Departamento de Computación,
Facultad de Ciencias Exactas y Naturales,
Universidad de Buenos Aires

Trabajo Práctico 2

Teoría de las Comunicaciones

Primer Cuatrimestre de 2014

Apellido y Nombre	LU	E-mail
Delgado, Alejandro N.	601/11	nahueldelgado@gmail.com
Lovisolo, Leandro	645/11	leandro@leandro.me
Petaccio, Lautaro José	443/11	lausuper@gmail.com

Índice

1. Introducción

Presentamos una heurística y su análisis de efectividad para realizar la detección de enlaces submarinos en la traza de rutas entre dos hosts conectados a internet, utilizando los datos estadísticos resultantes del round trip time (RTT) conseguidos mediante una implementación propia de la herramienta traceroute, comunmente encontrada en los SO.

Para el análisis estadístico, los RTT son estudiados como el *z-score* o valor standard (ZRTT) respecto a las variaciones de los valores RTT promedios entre dos nodos continuos.

El método propuesto propone, en base a los datos estadísticos, la utilización de un umbral para la identificación de enlaces submarinos según los ZRTT relativos obtenidos para cada enlace. El uso de los ZRTT relativos presenta una manera detallada de identificar variaciones de tiempo entre enlaces, pudiendo identificar nodos con una diferencia de RTT mayor al promedio, como tendría un enlace submarino debido a la distancia que recorren los datos.

2. Desarrollo

Se implementó una herramienta en lenguaje Python para medir el *round-trip time* (RTT) hacia el host destino y cada hop intermedio durante una cantidad de tiempo determinada por el usuario. La herramienta se basa en el protocolo ICMP [2] tanto para descubrir la cantidad de hops y los gateways en cada hop hacia el host destino, como para medir los RTT hacia el host destino y cada hop intermedio (de manera similar a las herramientas `traceroute` [3] y `ping` [4], respectivamente.)

La herramienta hace uso la biblioteca Scapy [1] para la creación y comunicación de paquetes ICMP.

2.1. Medición de RTT hacia el host destino o un hop intermedio

Una medición consiste en enviar un paquete ICMP de tipo Echo Request al host destino, asignándole al paquete algún *time-to-live* (TTL) entre 1 y 30 inclusive[1], y tomar el tiempo que transcurre desde que se envía el paquete hasta que se recibe una respuesta. Las respuestas usualmente son de alguno de los siguientes tipos[2]:

- Un paquete ICMP de tipo Echo Reply en caso que el paquete emitido alcanzara el host destino, ó
- Un paquete ICMP de tipo Time Exceeded en caso que el paquete agotara su TTL antes de llegar al host destino.

Para medir el RTT hacia el host destino basta con enviarle un paquete a dicho host con un TTL lo suficientemente grande para asegurar que su TTL no se agote durante el envío del paquete, esperar hasta recibir un paquete ICMP de tipo Echo Reply proveniente del host destino y registrar el tiempo transcurrido.

Para medir el RTT hacia un hop intermedio i en la ruta al host destino, se le asigna al paquete un TTL de valor i. Esto produce que el paquete agote su TTL al llegar a un gateway en el i-éstimo hop, a lo cual éste responde con un paquete ICMP de tipo Time Exceeded. Finalmente se registra el tiempo transcurrido.

Tanto cuando se mide el RTT hacia el host destino o hacia un hop intermedio puede ocurrir que no se reciba ninguna respuesta, por ejemplo cuando el host o algún gateway está detrás de un firewall que bloquea el protocolo ICMP. Para evitar quedar esperando una respuesta durante una cantidad de tiempo indefinida, la herramienta descarta la medición si al cabo de un segundo no se recibió una respuesta.

[1] Se eligió 30 como máximo TTL posible porque es el valor de TTL máximo utilizado por defecto por la herramienta `traceroute`. Suponemos que la mayoría de las rutas en internet entre dos hosts cualquiera tienen menos de 30 hops, lo que hace adecuado dicho límite a los efectos de este trabajo.

[2] Es posible recibir respuestas de otros tipos, como por ejemplo paquetes ICMP de tipo Destination Unreachable en el caso que no se haya podido despachar el paquete a su destino por algún motivo, pero la herramienta ignora cualquier respuesta que no sea de los dos tipos mencionados.

2.2. Realizando múltiples mediciones en paralelo

Con el objetivo de maximizar el número total de mediciones realizadas y distribuir el impacto de picos de retraso en la conexión a internet entre las mediciones de RTT hacia todos los hops, la herramienta realiza mediciones hacia todos los hops de forma simultánea.

Las mediciones se hacen por baches: en un determinado momento se envían 30 paquetes al host destino, uno por cada TTL entre 1 y 30 y todos con TTL distinto, y se espera o bien hasta recibir las respuestas de todos los paquetes enviados, o bien hasta que transcurra un segundo y se den por perdidas las mediciones para las que no se recibieron respuestas. A continuación se registra el RTT hacia cada hop computando la diferencia entre el tiempo de recepción de una respuesta y el tiempo de envío del paquete de tipo Echo Request que la originó. Luego de esto se procede al siguiente bache de mediciones, o alternativamente, la herramienta termina su ejecución en caso de haber excedido el límite de tiempo de medición determinado por el usuario.

Para poder distinguir qué paquete produjo cada respuesta recibida, la herramienta le asigna un identificador único a cada paquete ICMP de tipo Echo Request emitido usando el campo *Identifier* (figura 1.) En la sección 2.2.2 se detalla el procedimiento implementado en la herramienta para obtener el valor del campo *Identifier* de un paquete ICMP de tipo Echo Request a partir de una respuesta de tipo Echo Reply o Time Exceeded.

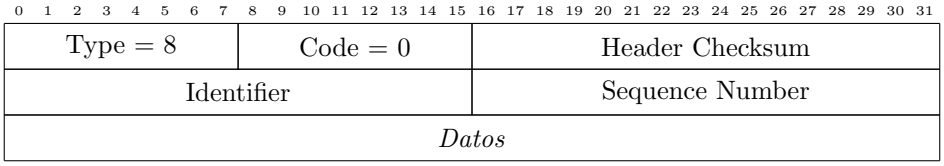

Figura 1: Paquete ICMP de tipo Echo Request

2.2.1. Construcción de un identificador único para cada paquete emitido

El identificador único se construye de forma tal que el TTL del paquete esté codificado dentro del identificador, y si se tiene un identificador cuyo paquete asociado se desconoce, sea posible deducir su TTL a partir del identificador. La siguiente es la fórmula utilizada por la herramienta para construir un identificador único dado el número de bache (comenzando desde 1) y el TTL del paquete:

$$\mathit{identificador}(\mathit{bache}, \mathit{ttl}) = 30 \times (\mathit{bache} - 1) + \mathit{ttl}$$

Luego para el caso en el que se tiene un identificador y se desea obtener el TTL del paquete asociado, se puede aplicar siguiente fórmula:

$$\mathit{ttl}(\mathit{identificador}) = (\mathit{identificador} - 1 \text{ mód } 30) + 1$$

Entonces por ejemplo, para el caso en el que $\mathit{bache} = 3$ y $\mathit{ttl} = 15$, se tiene que $\mathit{identificador}(\mathit{bache}, \mathit{ttl}) = \mathit{identificador}(3, 15) = 30 \times (3-1) + 15 = 75$. Luego para obtener el TTL a partir del identificador obtenido aplicando la segunda fórmula, se tiene que $\mathit{ttl}(75) = (75 - 1 \text{ mód } 30) + 1 = (74 \text{ mód } 30) + 1 = 14 + 1 = 15$.

2.2.2. Obteniendo el valor del campo *Identifier* del paquete que originó una respuesta

En el caso en que un paquete ICMP de tipo Echo Request haya llegado al host destino, éste contesta enviando un paquete ICMP de tipo Echo Reply (figura 2.) Este paquete también tiene un campo *Identifier*, que conserva el valor del mismo campo en el paquete ICMP de tipo Echo Request que lo originó.

