A conversation with Jesmin Jahan: Overcoming the compute versus communication scalability wall

INNOVATION LEADERS

by Bushra Anjum

Editor's Introduction

In this series of interviews with innovation leaders, Ubiquity Associate Editor and software engineer, Dr. Bushra Anjum sits down with Dr. Jesmin Jahan Tithi, a research scientist in the Parallel Computing Labs at Intel, to discuss overcoming the scaling wall that is thwarting application efficiency, specifically within high-performance computing.

A conversation with Jesmin Jahan: Overcoming the compute versus communication scalability wall

INNOVATION LEADERS
by Bushra Anjum

Dr. Jesmin Jahan Tithi is a research scientist in the Parallel Computing Labs at Intel, where she focuses on high-performance computing (HPC) and hardware-software co-design. She obtained her Ph.D. in computer science from Stony Brook University, New York (SUNYSB) and B.Sc. with Honors in computer science and engineering from Bangladesh University of Engineering and Technology (BUET). Jesmin has wide-ranging work experience; she taught as a lecturer at BUET after her bachelor's. During her Ph.D., she interned at Intel, Google, and the Pacific Northwest National Laboratory. Throughout her career, Jesmin has been supported by numerous conference travel awards, recognitions, fellowships, grants, and scholarships, e.g., the ACM-W Scholarship, Anita Borg scholarships, many NSF grants, a graduate fellowship, Division Recognition award from Intel, and several others. Jesmin was an ACM-Student Research Competition (SRC) finalist in 2015 and the best paper finalist at the International Symposium on Performance Analysis of Systems and Software (ISPASS) 2014. Jesmin is involved in Women in HPC and STEM workshops and has coached freshmen girls at SUNYSB about HPC. Apart from her main research, she is engaged in ethics in AI research as an associated faculty at the Goethe University Frankfurt. she organized the "Ethics in AI" workshop at the 7th Heidelberg Nobel Laureate Forum. Jesmin is a member of ACM-Future of Computing Academy, formed by ACM to support and foster the next generation computing professionals. In her free time, Jesmin likes to volunteer at the local mosque and other charity foundations, watch movies, or travel with family. The responses provided below are Jesmin's personal point of view and do not represent that of any organization or institution.

What is your big concern about the future of computing to which you are dedicating yourself?

I believe one of the main challenges for the future of computing is to overcome the "scaling wall." A scaling wall prevents applications (e.g., HPC or big-data workloads) to scale efficiently beyond a limit (e.g., beyond a single compute node). For example, usually, HPC/big-data applications are often too big to be efficiently solved (or even fit) on a single compute node. These applications need high-performance networking so that many compute nodes can work together to solve problems. The latest innovations on CPUs/GPUs/FPGAs/XPUs can accelerate a scaled-down version of an HPC or big-data application with very high efficiency if run on a single compute

node. However, the performance of the real-world scale (e.g. as it appears in practice) problem that runs on multiple compute nodes gets restrained by the scaling wall due to high latency and limited network communication bandwidth.

This scaling wall has emerged due to the disparity between compute and communication speed and capacity in state-of-the-art systems. Since 2010, the average computing power per node in a supercomputing cluster has increased by 19 times. However, the overall byte-per-flop ratio—total bandwidth in (GB/s)/total flops capacity in (Gflops/s) of a machine—has decreased by six times, causing a growing gap in the interconnect bandwidth. The access cost of a core's local memory is minimal compared to the access cost to a remote cluster node's memory. Recent research suggests that drastically different technological advancement (such as short-reach silicon photonics) is needed to communicate inside and outside the chip and across the network to supply the current demand of bytes-per-flop [1, 2, 3]. With recent technological advancements, it appears that soon we will be able to break (or at least significantly reduce the gap toward) the compute versus communication scalability wall using lightspeed interconnect. If that happens, the gap in access latency to thousands of compute nodes connected via network will be very similar to the access latency to the local memory of a compute node (i.e., in the range of nanoseconds). Algorithms that were limited by communication might not be defined by communication anymore. Provided the above becomes real, it will likely free us from all the painful tricks that we use today to improve scalability across compute nodes such as reducing remote accesses/communications to its minimum, coalescing data and sending large messages whenever possible, specialized data structure, etc. I am sensing a new paradigm shift on the horizon, which might completely change the way we program and optimize our code for supercomputers.

The current trend in machine learning (ML) and artificial intelligence (AI) demands on average 10 times increase in computing power each year [4]. To keep up with the growing compute and communication demand, there is a growing focus on co-designing hardware (processor, memory, network, storage) and software (compiler, programming models, algorithms, data structures) to satisfy the latency, bandwidth, compute and energy requirements. A specialized hardware shaves of unnecessary components and networks and adds more of what is needed and can be better utilized to improve overall efficiency. At the same time, the software stacks also need to change to leverage the new hardware components traditionally unavailable in general-purpose CPUs or GPUs. For example, new algorithms are being designed to sparsify ML models, to use less storage, and some recent research has shown to deliver 10–15 times speedup by doing so. I am working on some of these efforts to co-design hardware and software to embrace the above paradigm shift on the horizon.

How did you get introduced to the issues surrounding the “scaling wall”?

The concern emerged through my Ph.D. research, and then through the daily work I do at Intel.

