UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

TRABAJO FIN DE MÁSTER

ESTIMACIÓN Y COMPARACIÓN DE UN MODELO ESTADÍSTICO A-ESTABLE DE PRIMER ORDEN BASADO EN FLUJOS NETFLOW RESPECTO AL TRÁFICO AGREGADO EN REDES IP

Máster en Ingeniería Informática y de Telecomunicación

Matteo Stoppa

Julio de 2013

ESTIMACIÓN Y COMPARACIÓN DE UN MODELO ESTADÍSTICO A-ESTABLE DE PRIMER ORDEN BASADO EN FLUJOS NETFLOW RESPECTO AL TRÁFICO AGREGADO EN REDES IP

AUTOR: Matteo Stoppa
TUTOR: Jorge E. López de Vergara Méndez

Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Julio de 2013

Agradecimientos

No sera fácil resumir en esta sección la lista con todas las personas que merecerían estar incluidas. Sin embargo, reorganizando las ideas y esperando no olvidar a nadie, se puede empezar el elenco con:

Un particular agradecimiento y mi gratitud para el profesor Jorge E. López de Vergara Méndez por guiarme a lo largo de la investigación, además de su seguimiento y consejos, el tiempo, su completa disponibilidad y el interés demostrado; es decir todo lo que ha hecho posible el desarrollo de este TfdM.

Querría aprovechar para agradecer también a los profesores José Luis García-Dorado y Federico Simmross-Wattenberg por su ayuda, sus consejos y sus comentarios aclaratorios.

También quería agradecer a mi compañera de vida Laura por haberme ayudado con la revisión gramatical (a pesar de que no entendía mucho del contenido) y a mi compañero de piso Federico Ibáñez por ayudarme y tratarme como si fuera su hermano. Además gracias a los dos por haberme aguantado en los periodos mas tensos.

No puede faltar un agradecimiento a mi familia que a pesar de todo siempre han estado apoyándome y animándome.

Un “gracias” más va dirigido a un miembro de la familia que merece un trato a parte: mi abuelo, cuya sabiduría siempre es de inspiración.

Finalmente me gustaría agradecer a todos los que han ido llenando mis días y siempre me han dado la fuerza de concretar mis propósitos: todos los amigos cercanos o lejanos que están ahí.

Desde lo mas profundo de mi corazón un “Gracias a tod@s!”

Índice General

Índice de figuras

Índice de tablas

Glosario de acrónimos

SNMP: Simple Network Management Protocol

FFT: Fast Fourier Transform

AR: Auto Regressive

MA: Mobile Average

ARMA: Auto Regressive Mobile Average

TfdM: Trabajo Fin de Máster

Resumen

Este trabajo evalúa y compara los parámetros de una distribución α-estable para modelar el tráfico en una red a partir de la información agregada generada por el protocolo Simple Network Management Protocol (SNMP), y los flujos de red generados por el protocolo NetFlow de Cisco. Además, se presenta una comparación entre la información almacenada por los dos protocolos, los procesos necesarios para efectuar esta comparación, así como las operaciones que han sido necesarias a lo largo del trabajo. A continuación se extraen los parámetros de la distribución y se efectúa el test de bondad de Kolmogorov-Smirnov para calcular los errores entre las muestras disponibles, y para calcular los errores entre estas y la distribución calculada a partir de los resultantes parámetros de cada traza con la finalidad de ver cuál de los dos protocolos proporciona datos que mejor se adaptan a un modelo estadístico de tipo α-estable, y finalmente una prueba entre las distribuciones de los parámetros para compararlos. Como conclusión, se observa que es posible obtener resultados semejantes para los valores de los párametros, incluso con registros muestreados de NetFlow, dado que en el registro de este protocolo sólo se ha almacenado un flujo cada 100. Esto permite aprovechar este protocolo para estudiar las desviaciones del comportamiento habitual del tráfico de la red, reduciendo la carga que se pueda introducir en el *router*.

Capítulo 1

Introducción

1.1 Motivación

La gestión de red siempre ha sido una tarea muy compleja y un campo de investigación abierto. Además, a medida que crece el tráfico en la red este trabajo se puede volver más difícil. También hay que considerar la posibilidad de tener que enfrentarse a anomalías o ataques que intentan violar la privacidad de datos sensibles o causar interrupciones de servicio. En este último caso, la gestión debería comprender un sistema de monitorización con el objetivo de detectar rápidamente la anomalía/intrusión para poder resolver el problema.
Con el avance tecnológico y la mayor potencia de cálculo y almacenamiento de los equipos de red, aumenta la disponibilidad de información que es posible tener en cuenta para las tareas de monitorización y gestión de redes. Inicialmente, los equipos han venido utilizando el protocolo SNMP (Simple Network Management Protocol) para obtener esta información por parte de un gestor. No obstante, dicha información suele entregarse de manera agregada (e.g. Bytes que han entrado o salido por una interfaz de red). En los últimos años se han planteado otras posibilidades, como NetFlow [1], que proporciona un gran volumen de información del tráfico que atraviesa la red agregando la información en flujos.
La principal motivación de este Trabajo Fin de Máster es la de averiguar si es efectivamente posible utilizar la valiosa información de esta herramienta para modelar el tráfico de forma parecida a la de SNMP. En caso afirmativo, teniendo en cuenta las características de los registro de NetFlow (que se van a ver en el párrafo siguiente) y usando todas las informaciones contenidas, se podría desarrollar un robusto sistema de modelado y detección de ataques/anomalías que, una vez configurado, y con un mínimo impacto sobre el tráfico de red, proporcionaría y transmitiría los registros a las unidades de elaboración. De esta forma, la tarea de gestión ya no necesitaría tener que pedir la información agregada a intervalos periódicos, ahorrando tiempo y recursos de red. Para entender mejor la diferencia entre los dos protocolos y poder imaginar las mejoras en prestaciones que se podrían obtener vamos a ver más en detalle NetFlow.
NetFlow es un protocolo abierto desarrollado por Cisco para recoger datos de tráfico IP, que está incluido en los routers y en los conmutadores de Cisco y de otros fabricantes. NetFlow permite obtener información relativa al tráfico de red como un flujo de datos con origen, destino y protocolo en común. Por cada flujo NetFlow, se registra la fecha, hora de inicio y fin, los puertos y direcciones IP del remitente y del destinatario, el tipo de protocolo utilizado por el tráfico, el tipo de servicio proporcionado y las interfaces de red. De esta forma, la monitorización puede hacer uso de una información más detallada respecto a la que se puede obtener mediante SNMP. Esto permite incluso detectar eventuales anomalías o intrusiones más fácilmente, aunque esto tenga un coste computacional más elevado a la hora de utilizar los datos, pero este coste vendría siempre a ser inferior al estudio del tráfico a nivel de paquete y cargas útiles. Afortunadamente los equipos en los que se realizan los algoritmos de detección no son los mismos que los de los dispositivos de red, con lo cual pueden ser bastante más potentes. Tampoco hay que infravalorar el tiempo de ejecución

del estudio de los datos por cada una de las ventanas temporales en las que se divida la monitorización, que podría llegar a resultar demasiado largo para un sistema efectivamente viable.

1.2 Objetivos

Los objetivos del trabajo son los siguientes:

- Averiguar que sea efectivamente posible modelar la velocidad del tráfico de red a partir de la información presente en los dos registros con un modelo estadístico de tipo α-estable, aprovechando su ventaja de adaptabilidad a la alta variabilidad del tráfico de red.

- Comparar la información almacenada en los registros entre sí y con los modelos calculados y además confrontar estos últimos mutuamente para evidenciar la diferencias entre las dos soluciones.

- Examinar la bondad de ajuste con el test de Kolmogorov-Smirnov para valorar la diferencia existente entre los modelos.

- Sacar unas conclusiones de todo el trabajo y proponer unas soluciones eventuales alternativas que podrían mejorar el trabajo, así como sugerir posibles trabajos futuros que puedan utilizar aún más información que la almacenada en el registro de NetFlow y/o investigar más los limites de este protocolo.

1.3 Fases de realización

En esta sección se perfilan las siete fases de realización del trabajo:

- Una primera fase de recogida de registros;
- Una segunda de elaboración de los registros para homogeneizar la información contenida con su consiguiente sincronización;
- Una fase de comparación de las series temporales para evaluar la necesidad de otras elaboraciones;
- Una fase de filtrado de la serie temporal de NetFlow y una nueva sincronización con SNMP;
- Una fase de evaluación de los parámetros de la distribución α-estable;
- Una fase para efectuar el test de bondad;
- Una última fase donde se resumen los resultados obtenidos y las conclusiones deducidas, y donde se proponen mejoras y los trabajos futuros a llevar a cabo.

1.4 Estructura del documento

Finalmente vamos a ver la estructura de la memoria y todos los temas y fases que se describirán en los siguientes capítulos:

- En el capítulo 2 después de una breve introducción sobre varias distribuciones que se pueden utilizar para modelar el tráfico de red, se describe más en detalle la distribución α-estable, sus características y sus parámetros, y se concluye resumiendo las informaciones destacadas del capítulo y discutiendo las razones por las cuales se ha preferido esta distribución para realizar este TfdM.

- En el capítulo 3 primero se introducen las condiciones iniciales y seguidamente se tratan las fases de comparación inicial de los registros de SNMP y NetFlow, la sincronización de los mismos registros y el filtrado de las series temporales del registro de NetFlow. Estas dos últimas fases resultan necesarias debido a los resultados obtenidos en la primera. Finalmente se comparan las series temporales elaboradas evidenciando las mejoras obtenidas.

- En el capítulo 4 se describe la fase de evaluación de los parámetros de las distribuciones α-estables y se enseñan todas las diferencias encontradas a primera vista antes de la siguiente fase.

- En el capítulo 5 se reseña la evaluación de los errores realizando un test de bondad de Kolmogorov-Smirnov. También se resumen los resultados obtenidos en las dos pruebas y se discuten brevemente.

- En el capítulo 6 se resumen los resultados más significativos que se han encontrado a lo largo de todo el TfdM, se trazan las conclusiones deducidas tras la elaboración del trabajo, se discuten las consecuencias y se proponen posibles trabajos futuros.

Capítulo 2

Modelo estadístico

2.1 Introducción

Como se ve en el capítulo anterior la primera elección importante que se ha tenido que tomar para empezar este TfdM ha sido la de seleccionar un modelo para describir las series temporales de la velocidad del tráfico en la red, ya que la robustez de la información útil extraída depende de sus características. No es necesario decir que cuando mejor se consigue adaptar el modelo a las variaciones de la velocidad de tráfico, más fiables son los resultados de las fases sucesivas. Se sabe que el tráfico de red se puede considerar como una variable aleatoria y en cortos periodos de tiempo este se puede considerar también estacionario. Si presuponemos que los paquetes que llegan al nodo de la red son independientes entre si se pueden utilizar muchos modelos estadísticos conocidos, aunque esto en principio no sea del todo cierto pero ayuda, en este momento, a simplificar los cálculos y se puede demostrar que es una buena aproximación del caso real, a través de muchos ejemplos que se encuentran en la literatura. Son muchos los modelos que se han utilizado para extraer características del tráfico de red [3], desde los más simples hasta los más complejos, y cada uno tiene sus ventajas y sus limites. Sin embargo, cuanto mejor se adapta un modelo a la distribución de las muestras, más bajo será el valor del error y consecuentemente más representativos se vuelven sus parámetros a la hora de resumir las características del tráfico. Se ha descubierto que una característica del tráfico que afecta de forma consistente a la adaptación del modelo es la alta variabilidad. No son muchos los modelos que consiguen adaptarse a esta característica y entre estos consta la distribución α-estable. Este es el motivo por el cual se ha elegido este modelo de primer orden para desarrollar este TfdM.
En la siguiente sección se encuentra una introducción sobre las dos distribuciones más simples y entre las primeras utilizadas para modelar el tráfico de red. A continuación se describe más en detalle la distribución α-estable, sus características y sus parámetros. Finalmente se concluye el capítulo resumiendo los datos destacados del mismo y discutiendo las razones por las cuales se ha preferido esta distribución para realizar este TfdM.

2.2 Modelos para el tráfico

El modelo más simple y limitado es el modelo de Poisson. Este modelo fue además unos de los primeros utilizados en las telecomunicaciones modelando los tiempos de llegada. En este modelo sólo hay un grado de libertad: la tasa de llegada por unidad de tiempo λ, definida como el recíproco de la esperanza matemática. En la hipótesis de que el número de paquetes que llegan en un intervalo de tiempo T no dependa del instante inicial y final, sino sólo de la longitud del intervalo, y de que los tiempos de llegada fueran variables independientes e idénticamente distribuidas, este modelo tendría sentido. También se puede demostrar que bajo esta hipótesis el número de los paquetes que llegan a un nodo de una red está caracterizado por una variable aleatoria de Poisson [12]. Además si partimos el intervalo T en n intervalos suficientemente pequeños para que en cada intervalo llegue un sólo paquete con probabilidad p o ninguno con probabilidad 1-p, se puede escribir que:

$$np = \lambda T \qquad \text{(Eq. 2.1)}$$

Como estamos en las hipótesis para poder utilizar la fórmula de Bernoulli se puede decir que:

$$P(N(T)=k) = \binom{n}{k} p^k (1-p)^{n-k} \qquad \text{(Eq. 2.2)}$$

Donde P(N(T)=k) es la probabilidad que en el intervalo T lleguen un número de paquetes = k. Evidenciando la p de la primera ecuación y substituyendola en la segunda, desarrollando el término binomial, simplificando y con n que tiende a infinito se obtiene:

$$P(N(T)=k) = \frac{(\lambda T)^k}{k!} \mathrm{e}^{-\lambda T} \qquad \text{(Eq. 2,3)}$$

Esta última ecuación se utiliza para calcular la función de distribución de probabilidad a partir de su definición:

$$F_\tau(x) = P(\tau \leqslant x) = 1 - P(\tau > x) = 1 - P(N(x)=0) = 1 - \frac{(\lambda x)^0}{0!} \mathrm{e}^{-\lambda x} = 1 - e^{-\lambda x} \qquad \text{(Eq. 2.4)}$$

Esto dice que la función de densidad de probabilidad es igual a:

$$f_\tau(x) = \lambda e^{-\lambda x} \qquad \text{(Eq. 2.5)}$$

En la figura 2.1 se pueden ver diferentes ejemplos de distribución de Poisson con distintos valores de λ.

La distribución de Poisson a pesar de ser muy sencilla, presenta distintos inconvenientes a la hora de usarla para modelar el tráfico real. El límite principal es que teniendo un sólo grado de libertad no se puede evaluar media y varianza de forma independiente, dado que la segunda variable depende del valor de la primera. Además esta es una distribución simétrica, como se puede ver también en la figura, y este comportamiento generalmente no se refleja en las características del tráfico de red. Para obviar estos inconvenientes y mejorar el modelo se puede probar con una

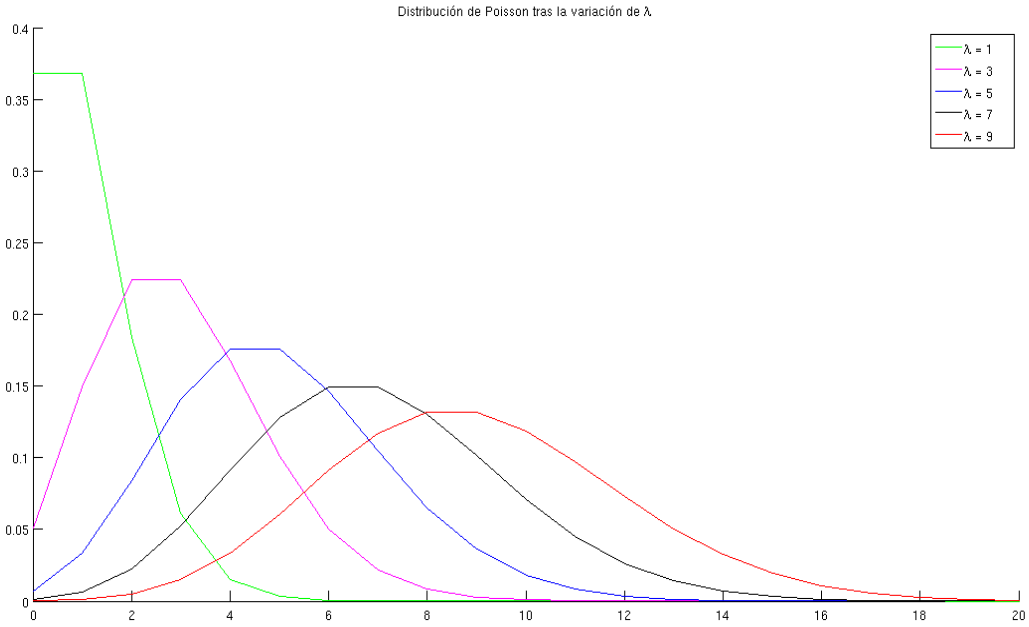

Figura 2.1 – Ejemplo de distribuciones de Poisson con diferentes valores de λ

distribución con más grados de libertad, aunque sea más compleja. Cuando el valor de λ es muy alto esta distribución tiende a converger hacia una distribución normal.
Justo esta última es un ejemplo de distribución con dos grados de libertad, en la cual la media y la varianza son dos parámetros independientes y por eso la variación de uno no afecta al otro. Esta distribución no sólo se ha usado hasta hace unos años para modelar el tráfico de una red sino que se utiliza para modelar muchas clases distintas de sucesos reales. De forma parecida a la de antes se podría calcular la función de densidad de probabilidad, pero en este caso el proceso sería más largo y complejo. Por eso se prefiere omitir los cálculos y llegar directamente al resultado. La resultante función de densidad de probabilidad de una variable aleatoria con distribución gausiana de media μ y desviación estandar σ es:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \qquad \text{(Eq. 2.6)}$$

Al contrario de lo que se obtenía antes, en este caso una variación de la media μ simplemente desplaza la distribución en correspondencia con el nuevo valor de la media y centrada con respecto a esta. Alternativamente una variación de la desviación deja la distribución centrada con respecto a la media, cambiando la anchura y consecuentemente la altura de la misma distribución. Unos ejemplos de distribuciones normales con media constante y distintos valores de varianza se encuentran en la Figura 2.2.
Como se puede ver a pesar de la mayor complejidad de la función, el haber añadido un parámetro a la distribución da más libertad a la hora de crear un modelo del tráfico agregado. De cualquier modo esta distribución sigue teniendo el límite de ser simétrica como la de Poisson. Esta asimetría, al igual que antes, consigue que el modelo y las muestras del tráfico de red no encajen perfectamente. Una vez más si se quiere eliminar este inconveniente, y en consecuencia mejorar el modelo, hay que añadir más grados de libertad y dicho esto, más complejidad a la distribución.