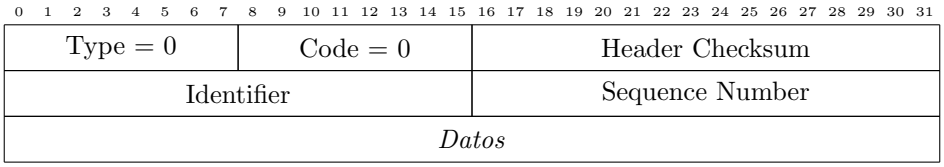

Figura 2: Paquete ICMP de tipo Echo Reply

Cuando un paquete (no necesariamente ICMP) agota su TTL antes de llegar al host destino, el último gateway al que llegó dicho paquete envía al host origen un paquete ICMP de tipo Time Exceeded (figura 3.) Éste paquete incluye el header IP y los primeros 8 bytes de datos del datagrama que agotó su TTL.

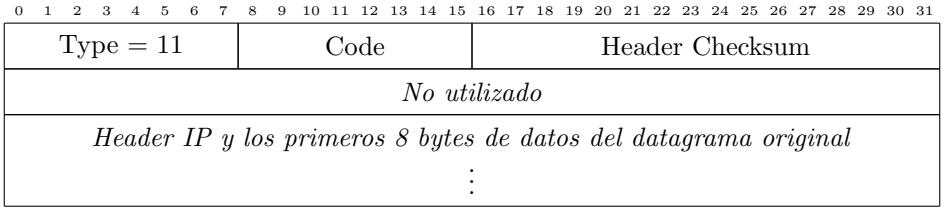

Figura 3: Paquete ICMP de tipo Time Exceeded

En particular, cuando el paquete que agotó su TTL es un paquete ICMP de tipo Echo Request, su header ICMP completo se incluye como parte de los 8 bytes de datos del datagrama original, del cual se puede extraer el valor del campo *Identifier* (ver figura 4.)

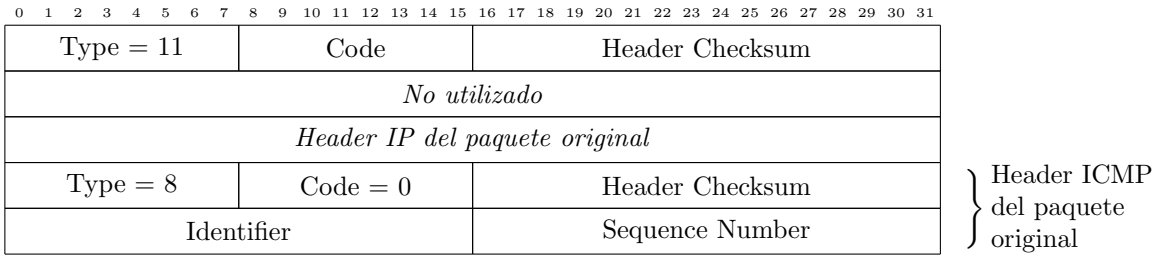

Figura 4: Paquete ICMP de tipo Time Exceeded como respuesta a otro paquete ICMP de tipo Echo Request

La herramienta entonces recibe paquetes ICMP de tipo Echo Reply o Time Exceeded, y para cada paquete, extrae el valor del campo *Identifier*, que coincide con el valor del mismo campo en el paquete ICMP de tipo Echo Request que lo originó. A partir del valor del campo *Identifier*, se obtiene el TTL del paquete que originó la respuesta y se almacena el RTT medido junto al resto de las mediciones del hop cuyo número coincide con dicho TTL.

2.3. Registro de mediciones y estadísticas computadas

Al finalizar la ejecución, la herramienta opcionalmente guarda a disco todas las respuestas recibidas que no fueron descartadas, junto al RTT medido para cada respuesta. En concreto, para cada respuesta recibida, se guardan los siguientes datos: TTL del paquete que la originó, IP del host que emitió la respuesta, tipo de la respuesta (valor del campo *Type* del header ICMP) y RTT expresado en milisegundos.

Junto a la herramienta desarrollada se provee una utilidad para leer los datos guardados a disco y generar estadísticas. Las estadísticas generadas son, para cada hop, RTT promedio (el RTT promedio de todos los paquetes recibidos provenientes de ese hop) y ZRTT.

La salida de dicha herramienta se incluye en la sección 3.

2.4. Experimentos realizados

Se eligieron como hosts destino los servidores web de tres universidades ubicadas en continentes distintos entre sí y respecto del continente desde el que se realizaron las mediciones, con la esperanza de atravezar uno o más enlaces submarinos distintos en cada traza obtenida. Las mediciones se realizaron desde Buenos Aires, Argentina, y las universidades elegidas fueron University of Oxford, The University of Sydney y Malaysia University of Science and Technology (Europa, Oceanía y Asia, respectivamente.)

Una vez determinados los hosts destino, se hicieron experimentos con versiones anteriores (y más simples) de la herramienta desarrollada. Tras cada experimento se modificó la herramienta para realizar mediciones más precisas, reflejando las conclusiones obtenidas en los experimentos anteriores. Este proceso se repitió hasta converger en la herramienta presentada en este trabajo.

2.4.1. Experimento 1: múltiples mediciones en paralelo, un único hop por vez

En esta primera iteración de la herramienta se enviaban simultáneamente 100 paquetes ICMP de tipo Echo Request al host destino con TTL 1 y se esperaba hasta recibir las 100 respuestas correspondientes, o bien hasta que transcurriera un segundo desde que se enviaron los paquetes. En este último caso, se descartan las mediciones correspondientes a los paquetes para los que no se recibieron respuestas. Luego se repite este proceso para los valores de TTL de 2 a 30 inclusive.

Esta técnica resultó estar muy sujeta a la congestión de la red en el momento que se tomaron las mediciones. Por ejemplo, si en el instante que se envían los paquetes para el i-ésimo TTL la conexión a internet de la computadora desde la que se realiza la medición sufre una congestión, pero la conexión se normaliza para el instante en el que se envían los paquetes para el $(i+1)$-ésimo TTL, es posible que el RTT promedio para el i-ésimo hop resulte muy superior al del $(i+1)$-ésimo hop. Esta anomalía puede producir un falso positivo en la etapa de detección de enlaces submarinos más adelante en el análisis.

Luego de repetir varias veces las mediciones para cada host destino y observar resultados muy distintos entre medición y medición para un mismo destino, se decidió modificar la herramienta de manera de esparcir uniformemente en el tiempo las mediciones para cada TTL, con la esperanza de suavizar las anomalías producidas por variaciones en la carga de la conexión a internet.

2.4.2. Experimento 2: múltiples mediciones en paralelo, todos los hops a la vez durante 1 minuto, identificadores de paquetes asignados al azar

Para este experimento se modificó la herramienta de forma de aplicar el procedimiento de envío de paquetes por baches a todos los hops en simultáneo descrito en la sección 2.2, pero con la diferencia que se le asignó a cada paquete un identificador elegido al azar en el intervalo $[0, 65535]$ equiprobablemente, y cada bache mantenía una tabla de identificadores y TTL del paquete asociado a cada identificador.

Los resultados para todas las mediciones fueron más consistentes que en el experimento anterior. Sin embargo aún se registraban pequeñas anomalías, que supusimos se trataban de errores de medición.

Para tener algún marco de referencia con el que comparar nuestras mediciones, se utilizó la herramienta `mtr` [6], que permite medir el RTT hacia cada hop en la ruta a un host destino y computar el RTT promedio para cada hop, entre otras estadísticas. El RTT promedio para cada hop computado por `mtr` resultó muy similar al computado por nuestra herramienta, con la salvedad que en los casos que un hop i registraba un RTT promedio más alto que el hop $i+1$, esta diferencia era notablemente más pequeña en la medición obtenida con `mtr` que con nuestra herramienta.

