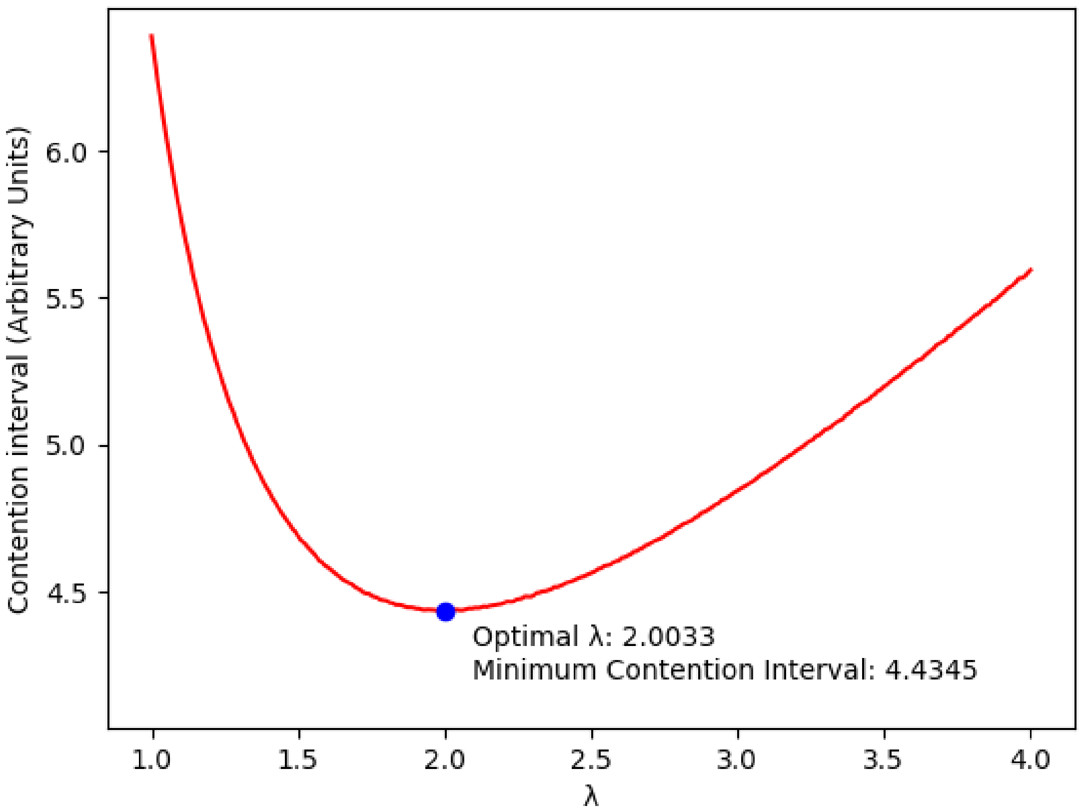
CSCD58 - Assignment 3 - David Traina - 1003914623

1.1



The above plot was generated using the included code with the command:

```
python simulate_contention_interval.py \
  --seed=123456789 \
  --iterations=10000000 \
  --start-lambda=1.0 \
  --stop-lambda=4.0 \
  --step-lambda=0.01
```

The plot shows the trend of lambda vs minimum expected contention interval and highlights the minimum.

1.2

We calculated that the minimum length of the contention interval is found to be approximately 4.4345 slot times (resulting from a lambda value of 2.0033). Suppose that the average transmission of 512 bytes takes 8 slot times (assuming 0 collisions).

Using our optimal contention interval, we can calculate what percentage of the link's theoretical bandwidth is actually available for transmission. Note that Ethernet alternates between contention intervals and successful transmissions.

Theoretical bandwidth of the Ethernet link:

=(512 bytes) / (8 slot times)

=64 bytes per slot time

Optimized "real-world" bandwidth using minimum contention interval of 4.4345 slot times:

=(512 bytes) / ((8 slot times) + (4.4345 slot times))

=512 bytes / 12.4345 slot times

~=41.175760987 bytes per slot time

So, the percentage of the theoretical bandwidth is available for transmission is equal to:

~=(41.175760987 bytes per slot time) / (64 bytes per slot time)

~=0.6433712654

~=64.34%

Note that this also simplifies as just (8 / (8 + 4.4345)) ~= 0.6433712654 ~= 64.34%

Therefore, given the minimum contention interval calculated in our simulation, about 64.34% of the theoretical bandwidth is available for transmission.

