



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

TP2: Rutas en Internet

08 de junio de 2016

Teoría de las Comunicaciones

Integrante	LU	Correo electrónico
Benitti, Raul	592/08	raulbenitti@gmail.com
Castro, Damian	326/11	ltdicai@gmail.com
Lizana, Helen	118/08	hsle.22@gmail.com
Grenier, Michelle	418/10	michelle.grenier@hotmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

Índice

1. Introducción	3
2. Marco teórico	3
2.1. Protocolo de Mensajes de Control de Internet- ICMP	3
2.2. Traceroute	3
2.3. Round Trip Time	4
3. Herramientas	5
3.1. Traceroute: idea	5
3.2. Detección automática de enlaces intercontinentales	5
3.3. Implementación	6
4. Experimentos	6
5. Resultados	7
6. Experimento 1: África	8
6.1. Experimento 2: EEUU	11
6.2. Experimento 3: Japon	14
6.3. Experimento 4: Rusia	17
7. Análisis de los resultados	20
7.1. Traceroute	20
7.2. Detección de enlaces intercontinentales	20
8. Conclusiones	20

1. Introducción

Internet es posible gracias a un conjunto de miles de redes interconectadas entre sí. La conexión entre redes de distintos continentes se realiza por medio de cables submarinos capaces de transportar grandes volúmenes de datos por segundo. En este trabajo experimentaremos con herramientas y técnicas frecuentemente utilizadas para el análisis de redes.

A partir de experimentos cuatro universidades ubicadas en diferentes partes del mundo como destino, analizaremos que tan factible resulta utilizar los datos conseguidos mediante traceroute para detectar saltos intercontinentales en las rutas que por las que se envían los paquetes en Internet.

2. Marco teórico

2.1. Protocolo de Mensajes de Control de Internet- ICMP

Forma parte del conjunto de protocolos IP (RFC 792). Los mensajes ICMP son comúnmente generados en respuesta a errores en los datagramas de IP para diagnóstico y ruteo. Estos mensajes son construidos en el nivel de capa de red y se encuentran dentro de los paquetes de IP estándar. En esta oportunidad nos concentraremos en 3 tipos de paquetes ICMP:

- ECHO_REQUEST (tipo 8): los paquetes ECHO_REQUEST son utilizados para solicitar a un host que responda con un paquete ICMP ECHO_REPLY. Esto sirve para saber, por ejemplo, si un host es alcanzable.
- ECHO_REPLY (tipo 0): este tipo de paquete se envía al recibir un paquete ICMP ECHO_REQUEST.
- TIME_EXCEEDED (tipo 11): indica al host origen que un paquete IP agoto su tiempo de vida (Time-to-live, TTL) y fue descartado antes de alcanzar el host destino.

2.2. Traceroute

Es una herramienta de diagnóstico utilizada para el diagnóstico de redes. También sirve para caracterizar la ruta por la que los paquetes de nivel de red deben pasar antes de alcanzar su destino final. En su versión más simple, devuelve una lista ordenada de los host pertenecientes al camino, junto con mediciones del RTT para cada host.

Existen varias maneras de implementar Traceroute, y cuál utilizar depende de la tecnología subyacente disponible. En términos generales, existen dos maneras de implementar traceroute: utilizar los paquetes ICMP ECHO_REQUEST/TIME_EXCEEDED/ECHO_REPLY, o modificar alguno de los protocolos para que provean las características necesarias.

Las primeras consisten en enviar paquetes ICMP ECHO_REQUEST, incrementando progresivamente el campo Time-To-Live (conocido como TTL, que sirve para que un paquete no permanezca en la red de forma indefinida) hasta recibir un paquete ECHO_REPLY del host destino o superar un TTL máximo predefinido. Cuando un host intermedio recibe uno de los paquetes, decrementa el TTL de éste en uno y realiza una de siguientes dos acciones:

- si el TTL resultante es mayor a cero, se continúa con el envío del paquete hacia el host destino;
- si el TTL resultante es igual a cero, se cancela el envío al host destino y se responde un paquete de tipo TIME_EXCEEDED al host inicial

De esta manera, el host origen puede ir reconstruyendo la ruta a medida que recibe los paquetes TIME_EXCEEDED. Este mecanismo solo requiere que los hosts de la red implementen ICMP, pero el hecho de enviar cada paquete por separado puede prestarse a comportamientos anómalos y resultados engañosos.

El otro tipo de implementaciones intenta solucionar estos resultados erróneos agregando más capacidades a los protocolos. Si bien esto resulta en métodos más eficientes (pueden requerir enviar menos paquetes) y fiables (pueden definir una ruta concreta), las implementaciones suelen ser más complejas y dependen de que todos los hosts de la red posean sus stacks de protocolos actualizados (lo que suele ser falso).

2.3. Round Trip Time

El RTT es el tiempo que tarda un paquete en ir y volver desde un nodo A (el origen) a un nodo B (el destino) dentro de una red. Cuando se trata de enlaces punto a punto, se define como $2 * \text{Delay}$. Si bien a nivel de enlace puede realizarse una estimación relativamente confiable del Delay a partir de variables conocidas (ancho de banda, velocidad de propagación del medio, etc), a nivel de red el RTT de un paquete IP queda sujeto a la ruta que éste toma. Es decir, el RTT de un paquete que viaja entre varias redes interconectadas depende de variables desconocidas de los enlaces intermedios, y empiezan a cobrar mayor importancia factores como la congestión de los routers intermedios.

3. Herramientas

3.1. Traceroute: idea

Implementamos nuestra propia herramienta de traceroute siguiendo la técnica del envío de paquetes ICMP ECHO_REQUEST/TIME_EXCEEDED/ECHO_REPLY. Consideramos dos posibles implementaciones: el algoritmo estandar, que consiste en enviar para cada TTL una ráfaga de paquetes; y una modificación en la que se envíe un paquete por TTL hasta alcanzar el host destino o superar el límite de saltos y repetir desde el principio. Elegimos la primera por simplicidad de la implementación al momento de calcular el valor del RTT.

Un punto a considerar cuando se realiza traceroute con ICMP es que cada paquete puede seguir una ruta distinta a la recorrida por los demás (y las rutas pueden variar incluso entre la ida y la vuelta de un mismo paquete). Por lo tanto, para un TLL dado podríamos obtener respuestas de varios hosts distintos. Para lidiar con este problema, decidimos considerar solo la ruta más probable. Para esto, por cada rafaga de paquetes echo request consideramos como nodo del camino aquel que haya respondido la mayor cantidad de veces (al calcular la frecuencia descartamos los timetouts que hayan sucedido).

Existe otro detalle a resolver una vez que quedan determinados los host del camino: para cada host tenemos una muestra de RTTs que pueden variar considerablemente. Teniendo en cuenta que el objetivo de nuestra herramienta es estimar un camino con los valores esperados de RTT entre nodos, sopesamos varias alternativas para aplanar los datos. Entre ellas analizamos las siguientes:

- Menor RTT
- RTT Promedio
- RTT Promedio, quitando previamente los outliers de la muestra (con el método de Cimbala)
- Mediana de RTT

La herramienta calcula todas ellas a modo de comparación, pero para los análisis nos decidimos por utilizar el RTT promedio pre-filtrado, pues esperamos que resulte en valores significativos que no se vean afectados por datos espurios.