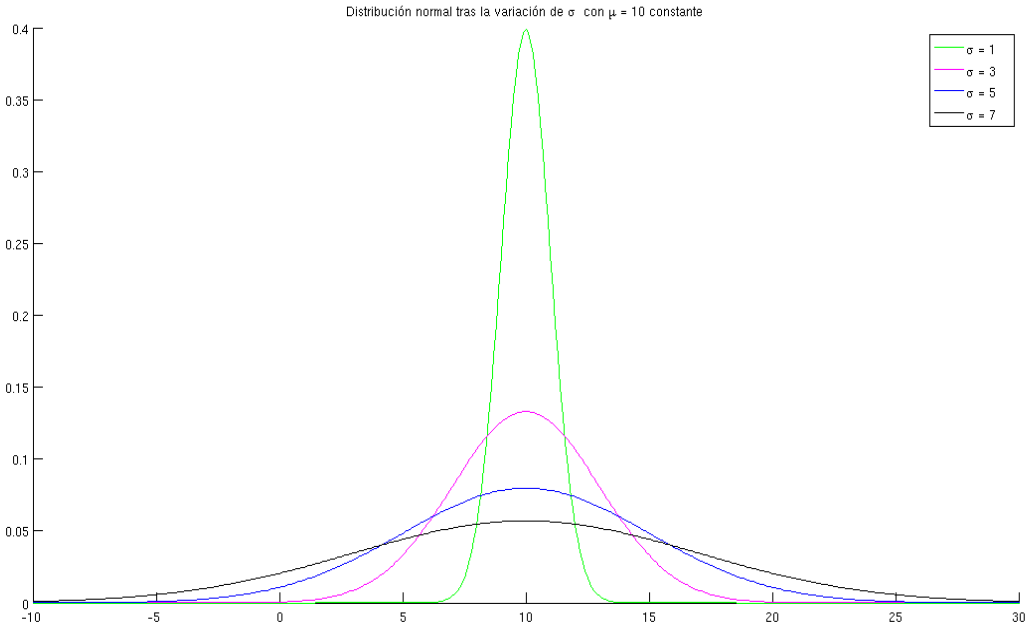

Figura 2.2 – Ejemplo de distribuciones normales con media constante y diferentes valores de varianza

Afortunadamente no faltan otros tipos de distribuciones con la libertad de configurar más parámetros. En la siguiente sección se van a describir las distribuciones α-estables que teniendo 4 grados de libertad pueden adaptarse a modelos asimétricos, siendo la asimetría controlada por otros dos parámetros adicionales. En la sección siguiente se da una visión general sobre las distribuciones α-estables.

2.3 Modelo α-estables

Las distribuciones α-estables se pueden ver como un superconjunto de la distribución gaussiana y surgen como solución al teorema del límite central cuando se admite la posibilidad de que los momentos de segundo orden no existan (es decir, cuando se considera que la varianza puede ser infinita) [4]. En términos de sucesos reales es obvio que éste nunca será el caso, sin embargo, en múltiples disciplinas tan dispares como la hidrología, la física de partículas o las telecomunicaciones, se observa que un modelo de varianza infinita se adapta mucho mejor a los datos que otros modelos existentes con varianza finita. Este fenómeno se suele denominar "alta variabilidad" o "efecto Noah" (Noé). El tráfico de red agregado es un ejemplo de este tipo de sucesos [5].
Esta familia de distribuciones ha sido descrita con detalle en la literatura [6]. No obstante, aquí se mencionan algunas de las propiedades de especial interés para este trabajo.
Las distribuciones α-estables están caracterizadas por cuatro parámetros: α, β, γ y δ (aunque algunos autores denominan a estos últimos σ y μ). α puede variar en el intervalo (0,2] y determina la forma de la curva que puede ser desde gaussiana cuando α=2, hasta una distribución degenerada cuando α→0. β pertenece al intervalo [-1,1] y determina la asimetría de la función de densidad de probabilidad dónde -1 indica asimetría total hacia la izquierda, 0 simetría y +1 asimetría total hacia la derecha. γ y δ son los parámetros análogos a la desviación típica y a la media de las gaussianas, respectivamente (de ahí que algunos autores les den el mismo nombre) pero γ nunca coincide con la desviación típica, ni siquiera en el caso α=2, mientras que δ coincide con la media solamente en el caso de que ésta exista (lo cual sucede cuando α>1).

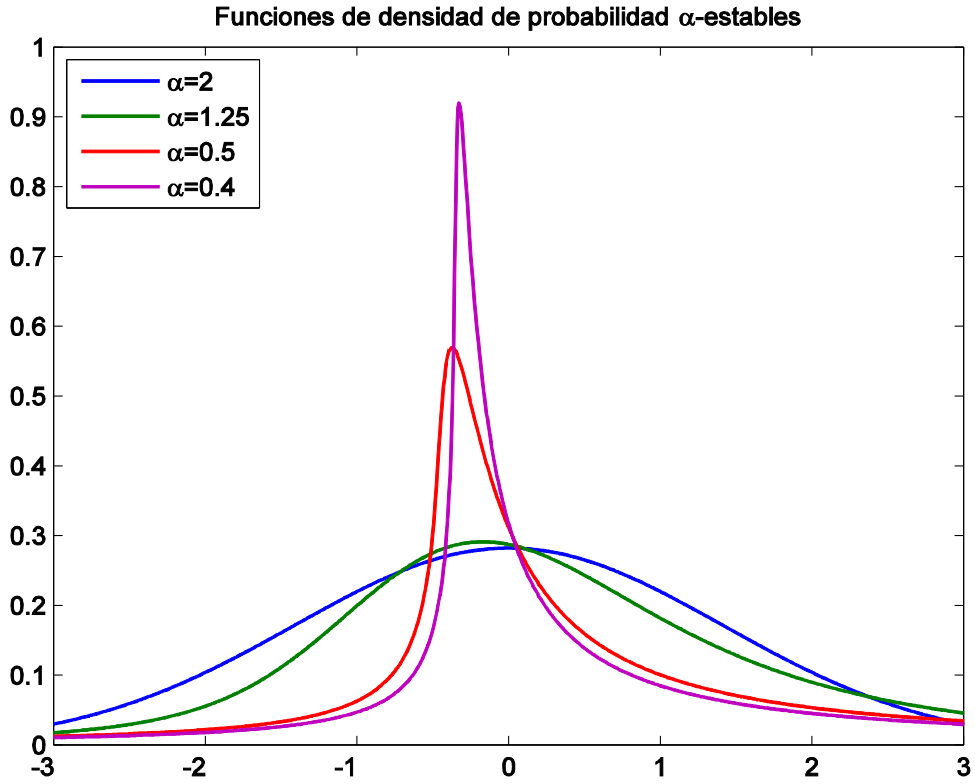

Figura 2.3 – Ejemplo de distribuciones α-estables para diferentes valores de α

Los cuatro parámetros de forma, asimetría, dispersión y localización, confieren a las distribuciones α-estables una gran flexibilidad, pero su capacidad para adaptarse a multitud de fenómenos reales no proviene de los cuatro grados de libertad sino de su estrecha relación con el teorema del límite central al igual que ocurre con las gaussianas. Las figuras 2.3 y 2.4 muestran algunos ejemplos de funciones de densidad de probabilidad α-estables para diversos valores de α y β respectivamente.

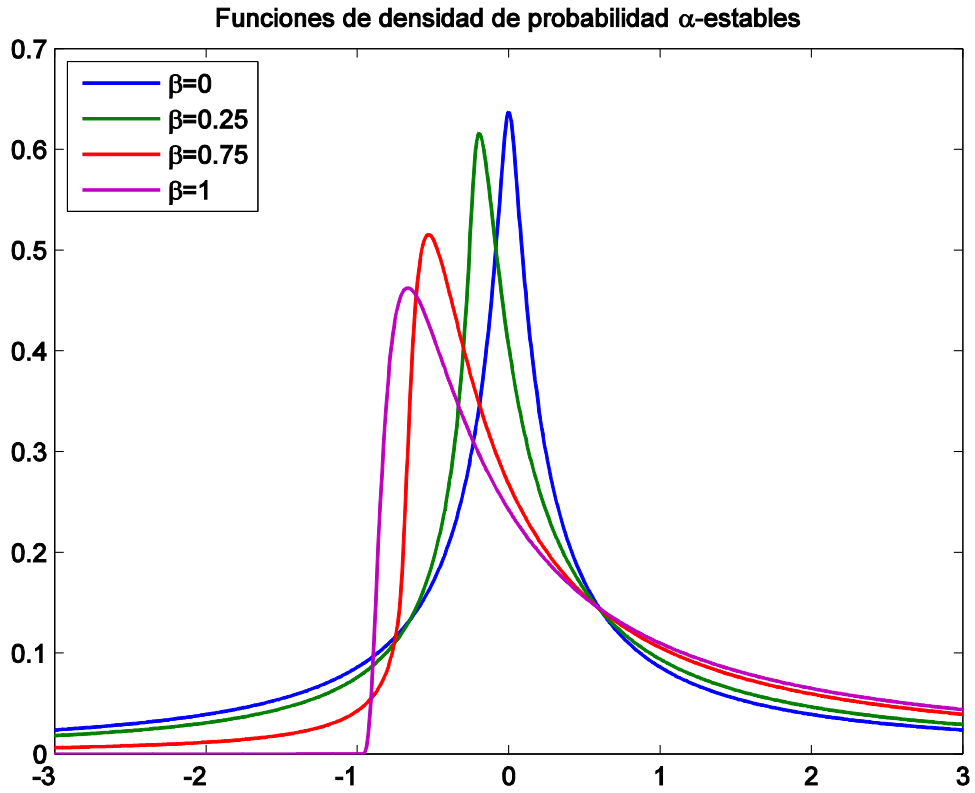

Figura 2.4 – Ejemplo de distribuciones α-estables para diferentes valores de β

A continuación, en la próxima sección, se concluye el capítulo con un resumen de lo que se ha visto hasta ahora, la elección de la distribución de primer orden para este trabajo y la solución que se va a adoptar para tener en cuenta otra característica que presenta normalmente el tráfico de red: la de la periodicidad a largo plazo.

2.4 Conclusiones

Para modelar información relativa al ancho de banda consumido en un enlace a partir de los registros de NetFlow y de cara a detectar tráfico anómalo se va a usar un modelo estadístico, concretamente una distribución α-estable, la cual, como se puede ver en [3], tiene la ventaja de adaptarse bien a la alta variabilidad del tráfico de red y permite identificar y distinguir cuándo el tráfico sigue un patrón normal o anómalo. Además gracias a los 4 grados de libertad que tienen estos tipo de distribuciones permiten adaptarse también a muestras que tienen distribuciones asimétricas. Como a lo largo del tiempo, la cantidad de datos que pasan por la red varía sensiblemente, será muy importante utilizar ventanas de tiempo en las cuales los parámetros del modelo se puedan considerar estacionarios y volver a calcular dichos parámetros para cada ventana de tiempo. Una vez calculados los parámetros será también posible poder comparar los resultados obtenidos con los encontrados a partir de registros de tráfico agregado en la misma red y en el mismo periodo temporal. De esta forma se podrá determinar la eventual mejora debida al uso de un conjunto de información más detallada.
Una última cuestión que hay que considerar es la periodicidad a largo plazo que presenta el tráfico de red, ya que generalmente sus características suelen repetirse principalmente cada semana, al igual que ocurre con periodos más largos, como por ejemplo anualmente. Como el objetivo de este trabajo no es el de modelar el tráfico, sino el de evaluar y comparar los parámetros de la misma distribución a partir de dos registros diferentes, aquí no se verán técnicas para introducir esta características en en modelo. En literatura existen varios modelos que pueden ser muy útiles para esta tarea, como modelos AR (Auto Regressive), MA (Mobile Average), ARMA (Auto Regressive Mobile Average) y muchos más. A la hora de detectar anomalías o ataques en la red sería aconsejables tener en cuenta esta periodicidad, aunque no sea con los modelos que se acaban de citar sino con una simple media entre las ventanas relativas al mismo día de la semana.
A continuación se presentan las características de los registros de SNMP y NetFlow empleados en la comparación. Posteriormente se explica el método empleado para sincronizar los registros. Tras ello, se muestra la necesidad de filtrar la serie temporal obtenida con los registros de NetFlow para obtener mejores resultados. Después se evalúan los parámetros estadísticos en ambos casos, así como los errores obtenidos. Finalmente se presenta un conjunto de conclusiones y se indican líneas de continuación.

Capítulo 3

Registros utilizados y operaciones previas

3.1 Introducción

Una vez elegido el modelo que se utilizará a lo largo de este trabajo, vamos a ilustrar las primeras operaciones que han sido necesaria para poder seguir la investigación. En la primera parte de este capítulo se comparan los registros de SNMP y de NetFlow para ver las principales diferencias y el procesado de las señales realizado para homogeneizar la información que se quiere modelar. Esta tarea es muy importante, ya que no sólo da la posibilidad de comparar la información sino que transformando los registros en series temporales, el tiempo de muestreo juega un rol importante a la hora de elegir la longitud de la ventana de análisis del modelo. De hecho un tiempo de muestreo más largo reduce el número de muestras disponibles para evaluar los parámetros de la distribución, mientras que un tiempo inferior a 5 segundos no añadiría ningún dato novedoso a la serie temporal de SNMP, disponiendo sólo de muestras (del total de los bytes recibidos o transmitidos) de este registro cada 5 segundos. Después se describe la primera operación de sincronización de los registros. Seguidamente se delinea la operación de filtrado de las series temporales de NetFlow, justificando el uso del filtro, así como se ilustra el banco de filtros utilizados y los resultados obtenidos a la salida de este y la variación del error cuadrático medio. Finalmente en las conclusiones se encuentra un resumen del las características que más destacan en el capítulo.

3.2 Comparación entre registros

Durante el trabajo se han considerado los datos de subida y bajada procedentes del router de la Universidad de Valladolid recogidos a través de SNMP y NetFlow durante un periodo de tiempo que va del 4 de junio del 2007 al 30 de julio de 2008 para los datos de SNMP y del 1 de septiembre de 2007 al 31 de diciembre del 2008 para los flujos de NetFlow. Las muestras se tomaron de manera independiente, siendo obtenidas por parte de un sistema de monitorización para el caso de SNMP, y generadas directamente por el router para el caso de NetFlow. Al ser dichas muestras independientes será necesario sincronizarlas posteriormente.
Para poder comparar los registros claramente se necesita evaluarlos durante un periodo común, por lo que se ha considerado un periodo entre el 1 de septiembre del 2007 al 30 de julio del 2008. Además, los protocolos utilizados no almacenan los datos relativos al tráfico de red de la misma forma, pues en el caso de SNMP sólo se ha obtenido la información del contador temporal y la suma de bytes totales en el instante de muestreo, mientras que la información disponible con NetFlow es más extensa, como ya se apuntaba anteriormente. A continuación se puede encontrar información más detallada para cada uno de los dos tipos de datos a comparar:

➔ Registros SNMP. El sistema de monitorización en la Universidad de Valladolid fue configurado para sondear de manera periódica de SNMP a la tabla de interfaces del router, y con los datos obtenidos generar series temporales formadas por dos campos: el primero almacena el valor del contador temporal con resolución de microsegundo y con muestras recogidas cada 5 segundos; el segundo campo almacena el valor del contador de bytes en el instante de muestreo.

➔ Registros NetFlow. Estos registros guardan más información que los primeros, ya que por cada flujo transmitido se almacenan la fecha y la hora (4 campos) de inicio de la transmisión, la fecha y la hora de fin de la transmisión (con resolución de milisegundos), protocolo de nivel 4, dirección IP (4 campos) y puerto de origen, tipo de servicio, dirección y puerto de destino, otros dos campos de información de servicio, y finalmente un campo para el número de paquetes y uno con el número de bytes [7]. También hay que tener en cuenta la tasa de muestreo de los registros NetFlow, los cuales no contienen toda la información del tráfico sino que se limita, en nuestro sistema de captura, a una muestra cada 100 paquetes para reducir el coste computacional en los dispositivos de red. En este trabajo comprobaremos empíricamente el impacto que dicho muestreo tiene en el cálculo de tasas de transmisión.

Como para sincronizar los flujos el instante inicial y la forma de almacenar los datos en ambos casos es distinta, inicialmente, en los registros de SNMP se han aislado las 7 semanas y 4 días de diferencia que hay entre las dos, para luego buscar la mejor sincronización encontrando el mínimo valor de error cuadrático medio, como vamos a ver en la sección siguiente.

