Research and Teaching Summary

Mithuna S. Thottethodi

June 18, 2018

1 Research Statement

Research Trajectory: In the early 2000s, the 'power wall' problem that limited clock scaling of uniprocessors drove the world to the multicore era. In that period, the main focus of my research was broadly in computer architecture including multicore memory hierarchies, on-chip interconnection networks. In addition, we worked on novel architectures for programmable microfluidics in an inter-disciplinary project that included aspects of mechanical engineering and chemistry.

The other major change starting from the mid 2000s was the shift towards cloud computing with warehouse-scale/datacenter-scale computing at the heart of it. My research focus over that period has accordingly shifted to datacenter scale systems. Below, I offer a brief summary of three ongoing projects in various stages of progress.

1.1 Application-Specific Placement and Routing for Communication optimization

While abstraction is a powerful tool to manage complexity, the strict layering of abstraction layers often hides opportunities for cross-layer optimizations. Specifically in the context of networks, application-level knowledge of stable communication patterns is not visible at the lower abstraction layers when placing communicating threads/processes on processors and when routing communication between such threads/processes. My research led to the development of systematic application-specific placement [3] and routing improvements [1, 2]) by leveraging such cross-layer knowledge.

1.2 SmartEdge: End-to-end Mobile Web Acceleration from Datacenters to the Edge

This research project focuses on the grand challenge of achieving low end-to-end latency (under 100 ms) for web applications. Sub-100-ms latencies makes web applications feel as responsive as local applications that run on desktops/mobiles, which has enormous implications for enhanced user experiences, faster adoption of cloud computing, and for e-commerce revenues. Achieving such low latency is especially challenging because (a) todays web download process is a poor fit to high-latency cellular networks, and (b) the trend in web applications toward more personalized, dynamic content which makes it difficult to cache mutable web content close to users; the storage is typically held in distant centralized data-centers (DCs). Much of the previous work in caching dynamic content offers only weak consistency guarantees (typically, eventual consistency), but many interactive scenarios require the stronger causal model.

We employ a comprehensive approach called SmartEdge which moves both the application functionality and the back-end storage to the network edge. To address the latency gap between cell devices and desktops, this research employs redundant execution on the cloud and proactively pushing data to the client. SmartEdge leverages the fact that redundant execution may be approximate in the context of web applications to drastically reduce the overheads of redundant computation, which enables the edge to scale to hundreds of thousands of clients [5]. To address the challenge of scalable, consistent backend storage, SmartEdge (a) only partially replicates data in a subset of DCs (unlike existing techniques which require full replication of all data in all the DCs) [4], and (b) supports large-scale and hierarchical edge caching (unlike existing geo-replicated storage systems that are limited to a small number of DCs). These techniques not only enable latency reduction by placing data closer to users but also achieve an order of magnitude lower replication cost. Our work from this project has been published in MobiCom 2017 [5] and IEEE TCC [4](accepted, to appear).

1.3 Ongoing and Future Plans

Online Big Data (OLBD) applications which are critical workloads in datacenter computing demand extreme low latency in datacenter networks. Remote Direct Memory Access (RDMA), which is a promising alternative to tra-

ditional TCP, significantly reduces datacenter network latencies by about an order-of-magnitude. However, RDMA adoption poses two major challenges as RDMA suffers from performance fragility under congestion, and RDMA incurs either wasted memory or significant programmer burden for typical OLBD traffic. This project develops two novel networking technologies – Dart and RIMA – which enable scalable datacenter networks that achieve the low latency benefits of RDMA while avoiding its drawbacks (performance fragility, programmer burden, and wasted memory).

Dart addresses performance fragility by decoupling edge-congestion and in-network congestion and specializing solutions for each case separately. Dart handles edge-congestion using receiver-directed congestion control (RDCC) unlike prior approaches where senders have to infer sending rates indirectly from round-trip-times and/or dropped packets. RDCC enables accurate and fast (within-one-round-trip-time) convergence, which leads to lower latency and higher throughput. Dart handles transient in-network congestion by using a flow-preserving deflection technique that deflects all packets of selected short-flows along longer yet less-congested paths.

