An Analysis of the Internet - Computer Communication Networks

Table of Contents

John Lahut - jlahut@albany.edu

Mahir Manir - mmanir@albany.edu

Bharghav Baddam - bbaddam@albany.edu

Introduction

PlanetLab is a global Internet testbed comprised of over 1000 machines distributed across the globe. Most of these machines are hosted by research institutions and all of them run a common software package on top of a Linux-based operating system. The objective is to experiment with real, worldwide Internet testbed, to gather first-hand measurements of Internet performance from geographically distributed machines and evaluate path properties in today's Internet. The evaluation you will be performing here is inspired by a paper on Internet path stability by Vern Paxson ([Paxson96]).

We also would test our knowledge and understanding in the Computer Networks with this project.

Motivation

During this experiment we aim to study the differences in various internet paths. We look at latency between nodes, and the different paths packets can take from pairwise nodes. That is, we analyze the differences that may occur for a packet to travel to its destination, and from that destination back to the source.

Path stability in the internet is crucial to understand. With today's internet connecting more devices than ever, there is more traffic than ever before. Forwarding and routing tables throughout networks must remain stable enough to handle this traffic. If paths begin to change frequently, networks can become slow and overloaded. We also target links with large geographical differences to study the differences that may occur to do physical distances.

Methodology

To collect all the data needed for analysis, we set up a framework that was deployed to all nodes. We chose a set of 10 nodes scattered across North America, South America, and Asia. To collect ping and traceroute data from these nodes, we deployed a set of shell scripts that ran indefinitely on each node. Each script was responsible for performing a single measurement, and outputting the results of the measurement to a file. These measurements were ran every 30 minutes on each node. Measurements were collected over a seven day period.

A secondary helper script was deployed onto our UAlbany node. This script was run periodically to copy files generated for each node back to the UAlbany server. At the end of our seven day period, we had over 14,000 measurement files. All nodes used for data collection are using a UNIX based system.

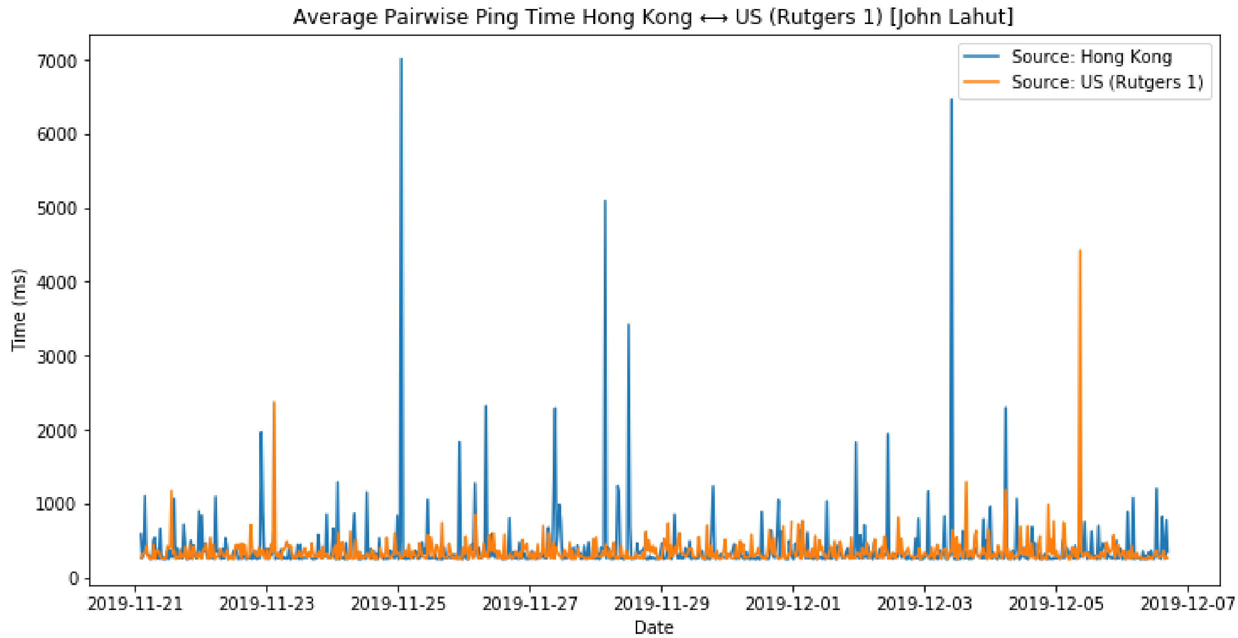
To analyze the files we heavily relied on the use of Python. In particular we used Jupyter notebooks, pandas, and matplotlib for analysis. This allowed us to produce meaningful plots and analyze the data in an efficient way. We utilized GitHub for collaboration, storage, and change management. All of the code, data, and libraries used can be found in the references section of this paper.

Analysis

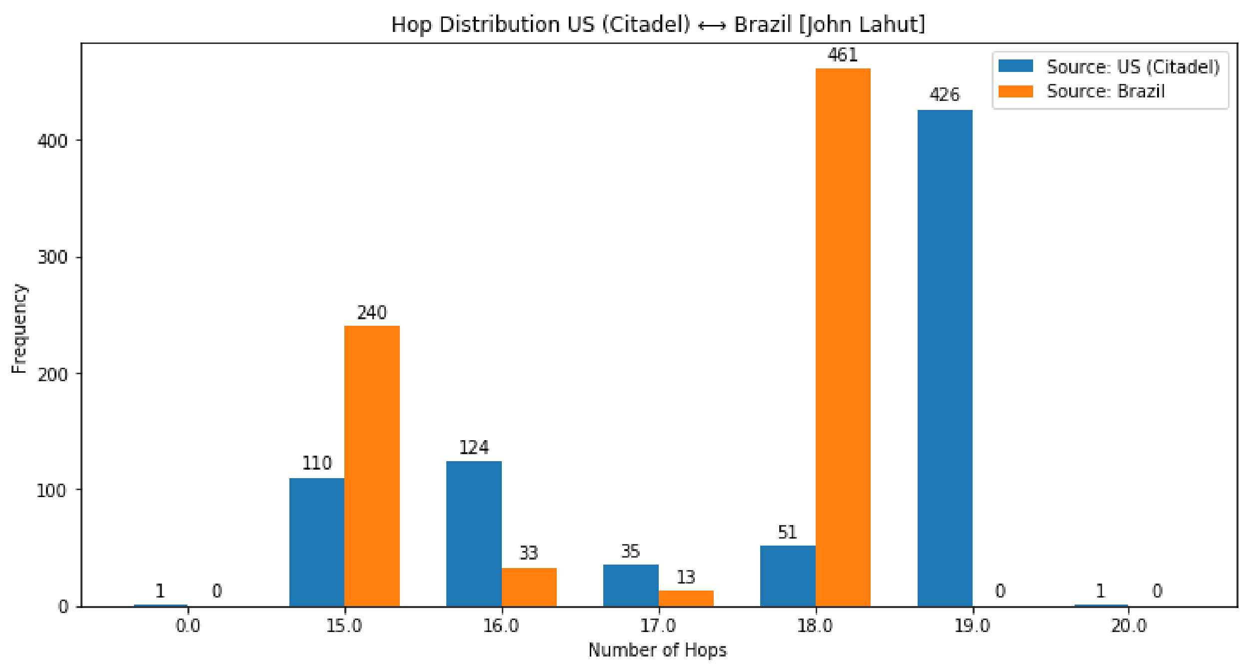
When we first started to analyze the data, it was surprising how variable connections between nodes can be. Some of our nodes had near constant response time and path lengths. Some of them had extreme variation. A summary of our results are below.

	US Citadel to Brazil	**US Citadel to US Rutgers 3**	**Japan to US Rutgers 3**	**Mexico to US Temple**	**Hong Kong to US Rutgers 1**
Minimum Ping Time (ms)	149.011	23.237	163.377	71.734	256.876
Average Ping Time	149.284	23.338	163.529	72.768	368.329
Max Ping Time	149.970	23.744	164.266	75.229	777.954
Standard Deviation	0.516	0.214	0.502	1.167	140.994

From the table above, we can see that Hong Kong to US Rutgers 1 has by far the most variable and volatile path. Ping times are high, and the standard deviation is huge. Below we can see a plot of the pairwise ping times:



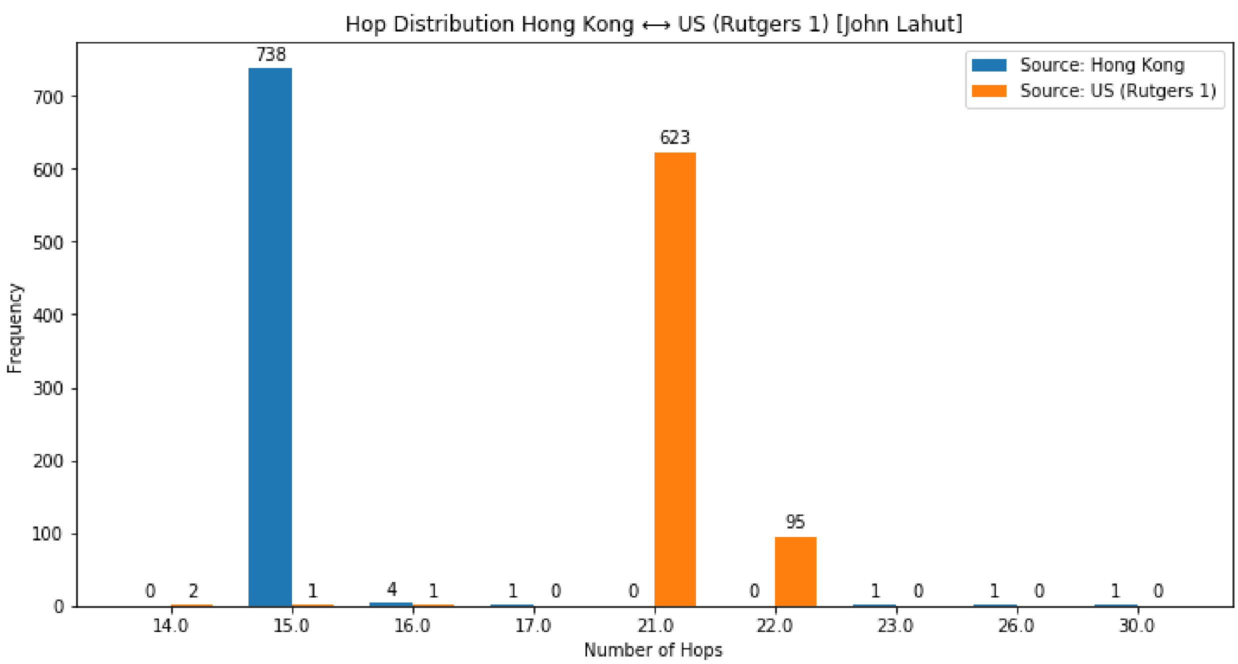
We did observe changes in path lengths during our measurements. Our path lengths from US Citadel to Brazil had the most variation out of all of our measurements. We recorded path lengths from 15 to 19. To try and explore why this happened, we take a closer look at the measurements that took 15 hops, and then compare with the measurements that took 18/19 hops. Additionally, in this case US to Brazil generally took 19 hops, while Brazil to US took 18 hops. A plot of the distribution is shown below.



