their environment on a day-to-day basis are of the same type, then there is no reason to diversify. Similarly, a large proportion of all computing resources today address some very similar problems, which can be solved quite well using general purpose architectures such as Intel Centrino chips. These tasks include the calculations that underlie familiar everyday tasks such as word-processing, and displaying web pages. But there do exist problems that have been previously thought to be very difficult for computers to solve, problems which seem out of reach of conventional computing. Examples of such problems are face-recognition, realistic speech synthesis, the discovery of patterns in large amounts of genetic data, and the extraction of 'meaning' from poetry or prose. These problems are like the trees and cliffs and oceans of our evolutionary landscape. The general purpose animals simply cannot exploit these features, they cannot solve these problems, so the problems are typically ignored or deemed 'too hard' for current computing platforms.

But there are companies and industries that do care about these problems. They require computing power to be harnessed for some very specific tasks. A few examples include extracting information from genetic data in the biotechnology companies, improving patient diagnosis and medical knowledge of expert systems in the healthcare sector, improving computer graphics for gaming experiences in entertainment businesses, and developing intelligent military tools for the defense industry. These fields all require the searching and sorting of data in parallel, and the manipulation of data on a much more abstract level for it to be efficient and worthwhile. This parallel operation and abstraction is something that general purpose processors are not very good at. They can attempt such a feat, but it takes the power of a supercomputer-size machine to tackle even very small instances of these specialized problems, using speed and brute force to overwhelm the difficulty. The result is very expensive, very inefficient, and does not scale well to larger problems of the same type.

It is this incorporation of variety and structure, the addition of trees, cliffs and oceans, into our computational problems causes our general-purpose processors to be very inefficient at these tasks. So why not allow the processors to specialize and diversify, just like natural selection explores the problem environment defined by our biological needs?

Following nature's example

Co-design attempts to address this problem. It tries to design solutions around the structure of the problem type, resulting in an ability to solve that one problem very well indeed. In practice this is done by meticulous crafting of both software and hardware in synchrony. This allows software which complements the hardware and utilizes subtleties in the construction of the processor to help speed things up, rather than software which runs on a general architecture and incurs a much larger overhead. The result is a blindingly fast and efficient special purpose architecture and algorithm that is extremely good at tackling a particular problem. Though the resulting processor may not be very good at certain tasks we take for granted using general-purpose processors, solving specialized problems instead can be just as valuable, and perhaps will be even more valuable in the future.

A selection of processors which are starting to specialize are discussed in the HPCwire article. These include MDGRAPE-3, which calculates inter-atomic forces, and Anton, a system specifically designed to model the behaviour of molecules and proteins. More common names in the processor world are also beginning to explore possible specializations. Nvidia's GPU based architectures are gaining in popularity, and FPGA and ASIC alternatives are now often considered for inclusion in HPC systems, such as some of Xilinx's products. As better software and more special purpose algorithms are written to exploit these new architectures, they become cheaper and smaller than the brute-force general purpose alternatives. The size of the market for these products increases accordingly.

The quantum processors built by D-Wave Systems [2] are a perfect example of specialized animals, and give an insightful look into some of the ideas behind co-design. The D-Wave machines don't look much like regular computers. They require complex refrigeration equipment and magnetic shielding. They use superconducting electronics rather than semiconducting transistors. They are, at first inspection, very unusual indeed. But they are carefully designed and built in a way that allows an intimate match between the hardware and the software algorithm that they run. As such they are very specialized, but this property allows them to tackle very well a particular class of problems known as discrete optimization problems. This class of problems may appear highly mathematical, but looks can be deceiving. It turns out that once you start looking, examples of these problems are found in many interesting areas of industry and research. Most importantly, optimization forms the basis of many of

the problems mentioned earlier, such as pattern recognition, machine learning, and meaning analysis. These are exactly the problems which are deemed 'too hard' for most computer processors, and yet could be of incredible market value. In short, there are many, many trees, cliffs and oceans in our problem landscape, and a wealth of opportunity for specialized processors to exploit this wonderful evolutionary niche!

Co-design is an important ideas in computing, and hopefully it will open peoples' minds to the potential of new types of architecture that they may never have imagined before. I believe it will grow ever more important in the future, as we expect a larger and more complex variety of problems to be solved by our machines. The first time one sees footage or takes a vacation trip to a tropical rain forest, one can but stare in awe and at the wonders of never-before-seen species, each perfectly engineered to efficiently solve a particular biological problem. I hope that in the future, we will open our eyes to the possibility of an eco-sphere of computer architectures, populated by similarly diverse, beautiful and unusual creatures.

[1] <http://www.hpcwire.com/features/Compilers-and-More-Hardware-Software-Codesign-106554093.html>

[2] <http://www.dwavesys.com/>