3.2. Detección automática de enlaces intercontinentales

Una vez determinado un camino y los RTT correspondientes, estamos en condiciones de comenzar el análisis para intentar detectar automáticamente los enlaces intercontinentales de larga distancia basandonos en la técnica de estimación de outliers propuesta por Cimbala. Para ello, obtenemos los RTT relativos entre hops consecutivos y aplicamos el algoritmo de Cimbala a fin de detectar outliers. Nuestra hipótesis es que los saltos que tomen mas tiempo, detectados con el metodo de outliers, son los posibles saltos intercontinentales.

Resulta importante considerar la posibilidad de que algunos hops no tengan definido su RTT. Esto puede suceder cuando, por ejemplo, el hop no implementa ICMP o se encuentra detrás de un firewall que bloqueaba este protocolo. Contemplamos la opción de interpolar estos faltantes, pero concluimos que la falta de información nos posibilita solo a aplicar un interpolado lineal que resultaría en información "suavizada" que juegue negativamente al momento de aplicar Cimbala. Por esto, decidimos utilizar solamente los hops con RTT definido: si detectamos un posible salto intercontinental y vemos que los hops no son consecutivos, al menos podemos deducir que el salto ocurre entre esos host.

3.3. Implementación

La herramienta fue desarrollada en Python utilizando el paquete `scapy`, y permite definir los siguientes parámetros de ejecución:

- `MAT_RAFA`: tamaño de la ráfaga para cada TTL
- `MAX_TTL`: cantidad máxima de saltos esperados
- `TIMEOUT`: tiempo de espera, medido en segundos
- `P` (Tolerancia a timeouts): cantidad de timeouts seguidos que se toleran antes de decidir que no hay respuesta, medido en porcentaje del tamaño de la ráfaga.
- `OUTPUT`: identificador para generar los nombres de los archivos de salida

El código se divide en tres funciones principales

- `rastrear()`: es la implementación del algoritmo de traceroute utilizando paquetes ICMP. Devuelve un muestreo de RTTs por TTL.
- `generar_camino()`: a partir del muestreo devuelto por `'rastrear'`, decide cual es el camino más probable y calcula los RTT correspondientes a cada hop.
- `detectar_enlaces_intercontinentales()`: dado un camino devuelto por `generar_camino()`, calcula los RTT entre hops y marca aquellos que pueden ser saltos intercontinentales mediante la detección de outlier según el algoritmo de Cimbala.

Para el mapeo de IP a País utilizamos una base de datos obtenida de MaxMind actualizada al 09/07/2016, accedida en el código por medio del paquete `geoip2` de Python. Descartamos otras fuentes de datos que, debido a que presentaban limitaciones de performance, no proveían una API o se encontraban desactualizadas.

4. Experimentos

A fin de probar el comportamiento del algoritmo propuesto, recolectamos los resultados de ejecutar el programa tomando como destinos a cuatro universidades en distintos continentes. Realizamos 3 corridas para cada destino, variando el tamaño de la ráfaga en 50, 150 y 300 paquetes. Los destinos elegidos fueron:

Universidad	Host	País	Contiente
Universidad de Oregon	<code>www.cs.uoregon.edu</code>	Estados Unidos	América del Norte
universidad de Tokio	<code>www.u-tokyo.ac.jp</code>	Japón	Asia
Universidad de Moscú	<code>msu.ru</code>	Rusia	Europa del Este
universidad de Sudáfrica	<code>www.unisa.ac.za</code>	Sudáfrica	África

Cuadro 1: Saltos y sus RTT en el camino desde Buenos Aires a la Universidad de Sudáfrica (África).

Exp	Destino	Comando
1	Sudáfrica	<code>sudo python traceroute.py www.unisa.ac.za -tr n -p 0.3 -m 30 -o AFR_n</code>
2	EEUU	<code>sudo python traceroute.py www.cs.uoregon.edu -tr n -p 0.3 -m 30 -o EEUU_n</code>
3	Japon	<code>sudo python traceroute.py www.u-tokyo.ac.jp -tr n -p 0.3 -m 30 -o JAP_n</code>
4	Rusia	<code>sudo python traceroute.py msu.ru -tr n -p 0.3 -m 30 -o RUSIA_n</code>

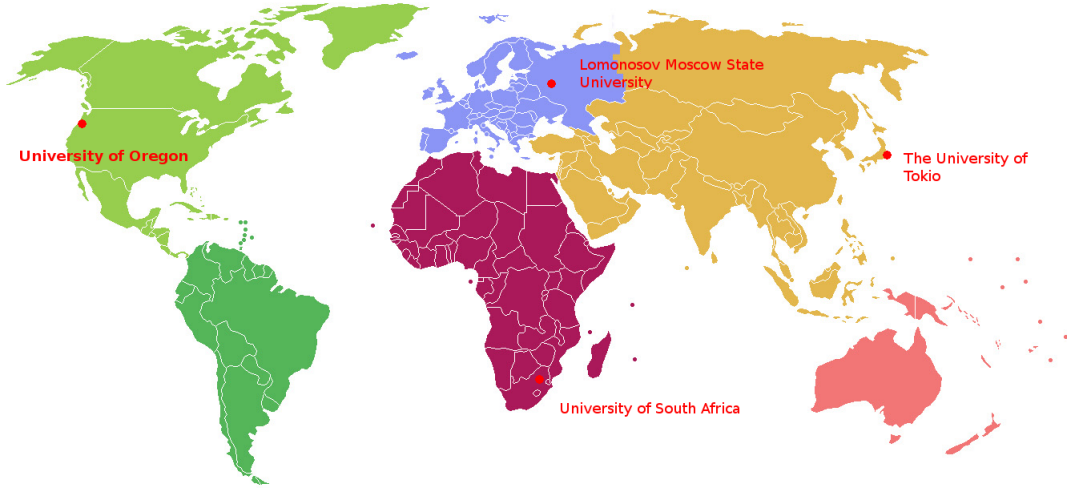


Figura 1: Ubicación de las universidades elegidas

Los experimentos se realizaron utilizando en una computadora con Linux Mint conectada a Internet (provisto por Fibertel) por medio de un enlace WiFi. Se utilizaron los siguientes comandos (donde n es el tamaño de la ráfaga) Notar que la herramienta debe ejecutarse con permisos de `root`.

5. Resultados

En las siguientes tablas mostramos los resultados obtenidos para distintos tamaños de ráfagas. Notar que solo estamos utilizando el RTT promedio, al que calculamos eliminando previamente los outliers. En las tablas de saltos se marcan entre corchetes los outliers según Cimbala.