3.3 Sincronización de los registros

Para comparar los registros generados a partir de ambos protocolos también hay que transformar sus muestras en cantidades homogéneas y sincronizarlas temporalmente. De dichos registros se va a generar la serie temporal que representa la evolución de la tasa de transmisión de datos a lo largo del tiempo. Esto se consigue en 3 simples pasos:

- Transformando el formato de los ficheros de los registros para poderlos importar directamente en una matriz con la herramienta de importación de Octave [8], importando el tipo de fichero en ASCII. Los ficheros son de texto plano y contienen signos de puntuación para dividir los campos y caracteres no reconocidos en correspondencia de periodos de tiempo normalmente pequeños, donde no han llegados los valores de los campos. Puede que esto suceda porque los paquetes hayan sido descartados o los equipos de red estuvieran momentáneamente caídos o desconectados. De hecho con esta importación los elementos de la matriz tienen que estar separados por uno o más espacios, mientras que las líneas tienen que estar separadas por una alimentación de línea (el carácter 0x0A, LF).

- Importando las matrices. Para reducir la dimensión de la matriz resultante del registro de NetFlow se ha decidido omitir los campos de la información de servicio, así como juntar los campos de la hora transformando todo a escala de milisegundos. De esta forma la matriz pasa de 26 a 16 columnas, ahorrando espacio de almacenamiento, carga computacional y como consecuencia, tiempo de elaboración. El registro SNMP, sin embargo, presenta jitter en los instantes de muestreo como queda también reflejado en los contadores de los bytes. Con el uso de un algoritmo se ha compensado el efecto de este componente de ruido. En las figuras 3.1 y 3.2 se puede ver un ejemplo de una serie temporal de SNMP a lo largo de 7 días antes y después de la compensación del jitter. Finalmente se calcula la misma variable para ambos tipos de registros, en este caso específico se trata de la tasa de transmisión en función de los intervalos de tiempo. En este momento se pueden crear varias matrices con intervalos cada 5, 10, 15, 20, 30 segundos y más para poder evaluar también como varía el error cuadrático medio en función a la dimensión del intervalo. Es necesario precisar que durante el cálculo del número de bytes en cada uno de los intervalos se supone que el flujo NetFlow tenga velocidad constante de principio a fin de la transmisión. Esta suposición reduce el coste computacional de la operación pero al mismo tiempo introduce un filtrado paso bajo a toda la serie temporal. La elección del tiempo de muestreo para la serie temporal de NetFlow llega a ser muy importante desde este punto de vista, ya que cuanto más largo es el tiempo de muestreo, más estrecha es la banda del filtro, así como cuanto más pequeño es este tiempo, más ancha será la banda del filtro, degradando menos las evolución de la serie temporal. Además el tiempo de muestreo juega un rol muy importante a la hora de elegir la longitud de la ventana de análisis del modelo. En un tiempo de muestreo más corto se aprovecha mejor la información, como acabamos de ver, y a paridad de longitud de la ventana que se va a considerar estacionaria, proporciona más muestras, dando más robustez al modelo. Mientras que un tiempo más largo reduce la calidad de la información almacenada y el número de muestras disponibles para evaluar los parámetros de la distribución. En las figuras 3.3 y 3.4 se puede observar el efecto del filtro paso bajo al variar el tiempo de muestreo. Al final considerando todo y encontrando otros factores no esperados que se comentarán en las sucesivas secciones, se ha preferido un intervalo de 5 segundos que coincidiendo con el tiempo de muestreo de SNMP, aprovecha toda la información de este protocolo sin empeorar de forma visible la información del otro.

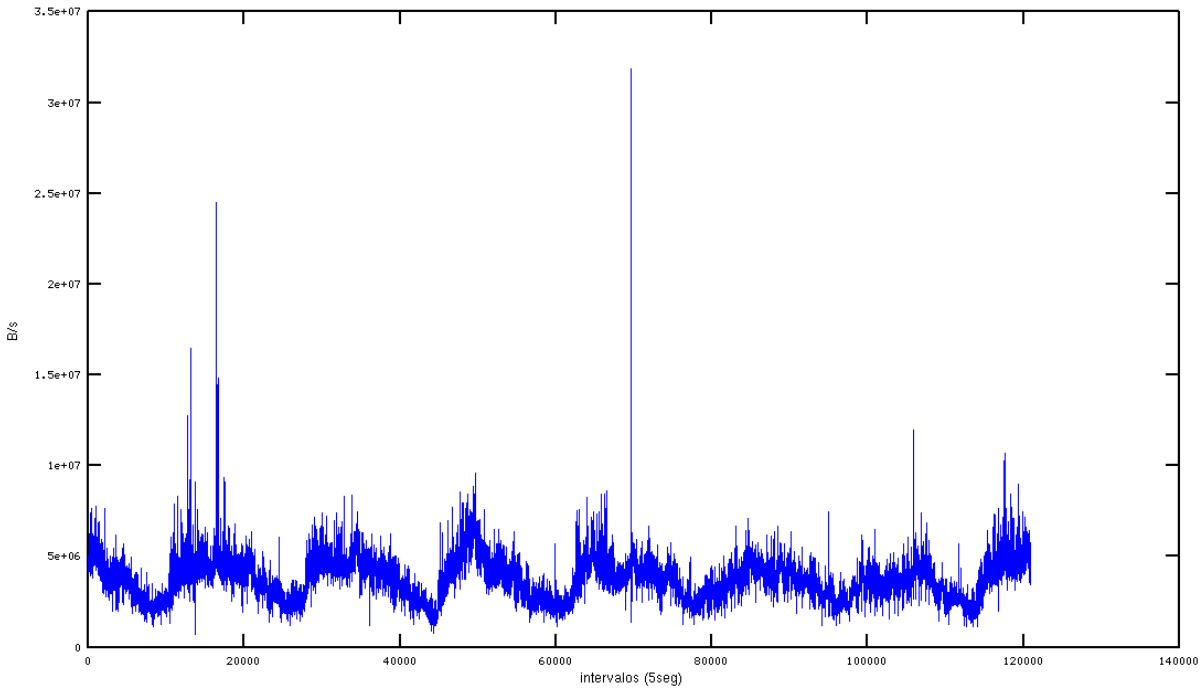

Figura 3.1 – Ejemplo de serie temporal de SNMP con jitter

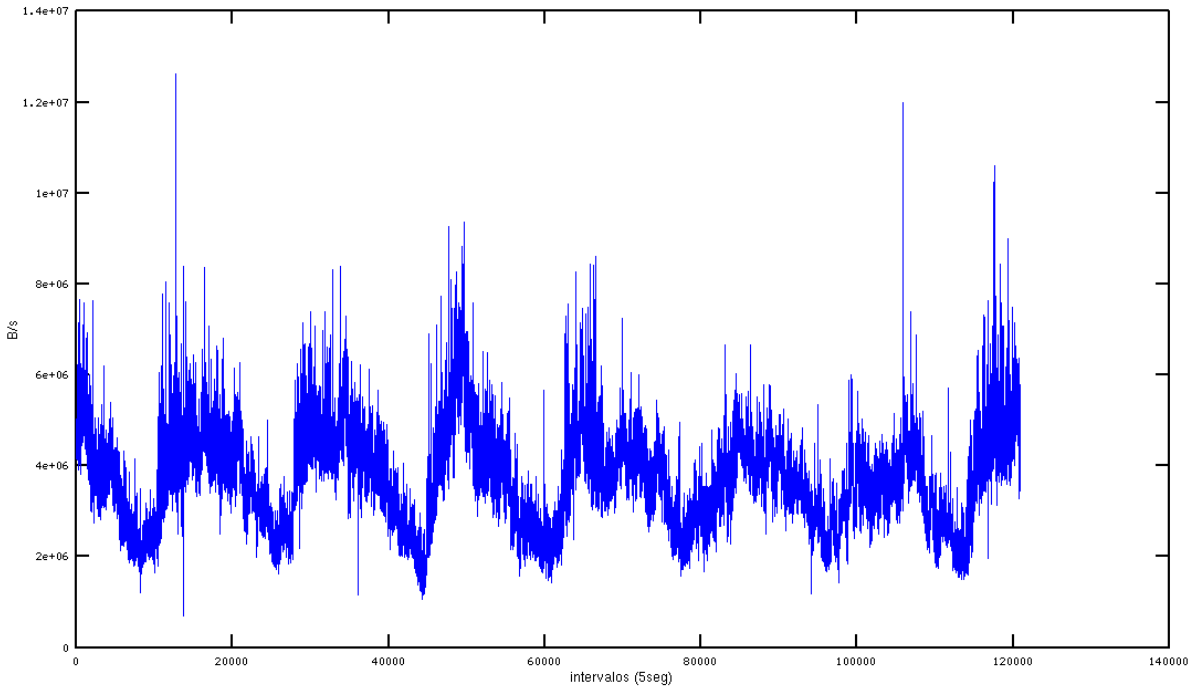

Figura 3.2 – Ejemplo de la misma serie temporal de SNMP de Figura 3.1 con jitter compensado

- Sincronizando los registros. Para encontrar la misma referencia temporal se ha operado de la siguiente manera: a partir de las fechas de inicio del almacenamiento (4 de junio y 1 de septiembre) y del periodo de transmisión de los contadores (5 segundos), se ha partido el fichero de registros de SNMP en dos, uno con las primeras 7 semanas y 4 días y el otro que empieza por la

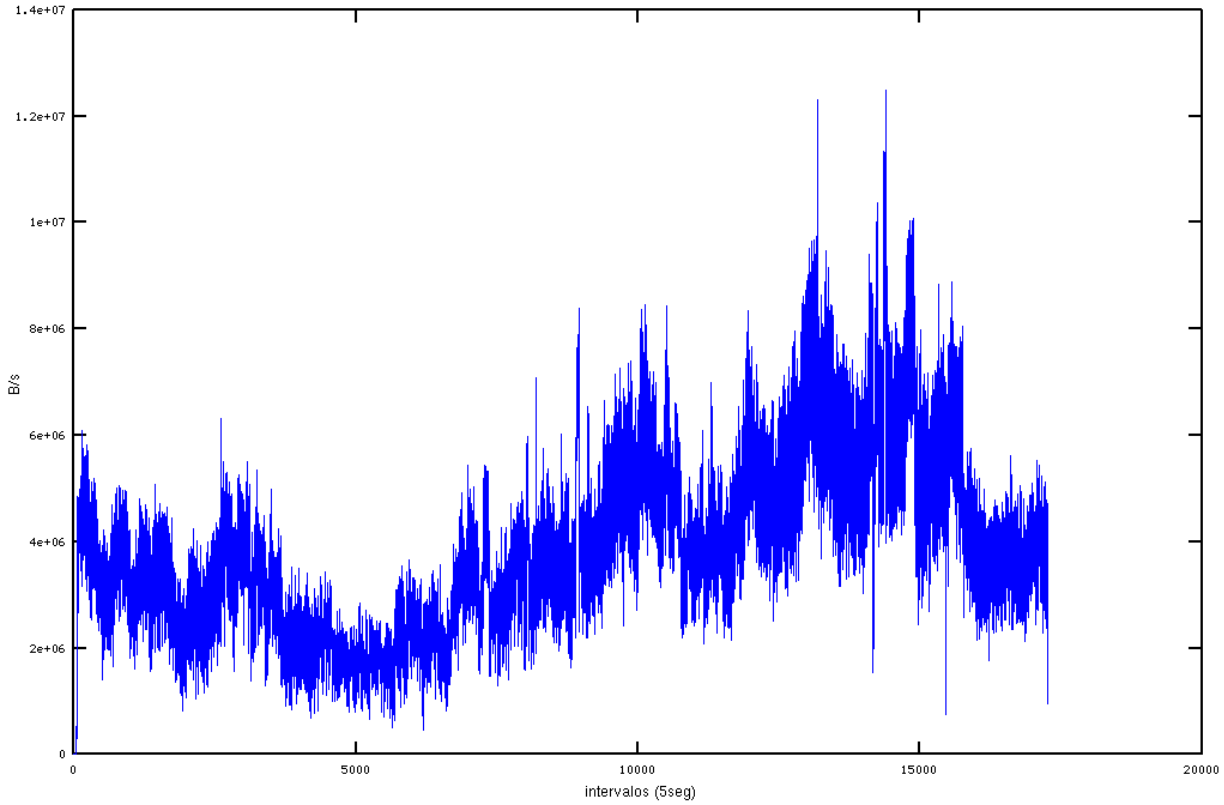

Figura 3.3 – Ejemplo de serie temporal de NetFlow con tiempo de muestreo de 5 segundos

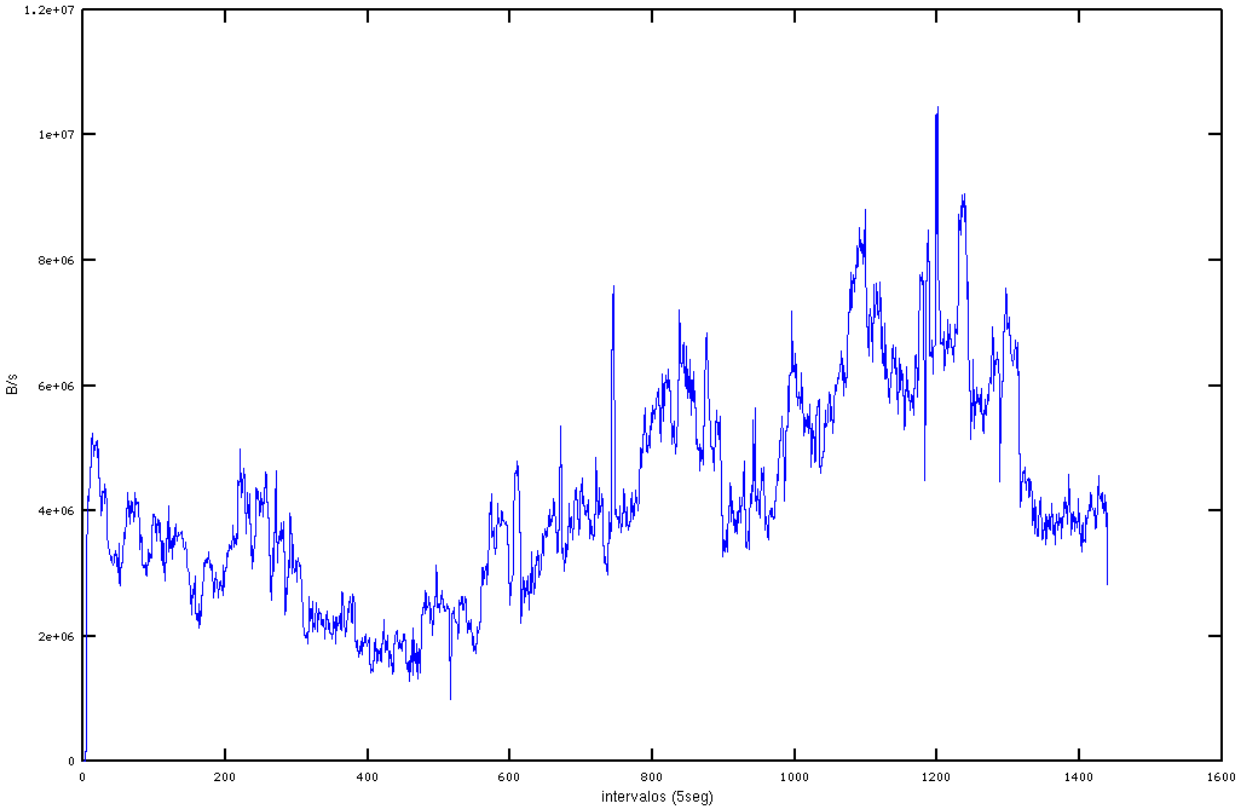

Figura 3.4 – Ejemplo de la misma serie temporal de la Figura 3,3 con tiempo de muestreo de 30 segundos

tarde del 31 agosto con el resto del fichero. Después, teniendo en cuenta que el primer flujo de NetFlow empieza a las 00:03:44 del 1 de septiembre, en una primera fase (aislando un día y medio de SNMP) se han comparado las posiciones del máximo absoluto y de unos máximos relativos para conocer la magnitud del desplazamiento restante. Finalmente, a partir del valor más frecuente se ha refinado el proceso calculando el error cuadrático medio para los valores de desplazamiento en un intervalo bastante ancho entre -20 y 20 muestras de diferencia, también para comprobar que el valor encontrado en la primera fase no hubiese sido afectado por el ruido. El error mínimo corresponde a la máxima semejanza de los registros. En la tabla 3.1 se resumen unos valores del error medio calculado alrededor del mínimo.

Desplazamiento	Error medio
- 4 muestras	0,021
- 3 muestras	0,018
- 2 muestras	0,020

Tabla 3.1 Error por desplazamiento

Como se puede ver en la tabla el ruido ha provocado que todavía hubiera una mínima diferencia entre los máximos relativos y la mejor sincronización de las series temporales, que ha sido detectada y corregida en esta fase. Además, como era de esperar, la ventana de desplazamiento entre -20 y 20 (-100 y 100 segundos de diferencia) muestras ha resultado ser muy ancha para el nivel de error que quedaba. Después de haber sincronizado los registros, como conocemos el tiempo absoluto se han ajustado también el final y el principio de los ficheros de los datos de SNMP para dividirlos justo a la medianoche del 1 de septiembre. Una última cosa que merece la pena comentar es una característica de la serie de NetFlow que se puede apreciar en la figura 3.5 y que hace que sea necesario un filtrado que se va a describir en la sección sucesiva.