Looking at the data for outgoing files from Brazil to the US, we see that the RTT for remains about the same for 15 hops versus 19 hops. The extra three hops appear to consistently appear prior to leaving Brazil, and prior to what looks like a transatlantic hop (RTT time jumps 20x). Additionally, these differences in hop counts do not seem to follow a particular pattern on time of day, or if there was a point at which the path changes. We observe mild oscillation within these files as well. Given that the RTTs remain fairly constant, we assume that shortest path algorithms change throughout the measurement period for this link, which is why the number of hops changes during our measurements.

Path Failures

We did observe different hop counts for pairwise nodes. In some of these cases, the hop count varied from both ends (i.e. The US Citadel and Brazil examined earlier). In other cases, one node consistently had one number of hops, where the other had a different number. We will examine the case of Hong Kong to US Rutgers. In this case, almost all of the routes from Hong Kong took 15 hops, whereas from US Rutgers, the routes took 21 hops. The distribution is shown below. All other pairwise hop distributions are shown in the appendix.



It is clear that the paths are consistent between these nodes, but only in one direction. In almost every single traceroute originating from Hong Kong, after getting into the US the packets time out. This is shown by a series of asterisks right after the transpacific hop happens. What is curious is that the final packet eventually reaches US Rutgers, but there are no RTTs for the intermediate hops, except the first few packets that arrives in the US. When originating from US Rutgers, we consistently see reliable packet delivery, but with many more hops. We observe that the US Rutgers path makes an additional 4-5 hops once arriving after the transpacific hop. We are unsure of the reason behind the additional hops once arriving in Asia. We do wonder however if there are any governmental policies behind the scenes to track or trace any incoming packets from a foreign country given the recent political climate in Hong Kong.

Temporary and Long Term Outage:

The frequency of outages is different for different paths.

The following are the percentages of outages (temporary or long term outages) which are observed among the nodes.

Calculations below are for bidirectional.

Nodes		Fraction of Outages
US - Brazil		3.8
US(Citadel) - US (Rutgers)		64.2
Japan - US		66.2
Mexico - US		100
HongKong - US		72.5

And the total number of temporary outages from to and fro are –

Calculations below are for bidirectional.

Nodes		Temporary Outages
US - Brazil		11
US(Citadel) - US (Rutgers)		703
Japan - US		745
Mexico - US		965
HongKong - US		746

We cannot determine the likely cause based on the measurements. The failures could be because of a router which is working but doesn't respond to host or due to a firewall or due to packet drop. So definitely we need additional information such as the location of the router of failure or something of similar kind mentioned above to find the cause of failure.

The Traceroute itself is not enough to find the cause of failure.

And the long term outages with 10's of seconds delayed with destination reached :

Nodes (Bidirectional)		Long Term Outages with destination
US - Brazil		0
US(Citadel) - US (Rutgers)		10
Japan - US		10
Mexico - US		13
HongKong - US		10

And the long term outages without destination reached :

Nodes (Bidirectional)		Long term Outages without destinations
US - Brazil		0
US(Citadel) - US (Rutgers)		0
Japan - US		0
Mexico - US		1
HongKong - US		1

The long term outages could be those with 10's of seconds delayed but eventually reached the destination and the one's without destination.

The reason could be either the router or destination is not working and dropping all the packets or a part of network and firewall is not allowing to packets to propagate and dropping them.

So, it is difficult to determine why the long term outage is caused by.

Core and Edge Failure Percentage:

The Percentages of failures in Edge and Core **(A – B)**.

Nodes (A - B)		Edge Percentage
US - Brazil		16
US(Citadel) - US (Rutgers)		20.3
Japan - US		20.7
Mexico - US		61.1
HongKong - US		20.3
Nodes(A-B)		Core Percentage
US - Brazil		84
US(Citadel) - US (Rutgers)		79.6
Japan - US		79.2
Mexico - US		38.8
HongKong - US		79.6

The percentages of failures in Edge and Core **(B -A)**

Nodes (B - A)		Edge Percentage
Brazil - US		52.3
US (Rutgers) - US(Citadel)		29.7
US - Japan		20.8
US - Mexico		41.9
US-HongKong		3.22
Nodes (B - A)		Core Percentage
Brazil - US		47.6
US (Rutgers) - US(Citadel)		70.2
US - Japan		79.1
US - Mexico		58.01
US-HongKong		96.7

The major percentages of failures have occurred in the core network. It may be the reason that majority of routing path is in the core network compared to edge network, so it is expected the core which is with more routers may have more failures than Edge but not often edge also has more failures than core network.

Fluttering:

Yes I have detected route fluttering, lets take some help from the fluttering graphs done by Mahir.

From US (Citadel) – Brazil Flutter Graph:

There are only few routes which have fluttering hops compared to the number of routes without flutter in birection. The number of traceroutes with fluttering are 305 both directions combined.

From US(Citadel) – US(Rutgers 3)

The route from Citadel to Rutgers do not have Fluttering except one traceroute file but the path from Rutgers to Citadel has large number of traceroute files with fluttering. More frequent in this path and the number of traceroutes with fluttering are 699 both directions combined.

From Japan – US(Rutgers 3)
It's the same case as above, the path from Japan to Rutgers do not have fluttering except one traceroute file, but the path from Rutgers to Japan is having large number of traceroute files with fluttering. More often in this path and the number of traceroutes with fluttering are 744 both directions combined.

From Mexico to US (Temple)
Here both the routes i.e., Mexico to US and US to Mexico have large number of traceroute files with Fluttering. More frequent in both the paths and the number of traceroutes with fluttering are 1473 both directions combined.

From Hong Kong to US(Rutgers 1)
The path from HongKong to US(Rutgers 1) does not have any fluttering. But the path from US (Rutgers 1) to Hong Kong have large numbers of traceroutes with Fluttering.
The number of traceroutes with fluttering are 722 both directions combined.

Path Reliability between Continental Links and Inter-Continental Links

We were tasked with finding if routes show any significant differences in reliability between continental and inter-continental links. To define reliably we have looked into several factors according to the finds in [Paxson96]. We first any instances of Forwarding Loops in out traceroute files. Forwarding loops as explained in [Paxson96] are routers that receive packets, only to be forward to the same router in sequence. However they also mentioned that this could just be a result of an upstream route change. Thus we followed the methodology from [Paxson96] and defined a forwarding loop as a "traceroute measurement that show the same sequence of routers at least 3 times". To test this we split our connections into three categories, Continental Links, cross Atlantic links, and cross Pacific links. Please refer to *Reliability of Continental Links* for the node names, locations and index numbers. We then wrote a script to find what percent of traceroute files contain forwarding loops for each bi-directional connection. As we can see from the data links (5-1/1-5), (3-5/5-3) and (6-8/8-6) had about half of their traceroutes contain forwarding loops. Moreover both pacific links have the most amount of forwarding loops. We also tested for persistent loops, which is defined in [Paxson96] as trace routes that are unresolved by then end of the trace route. This mean the traceroute doesn't exactly fail, however it also never full reaches its destination. We again tested this using the same three categories and a script that find the percentage of persistent loops present for the same bi-lateral connections. As we can see the majority of persistent loops occurred for trans-pacific routes. These routes failed to meet their destination about 20% of the time while the rest of the categories seem pretty reliable. The third way we can determine reliability is to see if the traceroute destinations are unreachable of if they fail outright. Again we used the same categories and scripting methodology to find the unreachable traceroutes. This test however results in very few errors with only two files in trans-pacific categories results are unreachable.

Reliability of Continental Links -Mahir Manir

	Forwarding Loops	**Persistent Loops**	**unreachable**
US Citadel to US Rutgers 3 (5-1/1-5)	**49.96%**	**3.34%**	**0.0%**
Mexico to US Temple (4-7/7-4)	**00.07%**	**0.07%**	**0.0%**
US Citadel to Brazil (1-2/2-1)	**0.02%**	**3.34%**	**0.0%**
Japan to US Rutgers 3 (3-5/5-3)	**49.96%**	**20.27%**	**0.07%**
Hong Kong to US Rutgers 1 (6-8/8-6)	**50.89%**	**17.51%**	**0.07%**

Key: Continental link, Atlantic link, Pacfic Link

Lastly reliability can be observed though cases of fluttering. Fluttering as explained by [Paxson96] is a rapidly oscillating router. This means that certain hops in the traceroute will contain more than one router. While fluttering can is allowed and can be normal, constant fluttering shows the route is unstable. Moreover if the traceroute flutters in only one direction, then the path is unstable [Paxson96]. To test this we wrote script to detect the frequency of fluttering hops in our bi-directional traceroute files. If we refer to the series of figures titles *"Packet fluttering distribution"* in the appendix we can see some interesting results. It seems the most frequent route fluttering actually happened between Temple(US) and Mexico. [nodes 4-7/7-4]. Moreover the transpacific routes from Japan or Hong Kong virtually had no cases of fluttering. Even the Trans-Atlantic had little to no cases of fluttering. Although our data shows that continental links and trans-Atlantic links are more reliable than trans-Pacific links based on the frequency of router loops, we cannot come to that conclusion. This is not only due the lack of evidence from the fluttering routes and the small amount of nodes used. In fact it could be due to a faulty node itself. It seems that Rutgers3 and even Rutgers1 might be the cause of the instability. Almost all of the cases of forwarding loops involved Rutgers3 regardless of it was a continental link or not. The most damning evidence comes from our analysis of fluttering. Every single traceroute leaving Rutgers has around 6-8 instances of fluttering regardless of the type of link. This is very evident in the trans - Pacific routes. While routes from Japan and Hong Kong have near perfect routes, both Rutgers nodes have between 6 and 8 instances of fluttering per traceroute. This leads to an asymmetrical route.