Remote Indirect Memory Access (RIMA) addresses the second challenge by enabling reactive, on-demand memory allocation as opposed to RDMAs proactive memory allocation for the worst case, which minimizes the memory footprint without programmer effort. Together, Dart and RIMA enable extreme low datacenter network latency for OLBD applications. (Work from this project is currently under review.)

2 Teaching Statement

Teaching and learning is a delicate process where the student-driven choice of courses to match their interests must be balanced with faculty-driven curriculum construction to ensure a complete program of study. Further, at the individual course-level, the content must be faculty-driven so they can bring their expertise to bear. Mismatches in balancing these concerns can potentially result in expert educators trying to teach disinterested students (from a faculty point-of-view) and bright students being forced to sit in classrooms they have no interest in (from a student point-of-view).

My teaching strategy flows from my belief these challenges can be solved by appropriate strategies at both the curricular level and at the level of individual courses. At the curricular level, I believe that institutions should strive to accommodate a breadth of choices to empower students' varied interests. On this front, the challenge in Purdue ECE is that there are limited options offered for the senior capstone design project. Effectively, all options are variants of microcontroller-based design projects. To address this problem, I have initiated alternative software-based design options in a pilot offering. The initial team developed a system that leverages modern deep learning techniques to automatically track classroom attendance via facial recognition (with an in-class camera setup). More importantly, when scaled up, this software-based capstone design course offering will accommodate the preferences of many ECE students whose interests may not lie in embedded systems.

Once students have adequate choices at the curricular levels, the goal must be to deliver a careful combination of foundational principles as well as timely and topical course content. While foundational principles are often long-lasting and (Self-evidently) important to teach, continuously upgrading course material to include topical content is also important; and has the added advantage of industry relevance. With this philosophy in mind, I have continuously modified the Parallel Computer Architecture graduate course to include topics on warehouse/datacenter computing, programming models for datacenters (e.g., mapreduce) and GPU-based computing.

Beyond the technical and educational aspects, there is a significant inter-personal dynamic that has a huge impact on teaching. Students are inherently more receptive if they believe that the teacher's self-identified primary goal is to help the student as an ally. I am pleased to have received multiple teaching awards in recognition of my efforts from the department (2017 Motorola Excellence in Teaching Award) as well as from the student honor society (Outstanding Professor, Fall 2013 and Fall 2014, Eta Kappa Nu, Beta Chapter).

References

[1] Ahmed H. Abdel-Gawad and Mithuna Thottethodi. Transcom: transforming stream communication for load balance and efficiency in networks-on-chip. In Carlo Galuzzi, Luigi Carro, Andreas Moshovos, and Milos Prvulovic, editors, *44rd Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2011, Porto Alegre, Brazil, December 3-7, 2011*, pages 237–247. ACM, 2011.

[2] Ahmed H. Abdel-Gawad and Mithuna Thottethodi. Scalable, global, optimal-bandwidth, application-specific routing. In *24th IEEE Annual Symposium on High-Performance Interconnects, HOTI 2016, Santa Clara, CA, USA, August 24-26, 2016*, pages 9–18. IEEE Computer Society, 2016.

[3] Ahmed H. Abdel-Gawad, Mithuna Thottethodi, and Abhinav Bhatele. RAHTM: routing algorithm aware hierarchical task mapping. In Trish Damkroger and Jack J. Dongarra, editors, *International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2014, New Orleans, LA, USA, November 16-21, 2014*, pages 325–335. IEEE Computer Society, 2014.

[4] Tariq Mahmood, Shankaranarayanan Puzhavakath Narayanan, Sanjay Rao, T. N. Vijaykumar, and Mithuna Thottethodi. Karma: Cost-effective geo-replicated cloud storage with dynamic enforcement of causal consistency. *IEEE Transactions on Cloud Computing*. (Accepted March 2018, to appear.).

[5] Ashiwan Sivakumar, Chuan Jiang, Yun Seong Nam, Shankaranarayanan Puzhavakath Narayanan, Vijay Gopalakrishnan, Sanjay G. Rao, Subhabrata Sen, Mithuna Thottethodi, and T. N. Vijaykumar. Nutshell: Scalable whittled proxy execution for low-latency web over cellular networks. In Kobus van der Merwe, Ben Greenstein, and Kannan Srinivasan, editors, *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking, MobiCom 2017, Snowbird, UT, USA, October 16 - 20, 2017*, pages 448–461. ACM, 2017.