# Research Progress and Vision

*Balakrishnan Chandrasekaran (b.chandrasekaran@vu.nl)*

The Internet is an extremely complex, engineered, and distributed system. Tracing even simple exchanges over the Internet such as fetching of a web page from a browser is a non-trivial exercise: The mental model it conjures up in this case comprising a browser requesting the page, a server responding, and the interactions facilitated by a set of routers and switches connecting the two is far too simplistic. Numerous other entities (e.g., firewalls screening the request, caches attempting to reduce the response time, and clusters of servers sharing the load from similar requests) play a vital role in the interaction. As this example demonstrates, determining the contribution of the different entities (e.g., browsers, firewalls, caches, and servers) towards the overall performance of applications (e.g., the time it takes to load the page on the browser) is extremely hard. It is, nevertheless, *the* crucial first step in designing better systems. To this end, I have pursued an empirical, data-driven approach for analyzing the design [3], deployment [8, 10], and operations of networked systems [16].[1] My research has subsequently exploited the insights derived from these empirical analyses to improve the performance [3, 16] and security [13] of such systems. Below, I describe my current research thrust, discuss preliminary findings, and briefly hint at the long-term roadmap.

**The Foundations for Modern Transport Protocols**

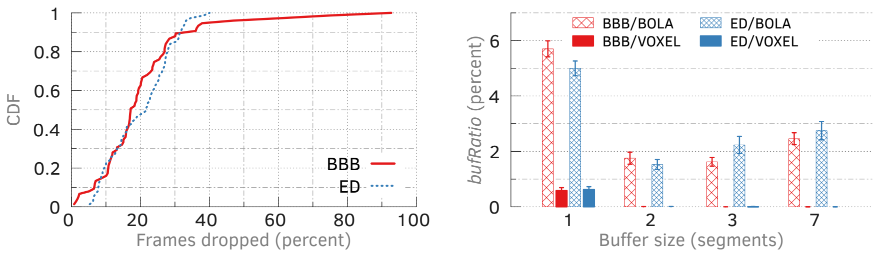
The vast majority of our day-to-day activities today depend on the Internet. We use a variety of applications—for work, education, health, entertainment, and participating in civil, political, and social life—that rely on network connectivity. Many researchers and practitioners, thus, consider the Internet and its underlying network infrastructure as "critical" infrastructure. Hence, *efficiently*, *equitably*, and *safely* using this shared infrastructure is of utmost importance today. We fail, nevertheless, on all these fronts today, because the transport protocols or algorithms (e.g., TCP, UDP, and QUIC) responsible for moving data over the network are largely *deficient* by design. These deficiencies are deeply rooted in the design, which strictly adheres to the architectural guidelines, represented as layers of abstractions, of the TCP/IP model designed for the 1970s Internet. My current efforts focus on designing a safe, systematic, and principled approach to address two fundamental architectural limitations in transport-protocol designs.

***1. Transport protocols are largely unaware of the performance "needs" (or requirements) of applications.***

Different applications have, unsurprisingly, different performance (e.g., latency, bandwidth, reliability, privacy, and security) requirements. Online games, for instance, typically require low bandwidth (e.g., for exchanging player positions or action updates) as well as low latency (e.g., for supporting fast-moving actions). Watching a video stream (e.g., Netflix or YouTube), in contrast, requires high bandwidth (e.g., for streaming high-resolution videos), but not low latency. These performance requirements may even vary over time for a given application. There simply exists, however, no standard mechanism for applications to share such key metadata (on their time-varying performance requirements) with the transport protocols.

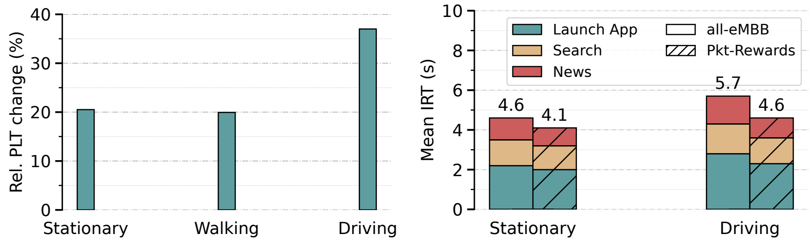
***Preliminary findings.*** The inadequacy of transport protocols manifests clearly for widely used applications such as video streaming [4, 12]. Streaming solutions are well-known to overwhelm the network, especially under challenging conditions: They do not, hence, equitably share bandwidth with other applications. Consequently, when most of us shifted to working remotely (i.e., over the Internet) during the pandemic in 2020, EU's regulatory authority allowed network operators to throttle video traffic, i.e., legitimize an unfair traffic discrimination practice, to make room for essential work-related traffic [6]. In our preliminary study, we simply asked whether transport protocols can alleviate the load on the network while still delivering high-quality video streams, *if* they were aware of the (performance) "needs" of the video stream.