Triangle Routing

Triangle Routing as defined by [Paxson96] is found when certain traceroute paths tend to take the alternative path as opposed to the shortest path. Although we did not explicitly look for router locations in our traceroute files we can still assume there is triangle routing. This is due to the *Hop Distribution* figures in the appendix. We can clearly see that any traceroutes take multiple routes to go to the same destination. This is especially evident in between Brazil and temple. There are many cases where it took many more hops to reach the destination that other trace routes. However it is important to realize this can also be a case of forwarding loops.

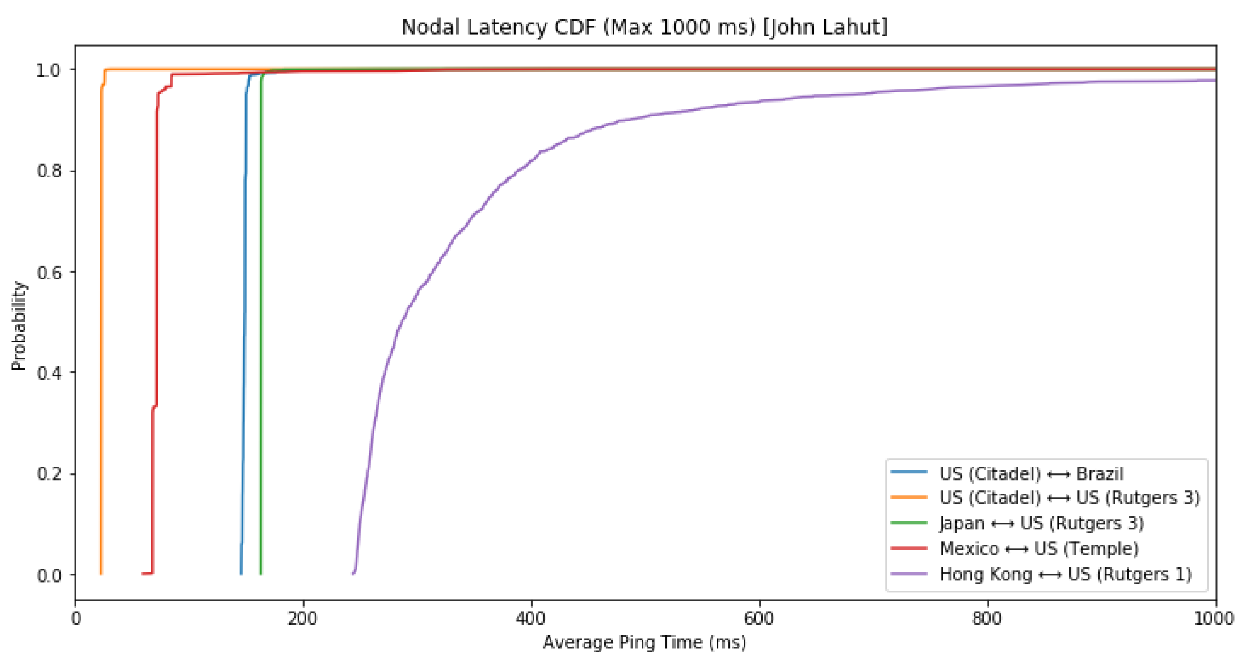
Path Loss and Latency

Our pairwise nodal latency varied quite a bit from pair to pair, which was expected given the geographical location of each of the nodes. Each of the ping times measured can be seen in a plot in the Appendix section of this paper.

Our intercontinental path, US Citadel to US Rutgers had the most consistent ping times. This was not surprising, and was expected given the geographical distance between the two nodes. On the opposite end of the spectrum, our Hong Kong to US Rutgers pair experienced great variability in terms of ping times. This particular case is interesting given how consistent our other transpacific nodal pair was (Japan to US Rutgers).

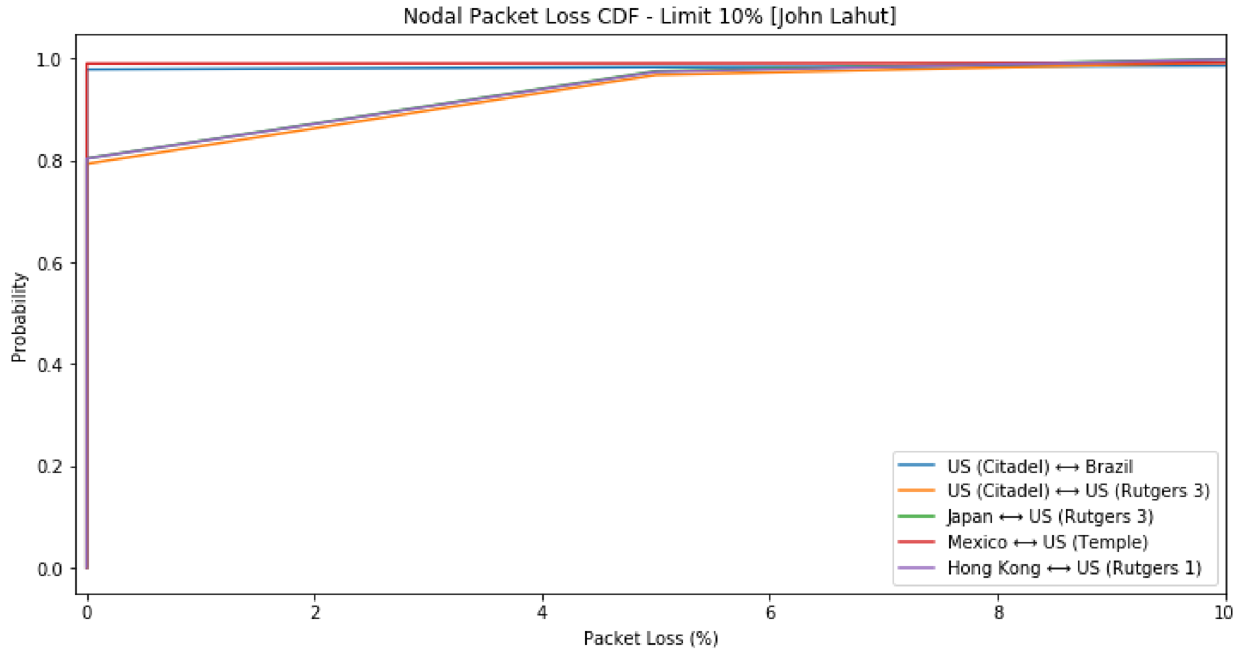
To get a better sense of the variability of ping times and packet loss for each of the nodal pairs, we plot a CDF for each of these attributes.

As seen in the below CDF of our nodal latency, the Hong Kong to US pair shows a smooth curve, signaling that the ping has a high variability, opposed to the other nodal pairs. We were quite surprised on how variable the ping times were for this node. All other nodes follow a sharp bend at the top of the “curve”. This indicates fairly consistent ping times between the nodes. Please note that this plot is limited to a maximum ping time of 1000ms to better display the most likely ping times. The full plot can be seen in the appendix.



Another pairwise set that continued to surprise use was the reliability and consistency of Japan to US Rutgers node. Nodal ping times remained consistent throughout the measurements, and path length distribution was one of the most constant paths that we measured.

Looking at pairwise packet loss, we follow a similar methodology to nodal latency. Below is a CDF of packet loss for each of the pairwise nodes.



We were surprised on how reliable these links were over a one week period in terms of reliable packet delivery. Almost every node had a near perfect reliable packet delivery. Two of the nodes, US Citadel to US Rutgers and Hong Kong to US experienced more packet loss than the other nodes. It was interesting that the intercontinental link was one of the worst performing pairs, given how close they are in geographical location. The other pair, Hong Kong to US was not surprising given how unreliable the ping times were and how much the connection varies.

Contributions

In this section we list contributions by team members in a tabular format. Each member was responsible for a certain number of questions, however we worked together and collaborated often on all questions.

Any plots produced by a team member will have their name attached to the plot itself.

