Research Statement

Ayush Mishra

(+41) 0782515146
aymishra@ethz.ch
ayushmishra.net

In many ways, computer networks help realize some of computing's best ideas - from the Internet itself to the large-scale datacenters that now make distributed deep learning possible. Systems perform better when the networks that support them evolve with the workloads and applications they run.

In this spirit, my research focuses on using mathematical models and empirical measurements to uncover key insights to guide the design of performant networks. I enjoy using these insights to question long-held assumptions and to inspire the design of novel solutions for real practical problems. My research so far has primarily explored the Internet transport and congestion control design space. It has progressed the state-of-the-art in understanding how these algorithms interact and co-exist with each other on the Internet.

My research has appeared as first-author publications in top networking conferences, including SIGCOMM, NSDI, IMC(3x), and SIGMETRICS. My PhD thesis was also awarded the IMDA Excellence in Computing Award (Best CS PhD thesis in the university) and was the runner-up for the SIGMETRICS Doctoral Dissertation Award. My research has also had practical impact; we have helped Meta make the congestion control algorithms in their transport stacks fairer, as well as caught the deployment of undocumented congestion control algorithms on the Internet by Google and Akamai, which has been confirmed by both of them.

Networking, by design, plays a supporting role in computer science research. In this aspect, I believe networking is highly inter-disciplinary. I therefore intend to work closely with systems researchers, control theorists, as well as engineers and developers from the Industry. Over the years, my research has greatly benefited from being discussed at more industry-adjacent venues like the IETF. I therefore also intend to work more with companies that actually deploy the networks researchers love to design and solve *real* and *practical* problems.

Previous Research

While traditional Congestion Control Algorithms (CCAs) were designed with the simple goal of utilizing bandwidth and preventing a congestion collapse, most CCAs that run on the Internet today are more nuanced. Application developers often use CCAs to improve performance as well as ensure robust operation across the networks in which they deploy. Over the years, the increased diversity of applications that run on the Internet, as well as the evolving nature of the Internet itself, has produced numerous congestion control algorithms. Moreover, congestion control research has seen a rennaissance in recent years, with several model-based and learning-based CCAs being proposed and deployed on the Internet. Network operators today can choose from a smorgasbord of CCAs to best meet their needs.

My PhD thesis studies the *extent* and *impact* of the heterogeneity in the modern Internet's congestion control landscape. In it, I proposed novel measurement techniques to identify CCAs deployed by popular websites on the Internet. I also devised mathematical models with a game theoretic formulation to understand how CCA adoption can be driven by their performance. Finally, I developed new metrics to measure how compliant re-implementations of standard CCAs are in newer transport stacks like QUIC.

One of the key drivers of CCA heterogeneity on the Internet today is BBR, a new congestion control algorithm proposed and deployed by Google in 2016. According to them, early deployments of BBR were boasting superior throughput and more consistent delays compared to traditionally popular CCAs like CUBIC. Motivated by this, we developed a CCA identification tool and used it to measure the 20,000 most popular websites on the Internet in 2019 [1]. Our measurements showed that just three short years since it was proposed, BBR was already deployed at 22% of the measured websites. Our more recent measurement study in 2024 [2], which extended its coverage by measuring websites over modern transport stacks like QUIC and realistic workloads like video streaming, confirmed that BBR's sizable share in Internet traffic still persists. Moreover, these measurements were also able to uncover the deployment of undocumented CCAs by Akamai and the early testing of BBRv3 by Google.

Most early adopters like Google, Spotify, and Dropbox, reported switching to BBR because of its throughput benefits. Indeed, during both our measurement studies, we saw BBR being particularly popular with websites that stream video and other throughput-hungry applications. This motivated us to ask a natural question - Could BBR's throughput benefits cause it to completely replace loss-based CCAs like CUBIC on the Internet? In an attempt to answer this question, I developed a mathematical model to predict how CUBIC and BBR interact with each other while sharing a common bottleneck link. I then analyzed this model from a Game Theoretic point of view to understand how BBR's throughput gains evolve as more flows in a network switch to it. Our model predicted that with increased adoption, BBR's throughput gains over CUBIC diminish exponentially - pushing the network towards a Nash Equilibrium with both CUBIC and BBR flows [3, 4]. Therefore, if throughput is a key consideration, it is likely BBR will never completely replace CUBIC and that the Internet will reach a stable mixed distribution of CUBIC and BBR flows. Indeed, our measurements between 2019 and 2024 have seen BBR's adoption on the Internet stagnate.

