On the Evolution of End-to-end Congestion Control in the Internet: An Introductory View

Sally Floyd
ICSI
February 2001

Outline of talk:

- The danger of congestion collapse, and the role of congestion control in the Internet.

- Change and heterogeneity as conditions of the Internet.

- Speculations on the future evolution of end-to-end congestion control in the Internet.

Sub-themes:

• The Internet is a work in progress, with no central control or authority, many players independently making changes, and many forces of change (e.g., new technologies, new applications, new commercial forces, etc.)

• So far, the success of the Internet has rested on the IP architecture's robustness, flexibility, and ability to scale, and not on its efficiency, optimization, or fine-grained control.

• The rather decentralized and fast-changing evolution of the Internet architecture has worked reasonably well to date. There is no guarantee that it will continue to do so.

• The Internet is like the elephant, and each of us is the blind man who knows only the part closest to us.

– The part of the Internet that I see is end-to-end congestion control.

What is congestion control?

- Data on the Internet is broken into chunks called *packets*, and a single transfer could have 10, 100, or 1,000 packets.

- It is not like a phone call, with a “pipe” of a fixed size connecting the two ends.
  – Each packet is sent separately, with the destination address contained in the *IP packet header*.

- So how fast should the sender send packets?
  – If the sender sends packets as fast as possible, they could just get lost (dropped) in the middle of the network.
  – The sender uses *end-to-end congestion control* to decide how fast to send packets.

Why do we need end-to-end congestion control?

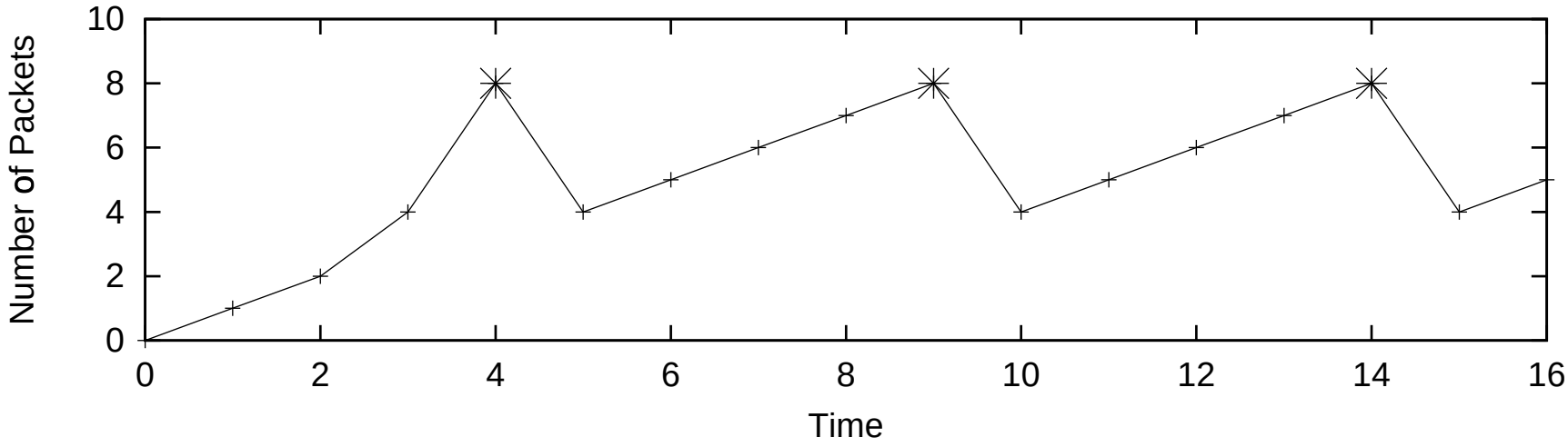
- To avoid congestion collapse.
  - (Rush hour traffic on a bad day...)
- For fairness between users.
- For the user to use the network in the best way it can.

How does congestion control work in the Internet?

- Most of the traffic in the Internet uses the TCP protocol.

- The data sender sends one packet, and waits for an acknowledgement (ACK) from the other end.
  - The data sender then sends two packets...
  - and then four packets... and then eight packets...