La observación anterior nos incentivó a realizar un nuevo experimento corriendo la herramienta durante una hora para cada host destino, bajo la suposición que al tomar mediciones durante más tiempo, las anomalías registradas se suavizarían.

2.4.3. Experimento 3: múltiples mediciones en paralelo, todos los hops a la vez durante una hora, identificadores de paquetes asignados al azar

En este caso la herramienta no sufrió modificaciones. La única variable actualizada fue el tiempo durante el que se realizaron mediciones, que esta vez fue de una hora.

Tal como se supuso, las anomalías mencionadas en el experimento anterior se suavizaron notablemente. Sin embargo este experimento introdujo dos nuevos problemas:

1. En algunas ejecuciones de la herramienta ocurrió que dos paquetes dentro de un mismo bache recibieron el mismo identificador elegido al azar. Esto produjo que se registraran mediciones confundiendo el hop, lo que afectaba el RTT promedio de esos hops notablemente.

2. En otras, ocurrió que una respuesta proveniente de un bache tardó más de un segundo en llegar al host en el que corre la herramienta. Luego, en el bache siguiente, un paquete recibió el mismo identificador que el paquete del bache anterior que originó la respuesta que tardó más de un segundo. A continuación, la respuesta del bache anterior finalmente llega al host donde corre la herramienta, y ésta la confunde con la respuesta del paquete del bache actual que recibió el identificador repetido. La herramienta entonces registra esa respuesta para el hop correspondiente al identificador en el bache actual, que puede tener un RTT muy distinto al RTT promedio de ese hop hasta el momento, afectando el promedio considerablemente.

Suponemos que estos problemas fueron observados en este experimento y no en el anterior por el hecho de haber realizado un número mucho más grande de mediciones, lo que aumenta la probabilidad que estos fenómenos ocurrieran.

Ambos problemas se solucionan generando un identificador único para cada paquete enviado a lo largo de la ejecución de la herramienta. El siguiente y último experimento incorpora dicho cambio en la herramienta.

2.4.4. Experimento 4: múltiples mediciones en paralelo, todos los hops a la vez durante una hora, identificadores de paquete únicos

En esta iteración final de la herramienta, se aplica el procedimiento de envío de paquetes por baches a todos los hops en simultáneo con identificadores únicos tal como está descrito en la sección 2.2.

En efecto, los dos problemas mencionados en el experimento anterior no volvieron a ocurrir. Las mediciones realizadas con la herramienta en su versión final resultaron indistinguibles de las realizadas con la herramienta `mtr`.

Aun así observamos algunas anomalías que detallamos en la sección 4.

El resto de este trabajo se basa en los resultados de este experimento.

3. Resultados

Los resultados presentados a continuación corresponden al experimento 4 explicado en la sección 2.4.4.

En todos los gráficos, el gateway presentado para cada hop es el que presenta mayor frecuencia muestral en el caso que se hubieran registrado respuestas provenientes de más de un gateway.

3.1. University of Oxford

TTL	IP Addresses	Absolute RTT	Relative RTT	Relative ZRTT	Location
1	192.168.1.1	3.915 ms	3.915 ms	-0.088	*
2	190.194.57.1	164.263 ms	160.347 ms	2.231	Avellaneda, Argentina
5	200.89.166.105	44.386 ms	-119.877 ms	-1.554	Argentina
6	200.89.165.197	44.233 ms	-0.153 ms	0.063	Argentina
9	200.89.164.213	43.083 ms	-1.150 ms	0.050	Argentina
10	200.89.165.222	42.798 ms	-0.284 ms	0.062	Argentina
11	208.178.244.125	42.155 ms	-0.644 ms	0.057	United States
12	67.16.134.218	234.253 ms	192.099 ms	2.660	United States
13	4.68.111.121	173.618 ms	-60.636 ms	-0.753	United States
14	4.69.138.123	275.144 ms	101.526 ms	1.437	United States
15	4.69.140.142	189.482 ms	-85.662 ms	-1.091	United States
16	4.69.202.65	271.422 ms	81.940 ms	1.172	United States
17	4.69.148.106	276.174 ms	4.752 ms	0.130	United States
18	4.69.143.214	274.406 ms	-1.769 ms	0.042	United States
19	4.69.201.69	275.762 ms	1.356 ms	0.084	United States
20	4.69.137.65	278.957 ms	3.195 ms	0.109	United States
21	4.69.143.89	278.244 ms	-0.713 ms	0.056	United States
22	4.69.133.101	301.552 ms	23.307 ms	0.380	United States
23	195.50.119.98	262.045 ms	-39.506 ms	-0.468	United Kingdom
24	146.97.33.41	260.679 ms	-1.366 ms	0.047	London, United Kingdom
25	146.97.33.21	263.462 ms	2.783 ms	0.103	London, United Kingdom
26	146.97.37.206	262.942 ms	-0.519 ms	0.059	London, United Kingdom
27	193.63.108.129	263.214 ms	0.272 ms	0.069	United Kingdom
28	193.63.108.134	261.751 ms	-1.464 ms	0.046	United Kingdom
29	193.63.109.110	271.367 ms	9.617 ms	0.195	Wantage, United Kingdom
30	192.76.21.2	271.423 ms	0.056 ms	0.066	Oxford, United Kingdom