Besides BBR, in the recent years the Internet has also seen the adoption of QUIC - a user space transport stack built over UDP and set to be the default for HTTP3. While most QUIC stacks re-implement standard congestion control algorithms like CUBIC and BBR, these user-space implementations are prone to modifications by the developer. In order to capture such modifications, we bench-marked the CCA implementations in popular open source QUIC stacks deployed on the Internet [5, 6]. I developed new metrics to not only identify CCA implementations in QUIC that differed significantly from their TCP counterparts, but also proposed fixes to make them more conformant. We uncovered several non-conformant CCA implementations in QUIC that risk introducing unfairness and unpredictability on the Internet. We have since been contacted by Meta to patch their unfair QUIC implementation of BBR. In our paper, we provide fixes for non-conformant implementations in other QUIC stacks as well, including those deployed by Google and Cloudflare.

Current Research

The 5-year longitudinal study of the Internet's congestion control landscape presented in my thesis reveals two key things about the Internet - 1) The Internet's congestion control landscape is (and will likely remain) heterogeneous, and 2) This heterogeneity can introduce significant unfairness on the Internet. During my postdoc, I am working on building solutions that can directly address the challenges a heterogeneous mix of CCAs presents for the Internet.

The first of these works introduces *Santa* [7], a new multi-queue AQM for the Internet to improve inter-flow fairness. It is built on the principles of *Performance Isolation*, a new paradigm for fairness on the Internet inspired by a simple observation - people deploy different CCAs because they want different throughput-delay tradeoffs. In our upcoming NSDI paper, we argue that in an ideal world, these flows should be able to meet their throughput-delay tradeoffs regardless of who they compete with. While this can be achieved via Fair Queueing, it is impractical to give each flow its own queue at Internet scale. More practical *approximate* fair queuing approaches exist and are able to improve bandwidth fairness, but still make it possible to flows to inflict delays and losses on each other. *Santa* reaches a compromise by grouping flows with similar desired throughput-delay tradeoffs in the same queue. This way, it is able to achieve *approximate* performance isolation with just a handful of queues. We implemented *Santa* on a programmable switch to demonstrate that it is practical and scalable enough to provide approximate performance isolation for thousands of flows.

The Internet's increasing CCA heterogeneity also demands the re-examination of some fundamental questions about Internet congestion control. These concern 1) the deployability of new CCAs 2) understanding the safety and convergence properties of a system with an arbitrary mix of CCAs, and 3) building CCAs that are robust under contention from a variety of cross-traffic. This is particularly challenging because of the contrasting nature of the deployed CCAs on the Internet - from loss-based AIMD algorithms to more modern model-based CCAs. My current research makes a case simplifying this problem by viewing the current Internet CCA design space under a single framework. I am working on quantifying the *aggression* of a congestion control algorithm, and how it can be used to determine its deployability on the Internet. Furthermore, I am working on a simple mathematical model that can utilize this aggression framework to characterize an arbitrary mix of CCAs in a network. I also plan to demonstrate that *aggression* is an essential building block in designing tuneable CCAs that can adapt depending on the cross traffic they compete with.

Besides congestion control, I am also actively working on projects that explore transport and load balancing schemes for large-scale datacenters running ML workloads, improving the testing and development of ML-based network algorithms [8], and understanding the impact of Host Congestion in GPU clusters. I am also actively collaborating on a few Internet-scale measurement studies to better understand how flows compete with each other on the Internet, as well as inferring Internet usage patterns using publicly available IXP traffic volume data [9].

Future Directions

I plan to drive my research agenda based on emergent properties in networks we already run, as well as designing performant networks for new applications and workloads. The Internet's current CCA landscape motivates reviewing our ideas of stability, fairness, and deployability on the Internet. Distributed ML-jobs are the next emerging workload that networks need to optimize for. I am also interested in revisiting the applicability of ML-based techniques for improving networking. Over the next 5 years, I plan to pursue the following research directions:

Measuring the Internet. Network algorithms for the Internet are always trying to optimize for a moving target. Papers in this space often argue setting certain network parameters in their experiments while relying on outdated (and likely now inaccurate) measurement studies. This introduces a significant amount of ambiguity in how we run and reproduce experiments in networking, especially in the Internet transport space. Having had much of my research inspired by measurement studies, I plan to introduce a large-scale and *consistent* measurement initiative to keeping an eye on how the Internet is evolving. This could range from measuring simple metrics like typical RTTs and per-flow bandwidths to more sophisticated measurement studies that try to monitor CCA deployment and application-level behavior.