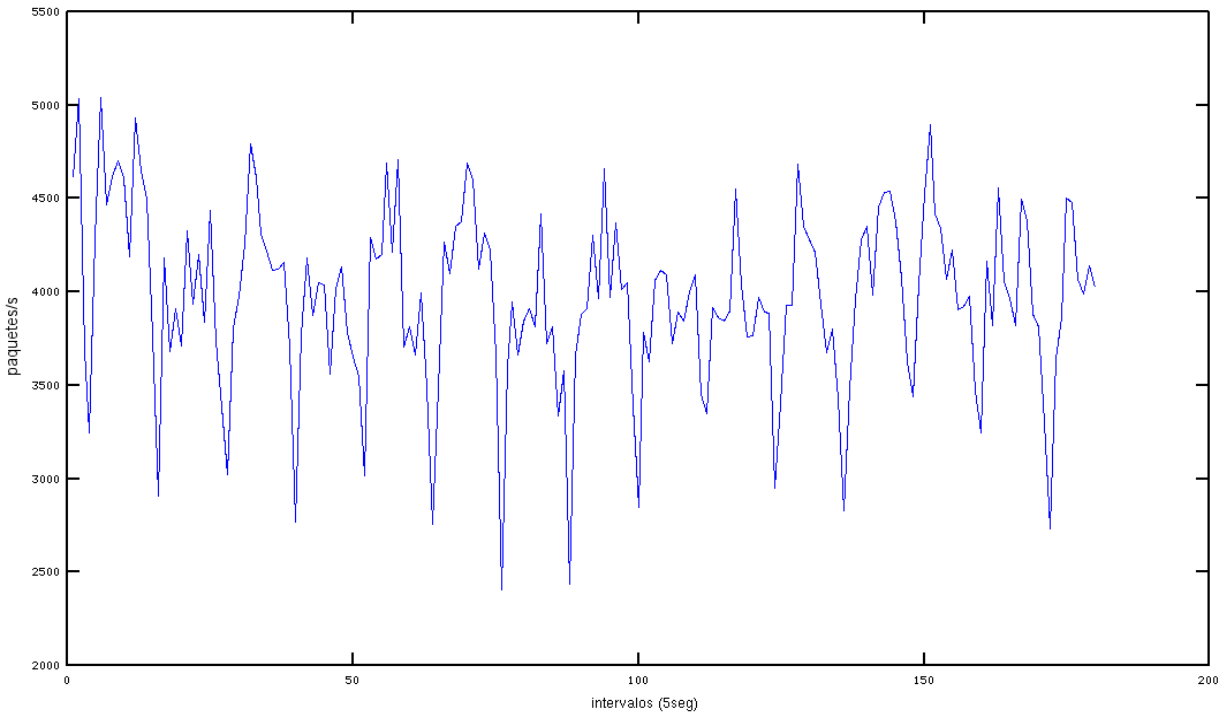

Figura 3.5 – Ejemplo de un trozo de la serie temporal de NetFlow donde se ve la improvisa y periódica bajada del número de paquetes que llegan/salen del router

Esta periódica disminución del número de paquetes generados por el router, al igual que la operación de filtrado, se va a detallar en la próxima sección donde será analizada más detalladamente enseñando sus consecuencias, se intentará descubrir su causa, algunas de las cuales serán justificadamente descartadas.

3.4 Filtrado de la serie temporal de NetFlow

La Fig. 3.5, en la sección anterior, muestra la tasa de transmisión de paquetes y como ya se percibía cada 12 muestras se manifiesta una bajada del número de paquetes generados. Esta bajada genera unas componentes de ruido que afectan negativamente a las medidas de los parámetros de la distribución, así como empeora la comparación de las dos series temporales haciendo aumentar el valor del error. Para resaltar el efecto de esta inesperada característica de la serie de NetFlow se puede ver la figura 3.6 en la cual se propone una gráfica solapada de la tasa de transmisión del enlace, calculada mediante ambos mecanismo de medida a lo largo de un día con muestras cada 5 segundos.

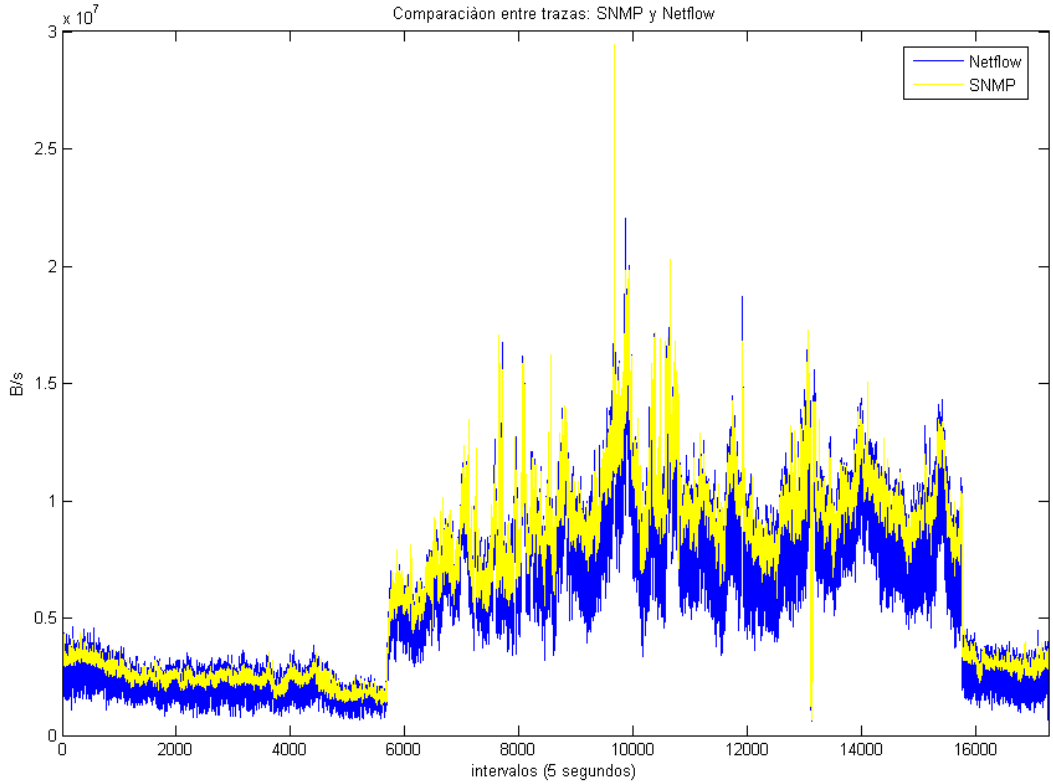

Figura 3.6 – Ejemplo de solapamiento entre las series temporales de NetFlow (en azul) y SNMP (en amarillo) a lo largo de un día, donde se puede ver el efecto del ruido

Como puede observarse, la serie temporal obtenida a partir de NetFlow contiene mucho ruido respecto a la obtenida por SNMP. Para estudiar este ruido, a partir de los registros se ha calculado la FFT (Fast Fourier Transform) de ambas series temporales, calculando así la densidad de potencia en el dominio transformado. Los resultados se pueden ver en la Fig. 3.7. Para apreciar la diferencia y resaltar los picos de alta frecuencia, se ha dibujado la figura usando una escalera logarítmica para el eje de la densidad de potencia. Además siendo las señales reales, su transformada es simétrica por lo que se muestran sólo las frecuencias positivas.
Como se puede ver, la serie temporal obtenida a partir de los registros de NetFlow está caracterizada por la presencia de componentes espectrales de alta frecuencia con un considerable nivel de energía, lo que se traduce en una variabilidad más alta en el dominio del tiempo. Estas

componentes son las que generan el ruido y añaden un valor de error cuadrático medio muy alto. Debido a este hecho, y para mejorar los resultados que se pueden obtener a la hora de extraer los parámetros de una distribución estadística, se ha procedido a filtrar la serie temporal obtenida a partir de los registros de NetFlow. De esta manera se consigue una similitud mayor con la serie temporal obtenida por SNMP.

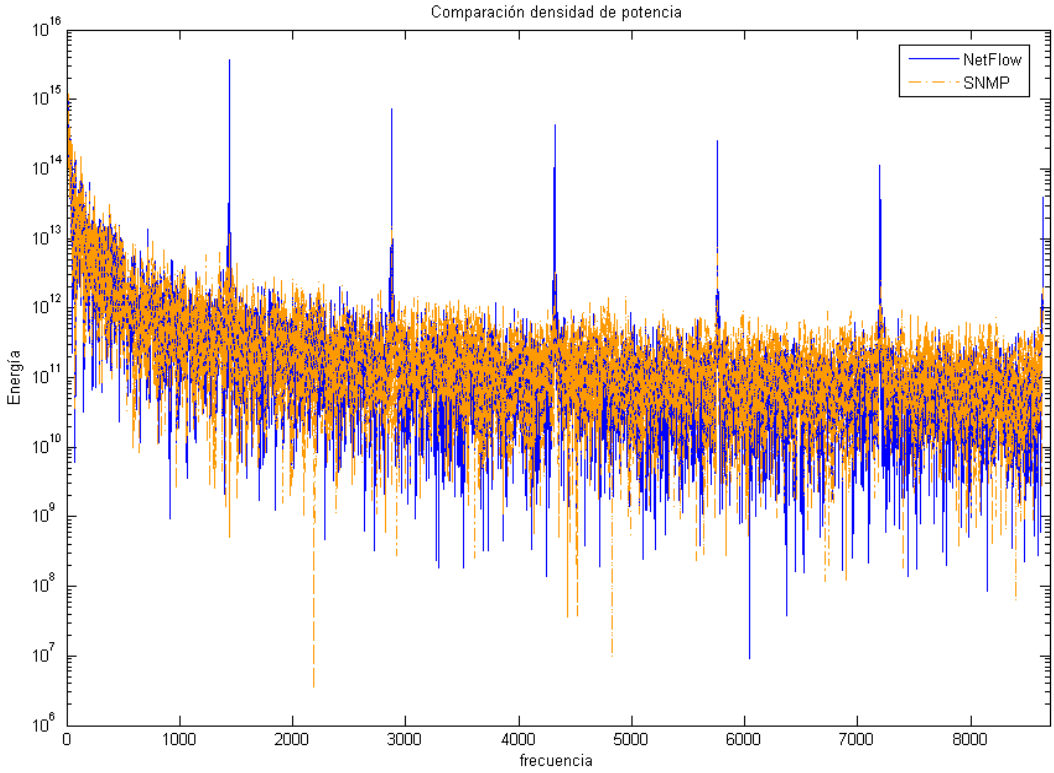

Figura 3.7 – Comparación entre la densidad de potencia de las series temporales de NetFlow (en azul) y SNMP (en amarillo)

La forma en que hay que definir el filtro no es trivial, pues tiene que eliminar el ruido de alta frecuencia pero al mismo tiempo no puede ser un filtro paso bajo, porque al atenuar mucho la banda de alta frecuencia, eliminaría también la información relativa a la alta variabilidad del tráfico de red, quitando a la distribución α-estable su característica principal, y como consecuencia empeorando los resultados de la extracción de los parámetros.
Comparando bien la densidad de potencia de las dos series temporales, lo más destacable es que las componentes de ruido se concentran alrededor de 6 frecuencias fijas y que según los cálculos serían las que se encuentran en correspondencia en los índices 1441, 2881, 4320, 5760, 7200 y 8640 (1/12, 1/6, 1/4, 1/3, 5/12 y 1/2 de la frecuencia de muestreo respectivamente) del vector de densidad de potencia.
Además se ha comprobado que el ruido se encuentra con las mismas características a lo largo de todos los días del periodo considerado y no se limita al día considerado en el ejemplo.
Los resultados que se encuentran en el dominio de las frecuencias se reflejan en el dominio del tiempo, donde se puede comprobar que cada minuto (12 muestras) la tasa de flujos que el router está almacenando en el fichero baja, probablemente como consecuencia de alguna tarea periódica configurada en el mismo router. A continuación se han efectuado otras pruebas para ver si estas componentes pueden de alguna forma depender de la limitación temporal de la serie temporal o de

efectos de aliasing. En el primer caso se ha aplicado una ventana de Hamming (0,54 + 0,46*sin(2πt/NT)) en el dominio temporal para eliminar por completo el primer lóbulo secundario en el dominio transformado. El resultado de esta operación no ha modificado la intensidad del ruido respecto a la de la señal. En el segundo caso se han cambiado la longitud de los intervalos de agregación de los flujos de NetFlow considerando intervalos de 2,5 y 10 segundos. En este caso la densidad de potencia calculada igualmente presentaba ruido pero con sólo 3 componentes en el caso de intervalos de 10 segundos y 12 componentes en el de 2,5 segundos, como se puede ver en las figuras 3.8 y 3.9 respectivamente.

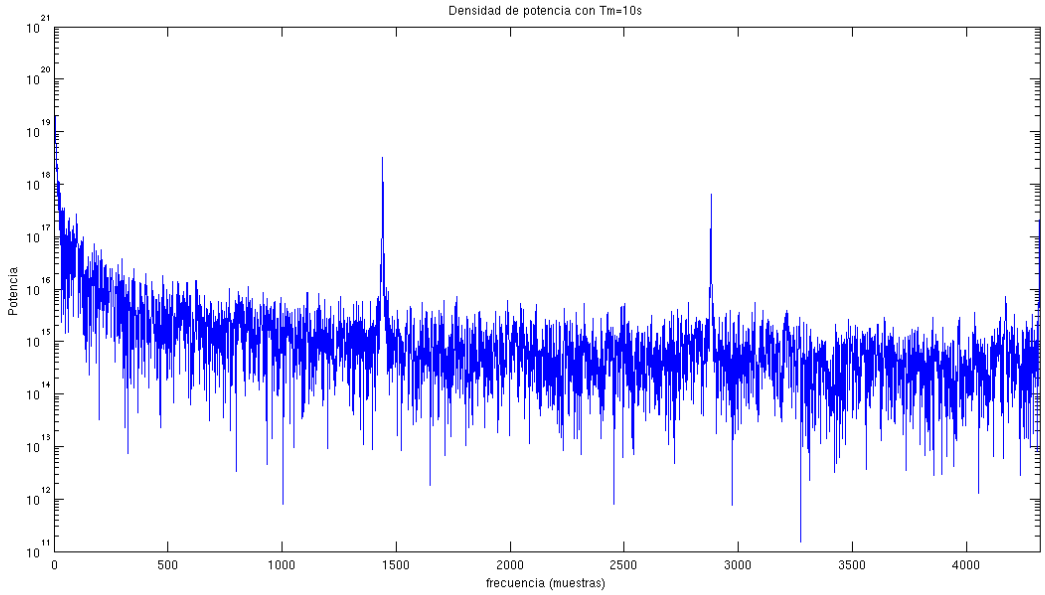

Figura 3.8 – Densidad de potencia de la serie de NetFlow con tiempo de muestreo cada 10 segundos

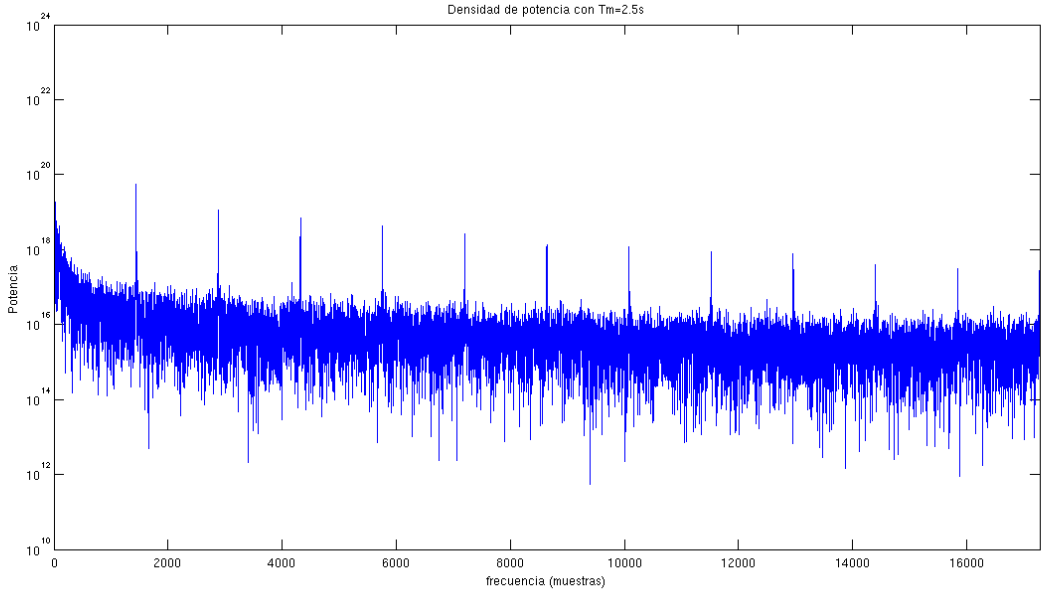

Figura 3.9 – Densidad de potencia de la serie de NetFlow con tiempo de muestreo cada 2.5 segundos

Una última prueba efectuada ha sido la de ver si esta característica se encuentra sólo en los registros de NetFlow generados en el router de Valladolid o si este ruido está presente en los registros de otras ciudades. Para averiguar eso se han considerado registros de otras 14 ciudades (Badajoz, Barcelona, Bilbao, Ciudad Real, Las Palmas, Madrid, Murcia, Oviedo, Palma, Pamplona, Santander, Santiago, Sevilla y Valencia). La densidad de potencia calculada para estas ciudades se encuentra en las figuras 3.10 – 3.23 respectivamente.