Figura 5: Traza hacia University of Oxford

Ruta principal

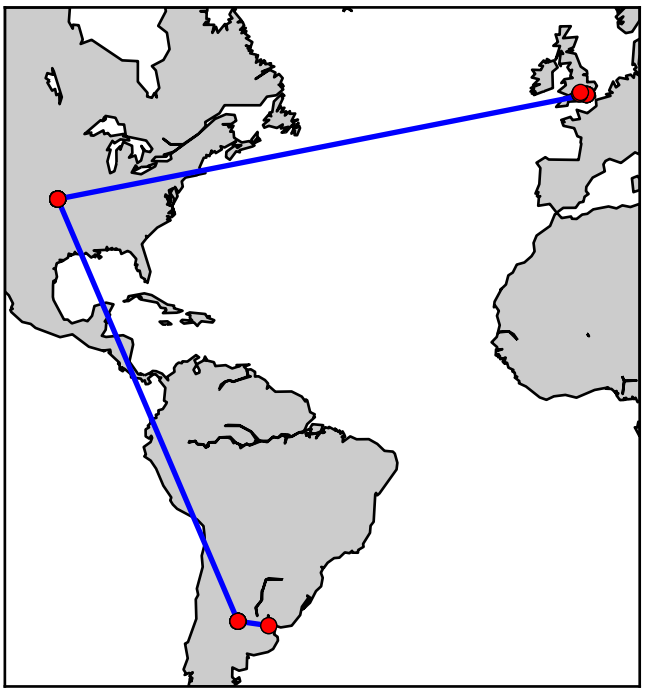

Figura 6: Ruta hacia University of Oxford

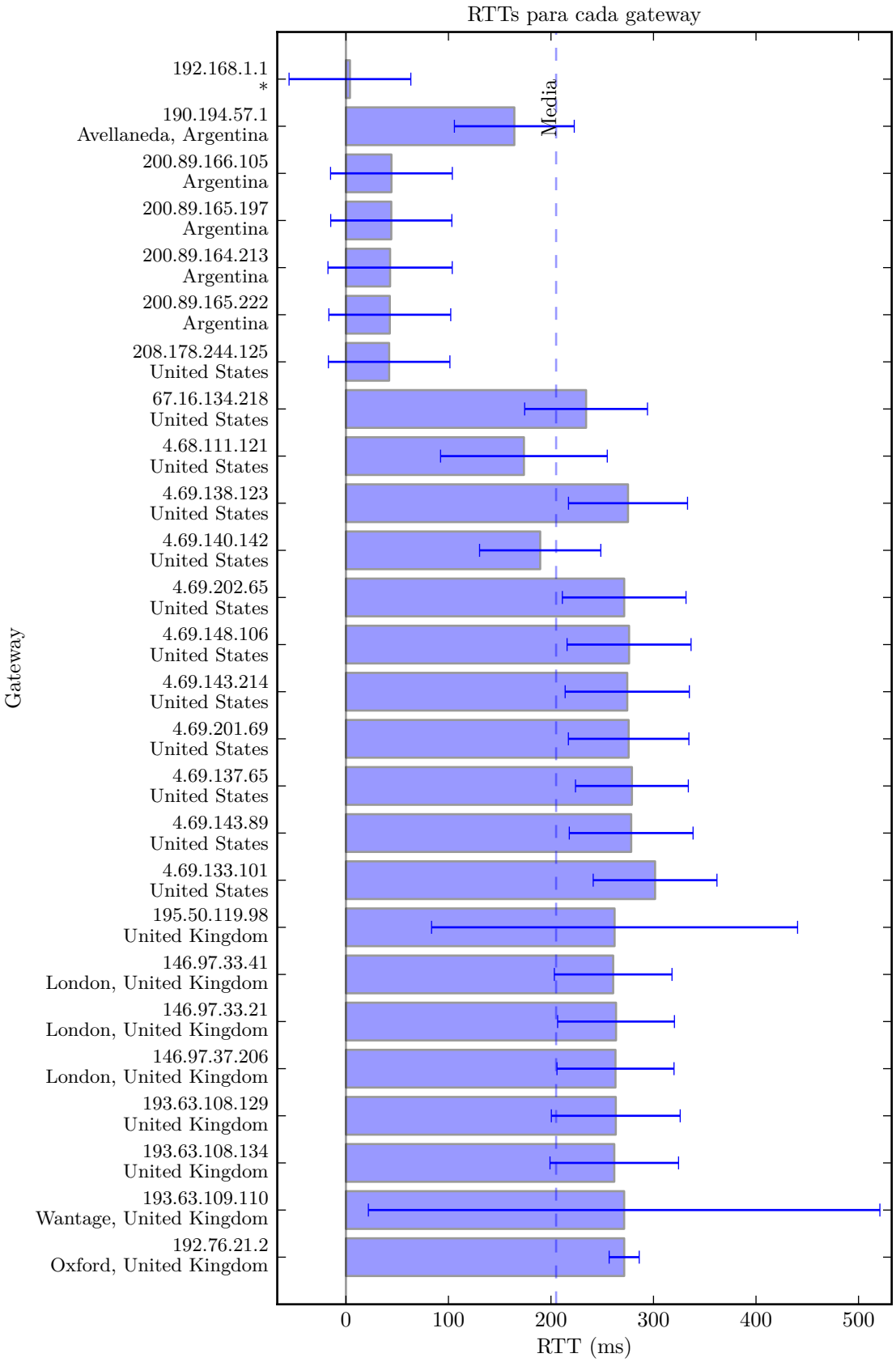

Figura 7: RTT de los gateways de la ruta hacia University of Oxford

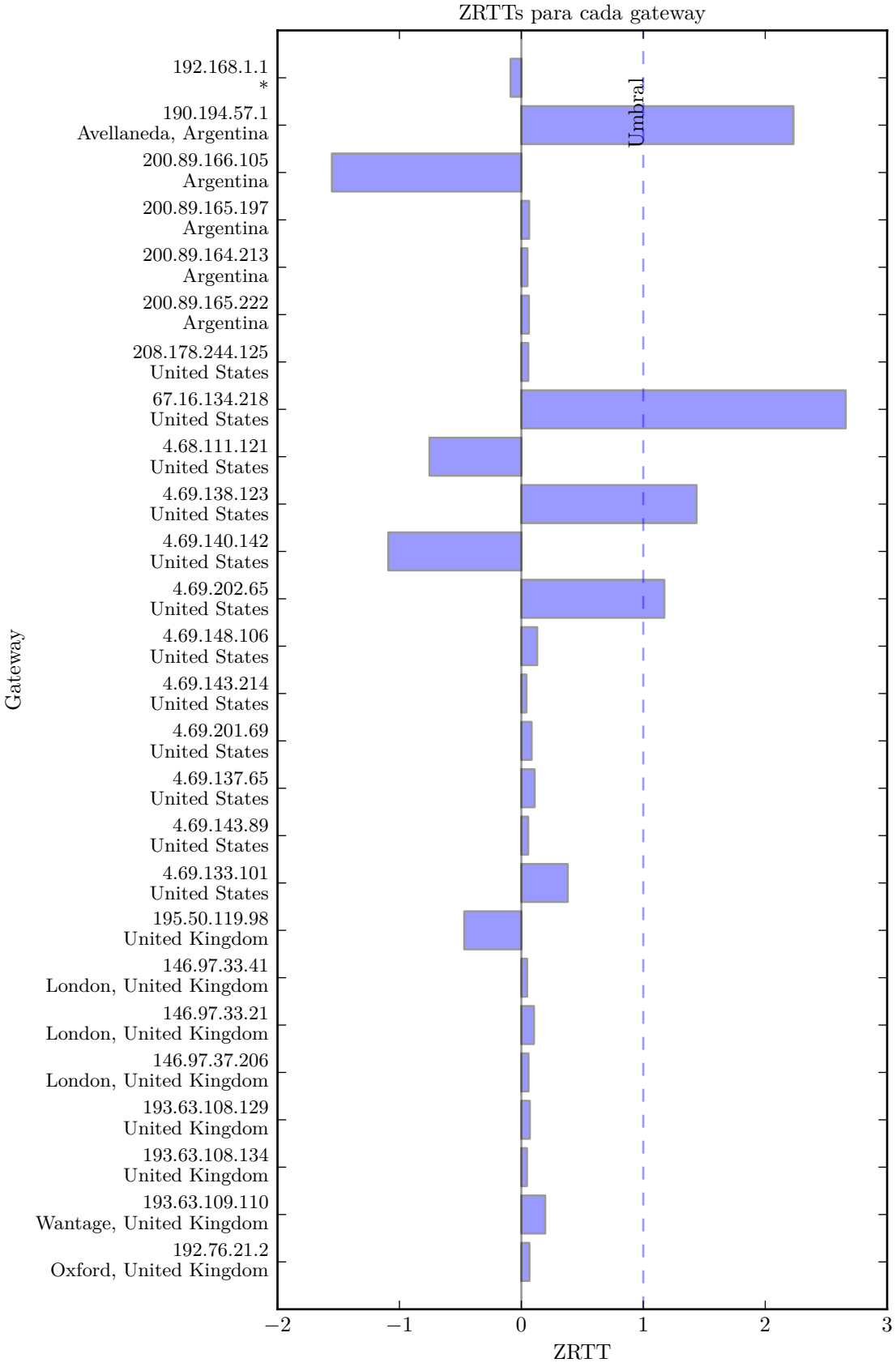

Figura 8: ZRTT de los gateways de la ruta hacia University of Oxford

3.2. The University of Sydney

TTL	IP Addresses	Absolute RTT	Relative RTT	Relative ZRTT	Location
1	192.168.1.1	2.495 ms	2.495 ms	-0.213	*
2	190.194.57.1	138.808 ms	136.313 ms	1.956	Avellaneda, Argentina
5	200.89.165.157	30.104 ms	-108.703 ms	-1.586	Argentina
6	200.89.165.130	29.463 ms	-0.641 ms	-0.024	Argentina
9	200.89.164.217	28.635 ms	-0.828 ms	-0.026	Argentina
10	200.89.165.222	28.665 ms	0.031 ms	-0.014	Argentina
11	159.63.53.213	36.210 ms	7.544 ms	0.095	United States
12	67.16.139.18	207.744 ms	171.534 ms	2.465	United States
13	129.250.9.117	192.087 ms	-15.657 ms	-0.241	Englewood, United States
14	129.250.3.172	195.526 ms	3.439 ms	0.035	Englewood, United States
15	129.250.3.174	194.925 ms	-0.601 ms	-0.023	Englewood, United States
16	129.250.2.168	231.937 ms	37.012 ms	0.521	Englewood, United States
17	129.250.2.230	228.437 ms	-3.500 ms	-0.065	Englewood, United States
18	204.1.253.166	228.255 ms	-0.181 ms	-0.017	Englewood, United States
19	202.158.194.172	352.907 ms	124.651 ms	1.788	Australia
20	113.197.15.68	352.569 ms	-0.337 ms	-0.019	Australia
21	113.197.15.66	375.784 ms	23.215 ms	0.321	Australia
22	113.197.15.65	358.409 ms	-17.375 ms	-0.265	Australia
23	202.158.194.197	386.693 ms	28.284 ms	0.395	Australia
24	202.158.205.165	387.616 ms	0.923 ms	-0.001	Australia
25	113.197.9.186	361.201 ms	-26.415 ms	-0.396	Lidcombe, Australia