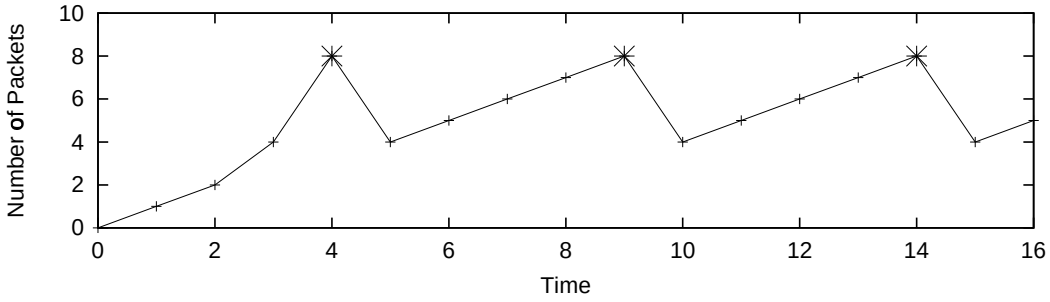
- When a packet is lost in the network, the sender slows down.
  - Then the sender gradually sends faster again.



What is congestion collapse?

- Before 1988, TCP did not use congestion control.
  - In 1986, the Internet had a series of congestion collapses.
  - One form of *congestion collapse* is when the links in the network are busy carrying packets that will only be dropped later in the network.

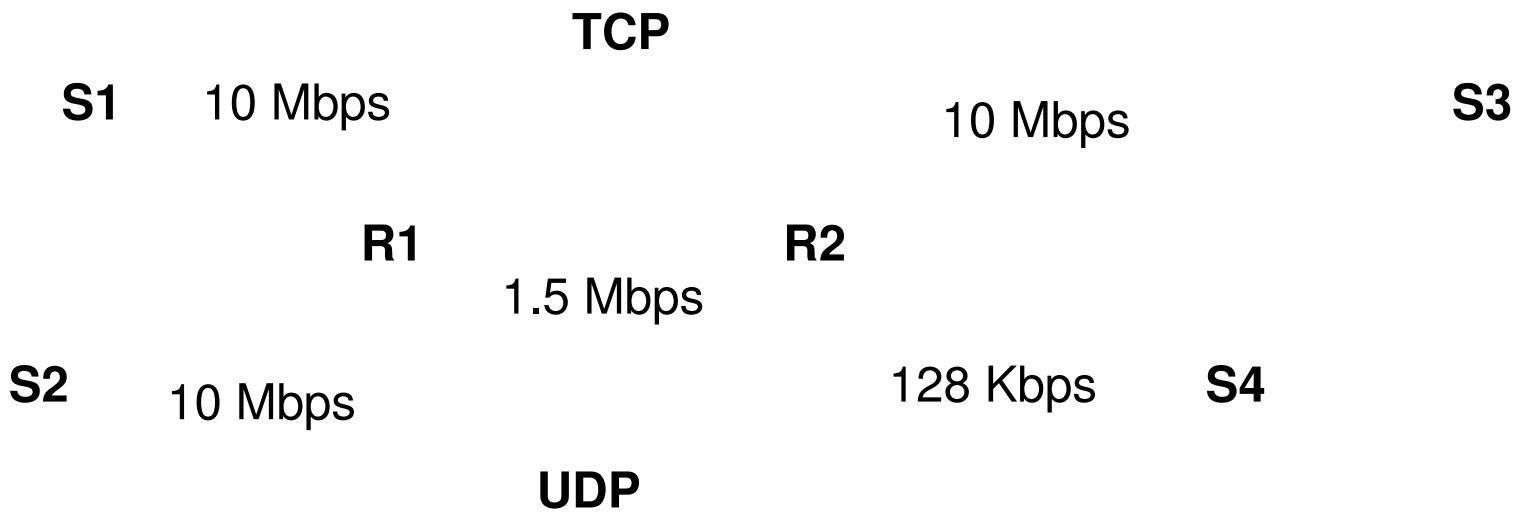
- Congestion control was added to TCP in 1988 by Van Jacobson.
  - Lost packets were taken as indications of congestion.
  - The *congestion window* tells the sender how many packets it can send at once.
  - Double the congestion window each round-trip time until the first loss.
  - Halve the congestion window after a lost packet.
  - Otherwise, increase the congestion window by one each round-trip time.



Congestion collapse from undelivered packets:

Problem: Paths clogged with packets that are discarded before they reach the receiver [Floyd and Fall, 1999].