Internet Congestion Control. The CCA heterogeneity on the modern Internet *resets* some of the things we thought we had figured out about congestion control. Perhaps the most consequential of these research questions underpin the fairness and deployability standards for the Internet. The old TCP friendliness paradigm does not apply to the current ecosystem of Internet congestion control algorithms. I am interested in concretely defining these modern deployability standards and finding ways to effectively enforce them. I expect to form these deployability standards around a CCA's measured *aggression* - which is the main thrust of my current research. In the meantime, we also need to develop ways to protect flows from CCAs that violate these deployability guidelines. *Santa* could be a part of the solution here (be isolating overly aggressive flow in their own low priority queue). In the past, we have also shown that we can easily limit the aggression of TCP flows by modifying their receive window field in the TCP header [10].

Building performant ML clusters. Large-scale distributed machine learning jobs are an emerging workload in datacenters that challenge several existing assumptions in network design. For example, most packet switched networks are built on the idea of statistical multiplexing - a demand-based multiplexing strategy that assumes that no single user is capable of fully utilizing the entire link. This is not true in RDMA networks supporting large machine learning jobs, where GPUs are more than capable of utilizing the entire link capacity for several seconds. Trends like these motivate re-examining alternate reservation and circuit-based network designs. Modern ML clusters are also becoming massive in scale, and present new challenges in maintainable topologies, improving power efficiency, and working around the physical constraints of building such massive networks. I believe these characteristics provide opportunities in designing novel modular topologies, and fault-resilient load balancing and transport algorithms. ML-traffic is also produced as a result of reasonably predictable patterns or *communication collectives*. I believe there is an opportunity here to measure, model, and characterize these communication patterns so that we can leverage their predictability in designing more performant network algorithms. I expect load balancing, traffic engineering, job scheduling, and transport to greatly benefit from a better understanding of how ML-traffic behaves.

ML for Networking. I am also interested in revisiting how we apply ML-based techniques in network algorithms. I plan to explore hybrid approaches in congestion control, traffic management, and traffic classification. While there are many proposals for learning-based approaches in these fields, I feel they fall short because they fail to leverage classic inference techniques that already work well. I am interested in applying ML-based techniques not to replace classical approaches, but to supplement their inference mechanisms and make them more robust.

References

[1] Ayush Mishra et al. "The Great Internet TCP Congestion Control Census". In: *Proceedings of SIGMETRICS*. 2019.

[2] Ayush Mishra et al. "Keeping an Eye on Congestion Control in the Wild with Nebby". In: *Proceedings of SIGCOMM*. 2024.

[3] Ayush Mishra et al. "Conjecture: Existence of Nash Equilibria in Modern Internet Congestion Control". In: *Proceedings of APNet*. 2021.

[4] Ayush Mishra, Wee Han Tiu, and Ben Leong. "Are we heading towards a BBR-dominant Internet?" In: *Proceedings of IMC*. 2022.

[5] Ayush Mishra, Sherman Lim, and Ben Leong. "Understanding Speciation in QUIC Congestion Control". In: *Proceedings of IMC*. 2022.

[6] Ayush Mishra and Ben Leong. "Containing the Cambrian Explosion in QUIC Congestion Control". In: *Proceedings of IMC*. 2023.

[7] Ayush Mishra et al. "Managing Congestion Control Heterogeneity on the Internet with *Approximate Performance Isolation*". In: *Proceedings of NSDI*. 2026.

[8] Benjamin Hoffman et al. "Into the Wild: Real-world Testing for ML-based ABR". In: *Proceedings of PACMI*. 2025.

[9] Ege Cem Kirci, Ayush Mishra, and Laurent Vanbever. "Five Blind Men and the Internet: Towards an Understanding of Internet Traffic". In: *ArXiv*. 2025.

[10] Ayush Mishra et al. "POSTER: Expanding the Design Space for in-Network Congestion Control on the Internet". In: *Proceedings of ACM SIGCOMM: Posters and Demos*. 2024.