6. Experimento 1: África

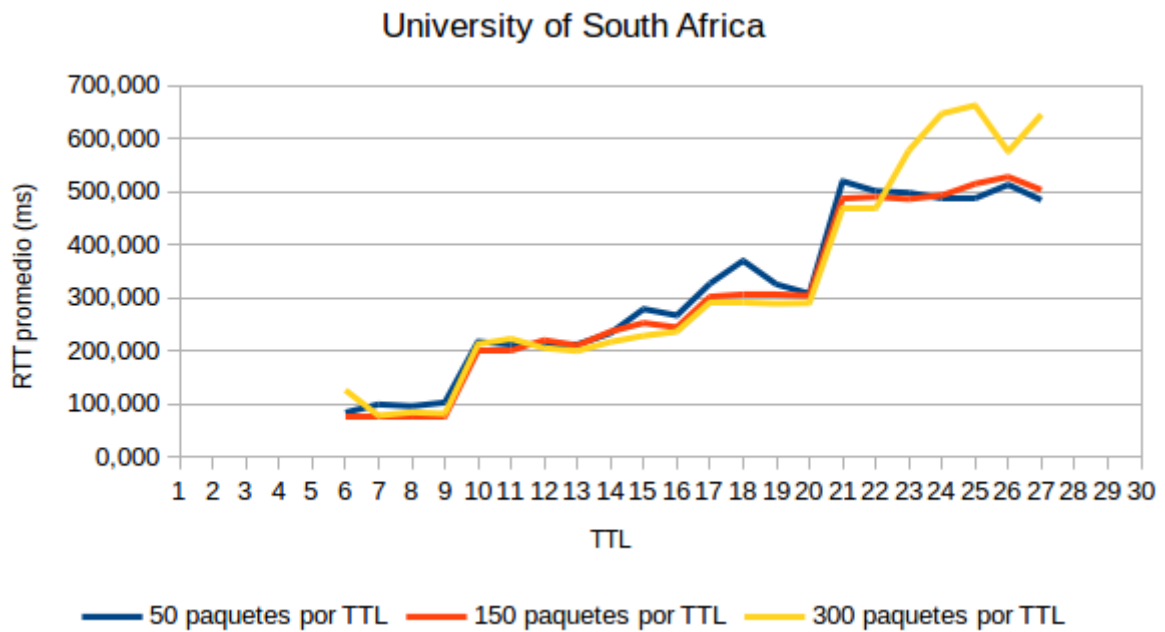
Cuadro 2: Camino estimado desde Buenos Aires a la Universidad de Sudáfrica (África).

TTL	IP	Pais	Rafaga = 50 RTT Filtrado Prom	Rafaga = 150 RTT Filtrado Prom	Rafaga = 300 RTT Filtrado Prom
1	192.168.0.1	Local	63.183 ms	63.66 ms	60.306 ms
2	None	Unknown	Unknown	Unknown	Unknown
3	None	Unknown	Unknown	Unknown	Unknown
4	None	Unknown	Unknown	Unknown	Unknown
5	None	Unknown	Unknown	Unknown	Unknown
6	200.89.164.129	Argentina	75.735 ms	69.777 ms	72.186 ms
7	200.89.165.130	Argentina	81.6 ms	69.322 ms	75.228 ms
8	200.89.165.222	Argentina	82.939 ms	69.796 ms	74.685 ms
9	195.22.220.172	Italy	72.147 ms	66.17 ms	72.982 ms
10	89.221.41.161	Italy	201.218 ms	197.541 ms	202.749 ms
11	89.221.41.161	Italy	199.258 ms	195.86 ms	203.352 ms
12	154.54.9.17	United States	201.047 ms	209.585 ms	203.779 ms
13	154.54.24.233	United States	198.495 ms	201.588 ms	197.17 ms
14	154.54.24.197	United States	219.718 ms	228.412 ms	211.934 ms
15	154.54.24.221	United States	224.876 ms	225.9 ms	221.758 ms
16	154.54.40.109	United States	241.303 ms	231.854 ms	228.356 ms
17	154.54.42.86	United States	294.135 ms	288.052 ms	287.978 ms
18	154.54.58.186	United States	298.166 ms	289.808 ms	287.868 ms
19	154.54.56.238	United States	297.178 ms	289.593 ms	287.225 ms
20	149.14.80.210	United States	286.71 ms	289.823 ms	287.583 ms
21	196.32.209.174	South Africa	472.497 ms	469.31 ms	466.299 ms
22	155.232.6.65	South Africa	480.99 ms	471.961 ms	467.52 ms
23	155.232.6.37	South Africa	470.571 ms	469.022 ms	465.979 ms
24	155.232.6.33	South Africa	475.67 ms	472.859 ms	470.095 ms
25	155.232.6.142	South Africa	471.279 ms	475.578 ms	464.099 ms
26	155.232.6.145	South Africa	498.626 ms	493.943 ms	498.355 ms
27	155.232.6.138	South Africa	463.824 ms	473.392 ms	511.219 ms
28	None	Unknown	Unknown	Unknown	Unknown
29	None	Unknown	Unknown	Unknown	Unknown
30	None	Unknown	Unknown	Unknown	Unknown

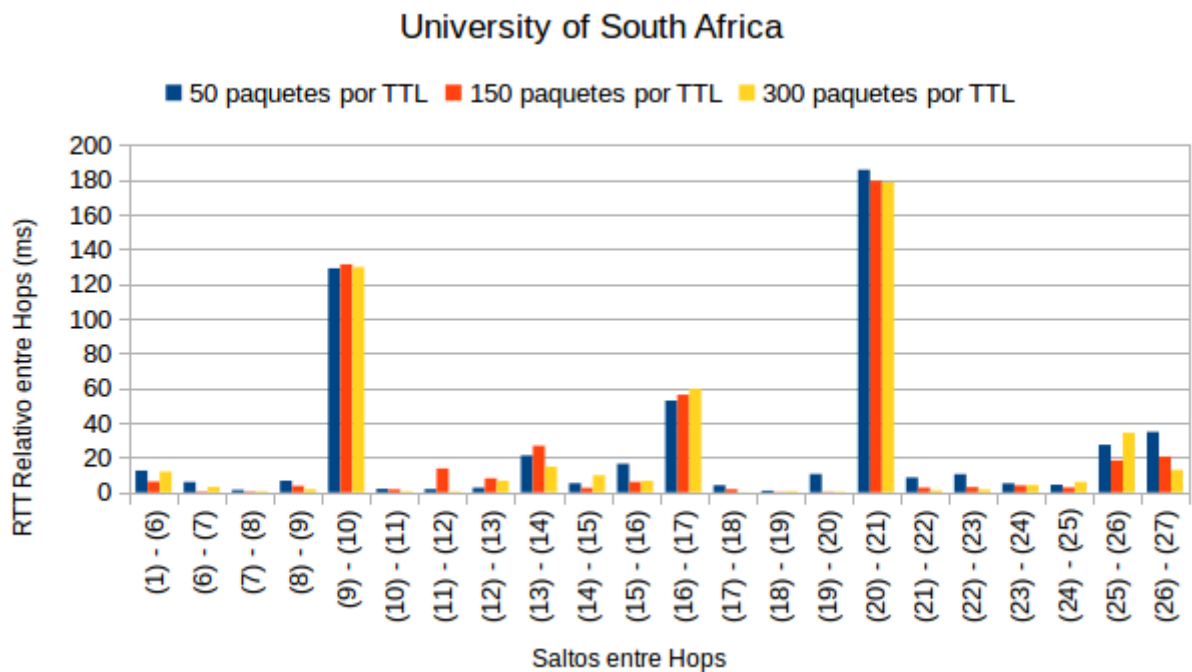
Cuadro 3: Saltos y sus RTT en el camino desde Buenos Aires a la Universidad de Sudáfrica (África).