Figura 9: Traza hacia The University of Sydney

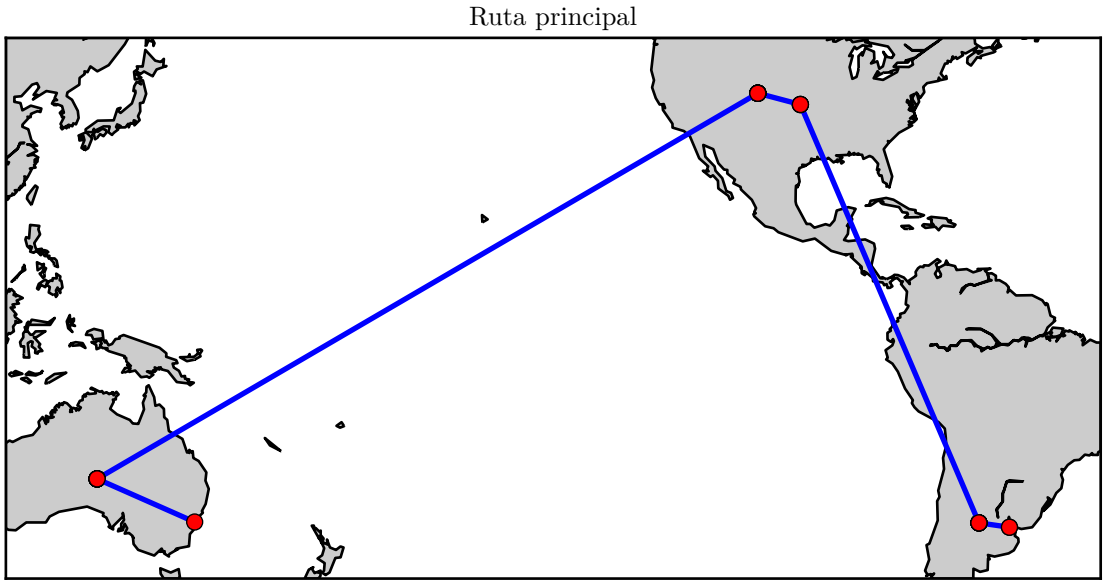

Figura 10: Ruta hacia The University of Sydney

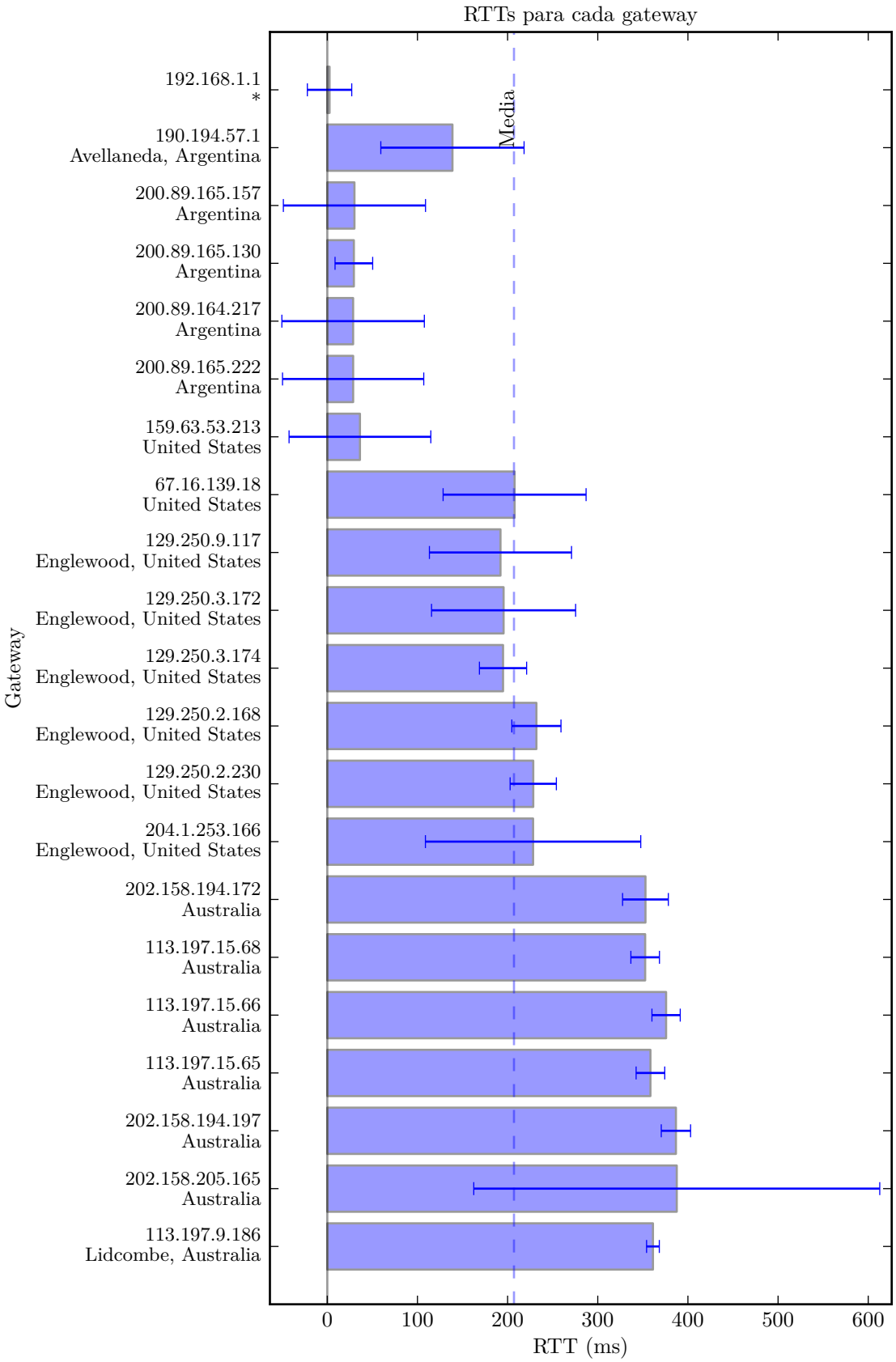

Figura 11: RTT de los gateways de la ruta hacia The University of Sydney

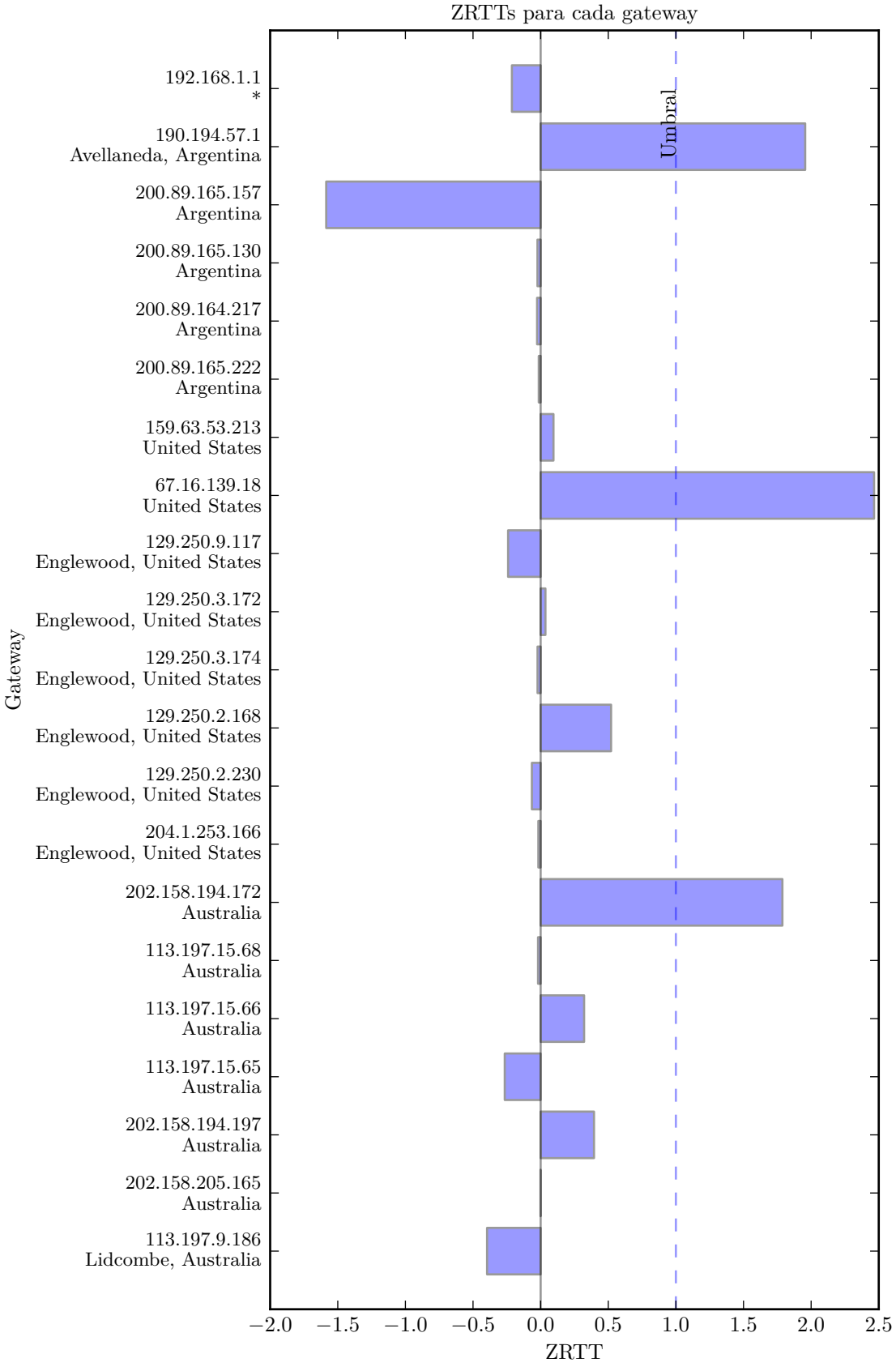

Figura 12: ZRTT de los gateways de la ruta hacia The University of Sydney

3.3. Malaysia University of Science and Technology

TTL	IP Addresses	Absolute RTT	Relative RTT	Relative ZRTT	Location
1	192.168.1.1	5.278 ms	5.278 ms	-0.134	*
2	181.28.111.1	121.930 ms	116.651 ms	2.043	Argentina
6	200.89.166.121	28.317 ms	-93.613 ms	-1.623	Argentina
7	200.89.165.86	28.037 ms	-0.280 ms	0.004	Argentina
8	64.214.130.253	45.763 ms	17.726 ms	0.318	United States
	208.178.245.21				United States
9	67.17.192.6	171.696 ms	125.933 ms	2.205	United States
10	203.208.172.189	174.447 ms	2.751 ms	0.057	Singapore
11	203.208.183.145	271.733 ms	97.286 ms	1.705	Singapore
	203.208.171.137				Singapore
	203.208.149.61				Singapore
	203.208.182.125				Singapore
	203.208.182.77				Singapore
	203.208.149.73				Singapore
	203.208.172.101				Singapore
	203.208.149.25				Singapore
	203.208.153.121				Singapore
	203.208.149.37				Singapore
	203.208.171.85				Singapore