Team Member	Task
John	Set up GitHub repository
John	Wrote shell scripts and deployed to nodes

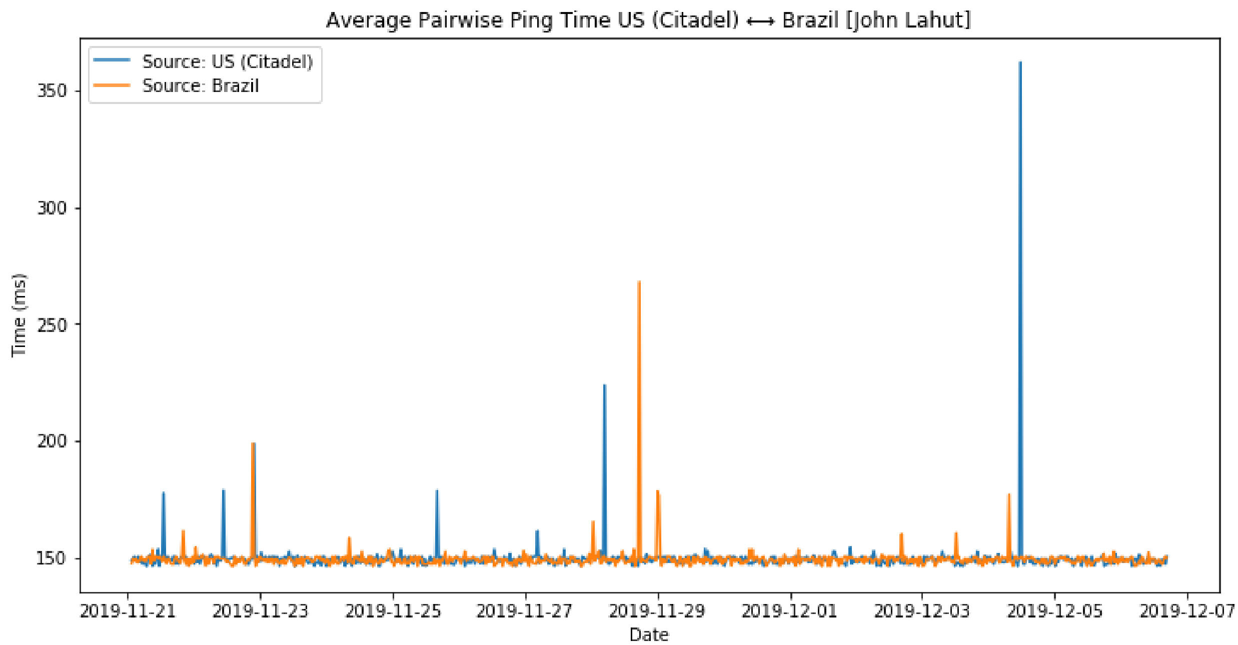
John	Set up initial data analysis framework
John	Question 1
Bharghav	Question 2
Bharghav	Question 3
Mahir	Question 4
Bharghav	Question 5
John	Question 6
Mahir	Question 7
John	Question 8
Mahir	Question 9
Bharghav	Introduction
John	Methodology Section
John	Motivation Section
Mahir	Conclusion

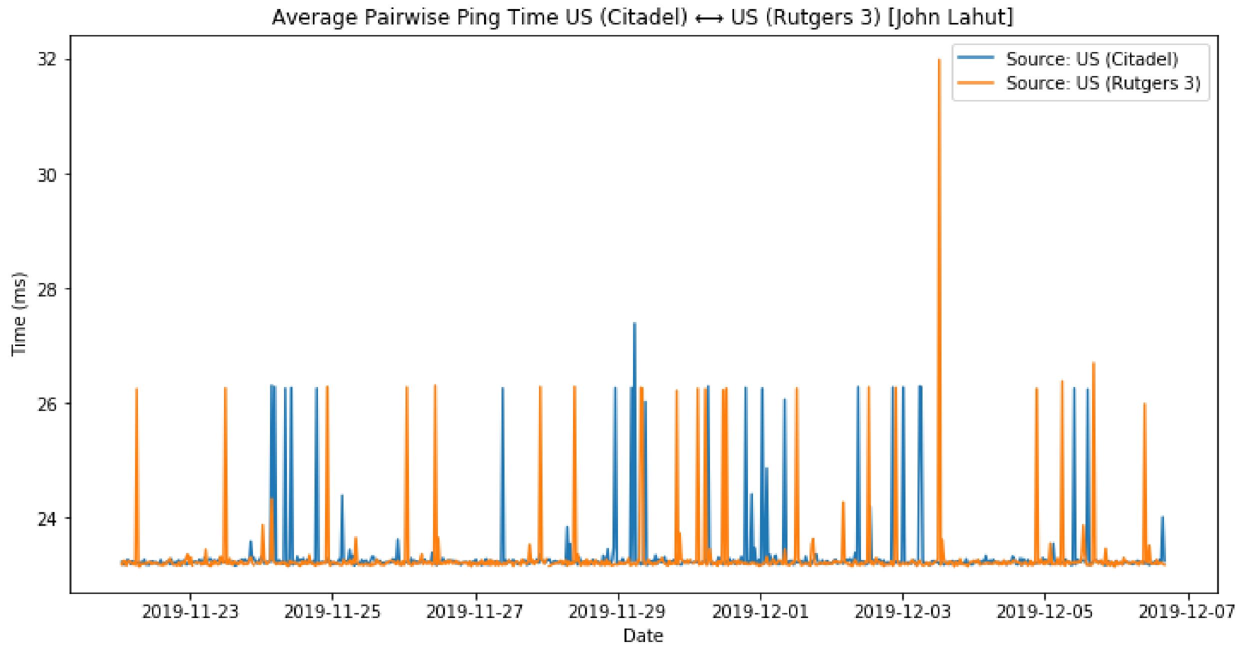
Conclusion

In the paper *End to End Routing Behavior in the Internet* by Vern Paxson, he concluded that internet routing has become less predictable between 1994 and 1995 by an increase of 2.1%. Using his finding and adapting his methodology we can conclude that internet routing instability has increased. From our findings in ping times, traceroute hop frequency, forward and persistent looping and especially fluttering; we have found an alarming rate of infrequencies across the board. Although many cases could be due to our selection of two Unstable Rutger nodes, can we not also claim that as further proof of router instability?

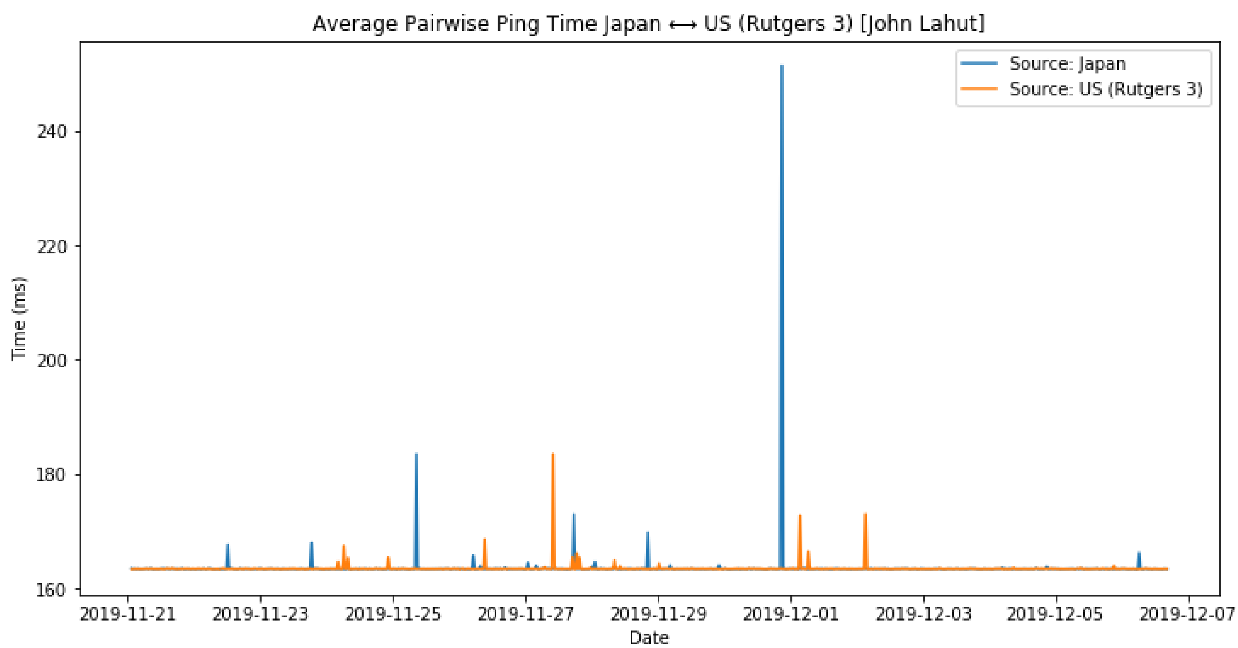
Appendix

All implementations and data files can be found at github.com/johnlahut/planet-lab. The readme file at the root of the repository explains the project structure.

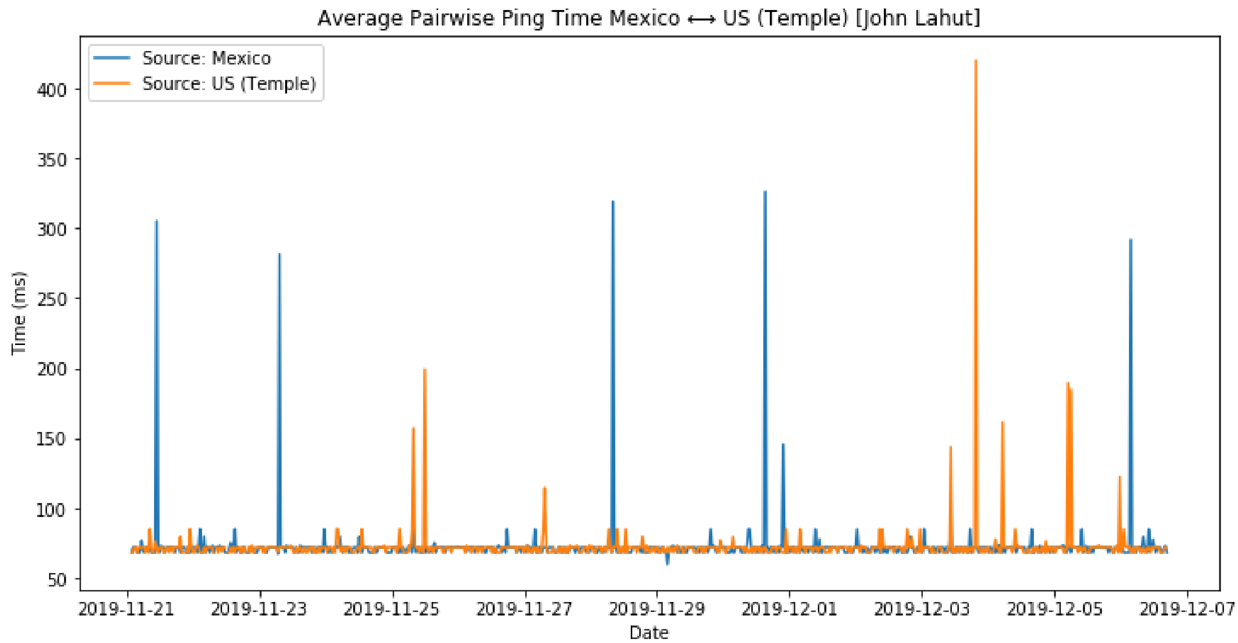

Average Pairwise Ping Time US (Citadel) ⟷ Brazil [John Lahut]
Source: US (Citadel)
Source: Brazil
Time (ms)
350
300
250
200
150
2019-11-21
2019-11-23
2019-11-25
2019-11-27
2019-11-29
2019-12-01
2019-12-03
2019-12-05
2019-12-07
Date