I learned the concepts of parallel programming during graduate school. I took two courses in grad school—Parallel Algorithms and Supercomputing—and that's where it all started. Almost all computers, even smartphones by that time had multiple CPUs (or cores). For example, my first laptop at grad school had an Intel Core i3 processor that had two cores. So, it made perfect sense to me that we need to write parallel programs to be able to use those cores efficiently and run programs fast. Through the course work, I got access to the Extreme Science and Engineering Discovery Environment (XSEDE) supercomputing cluster as well as AWS cloud servers, which were equipped with clusters of high-end servers containing many more computing cores.

In the Parallel Algorithms class, I picked up a project on parallelizing molecular dynamics simulation kernels (a program to compute molecule-molecule interaction energy). My task was to parallelize the kernel efficiently on a cluster of multicores containing hundreds of cores. By this time, I already knew the steps to parallelize an algorithm. I noticed, however, that even inside a single multicore, as soon as I used cores from different sockets to get more parallelism, my program did not get perfect scaling. And that happened due to the gap of commute and communication capability that I mentioned earlier. This time, I first encountered some impacts of the "scaling wall" and understood why this happens.

After that, I had to extend that algorithm to run on a cluster of multicores. In this case, the compute nodes do not share the memory; hence they need to communicate via message passing through the network, which takes much longer than communicating through memory (across sockets) or cache (inside a socket). Again, since computing is faster than communication, I had to make sure that I avoid communicating as much as possible to improve efficiency. At the end of the project, I submitted the result at the Supercomputing (SC) 2012 conference, and it got accepted. It was my first SC conference, and I got so much inspired by the scale of the conference and by the problems discussed there. The theme was "HPC matters," and I saw how much compute and communication it needs to solve those real-world truly large-scale problems.

Over the years, I have worked on many research projects, and it has always been a challenge to get good scaling across sockets and multiple compute nodes due to the disparity between computation and communication. Since then, the computing power of the cores has increased at least by order (greater than 10 times), but the communication has not. At Intel, I had been working on a new type of non-Von Neuman architecture, which could give an order of magnitude speedup in many of the applications that we care about. But as soon as we move to a different socket or a different compute node, the performance still gets limited by the communication speed and bandwidth. All these made me really concerned about the scaling wall.

What project are you currently leading that address some of the emerging challenges due to the disparity between computation and communication?

Over the last two years, I have been working on the DARPA HIVE project [1], which aims to co-develop hardware and software for graph problems to gain around 1,000 times performance boost over the traditional cluster of cores. Under HIVE, we have designed a system called Programmable Unified Memory Architecture [1] that aims to close the gap between computing and communication using innovative technology and software-hardware co-designing, in general.

I have also been focusing on machine learning algorithms. The current trend in deep learning demands a 10 times increase in compute [4], use of bigger models and larger datasets. This necessitates the use of both models- and data- parallelism and, as a consequence, also demands highly efficient communication inside the chip and across the network, higher bandwidth, lower latency, and lower energy cost at the same time. I am working to accomplish that goal by designing new algorithms and by catalyzing the hardware innovations at the same time.

Aside from my main projects, I am also very concerned about the unethical and irresponsible usage of AI. I truly believe that countries, companies, governments, and non-government organizations alike, together with individual users, researchers, and developers of AI systems, should be educated and work together to achieve ethical, fair, just, and responsible usage of AI. For these reasons, I am focusing on tools and methods to systematically analyze the ethical implications of AI software. I am collaborating with a research team led by Dr. Roberto V. Zicari on [Z-inspection](#), a methodology to assess ethical AI.

If you are interested to know more or contribute in any of these works, please contact me at shraban03@gmail.com or get connected to me at LinkedIn at <https://www.linkedin.com/in/jesmin-jahan-tithi>.

References

- [1] Schor, D. [DARPA ERI: HIVE and Intel PUMA graph processor](#). WikiChip Fuse. Aug. 4, 2019.
- [2] Feldman, M. [On-chip optical links are one step closer to reality](#). *The Next Platform*. Sept. 11, 2019.
- [3] Tithi, J. J. et al. High-performance energy-efficient recursive dynamic programming with matrix-multiplication-like flexible kernels. *International Parallel and Distributed Processing Symposium*. IEEE, 2015.
- [4] Amodei, D. and Hernandez, D. [AI and compute](#). OpenAI. Blog. May 16, 2018.

Biography

Bushra Anjum is a software technical lead at Amazon in San Luis Obispo, CA. She has expertise in Agile Software Development for large scale distributed services with special emphasis on scalability and fault tolerance. Originally a Fulbright scholar from Pakistan, Dr. Anjum has international teaching and mentoring experience and has served in academia for over five years before joining the industry. In 2016, she has been selected as an inaugural member of the [ACM Future of Computing Academy](#), a new initiative created by ACM to support and foster the next generation of computing professionals. Dr. Anjum is a keen enthusiast of promoting diversity in the STEM fields and is a mentor and a regular speaker for such. She received her Ph.D. in computer science at the North Carolina State University (NCSU) in 2012 for her doctoral thesis on Bandwidth Allocation under End-to-End Percentile Delay Bounds. She can be found on Twitter [@DrBushraAnjum](#).

DOI: 10.1145/3388708