	203.208.171.234				Singapore
	203.208.182.41				Singapore
12	203.208.151.117	348.933 ms	77.199 ms	1.355	Singapore
	203.208.152.222				Singapore
	203.208.151.113				Singapore
	203.208.153.166				Singapore
	203.208.151.98				Singapore
	203.208.171.9				Singapore
	203.208.151.229				Singapore
	203.208.149.225				Singapore
	203.208.152.226				Singapore
	203.208.151.85				Singapore
	203.208.154.45				Singapore
	203.208.151.221				Singapore
	203.208.182.45				Singapore
	203.208.171.189				Singapore
13	203.208.183.14	350.153 ms	1.220 ms	0.030	Singapore
	203.208.183.153				Singapore
	203.208.153.254				Singapore
	203.208.174.82				Singapore
14	203.208.153.166	352.471 ms	2.318 ms	0.050	Singapore
	203.208.151.98				Singapore
	124.158.224.45				Malaysia
	203.208.152.222				Singapore
	203.208.152.226				Singapore
	203.208.182.45				Singapore
15	61.11.210.1	354.032 ms	1.560 ms	0.036	Malaysia
	203.208.174.82				Singapore
16	61.11.211.175	349.901 ms	-4.131 ms	-0.063	Malaysia
	124.158.224.45				Malaysia
17	124.158.228.58	356.670 ms	6.769 ms	0.127	Malaysia
	61.11.210.1				Malaysia
18	110.4.44.250	350.958 ms	-5.711 ms	-0.090	Penang, Malaysia
	61.11.211.175				Malaysia
19	110.4.45.250	352.772 ms	1.814 ms	0.041	Penang, Malaysia
	124.158.228.58				Malaysia