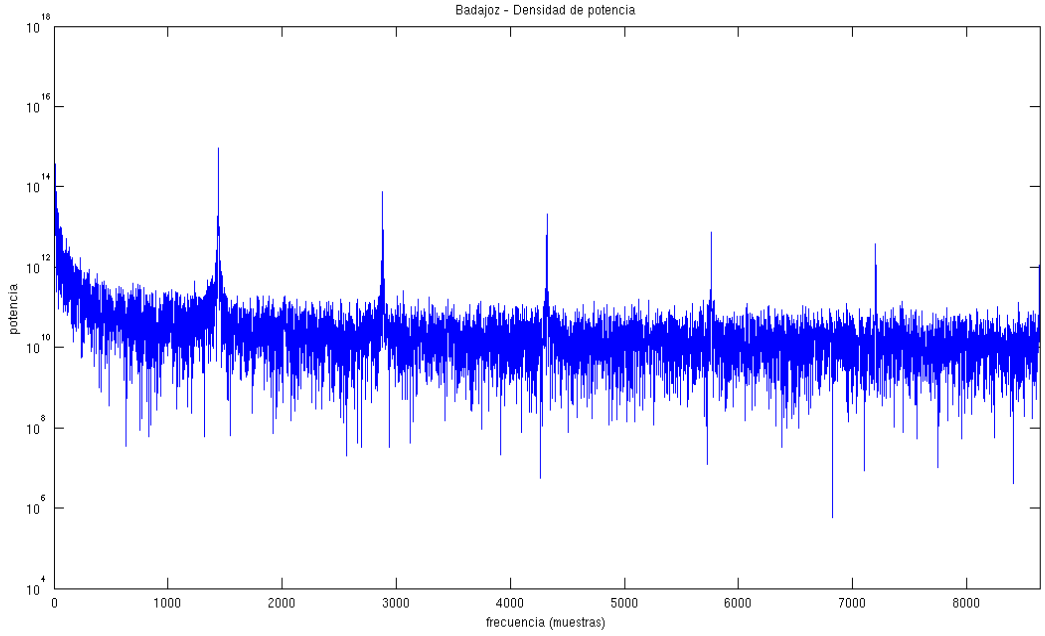

Figura 3.10 – Densidad de potencia de la serie temporal de NetFlow en Badajoz

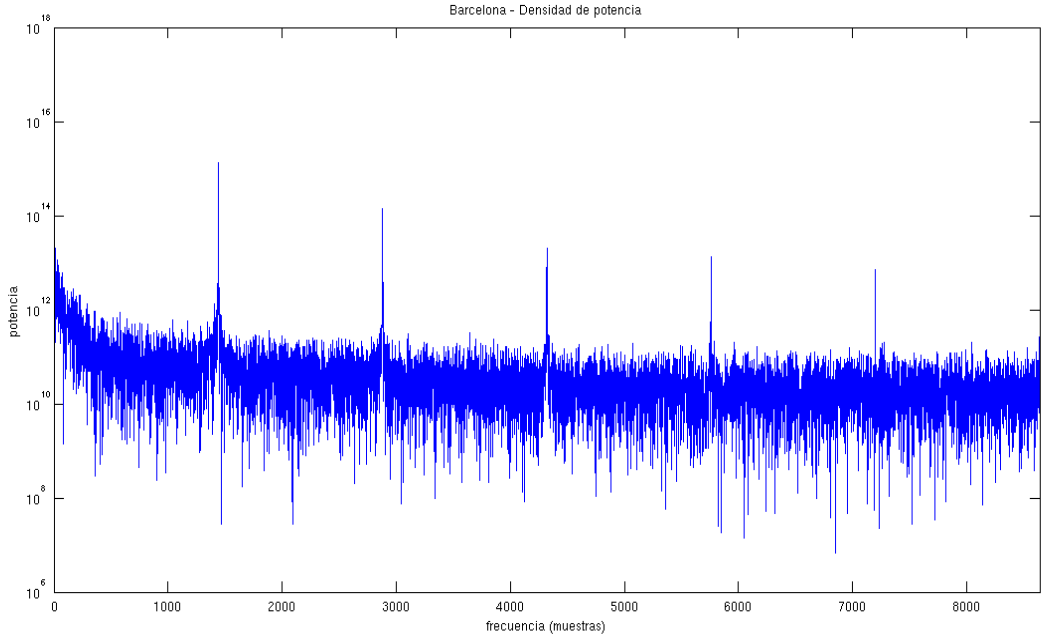

Figura 3.11 – Densidad de potencia de la serie temporal de NetFlow en Barcelona

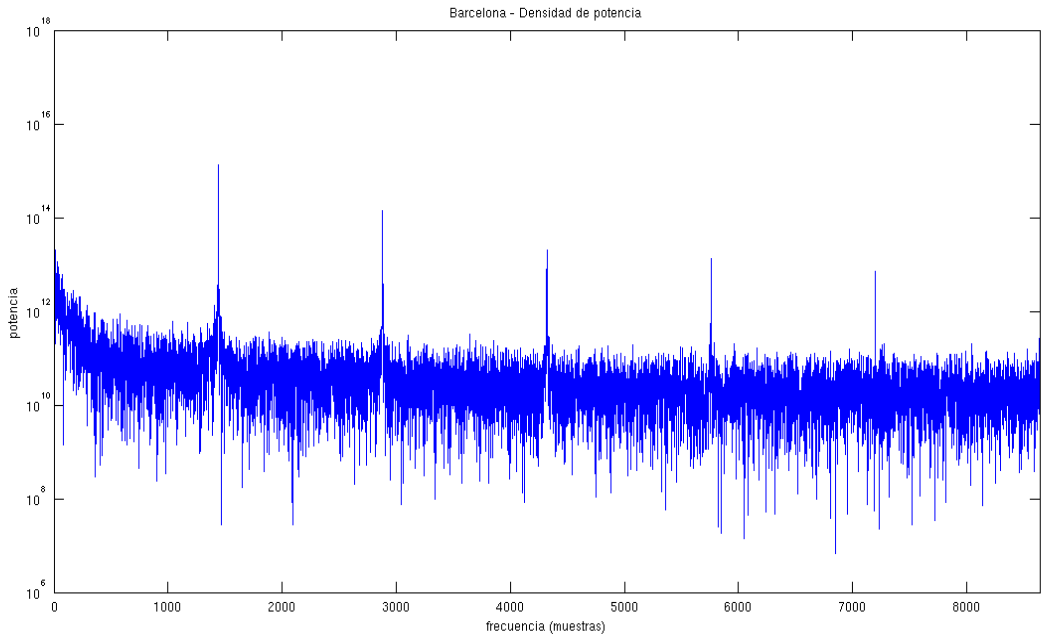

Figura 3.12 – Densidad de potencia de la serie temporal de NetFlow en Bilbao

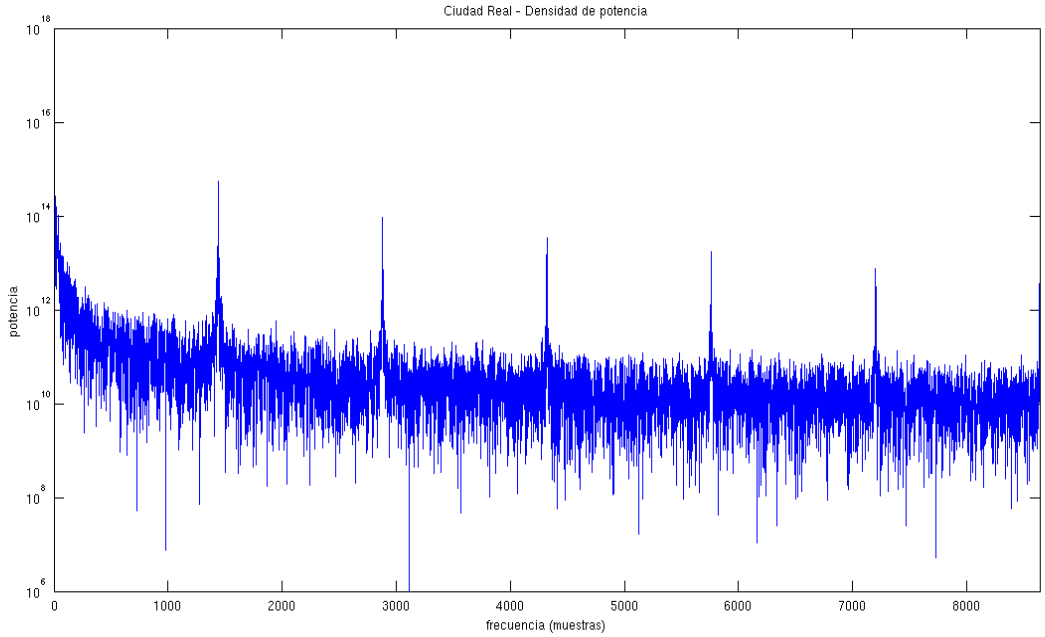

Figura 3.13 – Densidad de potencia de la serie temporal de NetFlow en Ciudad Real