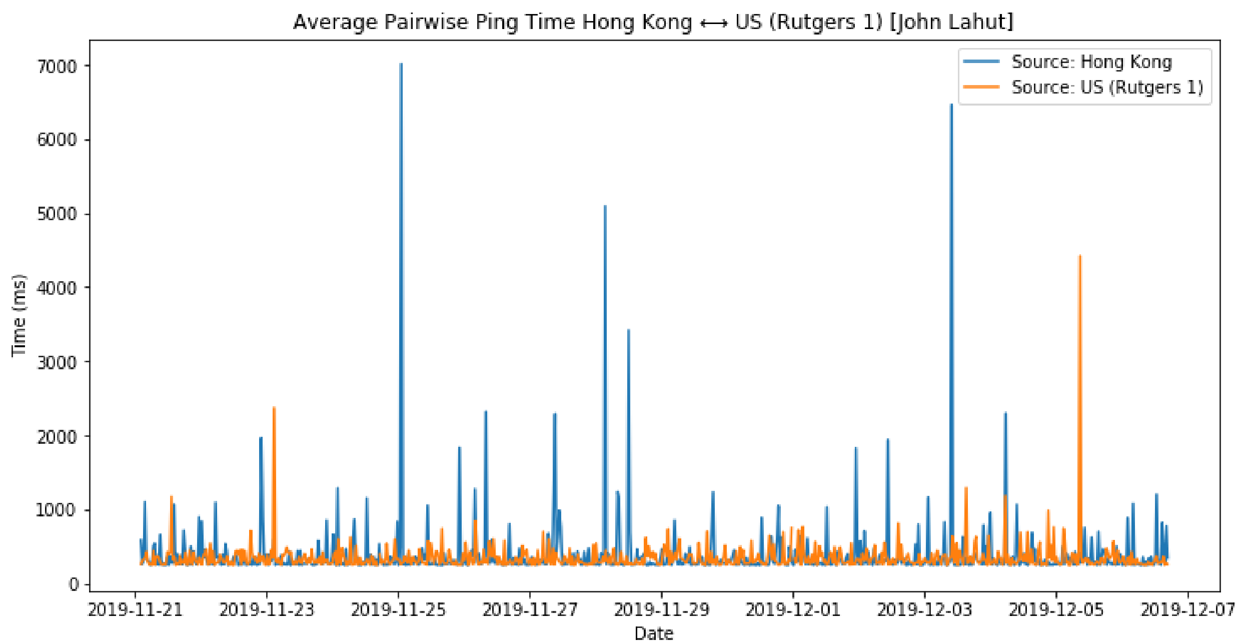


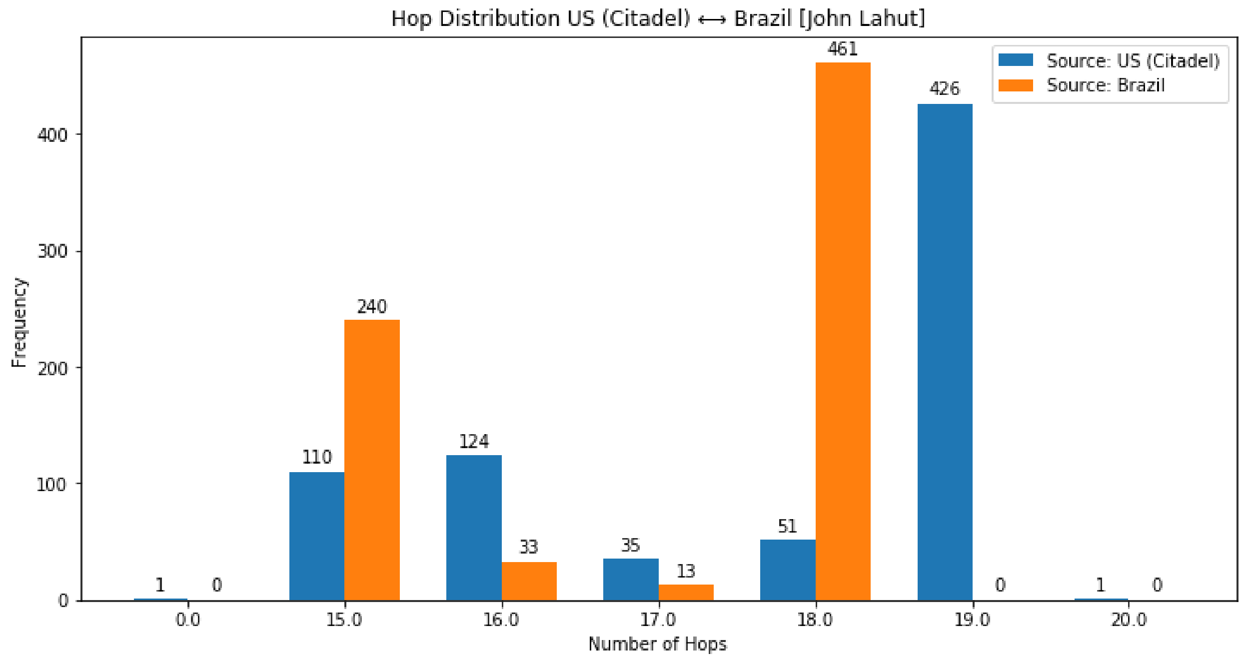
Average Pairwise Ping Time US (Citadel) ⟷ US (Rutgers 3) [John Lahut]
Source: US (Citadel)
Source: US (Rutgers 3)
Time (ms)
Date