Figura 13: Traza hacia Malaysia University of Science and Technology

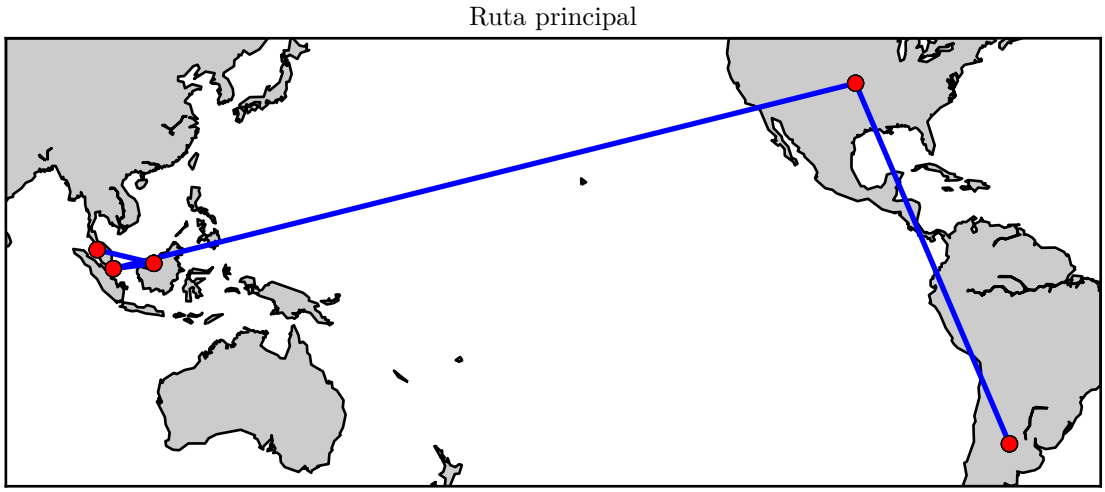

Figura 14: Ruta hacia Malaysia University of Science and Technology

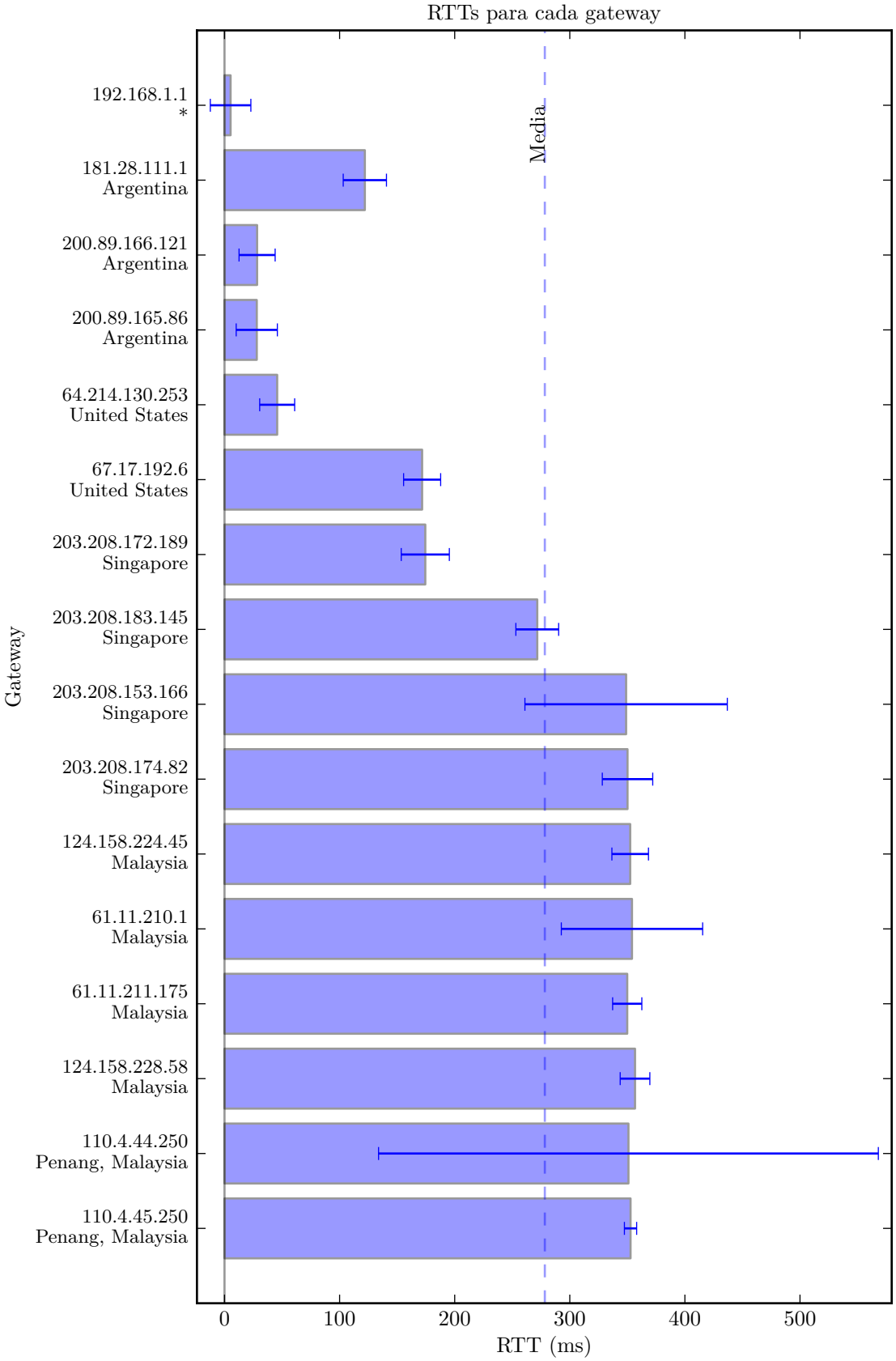

Figura 15: RTT de los gateways de la ruta hacia Malaysia University of Science and Technology

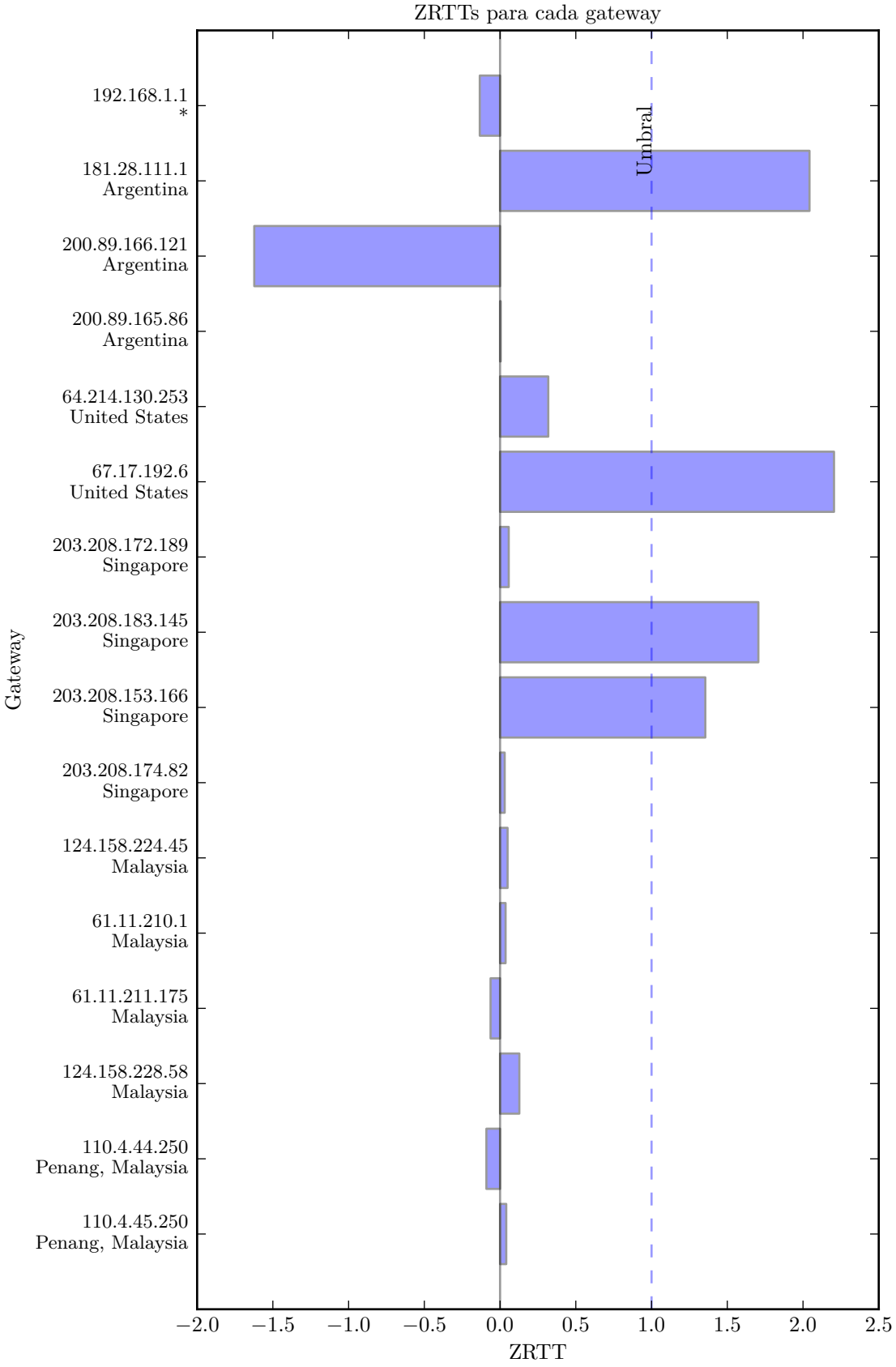

Figura 16: ZRTT de los gateways de la ruta hacia Malaysia University of Science and Technology

4. Discusión

Teniendo como referencia los resultados obtenidos, notamos que ocurrieron algunos fenónemos en común que creemos vale la pena mencionar.