Salto		50	150	300
Origen -> Destino		Promedio Filtrado	Promedio Filtrado	Promedio Filtrado
(1)	192.168.0.1 Local -> (6) 200.89.164.129 Argentina	12.552 ms	[6.118 ms]	[11.88 ms]
(6)	200.89.164.129 Argentina -> (7) 200.89.165.130 Argentina	5.865 ms	0.456 ms	3.043 ms
(7)	200.89.165.130 Argentina -> (8) 200.89.165.222 Argentina	1.339 ms	0.474 ms	0.544 ms
(8)	200.89.165.222 Argentina -> (9) 195.22.220.172 Italy	10.792 ms	3.626 ms	1.702 ms
(9)	195.22.220.172 Italy -> (10) 89.221.41.161 Italy	[129.071 ms]	[131.371 ms]	[129.767 ms]
(10)	89.221.41.161 Italy -> (11) 89.221.41.161 Italy	1.961 ms	1.68 ms	0.603 ms
(11)	89.221.41.161 Italy -> (12) 154.54.9.17 United States	1.789 ms	[13.724 ms]	0.427 ms
(12)	154.54.9.17 United States -> (13) 154.54.24.233 United States	2.552 ms	[7.997 ms]	6.609 ms
(13)	154.54.24.233 United States -> (14) 154.54.24.197 United States	[21.222 ms]	[26.824 ms]	[14.764 ms]
(14)	154.54.24.197 United States -> (15) 154.54.24.221 United States	5.158 ms	2.512 ms	[9.824 ms]
(15)	154.54.24.221 United States -> (16) 154.54.40.109 United States	[16.428 ms]	[5.954 ms]	6.598 ms
(16)	154.54.40.109 United States -> (17) 154.54.42.86 United States	[52.831 ms]	[56.199 ms]	[59.622 ms]
(17)	154.54.42.86 United States -> (18) 154.54.58.186 United States	4.031 ms	1.755 ms	0.111 ms
(18)	154.54.58.186 United States -> (19) 154.54.56.238 United States	0.988 ms	0.215 ms	0.643 ms
(19)	154.54.56.238 United States -> (20) 149.14.80.210 United States	10.468 ms	0.23 ms	0.359 ms
(20)	149.14.80.210 United States -> (21) 196.32.209.174 South Africa	[185.787 ms]	[179.487 ms]	[178.716 ms]
(21)	196.32.209.174 South Africa -> (22) 155.232.6.65 South Africa	8.493 ms	2.652 ms	1.22 ms
(22)	155.232.6.65 South Africa -> (23) 155.232.6.37 South Africa	10.419 ms	2.94 ms	1.54 ms
(23)	155.232.6.37 South Africa -> (24) 155.232.6.33 South Africa	5.099 ms	3.838 ms	4.115 ms
(24)	155.232.6.33 South Africa -> (25) 155.232.6.142 South Africa	4.391 ms	2.719 ms	5.996 ms
(25)	155.232.6.142 South Africa -> (26) 155.232.6.145 South Africa	[27.347 ms]	[18.365 ms]	[34.256 ms]
(26)	155.232.6.145 South Africa -> (27) 155.232.6.138 South Africa	[34.802 ms]	[20.551 ms]	[12.863 ms]

Figura 2: RTT estimado del traceroute a la Universidad de Sudáfrica



(a) RTT promedio por TTL



(b) RTT relativos entre saltos

6.1. Experimento 2: EEUU

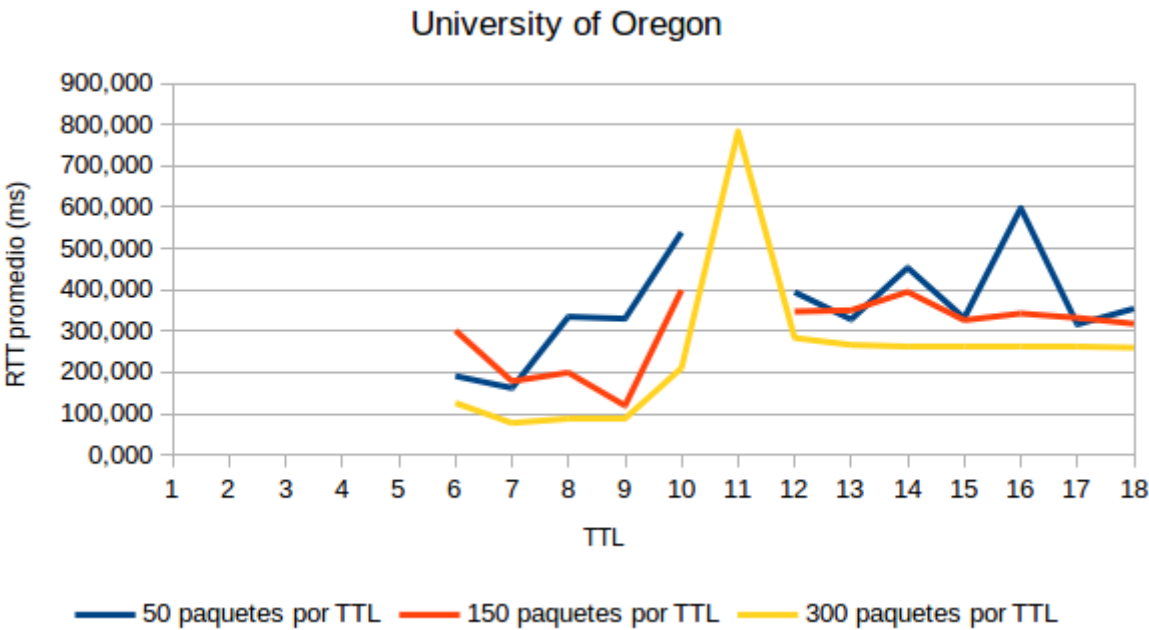
Cuadro 4: Camino estimado desde Buenos Aires a la Universidad de Oregon (America del Norte).

TTL	IP	Pais	50 RTT Filtrado Prom	150 RTT Filtrado Prom	300 RTT Filtrado Prom
1	192.168.0.1	Local	99.53 ms	103.514 ms	57.461 ms
2	None	Unknown	Unknown	Unknown	Unknown
3	None	Unknown	Unknown	Unknown	Unknown
4	None	Unknown	Unknown	Unknown	Unknown
5	None	Unknown	Unknown	Unknown	Unknown
6	200.89.165.141	Argentina	106.059 ms	102.341 ms	78.394 ms
7	200.89.165.130	Argentina	101.341 ms	103.905 ms	71.902 ms
8	200.89.165.222	Argentina	104.182 ms	103.432 ms	77.629 ms
9	190.216.88.33	Argentina	107.429 ms	101.819 ms	79.446 ms
10	67.17.94.249	United States	308.066 ms	303.806 ms	202.742 ms
11	4.68.72.66	United States	Unknown	Unknown	783.798 ms
12	4.69.132.149	United States	367.776 ms	305.802 ms	246.129 ms
13	4.69.132.149	United States	307.944 ms	307.181 ms	247.065 ms
14	4.53.200.2	United States	299.937 ms	309.681 ms	261.722 ms
15	207.98.64.165	United States	306.661 ms	305.197 ms	258.771 ms
16	207.98.68.182	United States	313.137 ms	306.078 ms	261.577 ms
17	128.223.2.1	United States	303.678 ms	308.079 ms	261.194 ms
18	128.223.4.25	United States	306.301 ms	306.542 ms	257.81 ms