2. Facebook Outage Overview

As we have learned in class, the internet is quite literally a "network of networks". Each of the smaller networks that comprise the internet (inter-network) is called an autonomous system, which is uniquely identified with an autonomous system number. Each autonomous system is a network (basically a group of routers) that is operated by a single organization or entity. These entities are usually large organizations like internet service providers (ISPs), universities, or large technology companies like Facebook. Border Gateway Protocol (BGP) is the technology that connects these autonomous systems together and allows them to exchange routing information. This exchange is done via "peering" in which neighboring autonomous systems exchange routing information over one or more TCP connections. Facebook's autonomous system is comprised of many data centers around the world, all connected via their "backbone network".

During the Facebook outage, a BGP configuration error caused Facebook to stop advertising the existence of many prefixes in the Facebook network (autonomous system). That is, it was not announcing some of its IP prefixes across the internet. This meant that ISPs and other networks were unable to route traffic to those IP prefixes because they were unaware of their existence. In order to assess the backbone network capacity, Facebook pushed a configuration change that unintentionally took down the entire backbone network. In response, many of

Facebook’s IP prefixes that relied on that backbone network declared themselves unhealthy by broadcasting a BGP UPDATE message to their peers to withdraw the prefix, and cutting those prefixes in the Facebook autonomous system off from the rest of the internet.

Among the prefixes that were withdrawn were those for all four of the authoritative name servers for facebook.com (a.ns.facebook.com, b.ns.facebook.com, c.ns.facebook.com, d.ns.facebook.com). This is the reason the configuration error was so catastrophic. Although there were still some prefixes being advertised over BGP in the Facebook autonomous system, the DNS servers to resolve facebook.com (or instagram.com, whatsapp.com, etc.) to an IP address were unreachable. It’s analogous to saying your friend’s phone is still connected to the cell network, but you accidentally deleted their contact information, and therefore you can’t call them (although many non-DNS related services were also cut off). If one happend to know the IP address of one of the still advertised Facebook machines, they might still be able to communicate with that machine. However, every browser or mobile app must first go to a DNS server to resolve the hostname to an actual IP, but those DNS servers were now disconnected from the internet, as if facebook.com didn’t exist at all.

The problem took so long to rectify because many of Facebook’s internal tools and servers were rendered inoperational either by the downed backbone network, or by the lack of DNS availability. Engineers had to be physically present at the datacenters to get things back online.

Let’s imagine we wanted to diagnose our connection issues during the Facebook outage. We’ll run through some commands we could have used to get an idea of what’s going on.

1. First, we can quickly verify that we are connected to the internet witha ping to Google
    a. ping -c 1 google.com
2. We try to resolve facebook.com to an IP with our default name server (8.8.8.8)
    a. dig facebook.com

        ;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 31322
        ;facebook.com.   IN   A
    b. We get back a SERVFAIL response. This happense when the authoritative nameservers are unreachable or fail to respond. Next we try a different DNS server (1.1.1.1) to see if the problem is with 8.8.8.8
    c. dig @1.1.1.1 facebook.com

        ;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 31322
        ;facebook.com.
    d. From this we can reasonably conclude there is some issue with Facebook’s authoritative nameservers. 1.1.1.1 and 8.8.8.8 get (and cache) their routing info from Facebook’s own authoritative name servers, so if both are independently returning SERVFAIL then either there is an issue with both 1.1.1.1 and 8.8.8.8, or there is an issue with Facebook (more likely). We know now that Facebook’s

nameservers were disconnected from the internet because they were no longer advertising routes to the authoritative nameservers over BGP. Therefore, other DNS servers could not reach the authoritative nameservers..

3. Next, we will work our way up from a well-known authoritative root name server to illustrate the issue further.
    a. dig www.facebook.com @a.root-servers.net
    b. We omit the output, but it gives us the authoritative name servers for the top-level domain com. We select a.gtld-servers.net and query for facebook.com
    c. dig www.facebook.com @a.gtld-servers.net
    d. We again get SERVFAIL back, so again we find no route to facebook.com (this time directly via the authoritative com. top level domain server server).
       From this brief investigation we can deduce that something is some kind of issue with Facebook's authoritative name servers.

Resources (not including course notes and textbook):

Bagley, Steve. "Why Did Facebook Go Down? - Computerphile" *Youtube*, uploaded by Computerphile, 5 October 2021, www.youtube.com/watch?v=Bie32IZIMtY.

Eater, Ben. "Why was Facebook down for five hours?" *Youtube*, uploaded by Ben Eater, 12 October 2021, www.youtube.com/watch?v=-wMU8vmfaYo.

Janardhan, Santosh. "More details about the October 4 outage." *Facebook Engineering*, https://engineering.fb.com/2021/10/05/networking-traffic/outage-details/. Accessed 8 November 2021.

Martinho, Celso, and Tom Strickx. "Understanding How Facebook Disappeared from the Internet." *Cloudflare*, www.blog.cloudflare.com/october-2021-facebook-outage/. Accessed 8 November 2021.