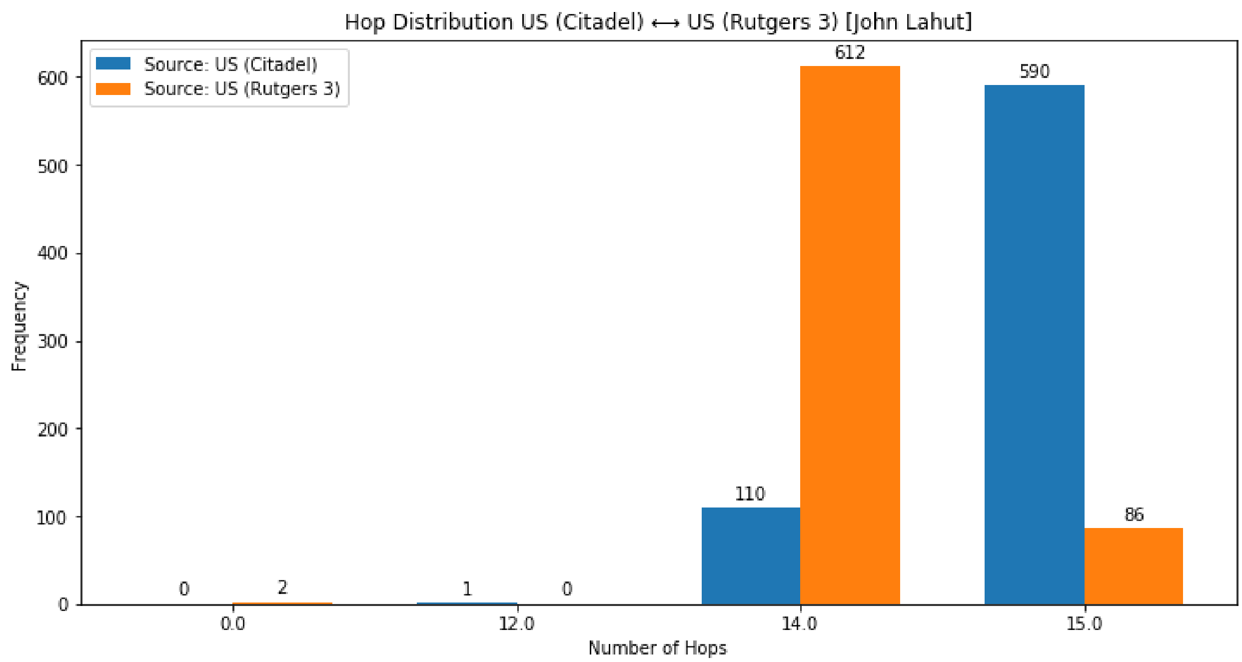


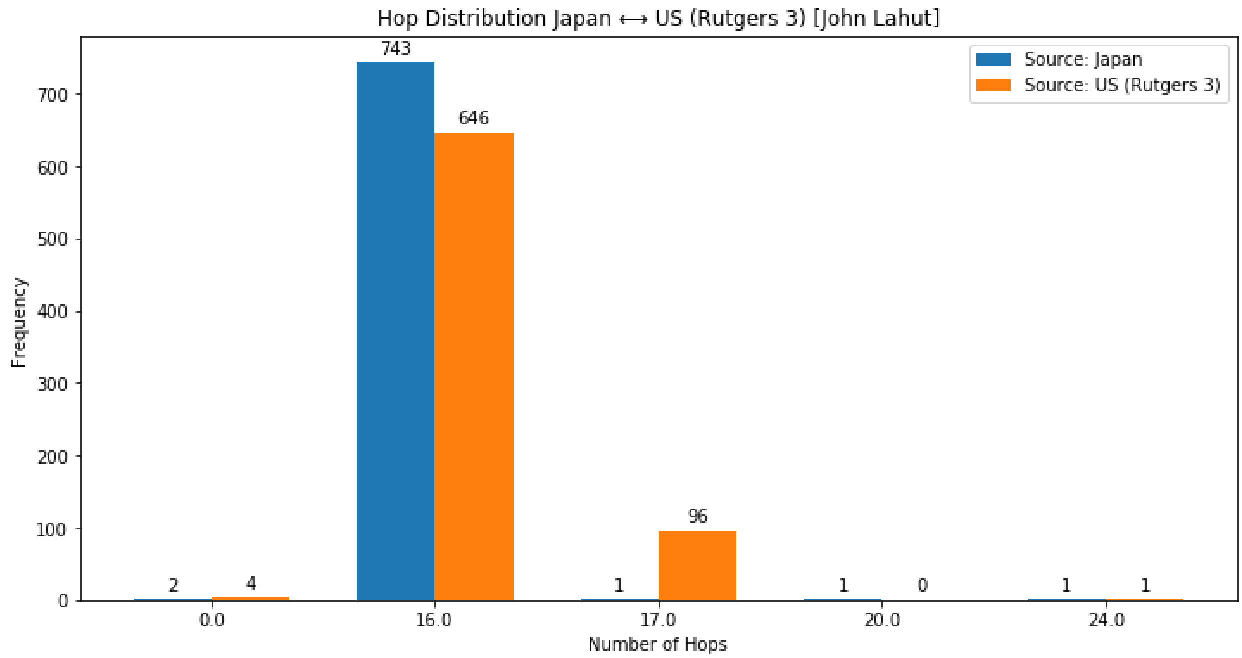
Average Pairwise Ping Time Japan ⟷ US (Rutgers 3) [John Lahut]
Source: Japan
Source: US (Rutgers 3)
Time (ms)
Date

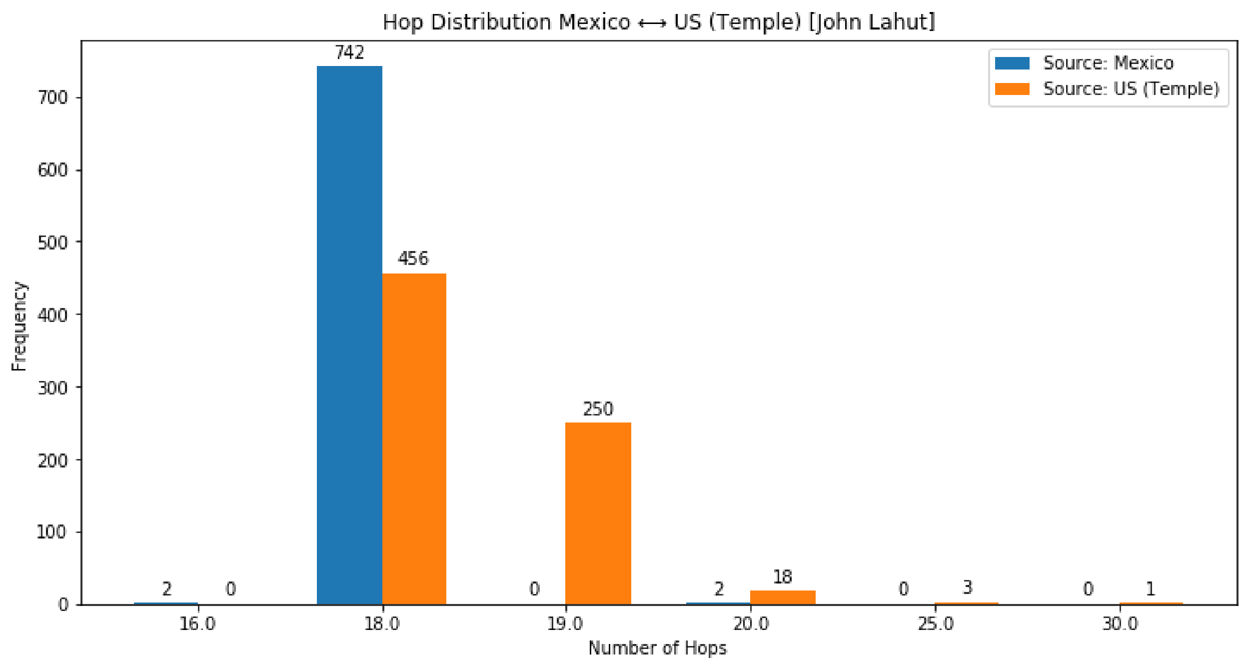

Average Pairwise Ping Time Mexico ⟷ US (Temple) [John Lahut]
Source: Mexico
Source: US (Temple)
Time (ms)
Date
50
100
150
200
250
300
350
400
2019-11-21
2019-11-23
2019-11-25
2019-11-27
2019-11-29
2019-12-01
2019-12-03
2019-12-05
2019-12-07

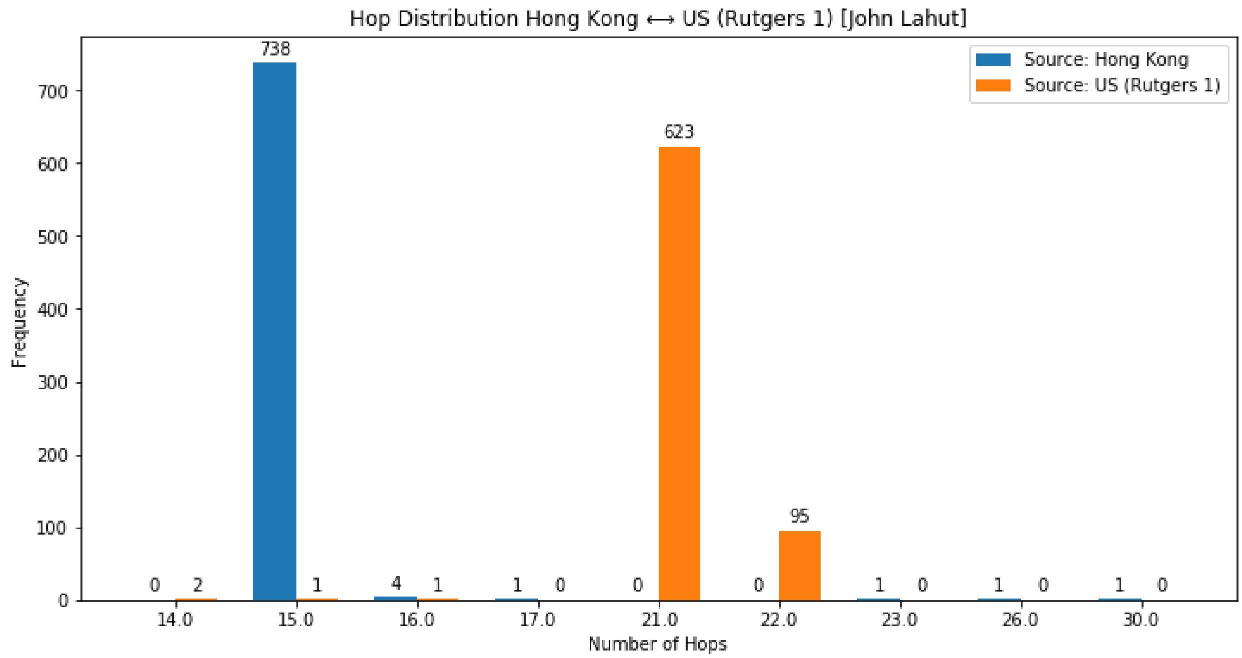

Average Pairwise Ping Time Hong Kong ⟷ US (Rutgers 1) [John Lahut]
Source: Hong Kong
Source: US (Rutgers 1)
Time (ms)
Date
0
1000
2000
3000
4000
5000
6000
7000
2019-11-21
2019-11-23
2019-11-25
2019-11-27
2019-11-29
2019-12-01
2019-12-03
2019-12-05
2019-12-07

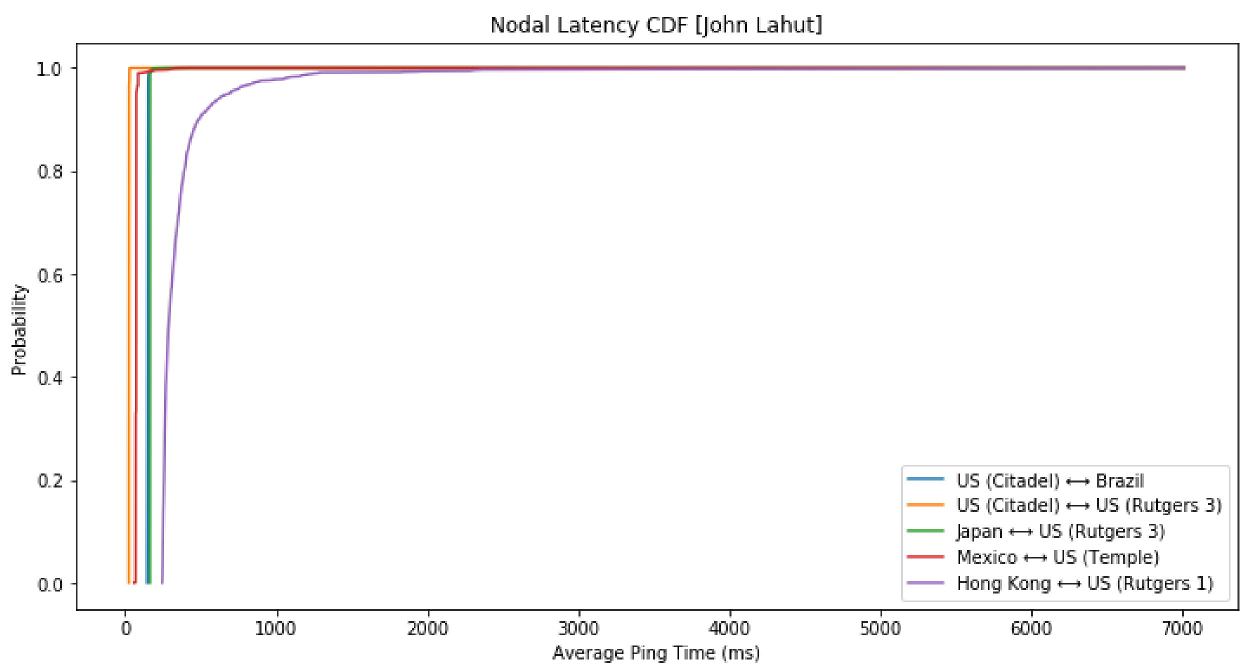

Hop Distribution US (Citadel) ⟷ Brazil [John Lahut]
Source: US (Citadel)
Source: Brazil
Frequency
Number of Hops
0
100
200
300
400
0.0
15.0
16.0
17.0
18.0
19.0
20.0
1
0
110
240
124
33
35
13
51
461
426
0
1
0