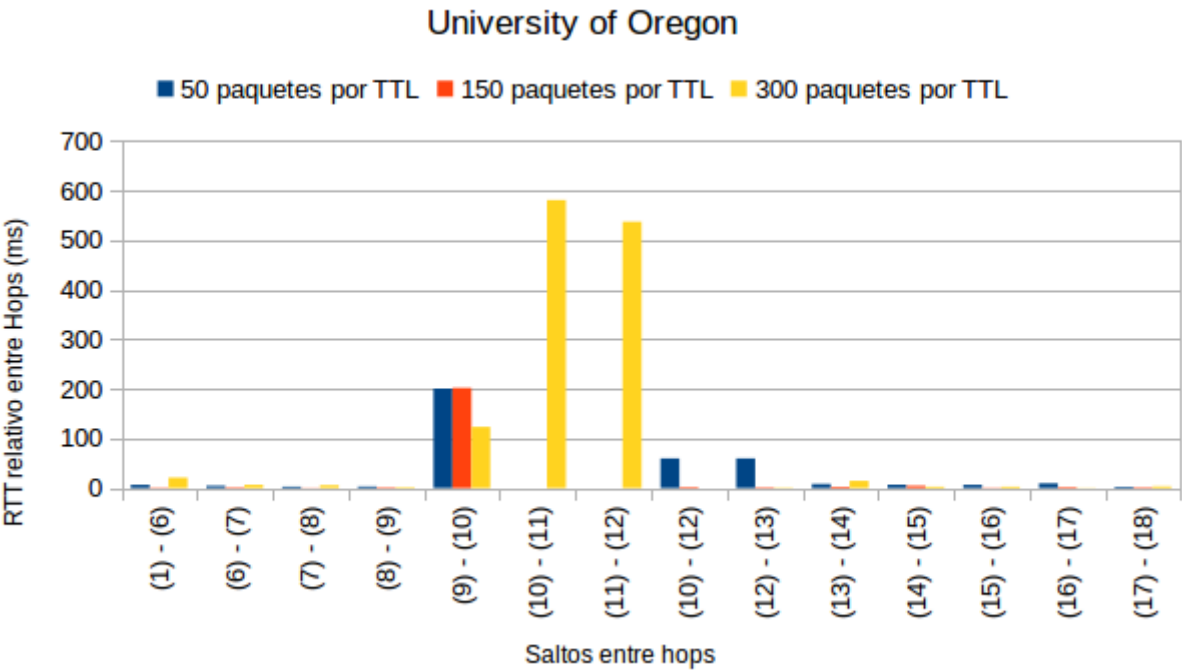
Cuadro 5: Saltos y sus RTT en el camino desde Buenos Aires a la Universidad de Oregon (Estados Unidos).

Salto		50	150	300
Origen -> Destino		Promedio Filtrado	Promedio Filtrado	Promedio Filtrado
(1)	192.168.0.1 Local -> (6) 200.89.165.141 Argentina	6.528 ms	1.173 ms	[20.933 ms]
(6)	200.89.165.141 Argentina -> (7) 200.89.165.130 Argentina	4.718 ms	1.564 ms	6.493 ms
(7)	200.89.165.130 Argentina -> (8) 200.89.165.222 Argentina	2.841 ms	0.472 ms	5.727 ms
(8)	200.89.165.222 Argentina -> (9) 190.216.88.33 Argentina	3.247 ms	1.613 ms	1.817 ms
(9)	190.216.88.33 Argentina -> (10) 67.17.94.249 United States	[200.638 ms]	[201.987 ms]	[123.297 ms]
(10)	67.17.94.249 United States -> (12) 4.69.132.149 United States	[59.71 ms]	1.996 ms	x
(10)	67.17.94.249 United States -> (11) 4.68.72.66 United States	x	x	[581.056 ms]
(11)	4.68.72.66 United States -> (12) 4.69.132.149 United States	x	x	[537.669 ms]
(12)	4.69.132.149 United States -> (13) 4.69.132.149 United States	[59.833 ms]	1.378 ms	0.935 ms
(13)	4.69.132.149 United States -> (14) 4.53.200.2 United States	8.006 ms	2.5 ms	[14.657 ms]
(14)	4.53.200.2 United States -> (15) 207.98.64.165 United States	6.724 ms	[4.484 ms]	2.951 ms
(15)	207.98.64.165 United States -> (16) 207.98.68.182 United States	6.477 ms	0.881 ms	2.807 ms
(16)	207.98.68.182 United States -> (17) 128.223.2.1 United States	9.46 ms	2.001 ms	0.384 ms
(17)	128.223.2.1 United States -> (18) 128.223.4.25 United States	2.623 ms	1.536 ms	3.384 ms

Figura 3: RTT estimado del traceroute a la Universidad de Oregon



(a) RTT promedio por TTL



(b) RTT relativos entre saltos

6.2. Experimento 3: Japon

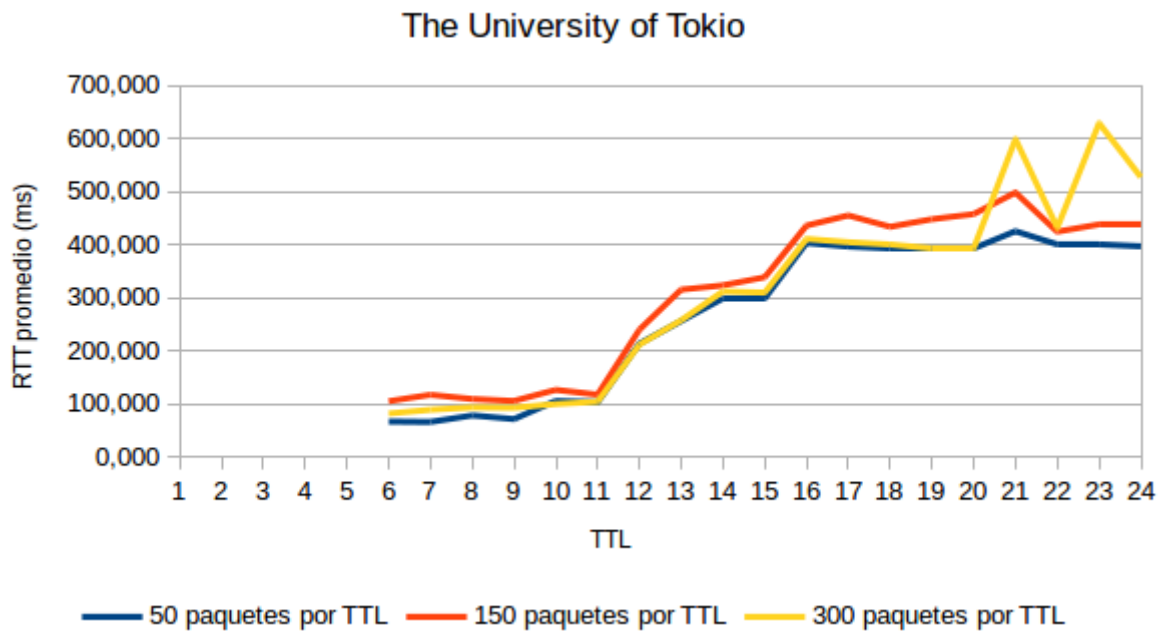
Cuadro 6: Camino estimado desde Buenos Aires a la Universidad de Tokio (Asia).