[1] Citing only publications since November 2020, with my VU Amsterdam affiliation.




*(a)* *(b)*

***Fig. 1:*** *(a) In 80% of all segments in two widely used videos, Big Buck Bunny (BBB) and Elephant's Dream (ED), we can drop at least 10% of frames while still guaranteeing excellent video quality (i.e., an SSIM score of 0.99). (b) Our pilot project, VOXEL, outperforms the state-of-the-art BOLA when streaming videos over Verizon LTE network; time spent in rebuffering instead of playing is substantially smaller in VOXEL than BOLA, even when video players buffer only one extra segment prior to playback.*




***(a)*** ***(b)***

***Fig. 2:*** *Instead of using only the eMBB channel for applications, if we systematically map the data packets to eMBB and URLLC, we can reap substantial improvements in performance: (a) reductions in page-load times (PLTs) for web browsing, measured across 200 different web sites, and (b) Android mobile application interaction response times (IRTs) for 3 types of applications.*

A stream typically comprises fixed-length *segments* or *chunks*, typically lasting between 2 and 10 seconds. The segments are further composed of *frames*. The loss (e.g., under poor network conditions) of some of these frames have less implications for the quality of the video, perceived by end users, than that of others [7]. We observed that typically a significant fraction of a segment can be dropped while still guaranteeing an "impeccable" video quality—pristine quality with imperceptible impairments per widely used objective metrics such as SSIM [15]. Fig. 1a shows the frame-drop tolerance of two widely used videos in the literature—Big Buck Bunny (BBB) and Elephant's Dream(ED)—though the inferences hold across a diverse range of videos [11]. *VOXEL*, our pilot project, exploits this frame-importance data to reduce *rebuffering* issues (i.e., "stalls") at the viewer's end [11] under challenging network conditions. VOXEL outperforms the state-of-the-art BOLA [14] by avoiding transmissions of "unimportant" frame data even under challenging network conditions (Fig. 1b): The ratio of time spent in rebuffering to playing is negligible in VOXEL, even when video players buffer only one segment prior to playback. That VOXEL can deliver less data while still presenting a high-quality video stream to users has many implications for content providers and content delivery networks.

***2. The "nature" (i.e., performance characteristics) of the underlying network is largely opaque to transport protocols.***

Different communication channels may have vastly different performance characteristics. Losses in WiFi networks and microwave links, for instance, correlate poorly with congestion [2, 3, 5]. End-to-end delays vary over time in satellite networks, irrespective of network utilization [9]. Understanding these characteristics is crucial for maximizing application performance. End hosts today are, morever, likely to have *concurrent* access to more than one type of network. Such rich connectivity options makes a compelling case for transport protocols to optimize the mapping between applications' performance requirements and the capabilities of the underlying network. There exists, however, no standard mechanism for specifying the performance characteristics of network channels (or how they differ from one another) to the transport protocols. Besides, the transport protocols do not "know" the applications' performance requirements to exploit the underlying network in the most efficient manner.

***Preliminary findings.*** The fifth generation technology standard for broadband cellular networks (5G) offers two contrasting communication channels: Enhanced Mobile Broadband (eMBB) and Ultra Reliable Low Latency Communications (URLLC) [1]. The former provides high-bandwidth and high-latency, while the latter offers very low-bandwidth (i.e., $0.4 - 16$ Mbps) and ultra low-latency (i.e., 1 ms round-trip between the mobile device and radio access network) [1]. Our preliminary work, *DChannel* shows that carefully mapping different data packets (or IP datagrams) of an HTTP session across these two channels improves page-load times by at least 20% compared to that when using only the high-bandwidth eMBB channel (Fig. 2a). We can similarly improve mobile application response times (refer Fig. 2b), thereby improving end-user's quality of experience. These results hold both in stationary and mobility scenarios. We demonstrated similar performance improvements over microwave networks [3]. These findings strongly attest to the need for transport protocols to "learn" applications' needs and accurately map them to the capabilities of the underlying communication channels.

