Statement of Purpose

Pol Rosello

December 2, 2014

Accelerating Machine Learning

Despite having been studied since the 1960s, the current success of deep learning artificial neural networks was delayed until the mid-2000s, in large part due to hardware bottlenecks resulting in long training times. It was only until the advent of multicore architectures and parallel programming paradigms that they finally gained traction, allowing breakthroughs in speech recognition or Stanford's recent successes at automatically generating image descriptions. Too often are good algorithms shackled by technology constraints. With Moore's Law coming to an end, computer scientists and computer engineers will need to work together to prevent future roadblocks to important machine learning advances. Traditional parallel programming will no longer be enough, and understanding novel heterogeneous computer architectures and special-purpose hardware will increasingly be the key to designing accurate algorithms with acceptable runtimes.

At Cornell University, I pursued a double major in Computer Science and Electrical and Computer Engineering in order to understand the link between high-level algorithms and the underlying hardware. In line with this relationship, I worked in Professor Christopher Batten's computer architecture research group, which follows a vertically-integrated design methodology encompassing algorithm design down to VLSI design. In the group, I mapped the k-Nearest Neighbors algorithm to a research microarchitecture which supports loop parallelization through explicit instructions. After implementing fast nearest-neighbor lookups through the use of an octtree to spatially partition the training set, I profiled the algorithm on a cycle-level simulator to find runtime bottlenecks and modified it to cut its execution time in half. I also implemented a multithreaded octtree generation algorithm which was used to benchmark a reconfigurable on-chip power distribution network for multicore processors which seeks to exploit uneven load balancing across cores. Finally, I expressed the octtree-backed k-NN algorithms using a research API which allows programmers to expose opportunities for parallelism to the underlying hardware. This way, the performance of the multiple algorithmic implementations of k-NN can be quickly evaluated on various microarchitectures. My contributions to both projects were acknowledged in two research papers authored by the group which were accepted in 2014 to the 47th IEEE International Symposium on Microarchitecture [1] [2].

My work in Professor Batten's research group has allowed me to realize that while general-purpose microarchitectures should continue to be optimized for algorithms, many breakthroughs lie in intelligently optimizing algorithms to exploit hardware to its fullest extent. In particular, algorithms with large search spaces frequently give up optimality in favor of acceptable execution times. Although there will always exist a balance between accuracy and speed, overlooking the role of the underlying hardware may lead to ignoring good algorithms which could perform well on special-purpose architectures. Indeed, the price of the delay in the spread of neural networks is too high to pay again. Designing new machine learning algorithms with knowledge about the underlying hardware in mind can bring exciting advances to the field and to society by accelerating the rate at which new algorithms become useful.

Biologically-Inspired Design

One reason behind the success of neural networks is that, in a way, they have been optimized over thousands of years. By drawing inspiration from neurological circuits, neural networks represent a simpler abstraction of a ubiquitous, successful method of learning in nature which has slowly evolved over time. I believe observing natural systems is frequently a good way to find optimized solutions to problems because living organisms are essentially the latest iteration of the longest-running genetic algorithm in existence.

Biologically-inspired design has driven a large part of my academic work as an undergraduate student. My interest in how the mind works has led me to take classes in Neuroscience, Cognitive Science, and Philosophy, and motivated me to become involved in research early on as a sophomore, when I joined

Professor Molnar's research group as a Research Assistant. In his group, I developed a testing platform for a research low-noise multi-electrode array for neural interfaces. I designed a printed circuit board to drive the chip and programmed an FPGA to configure the array for different types of neurophysiological recordings. I have also taken the opportunity to explore the relationship between computer science and biology whenever I have been given the freedom to develop open-ended design projects. In my Embedded Systems class, I implemented an ad-hoc wireless synchronization scheme for microcontrollers in C inspired by the mechanism by which groups of bioluminescent fireflies synchronize their flashes; in my Artificial Intelligence class, I designed a genetic algorithm to simulate the defense optimization of a colony of organisms in the presence of predators.

I believe that the relationship between computer science and biology, just like the relationship between computer science and computer architecture, can be extremely fruitful and symbiotic. Not only can computer science draw from biological systems to design new algorithms, but the tools it provides can be instrumental in new scientific discoveries. In particular, emerging fields such as computational neuroscience are an exciting opportunity for machine learning and data mining approaches to shed light on how complex biological systems work when traditional, reductionist approaches fall short.

Teaching

In addition to my experiences in undergraduate research, I have been active in assistant teaching multiple courses at Cornell. In the spring of my junior year, I was a consultant for Cornell's Data Structures and Algorithms class (CS 2110), holding office hours and grading student exams and assignments. The following semester, I was a Teaching Assistant for Embedded Systems (CS 3420/ECE 3140), in which in addition to grading projects and holding weekly office hours, I mentored students in their final open-ended design projects. Currently, I am a Teaching Assistant for Computer Architecture (CS 4420/ECE 4750), a senior-level, culminating design experience (CDE) course where students implement a quad-core multicore processor with fully-bypassed processors, a two-way associative cache and a ring interconnection network with an adaptive routing algorithm. Alongside holding office hours and grading projects, I am helping the course transition to three-person groups, allowing us to expand the verification components of the projects. Both Embedded Systems and Computer Architecture are crosslisted courses in the ECE and CS departments at Cornell, once again straddling the important link between software and hardware which I have sought to emphasize in my undergraduate career.

Conclusion

The opportunities afforded by the intersection of computer science, computer engineering, and biology are immense; machine learning in particular has a large amount of room to grow in each of the three fields. Stanford University has continually been at the forefront of machine learning research, embracing the kinds of interdisciplinary work which I believe are the key to interesting discoveries. Specifically, my interests in biology align with Stanford's strengths in bioinfiormatics and computational biology. By pursuing a Masters of Science at Stanford, I hope to continue to explore the relationship between computer science and computer engineering as well as its relationship with biology, to develop a more rigorous and mature understanding of machine learning approaches, and to acquire the tools to apply my knowledge in research and development in industry.

References

[1] Shreesha Srinath, Berkin Ilbeyi, Mingxing Tan, Gai Liu, Zhiru Zhang, and Christopher Batten. *Architectural Specialization for Inter-Iteration Loop Dependence Patterns.* 47th ACM/IEEE Int'l Symp. on Microarchitecture (MICRO-47), Dec. 2014

[2] Waclaw Godycki, Christopher Torng, Ivan Bukreyev, Alyssa Apsel, and Christopher Batten. *Enabling Realistic Fine-Grain Voltage Scaling with Reconfigurable Power Distribution Networks.* 47th ACM/IEEE Int'l Symp. on Microarchitecture (MICRO-47), Dec. 2014