

Reducing delays in wireless networks

April 10 2020, by Rob Matheson



To reduce lag times and increase quality in video streaming, mobile gaming, and other web services, researchers at MIT’s Computer Science and Artificial Intelligence Laboratory have designed a congestion-control scheme for time-varying wireless links, such as cellular networks. Credit: José-Luis Olivares, MIT

MIT researchers have designed a congestion-control scheme for wireless networks that could help reduce lag times and increase quality in video

streaming, video chat, mobile gaming, and other web services.

To keep web services running smoothly, congestion-control schemes infer information about a network's bandwidth capacity and congestion based on feedback from the network routers, which is encoded in data packets. That information determines how fast data packets are sent through the network.

Deciding a good sending rate can be a tough balancing act. Senders don't want to be overly conservative: If a network's capacity constantly varies from, say, two megabytes per second to 500 kilobytes per second, the sender could always send traffic at the lowest rate. But then your Netflix video, for example, will be unnecessarily low-quality. On the other hand, if the sender constantly maintains a high rate, even when network capacity dips, it could overwhelm the network, creating a massive queue of data packets waiting to be delivered. Queued packets can increase the network's delay, causing, say, your Skype call to freeze.

Things get even more complicated in [wireless networks](#), which have "time-varying links," with rapid, unpredictable capacity shifts. Depending on various factors, such as the number of network users, cell tower locations, and even surrounding buildings, capacities can double or drop to zero within fractions of a second. In a paper at the USENIX Symposium on Networked Systems Design and Implementation, the researchers presented "Accel-Brake Control" (ABC), a simple scheme that achieves about 50 percent higher throughput, and about half the network delays, on time-varying links.

The scheme relies on a novel algorithm that enables the routers to explicitly communicate how many data packets should flow through a network to avoid congestion but fully utilize the network. It provides that detailed information from bottlenecks—such as packets queued between cell towers and senders—by repurposing a single bit already available in

internet packets. The researchers are already in talks with mobile network operators to test the scheme.

"In cellular networks, your fraction of data capacity changes rapidly, causing lags in your service. Traditional schemes are too slow to adapt to those shifts," says first author Prateesh Goyal, a graduate student in CSAIL. "ABC provides detailed feedback about those shifts, whether it's gone up or down, using a single data bit."

Joining Goyal on the paper are Anup Agarwal, now a graduate student at Carnegie Mellon University; Ravi Netravali, now an assistant professor of computer science at the University of California at Los Angeles; Mohammad Alizadeh, an associate professor in MIT's Department of Electrical Engineering (EECS) and CSAIL; and Hari Balakrishnan, the Fujitsu Professor in EECS. The authors have all been members of the Networks and Mobile Systems group at CSAIL.

Achieving explicit control

Traditional congestion-control schemes rely on either packet losses or information from a single "congestion" bit in internet packets to infer congestion and slow down. A router, such as a base station, will mark the bit to alert a sender—say, a video server—that its sent data packets are in a long queue, signaling congestion. In response, the sender will then reduce its rate by sending fewer packets. The sender also reduces its rate if it detects a pattern of packets being dropped before reaching the receiver.

In attempts to provide greater information about bottlenecked links on a network path, researchers have proposed "explicit" schemes that include multiple bits in packets that specify current rates. But this approach would mean completely changing the way the internet sends data, and it has proved impossible to deploy.

"It's a tall task," Alizadeh says. "You'd have to make invasive changes to the standard Internet Protocol (IP) for sending data packets. You'd have to convince all Internet parties, mobile network operators, ISPs, and cell towers to change the way they send and receive data packets. That's not going to happen."

With ABC, the researchers still use the available single bit in each data packet, but they do so in such a way that the bits, aggregated across multiple [data packets](#), can provide the needed real-time rate information to senders. The scheme tracks each data packet in a round-trip loop, from sender to base station to receiver. The base station marks the bit in each packet with "accelerate" or "brake," based on the current network bandwidth. When the packet is received, the marked bit tells the sender to increase or decrease the "in-flight" packets—packets sent but not received—that can be in the network.

If it receives an accelerate command, it means the packet made good time and the network has spare capacity. The sender then sends two packets: one to replace the packet that was received and another to utilize the spare capacity. When told to brake, the sender decreases its in-flight packets by one—meaning it doesn't replace the packet that was received.

Used across all packets in the network, that one bit of information becomes a powerful feedback tool that tells senders their sending rates with high precision. Within a couple hundred milliseconds, it can vary a sender's rate between zero and double. "You'd think one bit wouldn't carry enough information," Alizadeh says. "But, by aggregating single-bit feedback across a stream of packets, we can get the same effect as that of a multibit signal."

Staying one step ahead

At the core of ABC is an algorithm that predicts the aggregate rate of the senders one round-trip ahead to better compute the accelerate/brake feedback.

The idea is that an ABC-equipped base station knows how senders will behave—maintaining, increasing, or decreasing their in-flight packets—based on how it marked the packet it sent to a receiver. The moment the [base station](#) sends a packet, it knows how many packets it will receive from the sender in exactly one round-trip's time in the future. It uses that information to mark the packets to more accurately match the sender's rate to the current [network](#) capacity.

In simulations of cellular networks, compared to traditional congestion control schemes, ABC achieves around 30 to 40 percent greater throughput for roughly the same delays. Alternatively, it can reduce delays by around 200 to 400 percent by maintaining the same throughput as traditional schemes. Compared to existing explicit schemes that were not designed for time-varying links, ABC reduces delays by half for the same throughput. "Basically, existing schemes get low throughput and low delays, or high throughput and high delays, whereas ABC achieves high throughput with low delays," Goyal says.

Next, the researchers are trying to see if apps and web services can use ABC to better control the quality of content. For example, "a video content provider could use ABC's information about congestion and data rates to pick the resolution of streaming video more intelligently," Alizadeh says. "If it doesn't have enough capacity, the video server could lower the resolution temporarily, so the video will continue playing at the highest possible quality without freezing."

More information: ABC: A Simple Explicit Congestion Controller for Wireless Networks: arXiv:1905.03429 [cs.NI] arxiv.org/abs/1905.03429

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

Provided by Massachusetts Institute of Technology

Citation: Reducing delays in wireless networks (2020, April 10) retrieved 6 June 2023 from <https://techxplore.com/news/2020-04-wireless-networks.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.