TTL	IP	Pais	50 RTT Filtrado Prom	150 RTT Filtrado Prom	300 RTT Filtrado Prom
1	192.168.0.1	Local	59.684 ms	62.26 ms	65.531 ms
2	None	Unknown	Unknown	Unknown	Unknown
3	None	Unknown	Unknown	Unknown	Unknown
4	None	Unknown	Unknown	Unknown	Unknown
5	None	Unknown	Unknown	Unknown	Unknown
6	200.89.164.129	Argentina	66.602 ms	80.867 ms	76.751 ms
7	200.89.165.5	Argentina	65.983 ms	85.396 ms	77.617 ms
8	200.89.165.250	Argentina	78.198 ms	81.414 ms	79.964 ms
9	185.70.203.56	Italy	71.299 ms	88.681 ms	75.557 ms
10	195.22.219.3	Italy	105.539 ms	105.86 ms	98.037 ms
11	195.22.219.3	Italy	103.2 ms	100.989 ms	99.468 ms
12	149.3.181.65	Italy	213.327 ms	217.664 ms	209.684 ms
13	129.250.2.227	United States	255.97 ms	261.183 ms	254.077 ms
14	129.250.4.13	United States	297.467 ms	310.539 ms	303.424 ms
15	129.250.2.54	United States	298.115 ms	307.844 ms	303.238 ms
16	129.250.3.86	United States	403.231 ms	418.358 ms	406.681 ms
17	129.250.6.188	United States	396.553 ms	416.043 ms	400.489 ms
18	129.250.2.255	United States	394.784 ms	404.056 ms	398.441 ms
19	61.200.80.218	Japan	392.899 ms	409.514 ms	391.58 ms
20	158205192173	Japan	391.804 ms	412.988 ms	393.182 ms
21	158.205.192.86	Japan	425.468 ms	438.572 ms	426.033 ms
22	158205121250	Japan	401.164 ms	424.37 ms	405.547 ms
23	154.34.240.254	Japan	400.949 ms	415.033 ms	572.7 ms
24	210152135178	Japan	397.01 ms	416.837 ms	401.609 ms

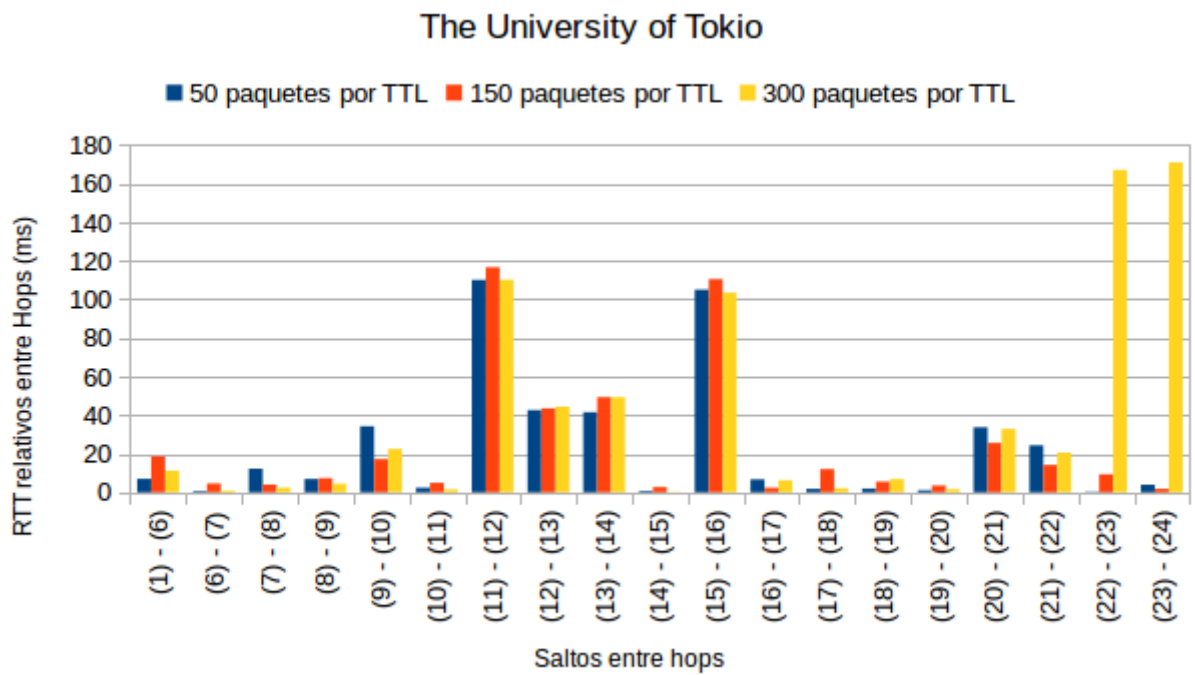
Cuadro 7: Saltos y sus RTT en el camino desde Buenos Aires a la Universidad de Tokio (Asia).

Salto		50		150		300	
Origen -> Destino		Promedio Filtrado		Promedio Filtrado		Promedio Filtrado	
(1)	192.168.0.1 Local -> (6) 200.89.164.129 Argentina	6.918 ms		[18.606 ms]		[11.22 ms]	
(6)	200.89.164.129 Argentina -> (7) 200.89.165.5 Argentina	0.619 ms		4.529 ms		0.867 ms	
(7)	200.89.165.5 Argentina -> (8) 200.89.165.250 Argentina	[12.215 ms]		3.982 ms		2.347 ms	
(8)	200.89.165.250 Argentina -> (9) 185.70.203.56 Italy	6.9 ms		[7.267 ms]		4.407 ms	
(9)	185.70.203.56 Italy -> (10) 195.22.219.3 Italy	[34.241 ms]		[17.178 ms]		[22.48 ms]	
(10)	195.22.219.3 Italy -> (11) 195.22.219.3 Italy	2.339 ms		4.87 ms		1.431 ms	
(11)	195.22.219.3 Italy -> (12) 149.3.181.65 Italy	[110.127 ms]		[116.675 ms]		[110.216 ms]	
(12)	149.3.181.65 Italy -> (13) 129.250.2.227 United States	[42.643 ms]		[43.519 ms]		[44.393 ms]	
(13)	129.250.2.227 United States -> (14) 129.250.4.13 United States	[41.497 ms]		[49.356 ms]		[49.347 ms]	
(14)	129.250.4.13 United States -> (15) 129.250.2.54 United States	0.648 ms		2.695 ms		0.186 ms	
(15)	129.250.2.54 United States -> (16) 129.250.3.86 United States	[105.116 ms]		[110.514 ms]		[103.443 ms]	
(16)	129.250.3.86 United States -> (17) 129.250.6.188 United States	6.678 ms		2.315 ms		6.192 ms	
(17)	129.250.6.188 United States -> (18) 129.250.2.255 United States	1.769 ms		[11.987 ms]		2.048 ms	
(18)	129.250.2.255 United States -> (19) 61.200.80.218 Japan	1.885 ms		5.458 ms		6.861 ms	
(19)	61.200.80.218 Japan -> (20) 158.205.192.173 Japan	1.095 ms		3.474 ms		1.603 ms	
(20)	158.205.192.173 Japan -> (21) 158.205.192.86 Japan	[33.664 ms]		[25.584 ms]		[32.85 ms]	
(21)	158.205.192.86 Japan -> (22) 158.205.121.250 Japan	[24.304 ms]		[14.202 ms]		[20.485 ms]	
(22)	158.205.121.250 Japan -> (23) 154.34.240.254 Japan	0.215 ms		[9.337 ms]		[167.152 ms]	
(23)	154.34.240.254 Japan -> (24) 210.152.135.178 Japan	3.939 ms		1.805 ms		[171.091 ms]	