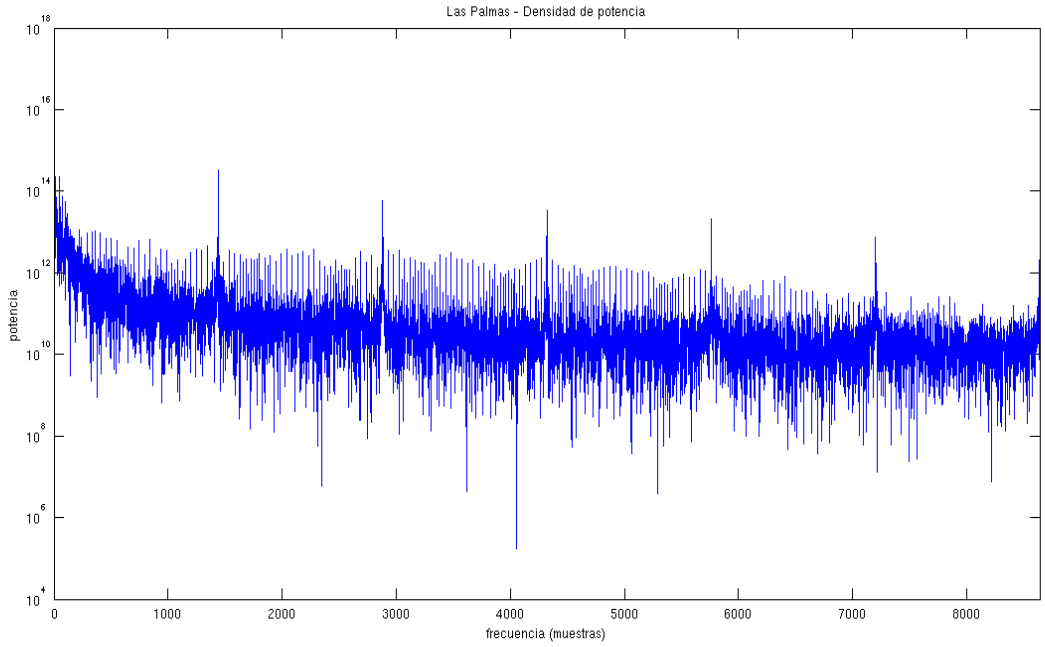

Figura 3.14 – Densidad de potencia de la serie temporal de NetFlow en Las Palmas

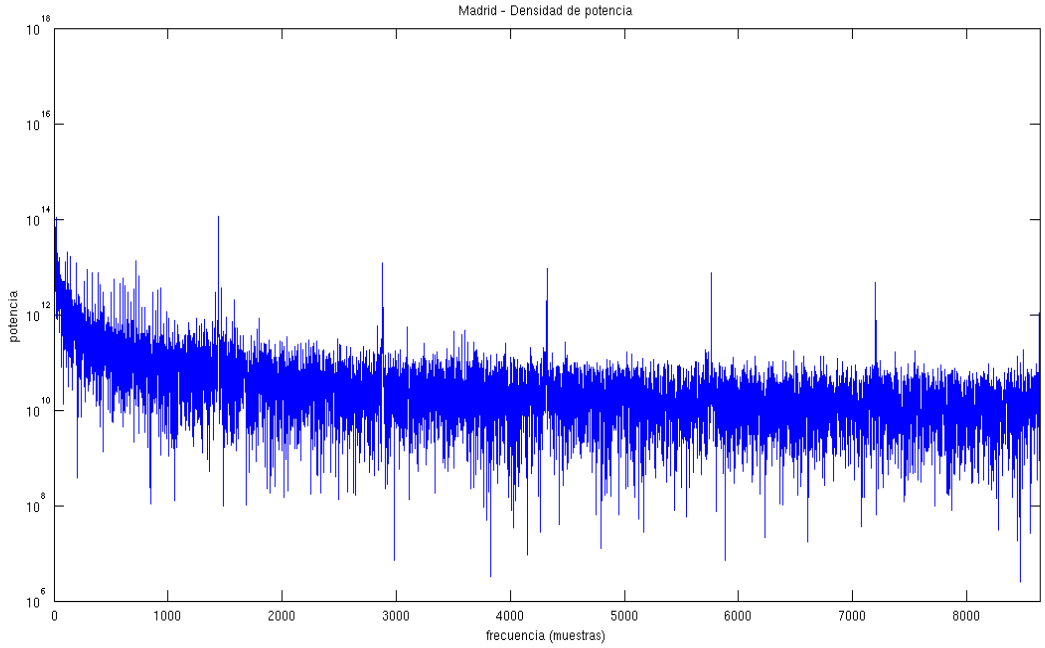

Figura 3.15 – Densidad de potencia de la serie temporal de NetFlow en Madrid

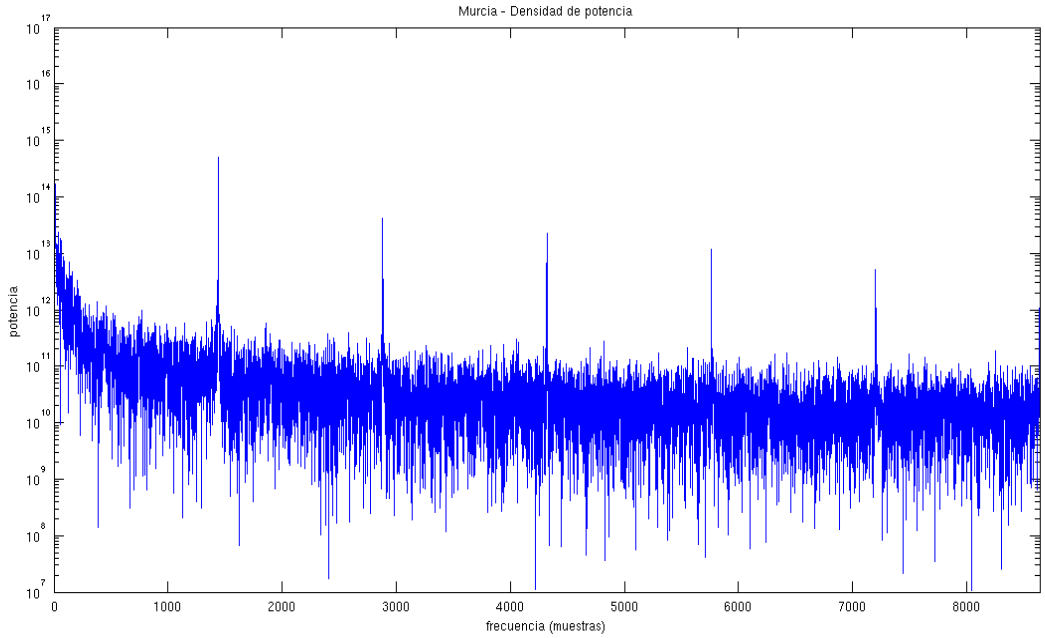

Figura 3.16 – Densidad de potencia de la serie temporal de NetFlow en Murcia

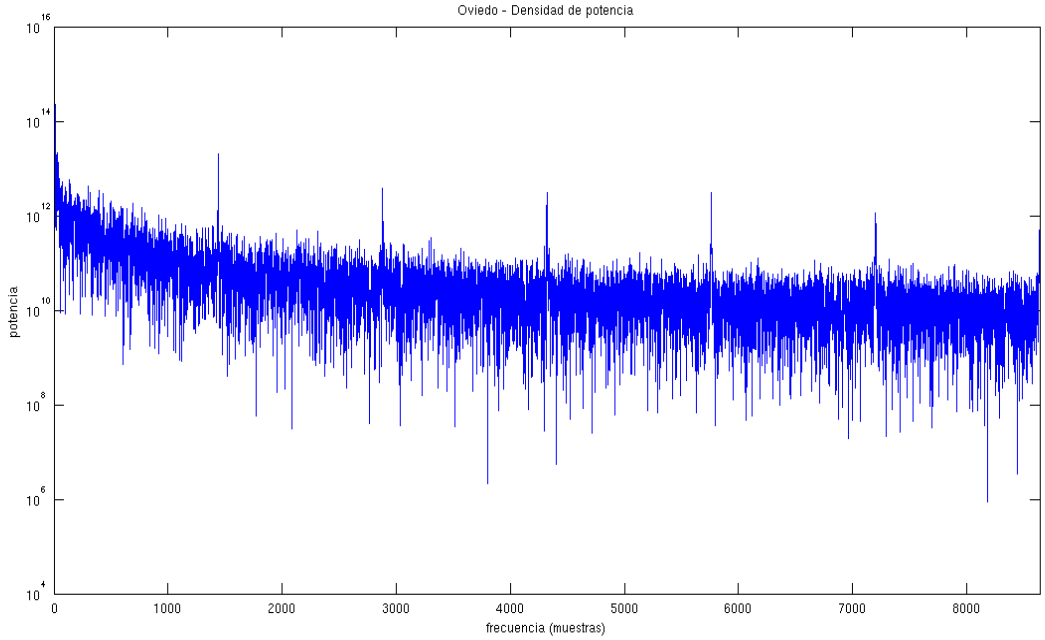

Figura 3.17 – Densidad de potencia de la serie temporal de NetFlow en Oviedo

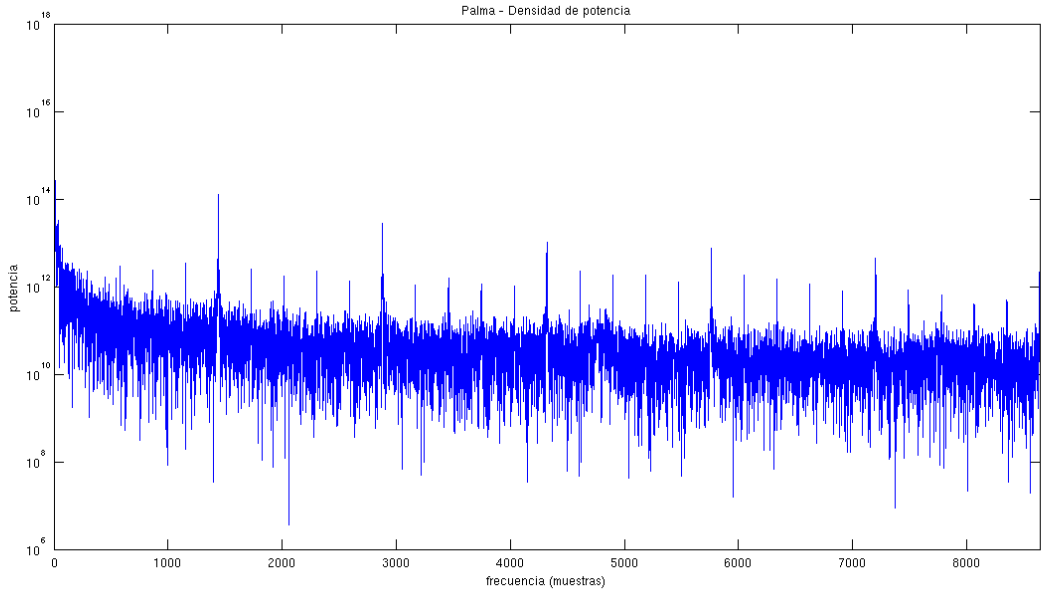

Figura 3.18 – Densidad de potencia de la serie temporal de NetFlow en Palma

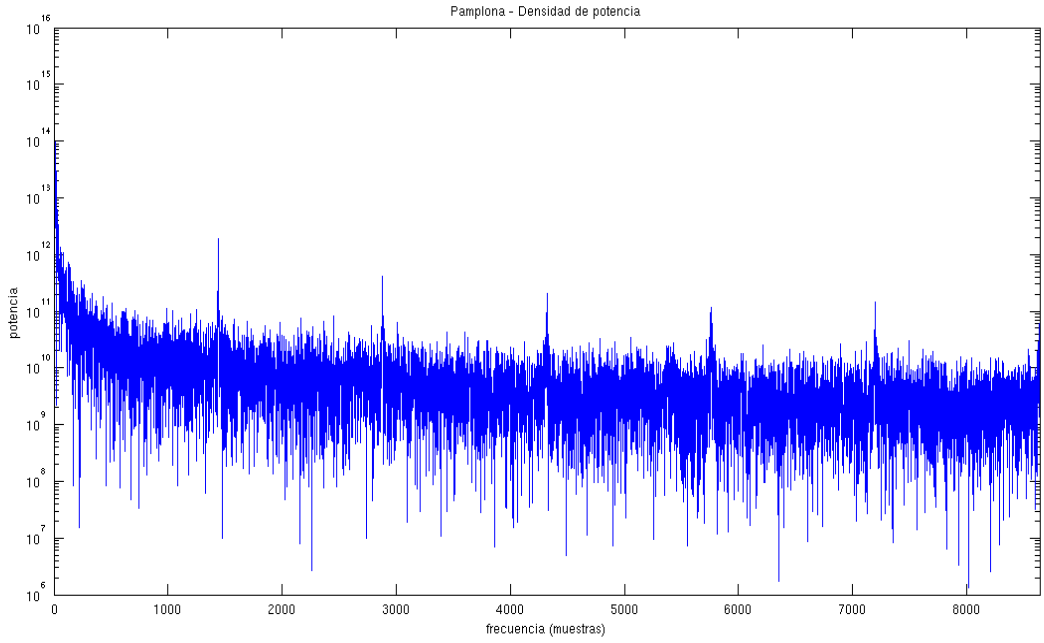

Figura 3.19 – Densidad de potencia de la serie temporal de NetFlow en Pamplona

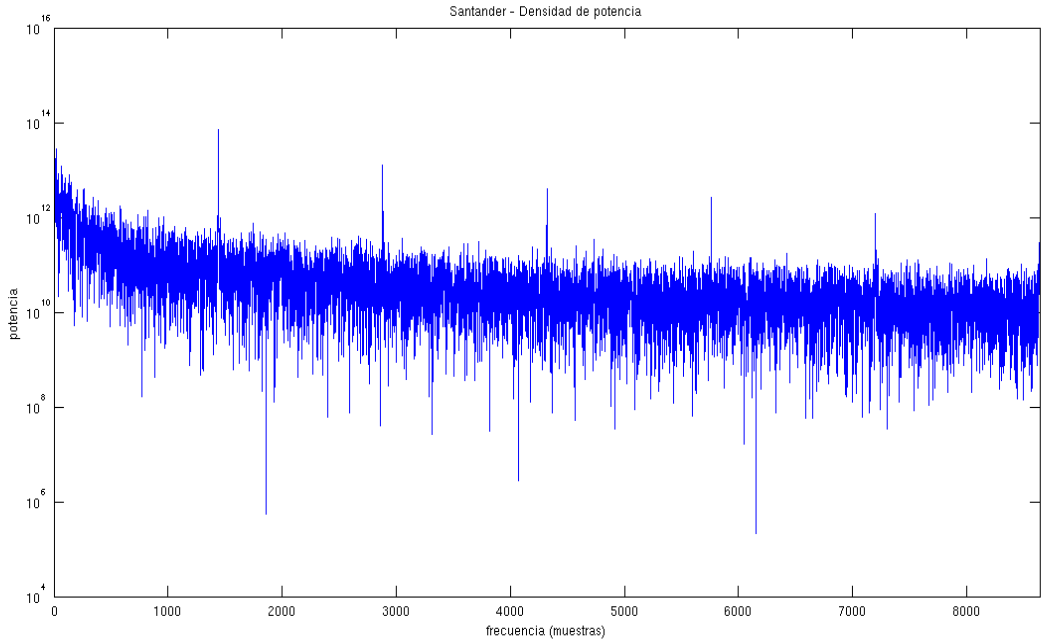

Figura 3.20 – Densidad de potencia de la serie temporal de NetFlow en Santander

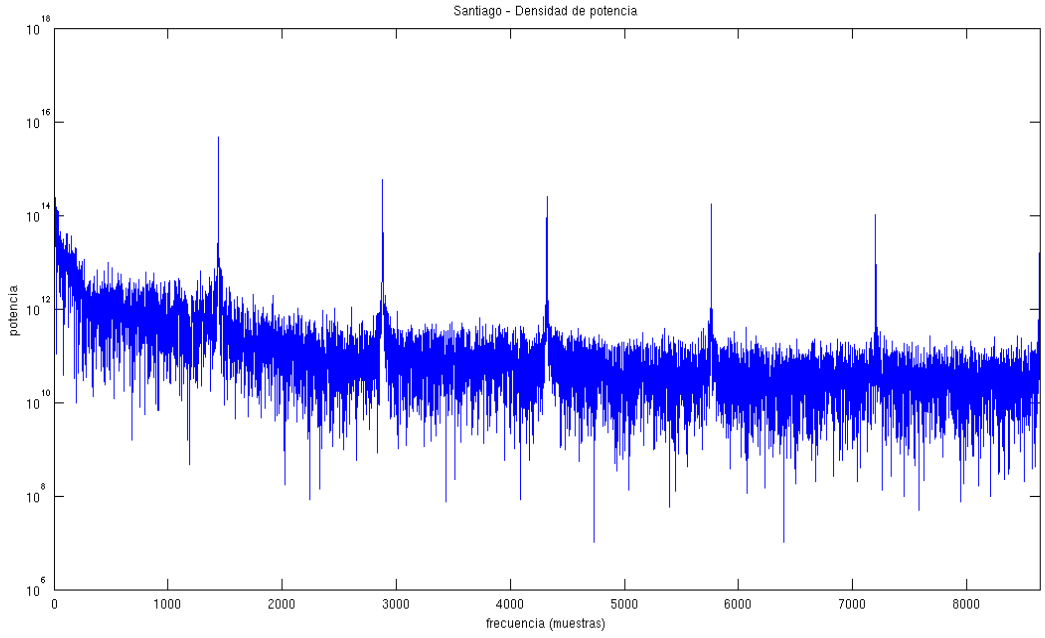

Figura 3.21 – Densidad de potencia de la serie temporal de NetFlow en Santiago

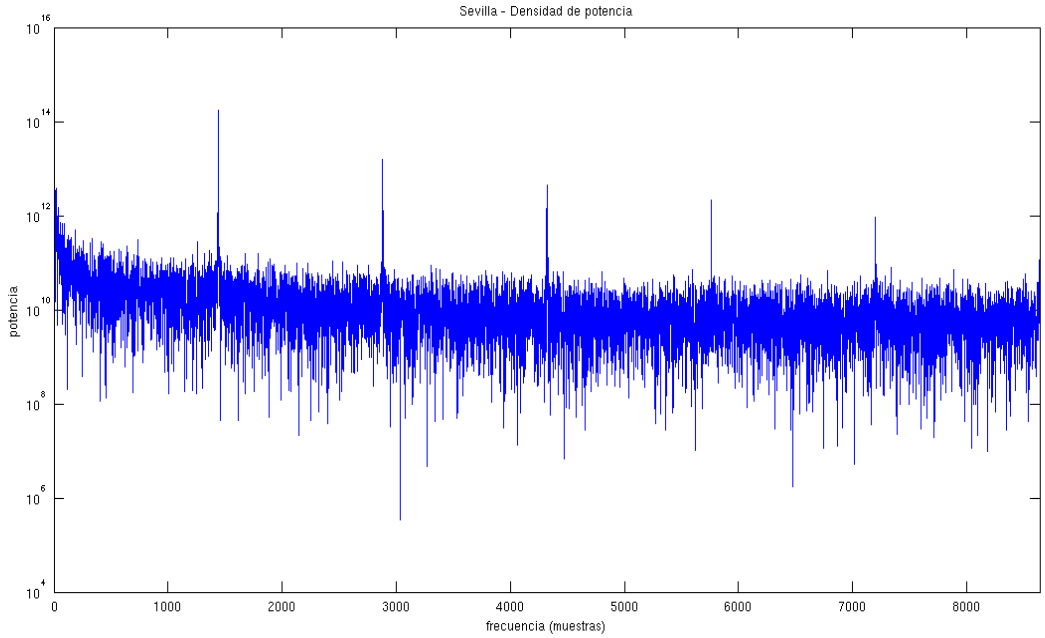

Figura 3.22 – Densidad de potencia de la serie temporal de NetFlow en Sevilla

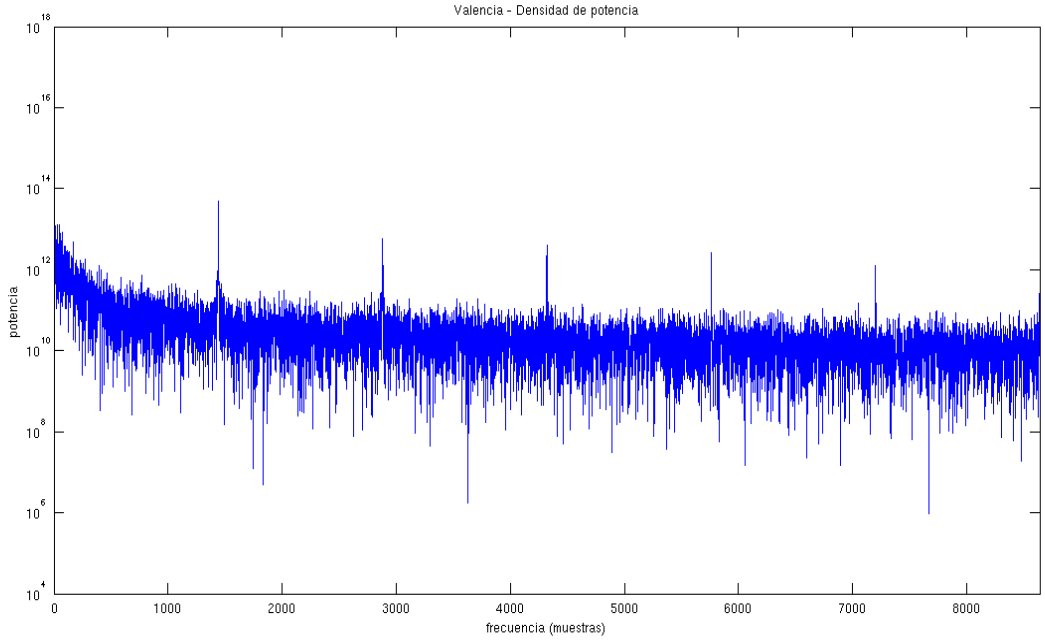

Figura 3.23 – Densidad de potencia de la serie temporal de NetFlow en Valencia

Aprovechando esta característica del ruido se han definido 6 filtros Notch, cada uno de los cuales está centrado en la frecuencia de un armónico. Para fijar la banda de cada filtro se han encontrado los puntos en los cuales el valor de la función de densidad sube de 3 dB (el doble) y ésta resulta ser de aproximadamente 81 muestras para los primeros 5 filtros y 40 para el último. Una vez definidos los filtros se pasa a filtrar la serie temporal obtenida a partir de los registros de NetFlow. En la figura 3.24 se puede ver la función de transferencia total obtenida de aplicar secuencialmente los 6 filtros Notch proyectados, mientras que en la tabla 3.2 se resumen los coeficientes de los filtros.

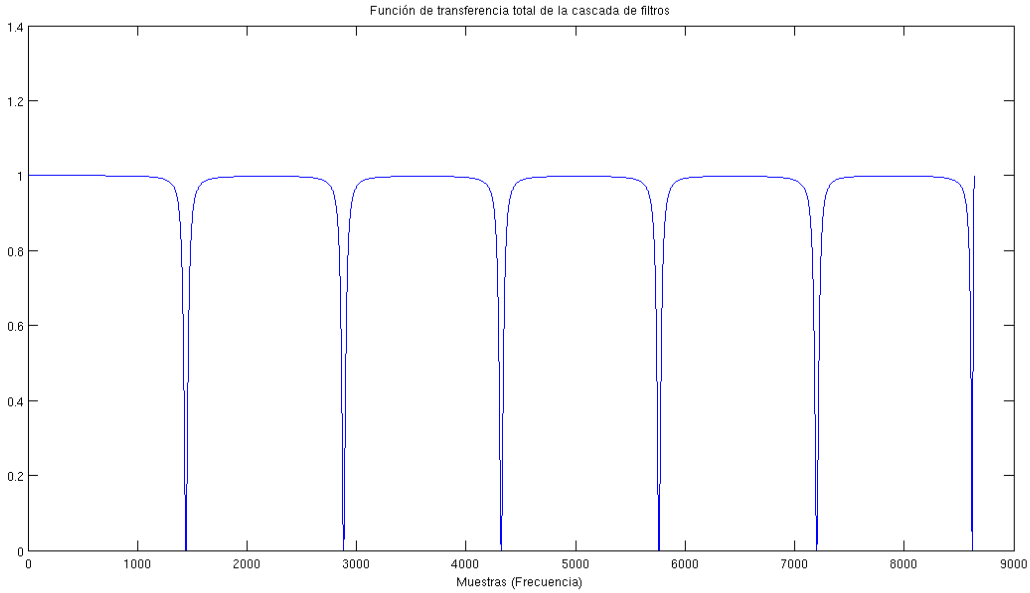

Figura 3.24 – Función de transferencia del banco de filtros Notch

Armónicas	Numerador			Denominador		
I	0,9927	-1,7190	0,9927	1	-1,7190	0,9853
II	0,9927	-0,9920	0,9927	1	-0,9920	0,9853
III	0,9927	-1,2156e-16	0,9927	1	-1,2156e-16	0,9853
IV	0,9927	0,9927	0,9927	1	0,9927	0,9853
V	0,9927	1,7193	0,9927	1	1,7193	0,9853
VI	0,9963	1,9926	0,9963	1	1,9926	0,9826

Tabla 3.2 – Resumen de los coeficientes de los filtros Notch

Para filtrar la serie temporal se han utilizado dos funciones distintas; la primera aplica a la vez numerador y denominador del filtro IIR (Infinite Impulse Response) a la función como en una operación normal de filtrado, mientras que la segunda, para proporcionar una fase plana, aplica el numerador en el sentido creciente del tiempo y el denominador en sentido inverso. Aunque la segunda función es más compleja, a la hora de extraer los parámetros, genera unos valores menos parecidos, especialmente para anchura y valor medio y como consecuencia un peor ajuste entre distribución teórica y empírica. Por esta motivación se ha preferido utilizar la primera función para la operación de filtrado.

En la Fig. 3.25 se puede ver la gráfica comparada entre la serie temporal de SNMP y la de NetFlow después del filtrado. Como se puede observar las dos series temporales después del filtrado (así como la densidad de potencia) se parecen más y como consecuencia también el error cuadrático medio ha bajado en un orden de magnitud.