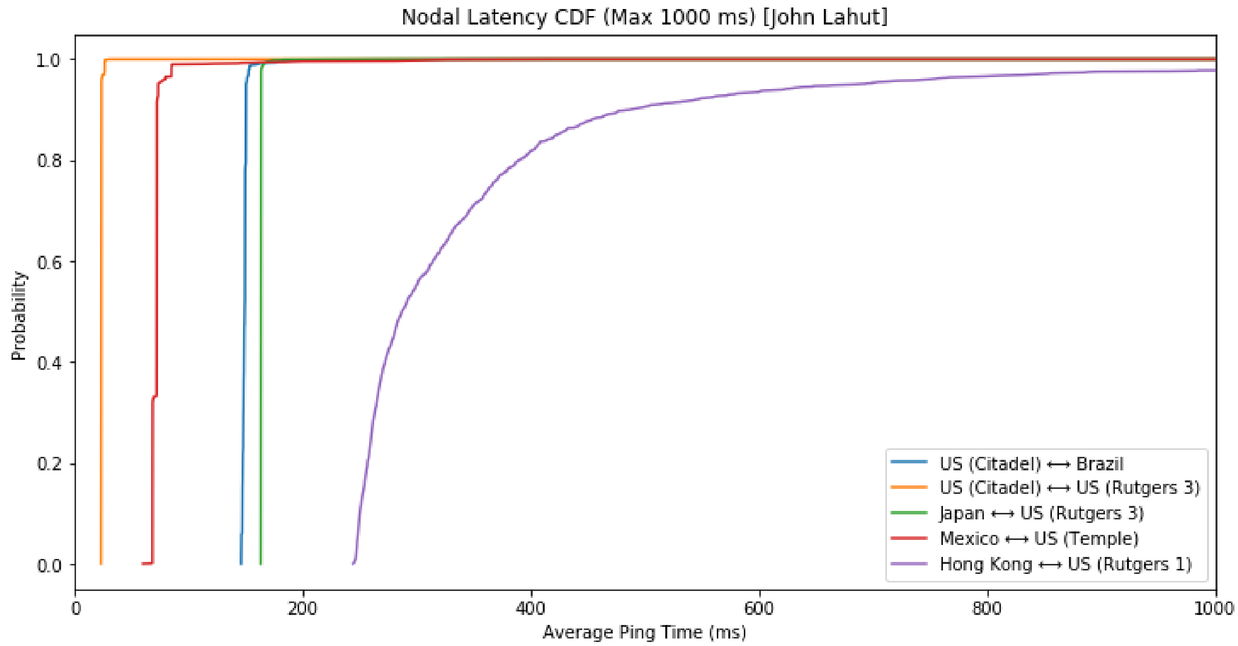

Hop Distribution US (Citadel) ⟷ US (Rutgers 3) [John Lahut]
Source: US (Citadel)
Source: US (Rutgers 3)
Frequency
Number of Hops
0
100
200
300
400
500
600
0.0
12.0
14.0
15.0
0
2
1
0
110
612
590
86

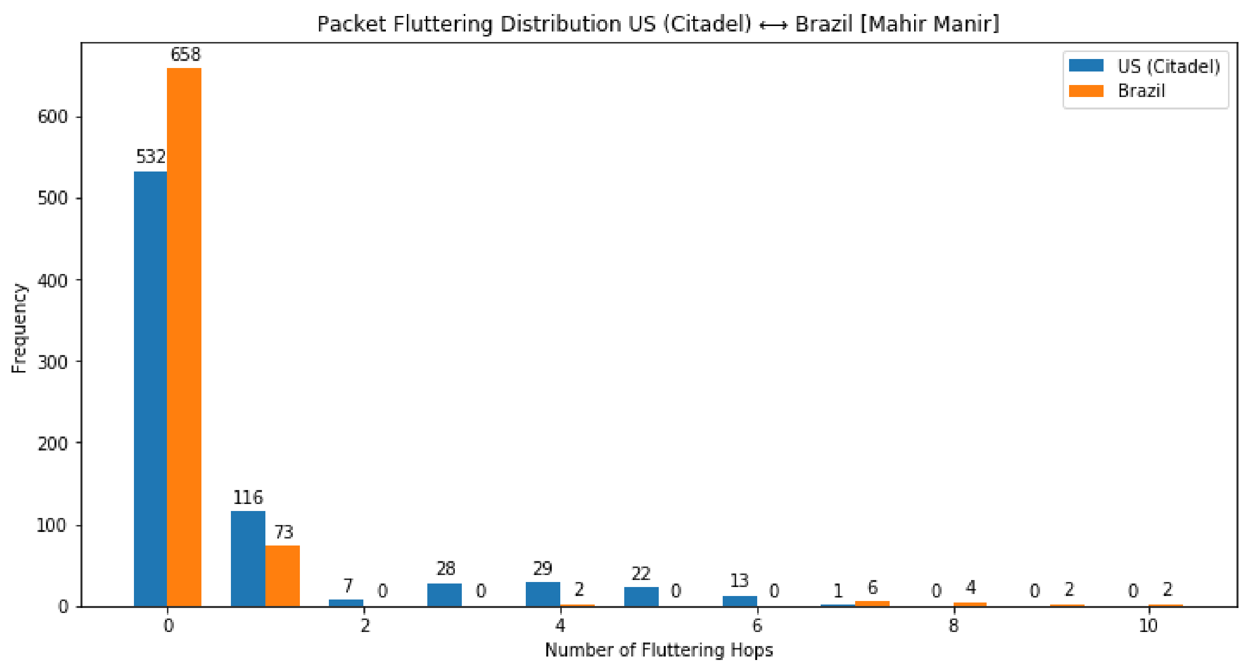

Hop Distribution Japan ⟷ US (Rutgers 3) [John Lahut]
Source: Japan
Source: US (Rutgers 3)
Frequency
Number of Hops
0
100
200
300
400
500
600
700
0.0
16.0
17.0
20.0
24.0
2
4
743
646
1
96
1
0
1
1

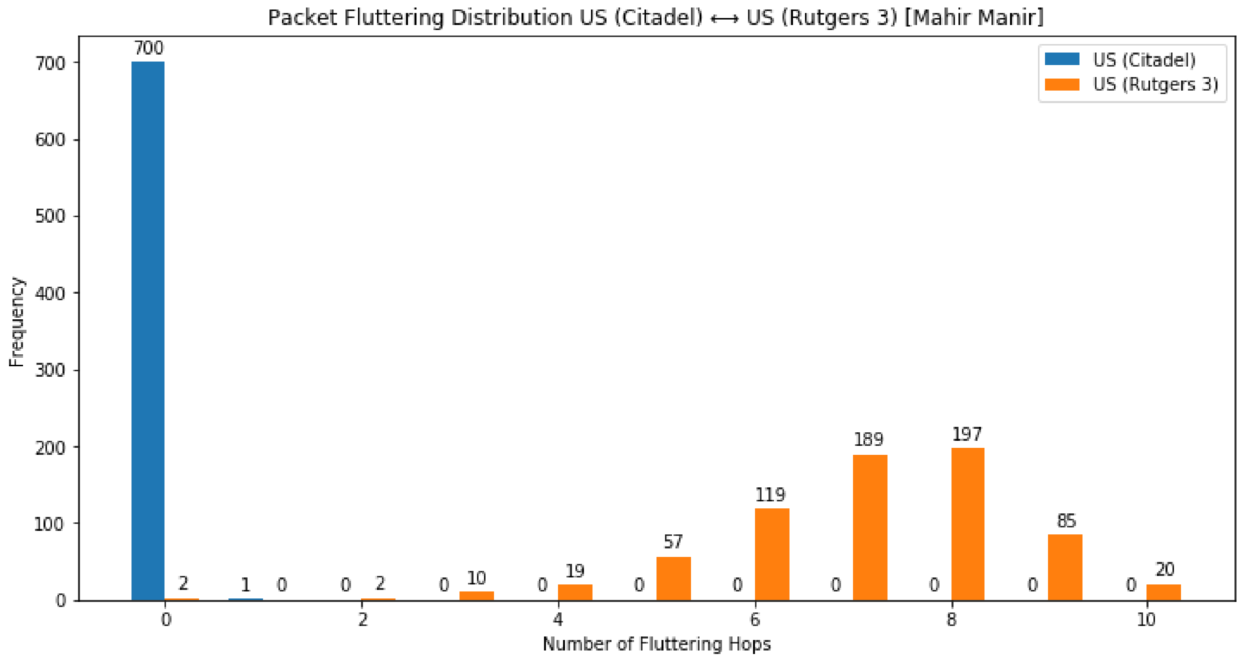

Hop Distribution Mexico ⟷ US (Temple) [John Lahut]
Source: Mexico
Source: US (Temple)
Frequency
Number of Hops
0
100
200
300
400
500
600
700
16.0
18.0
19.0
20.0
25.0
30.0
2
0
742
456
0
250
2
18
0
3
0
1