Figura 4: RTT promedio de traceroute a la Universidad de Tokio



(a) RTT promedio por TTL



(b) RTT relativos entre saltos

6.3. Experimento 4: Rusia

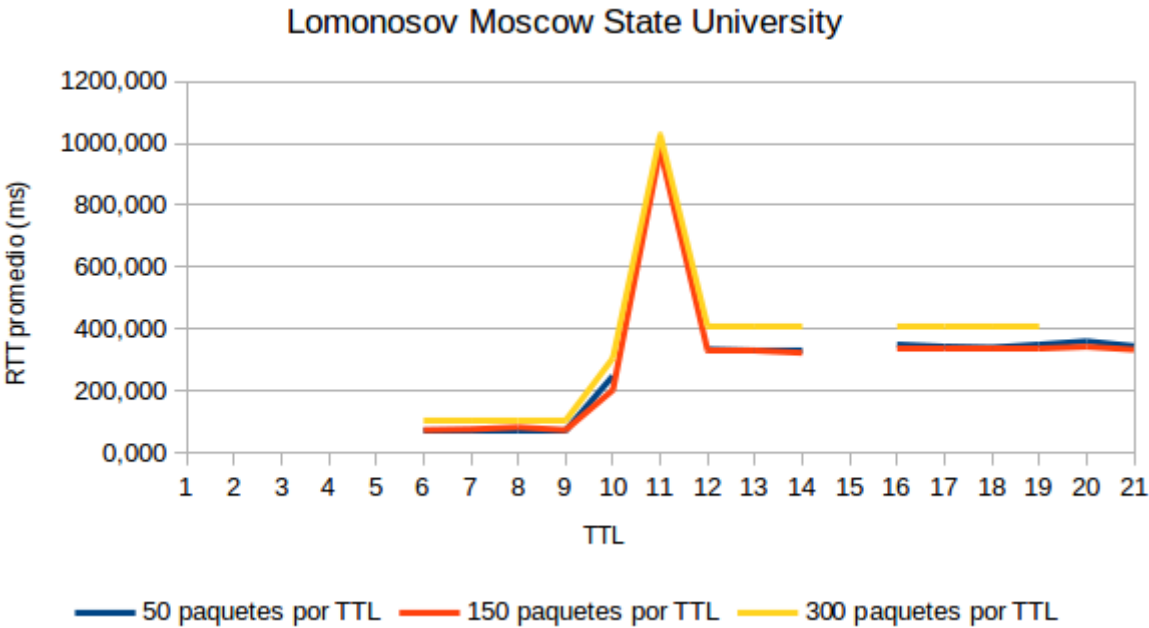
Cuadro 8: Camino estimado desde Buenos Aires a la Universidad de Moscú (Europa del este).

TTL	IP	Pais	50 RTT Filtrado Prom	150 RTT Filtrado Prom	300 RTT Filtrado Prom
1	192.168.0.1	Local	66.082 ms	101.009 ms	62.141 ms
2	None	Unknown	Unknown	Unknown	Unknown
3	None	Unknown	Unknown	Unknown	Unknown
4	None	Unknown	Unknown	Unknown	Unknown
5	None	Unknown	Unknown	Unknown	Unknown
6	200.89.164.141	Argentina	67.789 ms	103.426 ms	74.055 ms
7	200.89.165.130	Argentina	63.79 ms	101.435 ms	76.409 ms
8	200.89.165.222	Argentina	64.843 ms	103.571 ms	83.03 ms
9	190.216.88.33	Argentina	68.088 ms	101.673 ms	73.911 ms
10	67.17.99.233	United States	237.063 ms	306.126 ms	203.308 ms
11	4.68.72.66	United States	Unknown	1027.876 ms	980.007 ms
12	4.69.158.253	United States	326.02 ms	407.603 ms	327.569 ms
13	4.69.158.253	United States	325.194 ms	408.104 ms	327.74 ms
14	213242110198	United Kingdom	325.508 ms	408.248 ms	323.149 ms
15	None	Unknown	Unknown	Unknown	Unknown
16	194.85.40.229	Russia	338.62 ms	410.191 ms	336.288 ms
17	194190254118	Russia	336.555 ms	410.227 ms	335.396 ms
18	93.180.0.172	Russia	339.032 ms	409.307 ms	335.718 ms
19	188.44.33.30	Russia	341.636 ms	409.164 ms	337.758 ms
20	188.44.33.2	Russia	346.137 ms	408.97 ms	343.093 ms
21	188.44.50.103	Russia	340.466 ms	410.042 ms	333.481 ms

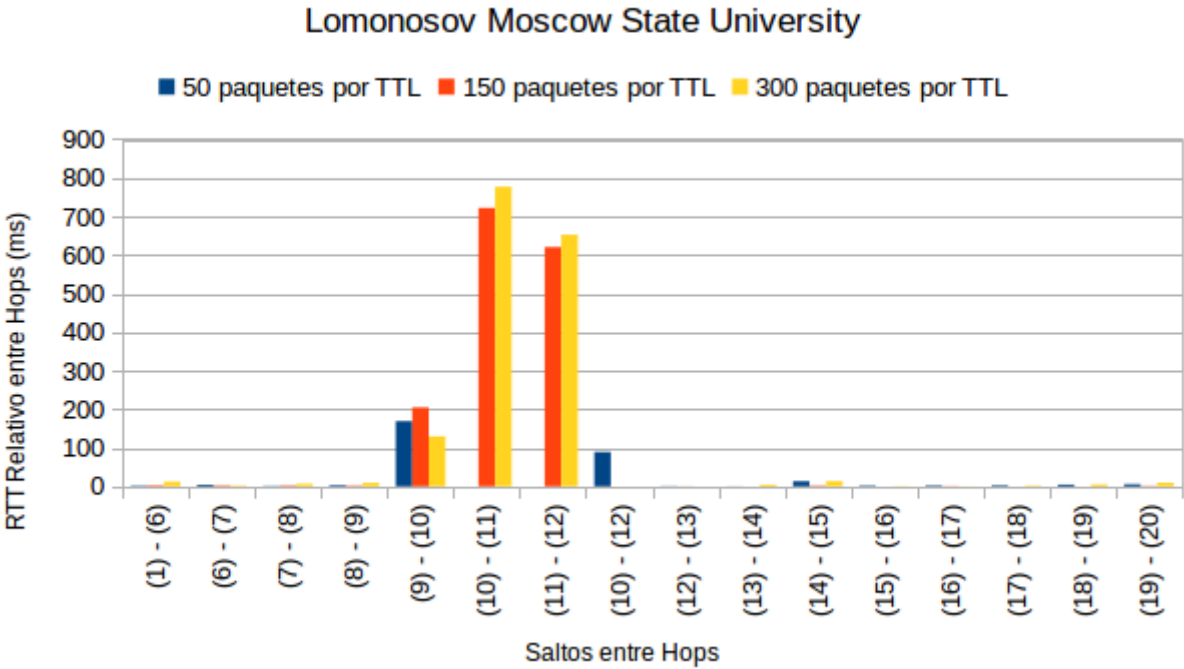
Cuadro 9: Saltos y sus RTT en el camino desde Buenos Aires a la Universidad de Moscú (Europa del Este).