4.1. Primer nodo externo

Podemos notar, en los resultados de todas las universidades analizadas, que el segundo hop posee un RTT promedio alto en comparación a los nodos más próximos a éste. Suponemos que esto se debe a que el gateway en ese hop tiene una prioridad baja para las respuestas a paquetes ICMP, haciendo que las respuestas tarden más de lo esperado.

4.2. Posible error de localización

Otro punto a tener en cuenta, también visible en los resultados de todas las universidades, es que la biblioteca elegida para la geolocalización (MaxMind GeoLite City[3]) sitúa muy probablemente la localización de los IP según la procedencia de la compañía que realice el enlace entre países. Pueden notarse en los resultados hops con bajo ZRTT relativo entre distintos países y luego el próximo nodo dentro del segundo país tiene un ZRTT relativo muy alto, indicando probablemente que se trata de un enlace de gran distancia.

4.3. Promedios de RTT

Un dato estadístico anómalo general es el de obtener, para un determinado nodo, un RTT absoluto menor al RTT absoluto del nodo anterior, lo que significaría que llegar a este nodo toma menos tiempo que llegar al anterior. Esta anomalía se debe a que los valores del RTT absolutos se calculan mediante el promedio de los RTT obtenidos para cada nodo, pudiendo el paquete ICMP haber tomado caminos diferentes y habiendo conseguido llegar de manera apenas más rápida en promedio.

4.4. Posibles enlaces submarinos

En la traza a la universidad de Oxford podemos observar en el gráfico del ZRTT relativo como la IP 67.16.134.218 asignado a Estados Unidos obtiene un ZRTT relativo alto estando rodeado por dos nodos cuyos ZRTT son bajos y su nodo anterior sufre del problema general de la geolocalización del cuál hablamos anteriormente (el nodo debería pertenecer a un router en Argentina), podemos decir que el IP mencionado pertenece al primer router luego de un enlace submarino.

Podemos observar también en el gráfico del ZRTT relativo de esta universidad como existen varios enlaces (4.69.138.123 y 4.69.202.65) de los cuales no es posible deducir con certeza la causa de sus altos valores, pero podemos especular de que, algún router posee una prioridad baja para contestar paquetes ICMP o que las IP sufren del problema de geolocalización indicado y estos saltos son entre Estados Unidos y algún país europeo y luego de el continente europeo a Reino Unido o posiblemente, una combinación de ambos (un salto a Reino unido y un router con prioridad de contestación baja). Esta deducción surge de que el análisis de los ZRTT relativos de los nodos siguientes muestra valores para los ZRTT muy bajos, incluso cuando la geolocalización muestra el cambio de paises.

La ruta a The University of Sydney muestra un gráfico de ZRRT relativos satisfactorio en cuánto al análisis de saltos submarinos. Podemos notar la IP del router 67.16.139.18 que sufre del problema de geolocalización y que al obtener un ZRTT relativo alto en relación a sus nodos vecinos y al estar próximo de Argentina, es posible identificarlo como salto submarino. El próximo salto notable es el de Estado Unidos a Australia de IP 202.158.194.172, donde podemos ver que esta IP no sufre del problema de geolocalización y marca un ZRTT relativo alto entre nodos cercanos además del cambio de país. Los demás routers del recorrido, a excepción del caso general del segundo nodo en la conexión, muestran ZRTT relativos esperables y bajos al no ser saltos submarinos, dejando como distinguidos los IP mencionados.

[3] http://dev.maxmind.com/geoip/legacy/geolite/

Por último, la traza obtenida a Malaysia University of Science and Technology muestra en su gráfico de ZRRT relativos, además del caso general del segundo nodo con alto ZRTT, 3 IP, 67.17.192.6, 203.208.183.145 y 203.208.153.166 los cuales podemos tomar como saltos submarinos. La IP 67.17.192.6 cae en el caso de geolocalización errónea y correspondería al salto de Argentina a Estados Unidos y las IP 203.208.183.145 y 203.208.153.166 que también sufren de lo mismo y que estimamos que sus ZRTT relativos representan un salto de Estados Unidos a Singapore y de Singapore a Malasia respectivamente. Los demás IP tienen ZRTT relativos bajos, indicando comunicaciones entre nodos cercanos.

4.5. Heurística para detección de enlaces submarino

Basándonos en el análisis realizado sobre la experimentación, proponemos como umbral en las mediciones de los ZRTT relativos para la detección de enlaces sumbarinos el valor 1. El umbral propuesto creemos que es suficiente para detectar grandes variaciones en relación al desvío estándar de RTT entre nodos.

Como lo planteamos anteriormente, los enlaces submarinos y los routers que asignan prioridad baja a las respuestas de paquetes ICMP muestran ambos un ZRTT alto, pero con la diferencia de que los routers que asignan una prioridad diferente hacen que el nodo siguiente tenga un ZRTT más bajo que el resto. Si bien se destacan del resto, usar únicamente un umbral positivo sobre los ZRTT presenta problemas a la hora de decidir si realmente pertenecen a un enlace submarino.

5. Conclusión

Concluimos que la técnica estudiada funciona bien sólo en los casos que la traza hacia el host destino no exhibe gateways que demoran más en responder con paquetes ICMP de tipo Time Exceeded que lo que demoran en reenviar paquetes al siguiente hop en la ruta a su destino, ya que en esos casos la técnica puede producir falsos positivos.

En el caso general, esta técnica resulta adecuada para identificar hops candidatos a saltos submarinos, pero suele ser necesaria una etapa de análisis adicional para filtrar los falsos positivos dentro del conjunto de candidatos identificado, de manera de obtener un subconjunto compuesto únicamente por hops correspondientes a saltos submarinos.

6. Trabajo futuro

Proponemos estudiar el siguiente método para filtrar el conjunto de candidatos a saltos submarinos identificado utilizando la técnica analizada en este trabajo.

En el caso en el que un hop i demora más en contestar con paquetes ICMP que lo que demora en reenviar paquetes al siguiente hop en la ruta y el hop $i + 1$ se comporta normalmente, el RTT absoluto promedio del hop i suele ser notablemente mayor que el del hop $i + 1$. Por lo tanto el RTT relativo promedio del hop $i + 1$ suele ser negativo, ya que su RTT absoluto promedio decrece respecto del hop anterior.

De manera análoga, el ZRTT del hop $i + 1$ será negativo; éste tendrá un valor más alejado de 0 en la medida que el hop i tenga un RTT absoluto mayor que el hop $i + 1$.

Una posible manera de identificar los hops que demoran más en responder paquetes ICMP que lo que demoran en reenviarlos al siguiente hop es, entonces, tomar un umbral negativo para el ZRTT de todos los hops, y si un hop i exhibe un ZRTT que cae por debajo de dicho umbral, entonces considerar el hop $i - 1$ como uno que presenta dicha anomalía.

Suponemos que combinando esta técnica junto con la estudiada en este trabajo puede ser posible identificar conjuntos de candidatos a saltos submarinos con menos falsos positivos.

Referencias

[1] *Scapy Project*. http://www.secdev.org/projects/scapy, Mayo de 2014.

[2] *RFC 792: Internet Control Message Protocol*. http://tools.ietf.org/html/rfc792.

[3] *Traceroute*. http://en.wikipedia.org/wiki/Traceroute, Mayo de 2014.

[4] *Ping (network utility) (Artículo en Wikipedia)*. http://en.wikipedia.org/wiki/Ping_(networking_utility), Mayo de 2014.

[5] *Internet Control Message Protocol (Artículo en Wikipedia)*. http://en.wikipedia.org/wiki/Internet_Control_Message_Protocol, Mayo de 2014.

[6] *mtr*. http://www.bitwizard.nl/mtr, Mayo de 2014.