Fix: Either end-to-end congestion control, or a "virtual-circuit" style of guarantee that packets that enter the network will be delivered to the receiver.



-
- Change and heterogeneity as conditions of the Internet.
-

So why is there so much work to do on congestion control?

- Understanding how this large, complex system behaves.
  – The tools: measurement, modeling, simulations, analysis.

- Making changes to TCP's congestion control.
  – Why? To get better behavior over high-bandwidth networks, wireless networks, satellite networks, with web traffic, ...

- Making changes to the IP header.
  – Why? So that *routers* in the middle of the network don't have to drop packets to tell end-nodes about congestion.

- Research on new protocols as alternatives to TCP.
  – Why? For Internet audio and video traffic. For multicast traffic.

- Research on how routers can better handle *crooks*, *mobs*, and *bullies*
  – (That is, misbehaving users, flash crowds, and Distributed Denial of Service attacks).

Understanding how the Internet behaves:

- Measurements:
  – End-to-end behavior of the Internet in terms of loss; out-of-order delivery of packets; bottleneck bandwidth; etc. [Vern]
  – TCP behavior in web servers. [Jitu, Sally]
  – *Heavy-tailed* distributions of file sizes, web items, etc. (While most web items are small, the large ones are quite large, so most of the web packets in the Internet are from the large items.) [Vern, Sally]
  – Network topology: How can we model the physical structure of the Internet? [Scott]

- Modeling, simulations, and analysis:
  – Just because we understand the rules, does not mean that we understand the behavior of a large, complex system following the rules.

Making changes to TCP's congestion control:

- Increasing the *initial window*, the number of packets TCP can send in the first round:
  - Before: one packet;
  - Now: two packets. Experimental: three or four packets.

- Helping TCP recover more quickly when a single packet is dropped from a small transfer.
  - (*Limited Transmit*: Sending a new packet in response to the first or second *duplicate acknowledgement*.)

- Helping TCP to be more robust with delayed or reordered packets.
  - When TCP thinks a packet is lost, it halves its sending rate.
  - If the packet turns out not to have been lost after all, the TCP sender could adjust its behavior.
  - (With the *D-SACK* extension, the TCP sender can learn if a packet was not lost, but just arrived at the receiver late or out-of-order.)

Making changes to the IP protocols:

- *Multicast* routing: Sending one packet to many different computers, by duplicating the packet in the network. [Mark]

- Explicit Congestion Notification (ECN):
  – A router used to have a certain fixed *buffer size* for packets waiting to be sent.
  – Routers are beginning to use *active queue management*, and to drop packets before the buffer fills up, to tell the end nodes to slow down.
  – With ECN, routers could set a bit in the IP packet header, instead of dropping the packet, to tell the end nodes to slow down. [Sally]

Proposing changes to the IP protocols should not be done lightly:

- "What simulations and measurements of prototype implementations do you have that show that it is better than alternatives? What objective concrete evidence do you have that it is worth the trouble of changing many 1,000,000s of hosts and many 100,000 routers?"
  – V. Schryver, 1999, Email to the end2end-interest mailing list.

- Most of my time today has been spent dealing with two proposals (one good, the other not) of additions to the document on ECN before the next big step of the standardization process. Is this research? architecture? engineering? politics? It is hard to say...

Making new protocols as alternatives to TCP:

• Most of the traffic in the Internet uses TCP, and for most of that traffic, TCP is a good fit.

• For some traffic, TCP's congestion control is not a good fit:

– Audio and video traffic that that would prefer not to halve its sending rate in response to a single packet drop. [Sally, Mark, Jitu, Joerg]

– Multicast traffic, with many receivers. [Mark, Joerg]

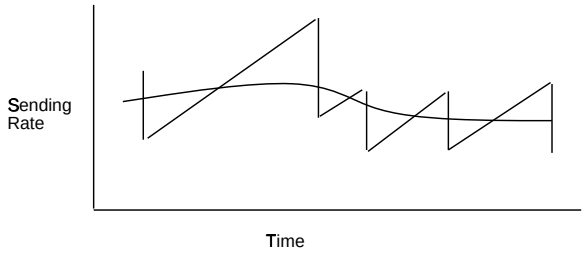
Equation-based congestion control: One new form of end-to-end congestion control

- The "steady-state model" of TCP:

  – The sending rate T of a single TCP transfer, as a function of the packet size B, round-trip time RTT, and packet drop rate p.

$$T = \frac{B}{RTT\sqrt{\frac{2p}{3}} + (2RTT)(3\sqrt{\frac{3p}{8}})p(1+32p^2)} \qquad (1)$$

  – J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, Modeling TCP Throughput: A Simple Model and its Empirical Validation Proceedings of SIGCOMM'98



Equation-based congestion control:

- Use the TCP equation characterizing TCP's steady-state sending rate as a function of the RTT and the packet drop rate.