Salto		50	150	300
Origen -> Destino		Promedio Filtrado	Promedio Filtrado	Promedio Filtrado
(1)	192.168.0.1 Local -> (6) 200.89.164.141 Argentina	1.707 ms	2.416 ms	11.913 ms
(6)	200.89.164.141 Argentina -> (7) 200.89.165.130 Argentina	3.999 ms	1.99 ms	2.354 ms
(7)	200.89.165.130 Argentina -> (8) 200.89.165.222 Argentina	1.053 ms	2.136 ms	6.621 ms
(8)	200.89.165.222 Argentina -> (9) 190.216.88.33 Argentina	3.245 ms	1.898 ms	9.119 ms
(9)	190.216.88.33 Argentina -> (10) 67.17.99.233 United States	[168.974 ms]	[204.453 ms]	[129.397 ms]
(10)	67.17.99.233 United States -> (12) 4.69.158.253 United States	[88.958 ms]	x	x
(10)	67.17.99.233 United States -> (11) 4.68.72.66 United States	x	[721.75 ms]	[776.699 ms]
(11)	4.68.72.66 United States -> (12) 4.69.158.253 United States	x	[620.273 ms]	[652.438 ms]
(12)	4.69.158.253 United States -> (13) 4.69.158.253 United States	0.827 ms	0.502 ms	0.172 ms
(13)	4.69.158.253 United States -> (14) 213.242.110.198 United Kingdom	0.314 ms	0.143 ms	4.592 ms
(14)	213.242.110.198 United Kingdom -> (16) 194.85.40.229 Russia	[13.113 ms]	1.943 ms	13.139 ms
(16)	194.85.40.229 Russia -> (17) 194.190.254.118 Russia	2.065 ms	0.035 ms	0.893 ms
(17)	194.190.254.118 Russia -> (18) 93.180.0.172 Russia	2.477 ms	0.919 ms	0.323 ms
(18)	93.180.0.172 Russia -> (19) 188.44.33.30 Russia	2.604 ms	0.143 ms	2.04 ms
(19)	188.44.33.30 Russia -> (20) 188.44.33.2 Russia	4.501 ms	0.195 ms	5.335 ms
(20)	188.44.33.2 Russia -> (21) 188.44.50.103 Russia	[5.671 ms]	1.072 ms	9.612 ms

Figura 5: RTT promedio del traceroute a la Universidad de Moscú



(a) RTT promedio por TTL



(b) RTT relativos entre saltos

7. Análisis de los resultados

7.1. Traceroute

Si bien todos los tamaños de ráfagas devuelven caminos con igual cantidad de hops, a veces algunos host no llegan a responder. Por ejemplo, el host con IP 4.68.72.66 (hop 11 tanto del camino a EEUU como a Rusia) parece no poder detectarse a menos que la ráfaga sea grande. Suponemos que el umbral de timeout que estamos utilizando puede ser demasiado bajo, por lo que no se alcanza a recibir una respuesta del host. Ahora, puede ser que el motivo por el cual este host no responda y sucedan timeouts suficientes como para descartar el hop sea que se están siguiendo caminos alternativos que no pasan por él, sino por otro nodo que ignora los paquetes ECHO_REQUEST.

También observamos la ocurrencia de comportamientos anómalos. Por ejemplo, en el camino a Rusia podemos ver que los hops 12 y 13 corresponden a la misma IP. Esto puede deberse a que se atraviesa una ruta MPLS que delega en ese host la tarea de enviar los mensajes TIME_EXCEEDED, o a que se toman caminos alternativos que terminan en ese mismo host para ambos TTLs. Dado que los RTT son muy similares para ambos hops, creemos que la primera explicación es mucho más probable.

http://www.net.in.tum.de/fileadmin/TUM/NET/NET-2012-08-1/NET-2012-08-1_02.pdf.

7.2. Detección automática de enlaces intercontinentales

En primer lugar, encontramos que el algoritmo de detección automática generalmente marcó más enlaces como posibles intercontinentales de lo que esperábamos. Analizando las tablas de saltos y sus respectivos gráficos, notamos que muchas veces se marcan como outlier variaciones que resultan poco significativas en comparación con los valores más altos. Aunque Cimbala considera que estos valores se alejan de lo esperado, parecen deberse más a factores como encolamiento o procesamiento en los hops que a enlaces de larga distancia. Este comportamiento se observó en todos los experimentos. Por ejemplo, en la ruta a Sudáfrica, entre los hops 12 y 20 se marcaron hasta 5 posibles saltos intercontinentales, pero todos pertenecen a una misma red y se encuentran, según la base de datos de geolocalización, dentro de Estados Unidos.

También observamos que el tamaño de las ráfagas hace variar mucho los RTTs, y por ende, los outliers encontrados. Esto lo podemos ver en las tablas de saltos entre hops, donde para distintos tamaños de ráfaga para un mismo salto a veces se considera outlier y otras veces no. Por ejemplo, para el experimento de Tokio, el RTT entre los hops 7 y 8, varía entre 2 y 12 ms, y se marca uno como outlier uno mientras que el otro no.

hay enlaces intercontinentales que son mucho más rápidos de lo que esperábamos (y cimbala no los detecta)

8. Conclusiones

Existen muchos factores que influyen en los tiempos de respuesta (el tipo de conexión a internet, la velocidad de internet, el ruteo de cada paquete, la congestión, los routers MPLS). Incluso usando una implementación de traceroute que mantenga un único camino, problemas como congestión pueden el resultado (si solo se miran los tiempos). Mirar tiempo + IP del hop da una mejor idea (por ejemplo, se puede desestimar saltos dentro de una misma red).

Queda probar considerar los saltos entre subredes, en lugar de entre hops. El servicio de geolocalización

puede ser erroneo.

Creemos que el metodo de Cimbala es demasiado sensible Suponemos que si los datos fueran más estables, el método de Cimbala podria ser más efectivo

Más tamaño de ráfaga tambien es mas riesgoso, porque hay mas chances de que varios paquetes sigan caminos distintos, haya mas varianza y el prefiltrado no funcione.