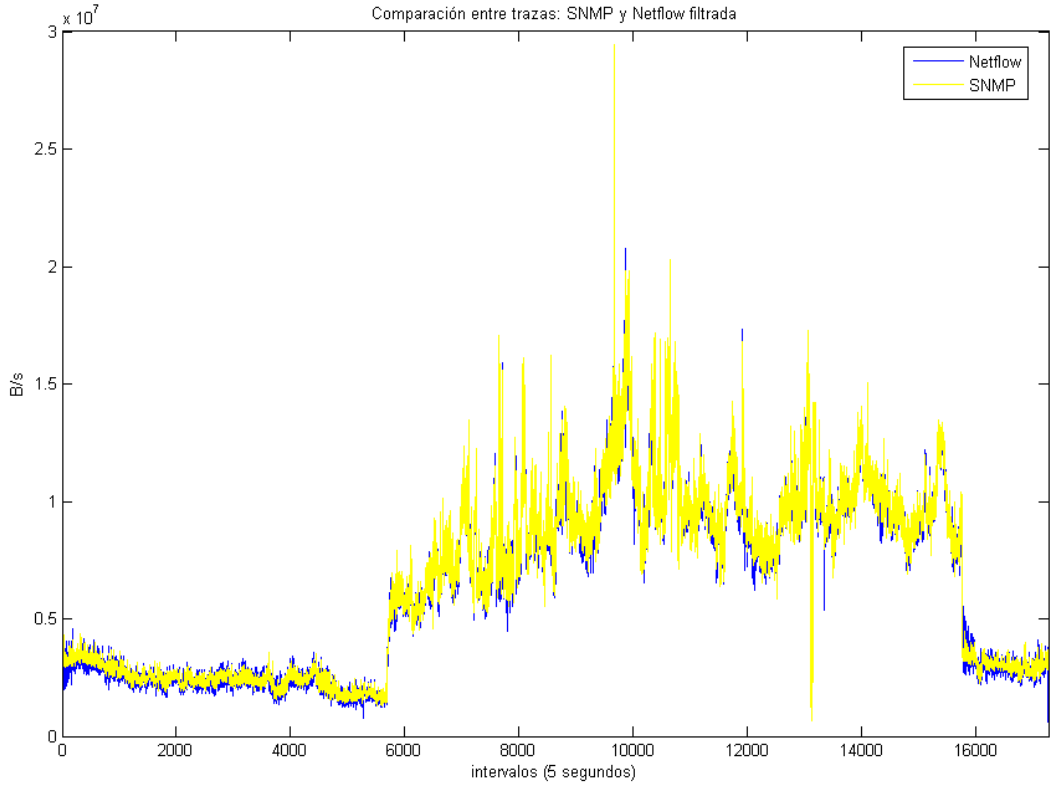

Figura 3.25 – Ejemplo de solapamiento entre las series temporales de NetFlow (en azul) y SNMP (en amarillo) de Figura 3.6 después haber filtrado la primera.

3.5 Conclusiones

Como se ha podido ver a lo largo de este capítulo, para poder comparar la información de los dos registros primero se ha empezado ilustrando las diferencias entre los dos protocolos utilizados prestando atención a la forma que utilizan ambos registros para almacenar esta información. A partir de esta diferencia ha sido posible transformar cada uno de los registros a través de distintos algoritmos en la relativa serie temporal. Como el registro de SNMP usa una referencia temporal relativa, mientras que el de NetFlow utiliza una referencia absoluta, surge la necesidad de sincronizar las series temporales. Esta tarea, como se ha podido ver en la sección precedente, se ha efectuado en dos fases: una primera para una sincronización encontrada a través de los máximos absolutos y locales y una segunda hecha minimizando el error cuadrático medio por diferentes desplazamientos para aislar eventuales errores residuos debidos al ruido. Una vez sincronizadas las dos series, se ha visto tras su comparación que todavía quedaban unas componentes de ruido de alta frecuencia en la serie temporal de NetFlow. Este ruido comporta una variabilidad temporal más alta con la consiguiente subida del valor del error. Con el propósito de reducir el error se ha analizado la densidad de potencia en el dominio de las frecuencias para encontrar las componentes de ruido e intentar filtrarlo y al mismo tiempo, si fuera posible, encontrar su causa. En el cálculo de la densidad de potencia resaltan seis componentes de alta frecuencia en correspondencia con los índices 1441, 2881, 4320, 5760, 7200 y 8640 (1/12, 1/6, 1/4, 1/3, 5/12 y 1/2 de la frecuencia de muestreo respectivamente). Conociendo la posición de las componentes espectrales del ruido se ha podido proyectar el filtro, o más bien la cascada de filtros, ya que con un sólo filtro no es posible eliminar selectivamente el ruido dejando la señal inalterada. Como anteriormente he comentado se ha elegido una distribución de tipo α-estable para modelar la tasa de bytes por unidad de tiempo para aprovechar de su característica de alta variabilidad. Un filtro paso bajo aunque elimine muy bien el ruido, eliminaría también componentes de la señal que permiten un mejor ajuste del modelo. Por eso se ha preferido una cascada de filtros Notch que bien diseñados tienen la ventaja de filtrar sólo el ruido. Además se ha tratado de encontrar las causas del ruido: después de dos pruebas que han sido útiles para demostrar que este comportamiento no depende ni de la limitación temporal de la trazas que genera aliasing (aplicando una ventana de Hamming a la serie temporal) ni del tiempo de muestreo (utilizando distintos valores temporales para los intervalos cambia la densidad pero no cambia el periodo temporal en el cual se manifiesta el ruido). Se han observado las características frecuenciales de otros registros de NetFlow procedentes de otras ciudades españolas. Calculando la densidad de potencia de las series temporales generadas a partir de los registros (en Badajoz, Barcelona, Bilbao, Ciudad Real, Las Palmas, Madrid, Murcia, Oviedo, Palma, Pamplona, Santander, Santiago, Sevilla y Valencia) y en todos los 14 casos se encuentran las mismas componentes espectrales de ruido, es decir, que a pesar de la ciudad considerada, las componentes del ruido se encuentran en correspondencia con los mismos índices y, como consecuencia, con la misma relación respecto a la frecuencia de muestreo.

Aunque esto no explica en absoluto la causa del ruido que afecta las series temporales de NetFlow, dejando como principal hipótesis la de una configuración en la red que interfiere periódicamente con el almacenamiento de los flujos en el registro, es muy importante saber que en cada caso se encuentra ruido añadido a la señal, con la consecuente necesidad de filtrar la traza, y que se puede utilizar el mismo banco de filtros para trazas procedentes de cualquier ciudad analizada.

Finalmente se puede ver la comparación entre las series temporales de SMNP y NetFlow filtrada, donde destaca la mejora obtenida con la aplicación del filtro que ha permitido reducir de un orden de magnitud el error cuadrático medio.

A continuación, en el siguiente capítulo, se describe como a partir de estas series temporales se van a extraer y comparar los parámetros de la distribución.

Capítulo 4

Evaluación de los parámetros de la distribución

4.1 Introducción

Después de las operaciones previas necesaria para obtener dos series temporales homogéneas y sincronizadas se puede pasar a la siguiente fase de evaluación de los parámetros. Probablemente la semejanza temporal entre las dos series no es una condición suficiente para que se parezcan también los parámetros de la distribución α-estable, pero seguramente es un buen punto de partida para que eso se verifique.
Uno de los objetivos de este trabajo es justo el de averiguar si usando el registro de NetFlow es posible encontrar parámetros iguales o parecidos a los encontrados a partir del registro SNMP. Así que la tarea de evaluación de los parámetros que se va a describir en este capítulo tiene mucha importancia. Además si la intención no es sólo la de modelar el tráfico de red, sino también la de detectar errores o anomalías, la importancia de esta fase es todavía más evidente.
Para extraer los parámetros de la distribución α-estable se ha utilizado Matlab, en concreto la función "stblfit", basada en el método de Koutrouvelis que se encuentra en el paquete STBL_CODE [10], que comprende todas las funciones relativas a esta distribución. En la página web de este paquete se pueden encontrar ulteriores informaciones sobre la distribución y los algoritmos comprendidos en el paquete, incluso las relativas a los algoritmos que no se han utilizados en este trabajo.
En la siguiente sección se introduce esta fase describiendo la elección de la ventana temporal y la operación de resincronización de las trazas. Esta última operación no estaba planeada desde el principio pero se necesita dado que a lo largo de 28 días el contador de SNMP (el cual usa un contador incremental y no una referencia temporal absoluta) presenta un ligero desfasamiento con respecto al contador temporal de NetFlow. En la mayoría de los casos la pérdida de sincronización se manifiesta en correspondencia de temporáneas ausencias de datos en el registro de SNMP, probablemente a causa de unos paquetes perdidos o dañados. Siempre en la siguiente sección se describe la evaluación y la comparación de los parámetros de la distribución. El capítulo se concluye con un resumen de los resultados y unas consideraciones generales.
A continuación se evalúan los parámetros, ilustrando los resultados obtenidos por las dos series temporales.

4.2 Evaluación de los parámetros del modelo α-estable

En esta sección, después de describir brevemente las condiciones iniciales para modelar el tráfico de red se muestran los resultados obtenidos.
En primer lugar, para la extracción de los 4 parámetros se necesitan ventanas temporales en las cuales el tráfico de red se pueda considerar estacionario. Exactamente como en [3], se ha elegido una duración de media hora para cada ventana. Como las series temporales tienen una resolución de 5 segundos, la longitud de las ventanas asegura un buen nivel de estacionariedad así como un número suficiente de muestras por cada ventana (1800 segundos/5 segundos por muestra = 360 muestras), de forma que se puede redefinir bien el modelo, disminuyendo el valor de error entre las funciones de distribución acumulada de la ventana con respecto a la distribución calculada a partir de los 4 parámetros extraídos.
En segundo lugar hay que corregir la pérdida de sincronía de la serie temporal de SNMP que se genera como consecuencia de errores en la secuencia de los contadores, como puede ocurrir cuando se pierden partes de datos en el registro de SNMP. Para mantener la referencia temporal se han vuelto a sincronizar ambas series temporales cada 12 horas. En la figura 4.1 se puede ver el efecto de la pérdida de sincronización, mientras que la figura 4.2 ilustra las dos series temporales después de la resincronización.
Una última cosa que hay que considerar es que a veces se pueden encontrar ventanas en las cuales faltan muestras de una o de las dos trazas. El cálculo de los parámetros, así como el de los errores, ha sido ignorado en dichas ventanas, ya que no serían de ninguna utilidad para evaluar las características del sistema y no proporcionarían información fiable a la hora de elegir si el tráfico es normal o anómalo en una ulterior fase de detección de errores o anomalías.

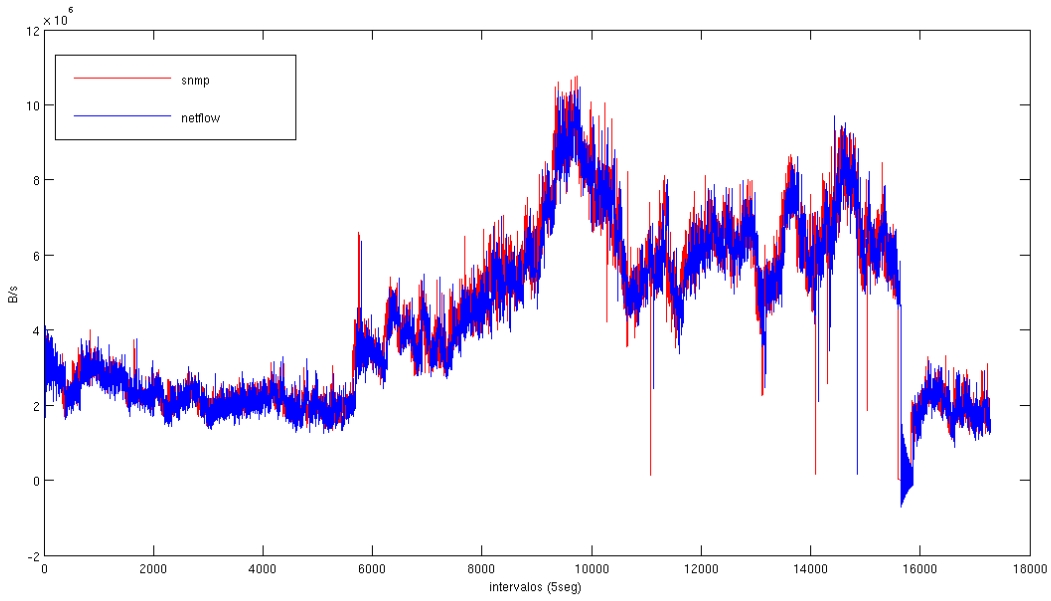

Figura 4.1 – Efecto de la perdida de sincronización entre las dos series temporales

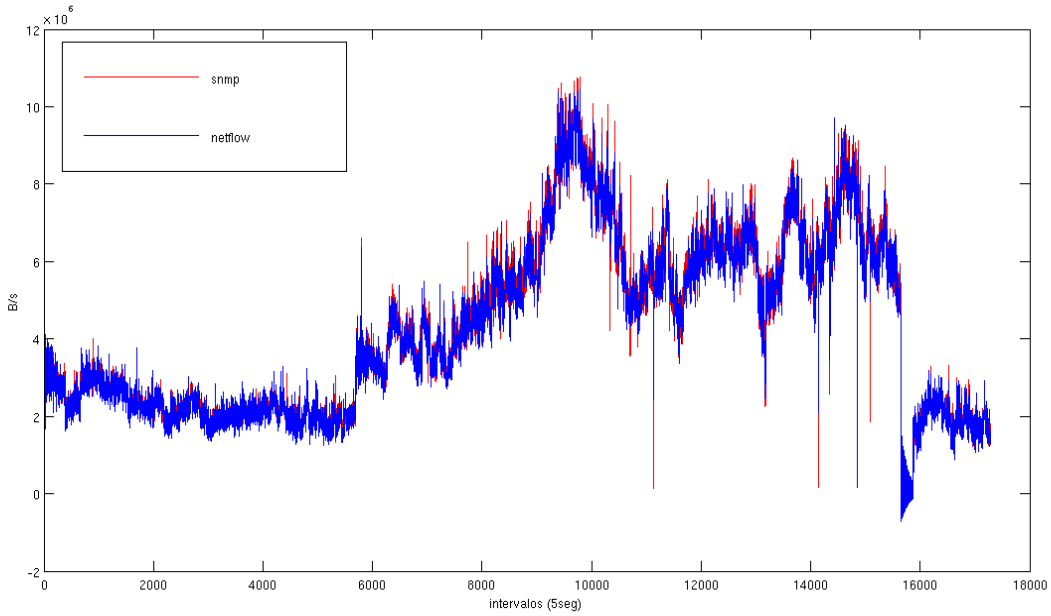

Figura 4.2 – Efecto de la resincronización de la serie temporal de SNMP

En las figuras 4.3, 4.4 y 4.5 se puede ver un ejemplo de la diferencia entre los parámetros de una distribución α-estable, estimados a partir de la serie temporal de SNMP y de la de NetFlow. De momento esta diferencia sólo permite una evaluación cualitativa entre los modelos calculados a partir de las dos series.

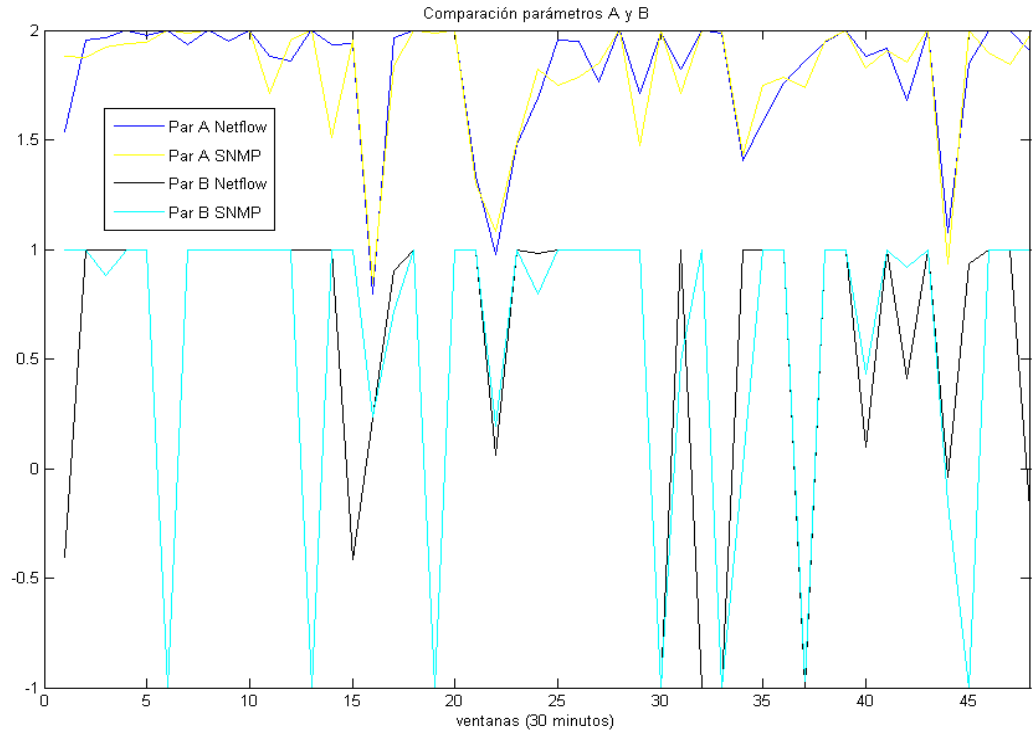

Figura 4.3 – Comparación entre los parámetros α (A) y β (B) de las series temporales