**Summary & Outlook.** Addressing these two transport-protocol limitations without overlooking the security implications of our solutions constitutes the major part of my research for the long term. To this end, I envision redesigning the network abstractions from first principles and addressing several key scientific challenges: (a) How can we systematically unearth the time-varying performance (e.g., latency, bandwidth, reliability, privacy, and security) requirements of different applications? Should application developers declare such needs, or could we automatically *learn* them? (b) How can the transport layer learn the characteristics or capabilities of the underlying network? Should we declare these capabilities or automatically infer them over time? (c) Where and how can we implement the mapping between applications' performance requirements and underlying communication channels' performance characteristics? (d) How can we change the design of congestion control algorithms, used by the transport protocols, by exploiting insights into applications' performance requirements and leveraging the characteristics of the underlying network channels.

## References


[1] 3GPP. 5G new radio (TR 21.915; Release 15). https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3389, September 2019.

[2] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. Katz. A comparison of mechanisms for improving TCP performance over wireless links. *IEEE/ACM Transactions on Networking*, 5(6), 1997.

[3] D. Bhattacherjee, W. Aqeel, S. A. Jyothi, I. N. Bozkurt, W. Sentosa, M. Tirmazi, A. Aguirre, B. Chandrasekaran, P. B. Godfrey, G. Laughlin, B. Maggs, and A. Singla. cISP: A Speed-of-Light Internet Service Provider. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI '22)*, Apr. 2022.

[4] Cisco. Cisco annual internet report (2018–2023). https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf, March 2020.

[5] A. DeSimone, M. C. Chuah, and O.-C. Yue. Throughput performance of transport-layer protocols over wireless LANs. In *Proceedings of GLOBECOM '93. IEEE Global Telecommunications Conference*, volume 1, 1993.

[6] European Commission - BEREC. Joint Statement from the Commission and the Body of European Regulators for Electronic Communications (BEREC) on coping with the increased demand for network connectivity due to the Covid-19 pandemic. https://www.berec.europa.eu/en/document-categories/berec/others/joint-statement-from-the-commission-and-the-body-of-european-regulators-for-electronic-communications-berec-on-coping-with-the-increased-demand-for-network-connectivity-due-to-the-covid-19-pandemic, March 2020.

[7] N. Feamster and H. Balakrishnan. Packet Loss Recovery for Streaming Video. In *12th International Packet Video Workshop*, April 2002.

[8] J. Fechner, B. Chandrasekaran, and M. X. Makkes. Calibrating the Performance and Security of Blockchains via Information Propagation Delays. In *37th ACM/SIGAPP Symposium on Applied Computing (SAC)*, 2022.

[9] S. Kassing, D. Bhattacherjee, A. B. Águas, J. E. Saethre, and A. Singla. Exploring the "Internet from Space" with Hypatia. In *Proceedings of the ACM Internet Measurement Conference*, IMC '20, 2020.

[10] J. Messias, M. Alzayat, B. Chandrasekaran, K. P. Gummadi, P. Loiseau, and A. Mislove. Selfish & Opaque Transaction Ordering in the Bitcoin Blockchain: The Case for Chain Neutrality. In *Proceedings of the ACM Internet Measurement Conference*, IMC '21, 2021.

[11] M. Palmer, M. Appel, K. Spiteri, B. Chandrasekaran, A. Feldmann, and R. K. Sitaraman. VOXEL: Cross-Layer Optimization for Video Streaming with Imperfect Transmission. In *Proceedings of the 17th International Conference on Emerging Networking EXperiments and Technologies*, CoNEXT '21, 2021.

[12] Sandvine. The Mobile Internet Phenomena Report. https://www.sandvine.com/hubfs/Sandvine_Redesign_2019/Downloads/2021/Phenomena/MIPR%20Q1%202021%2020210510.pdf, May 2021.

[13] E. Shi, W. Aqeel, B. Chandrasekaran, and B. Maggs. Puncturable Pseudorandom Sets and Private Information Retrieval with Near-Optimal Online Bandwidth and Time. In *Advances in Cryptology – CRYPTO 2021*, August 2021.

[14] K. Spiteri, R. Urgaonkar, and R. K. Sitaraman. BOLA: Near-optimal bitrate adaptation for online videos. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, April 2016.

[15] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4), Apr. 2004.

[16] X. Zhang, T. Sen, Z. Zhang, T. April, B. Chandrasekaran, D. Choffnes, B. M. Maggs, H. Shen, R. K. Sitaraman, and X. Yang. AnyOpt: Predicting and Optimizing IP Anycast Performance. In *Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '21, August 2021.