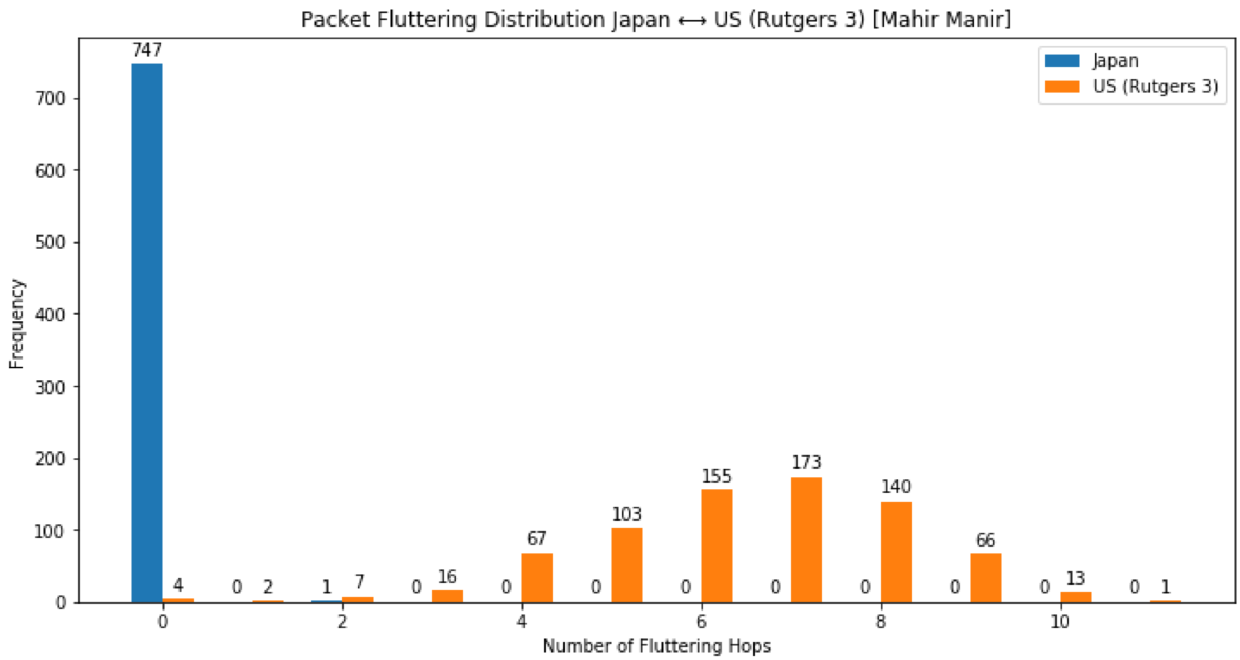

Hop Distribution Hong Kong ⟷ US (Rutgers 1) [John Lahut]
Source: Hong Kong
Source: US (Rutgers 1)
Frequency
Number of Hops
738
623
95
0 2
1
4 1
1 0
0
0
1 0
1 0
1 0
14.0
15.0
16.0
17.0
21.0
22.0
23.0
26.0
30.0
0
100
200
300
400
500
600
700

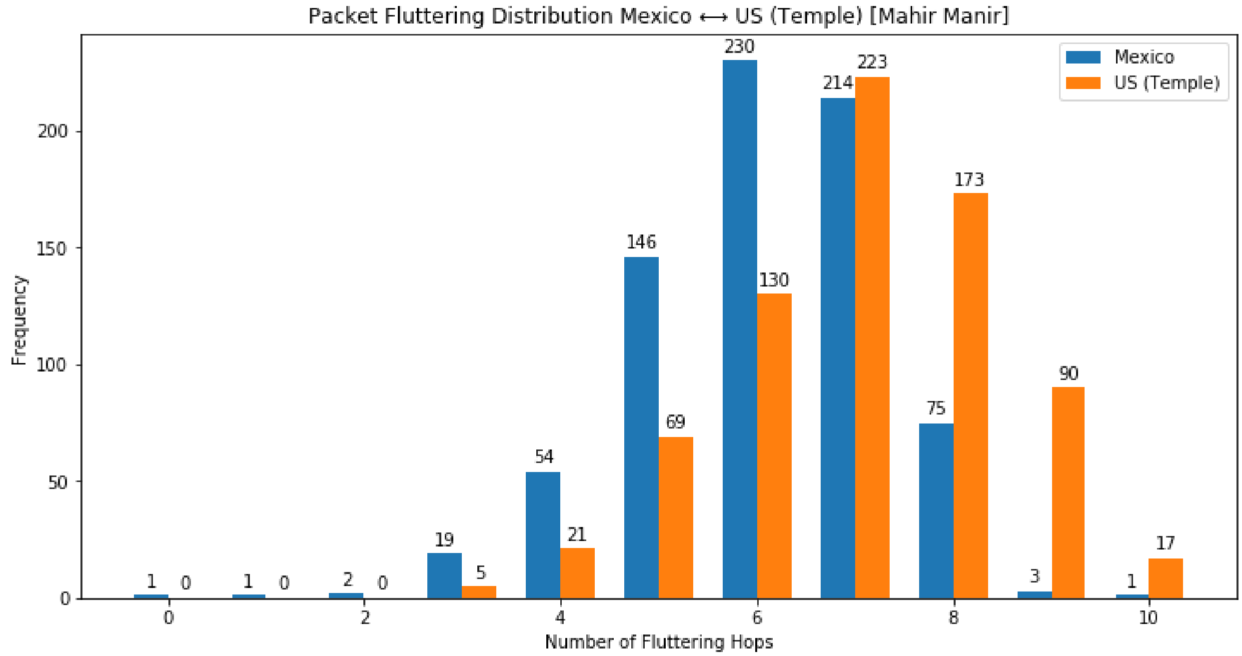

Nodal Latency CDF [John Lahut]
Probability
Average Ping Time (ms)
US (Citadel) ⟷ Brazil
US (Citadel) ⟷ US (Rutgers 3)
Japan ⟷ US (Rutgers 3)
Mexico ⟷ US (Temple)
Hong Kong ⟷ US (Rutgers 1)
0.0
0.2
0.4
0.6
0.8
1.0
0
1000
2000
3000
4000
5000
6000
7000

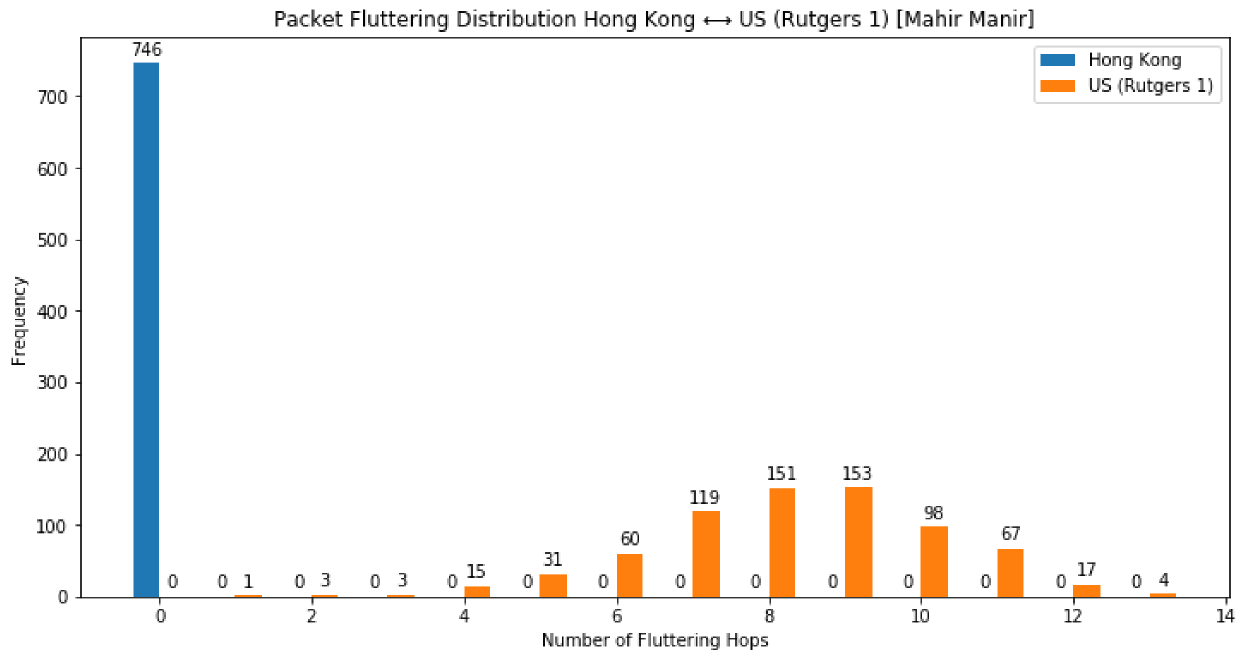

Nodal Latency CDF (Max 1000 ms) [John Lahut]
Probability
Average Ping Time (ms)
0.0
0.2
0.4
0.6
0.8
1.0
0
200
400
600
800
1000
US (Citadel) ⟷ Brazil
US (Citadel) ⟷ US (Rutgers 3)
Japan ⟷ US (Rutgers 3)
Mexico ⟷ US (Temple)
Hong Kong ⟷ US (Rutgers 1)


Packet Fluttering Distribution US (Citadel) ⟷ Brazil [Mahir Manir]
Frequency
Number of Fluttering Hops
US (Citadel)
Brazil
0
100
200
300
400
500
600
0
2
4
6
8
10
532
658
116
73
7
0
28
0
29
2
22
0
13
0
1
6
0
4
0
2
0
2


Packet Fluttering Distribution US (Citadel) ⟷ US (Rutgers 3) [Mahir Manir]
US (Citadel)
US (Rutgers 3)
Frequency
Number of Fluttering Hops
700
2
1
0
0
2
0
10
0
19
0
57
0
119
0
189
0
197
0
85
0
20


Packet Fluttering Distribution Japan ⟷ US (Rutgers 3) [Mahir Manir]
Japan
US (Rutgers 3)
Frequency
Number of Fluttering Hops
747
4
0
2
1
7
0
16
0
67
0
103
0
155
0
173
0
140
0
66
0
13
0
1





References

pandas - pandas.pydata.org

matplotlib - matplotlib.org

jupyter - jupyter.org

Work on this project is based off of analysis from [Paxson96] and tools provided by PlanetLab.