- Over longer time periods, maintain a sending rate that is a function of the measured roundtrip time and packet loss rate.

- The benefit: Smoother changes in the sending rate in response to changes in congestion levels.

- The justification: It is acceptable not to reduce the sending rate in half in response to a single packet drop.

- The cost: Limited ability to make use of a sudden increase in the available bandwidth.

Helping routers control bullies, mobs, and crooks:

• *Bullies*: Individual transfers that don't use end-to-end congestion control (and try to grab the network resources for themselves).

• *Mobs*: A large crowd trying to access a particular web site (the Olympics, or the Starr Report), for legitimate reasons, but clogging the network for everyone else.

• *Crooks*: Denial of Service attacks. This is malicious behavior.

– A malicious attack on a web server affects the legitimate users accessing that web server as well as other, unrelated traffic on the network.

• The recommendation, for all three cases:

Detect and control at the congested router.

-
-
-

- Speculations on the future evolution of end-to-end congestion control in the Internet.

The future of congestion control in the Internet: several possible views:

- View #1: No congestion, infinite bandwidth, no problems.
- View #2: The “co-operative”, end-to-end congestion control view.
- View #3: The game-theory, competing users view.
- View #4: The virtual-circuit, phone-company view.
- The darker views: Congestion collapse and beyond.

View #1: No congestion, infinite bandwidth, no problems.

- No congestion, essentially infinite bandwidth, no problems.

Well, if this happens, that is fine. I wouldn't want to count on it in all places all of the time.

View #2: The “co-operative”, end-to-end congestion control view.

- The ubiquitous use of end-to-end congestion control for best-effort traffic, encouraged by policing mechanisms at the routers.

- Improved end-to-end congestion control:
  – “Smoother” mechanisms for end-to-end congestion control, in addition to TCP.
  – Explicit Congestion Notification, to reduce packet drops.

- Quality of Service mechanisms for the subset of traffic that needs it.

- Traffic dominated by asynchronous communications, but with a mix of real-time audio and video traffic also.

- Evaluation: It has mostly worked so far, but how well will it scale?

View #3: The game-theory, competing-users view.

- Per-flow scheduling at the routers, protecting users from each other.

- End users each greedily trying to get the best service they can.

- A wide range of quality-of-service mechanisms, with pricing structures to match.

- Evaluation:
  – I believe that per-flow scheduling can still allow congestion collapse, and therefore would still require some form of end-to-end congestion control, but let's assume that is taken care of.
  – Are the benefits of the competing-users view worth the extra complexity and co-ordination that would be required in the network?

View #4: The virtual-circuit, phone-company view.

- A “virtual-circuit” style of coordination within the network, so that packets don’t enter the global network unless there are reasonable guarantees that they can be delivered to the end receiver.
  - Like a telephone call.

- With a virtual-circuit model, there is no need for end-to-end congestion control, and no danger of congestion collapse.

- Evaluation:

There are many costs of this approach, in terms of tight couplings in a far-flung global Internet, and missed opportunities for the opportunistic use of available bandwidth.

The darker views: Congestion collapse and beyond

- Periodic congestion collapse, because of an uneven use of end-to-end congestion control.

- The “Balkanization” of the Internet on ISP boundaries, resulting in effective congestion control and differentiated services only within ISP boundaries, and degraded performance for traffic that crosses ISP boundaries.

- No coherent global architecture, and therefore missed opportunities (in the development of differentiated services, of multicast capabilities, of coherent web caching architectures, etc.)

- Unrestrained “optimization” at all levels, and between levels, producing greater efficiency in the short term, but rigidity and an inability to accomodate change in the longer term.

- Short-term fixes are deployed, possibly blocking the path for longer-term evolution.

- Inherently difficult traffic patterns?