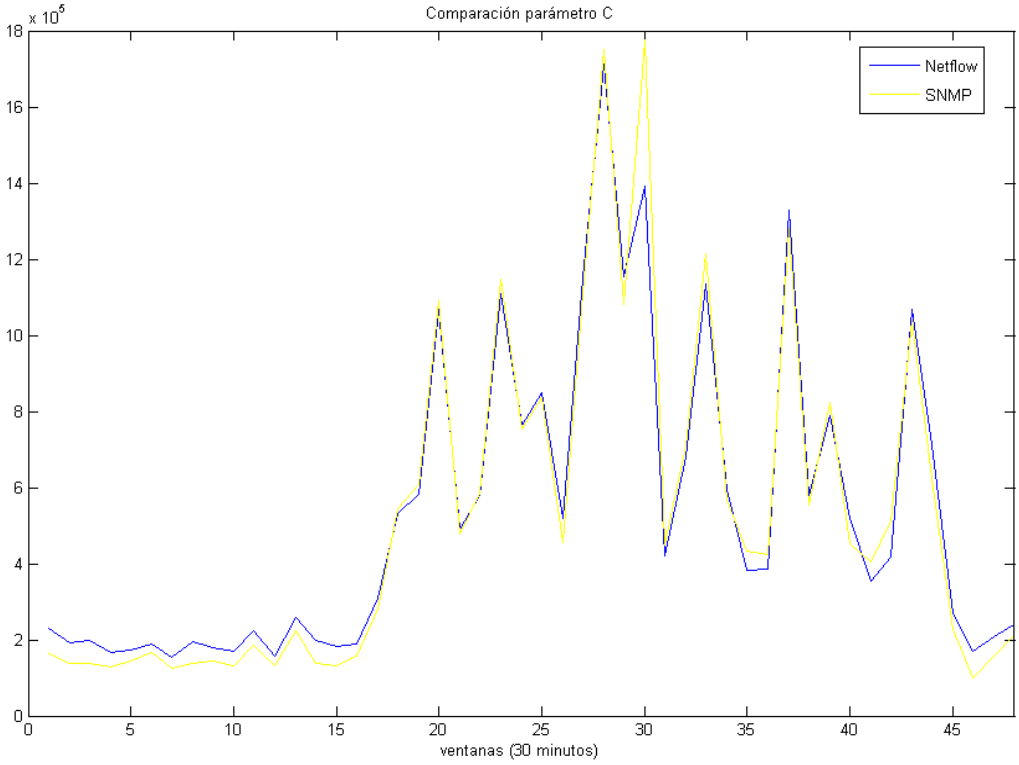

Figura 4.4 – Comparación entre el parámetro γ (C) de las series temporales

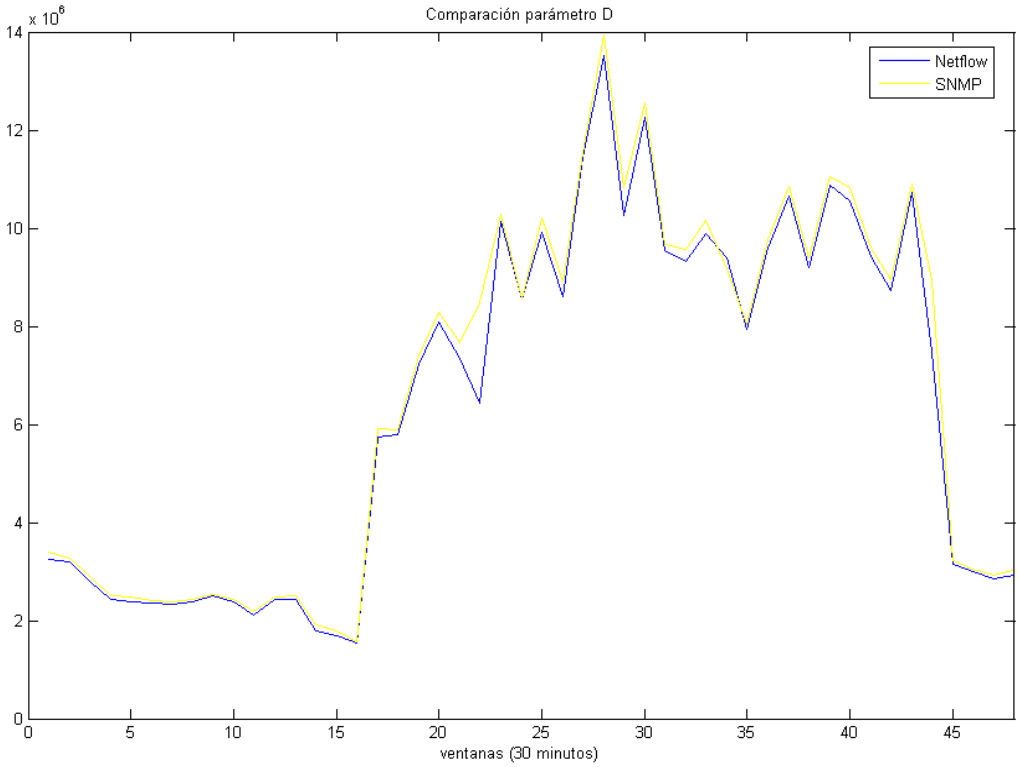

Figura 4.5 – Comparación entre el parámetro δ (D) de las series temporales

Una evaluación, siempre cualitativa, entre los modelos y las muestras reales se encuentra en las figuras 4.6, 4.7, 4.8 y 4.9, donde se puede ver un ejemplo de ajuste entre las funciones de distribución acumulada estimadas (a partir de los parámetros extraídos) y empíricas (a partir de las muestras de la ventana) par SNMP y NetFlow.

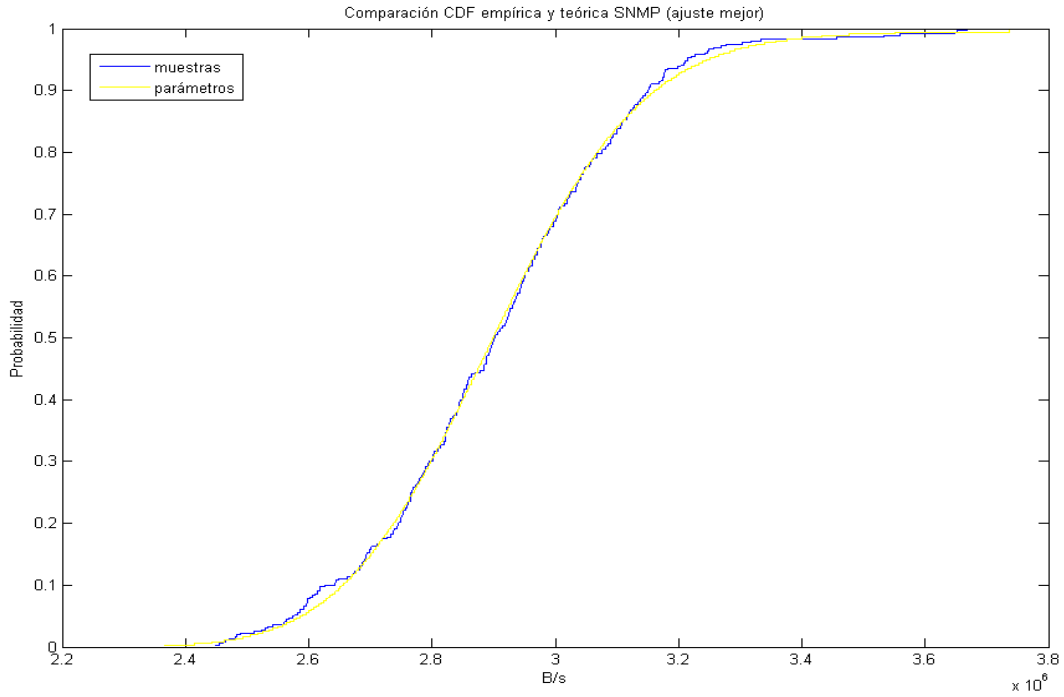

Figura 4.6 – Comparación entre las cdf calculadas a partir de los Parámetros y las muestras de SNMP en el caso en que este se ajuste mejor

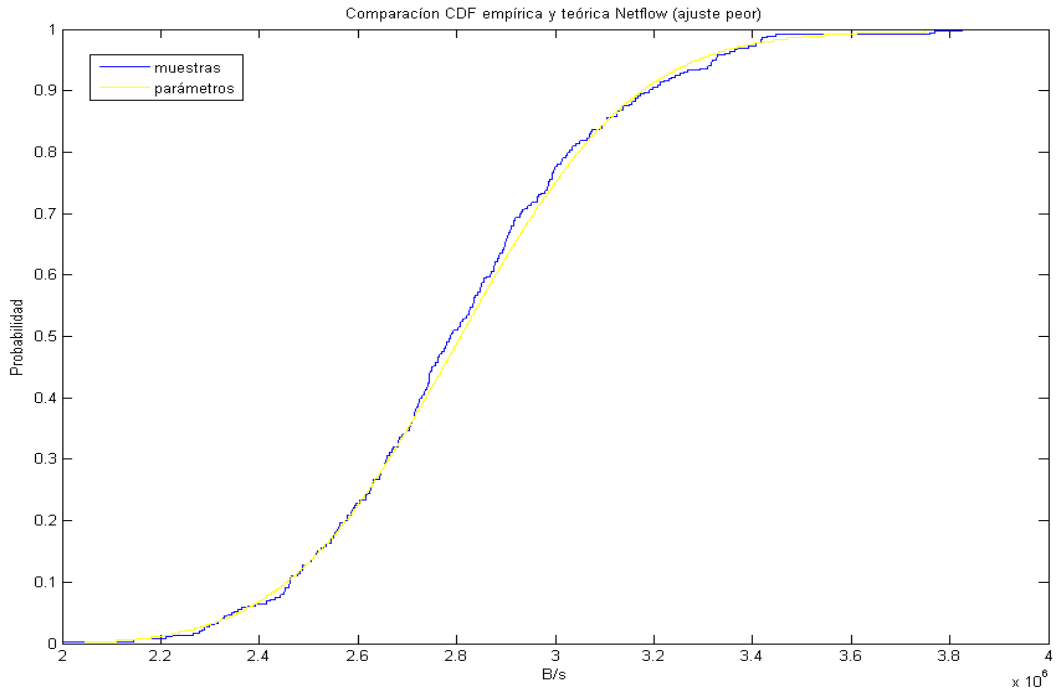

Figura 4.7 – Comparación entre las cdf calculadas a partir de los Parámetros y las muestras de NetFlow en el caso en que este se ajuste peor

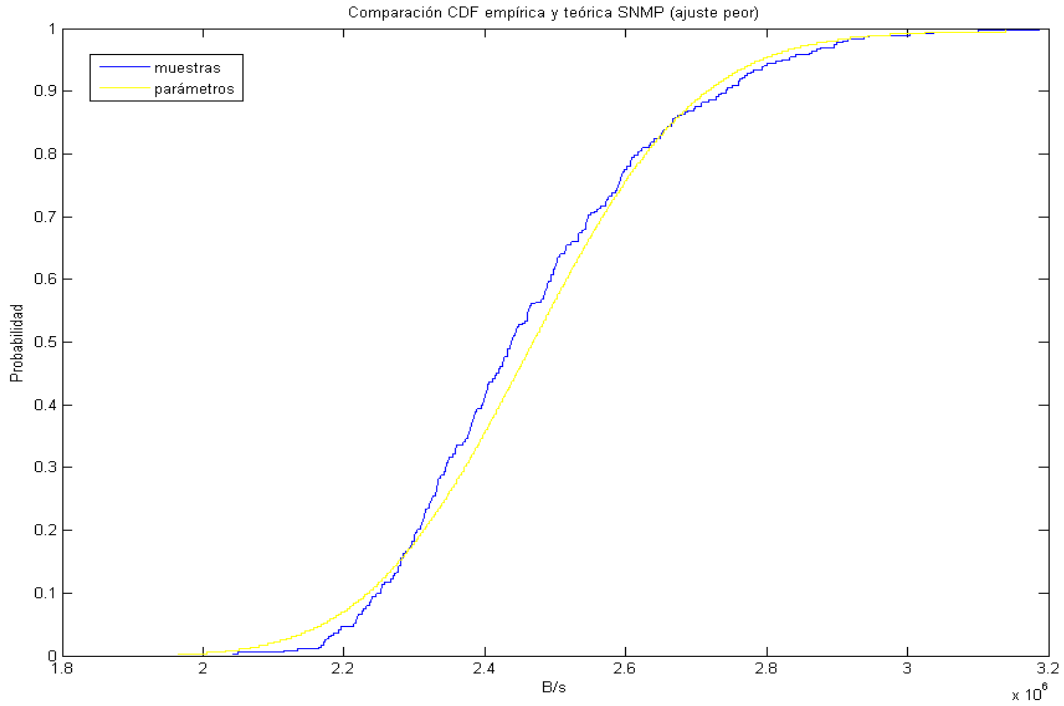

Figura 4.8 – Comparación entre las cdf calculadas a partir de los Parámetros y las muestras de SNMP en el caso en que este se ajuste peor

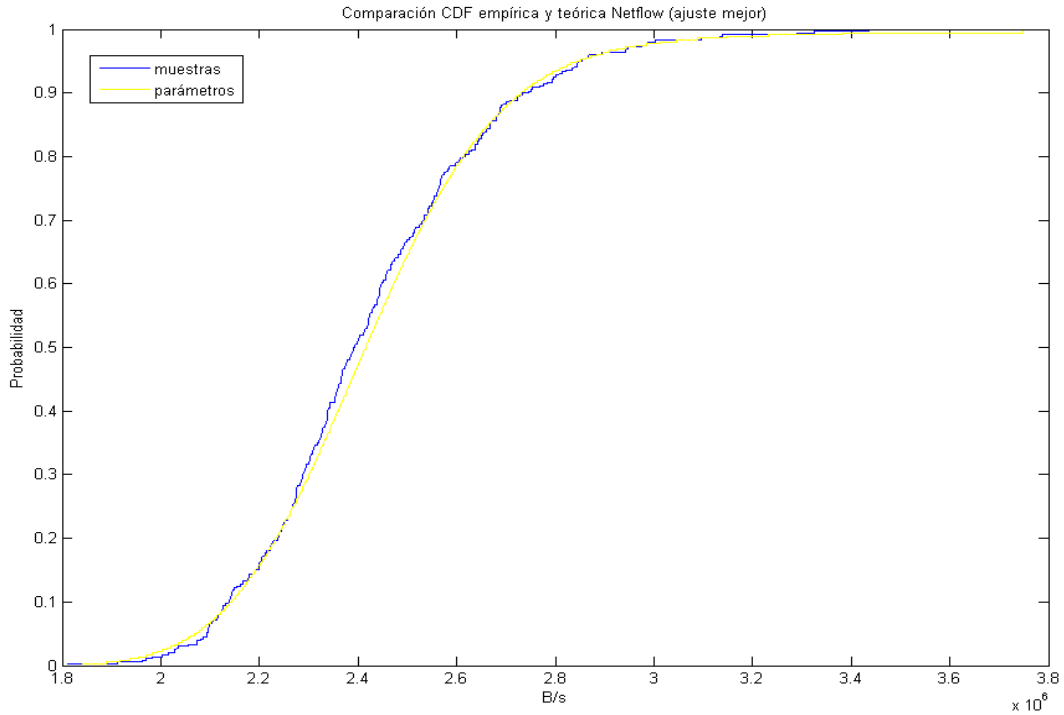

Figura 4.9 – Comparación entre las cdf calculadas a partir de los Parámetros y las muestras de NetFlow en el caso en que este se ajuste mejor

Se han elegidos dos casos, uno donde la serie temporal de SNMP proporciona una menor distancia entre las funciones y otro donde es la de NetFlow la que minimiza la diferencia. Finalmente en el capítulo siguiente se describe una evaluación cuantitativa entre modelos y de cada modelo respecto a las relativas muestras, donde se verá también que en cada caso la diferencia entre el error de ambas series temporales está por debajo del 5% en la mayoría de las ventanas.

4.3 Conclusiones

Como se ha visto en el capítulo, a partir de las series temporales elaboradas y filtradas según los procesos descritos en el capítulo anterior, se ha elegido la longitud de la ventana temporal, buscando un buen equilibrio entre un periodo de tiempo en el cual se pueda considerar el tráfico de red estacionario y el número de muestras comprendidas en el mismo intervalo. Un intervalo muy largo tendría un alto número de muestras que permiten dar robustez al modelo de la distribución pero, al mismo tiempo, el tráfico no se comportaría de forma estacionaria empeorando la evaluación de los parámetros. Por el contrario, un intervalo temporal más corto seguramente asegura la estacionariedad, pero no proporciona un número de muestras suficientes para una buena convergencia del modelo. Hay que tener en cuenta que a causa de la perdidas de paquetes que puede verificarse en la red hay partes de ventanas en las cuales partes de la muestras no están disponibles o no tienen sentido. Por ejemplo, en muestras de NetFlow que inicialmente estaban ausentes, y por eso forzadas a cero, y que por efecto del filtrado su valor se ve convertido en una cantidad que no tiene ninguna relación con la serie temporal ya que sólo representa las típicas ondulaciones de la respuesta al impulso de un filtro IIR. A la hora de elegir la longitud de la ventana sería mejor incluir un margen en el número de muestras para que en presencia de este caso se pueda seguir modelando bien la ventana. Considerando la totalidad, se ha elegido una longitud de media hora para las ventanas, que traducidas en número de muestras serían 360 muestras, dado que el periodo de muestreo es de 5 segundos. Además se ha elegido un periodo total de 28 días para asegurar un cierto nivel de fiabilidad a la hora de evaluar las diferencias y los errores. Comparando las dos trazas a lo largo de estas 4 semanas se ha descubierto una pérdida de sincronización entre las dos series temporales. Esto normalmente se genera después de partes dañadas del registro de SNMP y se ha corregido volviendo a sincronizar las trazas cada 12 horas. A continuación se han calculados los parámetros de la distribución α-estable que minimizan el error con respecto a las muestras con la función “stblfit” del paquete STBL_CODE de Matlab. Algunos ejemplos de los resultados obtenidos ya se han mostrado con anterioridad, y como se puede ver en las figuras relativas a la comparación entre parámetros, estos son muy parecidos, excepto el segundo ,β, que indica la parte de la distribución respecto a la media que tiene una varianza mayor y que en correspondencia de $\alpha = 2$ (distribución simétrica) su valor no altera la distribución. Además se puede ver como los últimos dos parámetros que representan la media y la varianza de la distribución α-estable se parecen todavía más, aunque la media de NetFlow es un poco inferior a la de SNMP. Una interesante curiosidad es que filtrando la traza de NetFlow con una función que proporciona a la señal una fase plana,a pesar de que la diferencia entre los dos primeros parámetros se queda prácticamente invariada, la diferencia entre los dos últimos parámetros resulta ser todavía mayor, motivo por el cual en principio se había excluido esta alternativa.
Después de esta comparativa cualitativa entre los parámetros evaluados, en el próximo capítulo se va a enseñar el cálculo del error para poder cuantificar la diferencia y ver si efectivamente es posible utilizar los registros de NetFlow en lugar de los de SNMP para desarrollar un sistema que modela el tráfico de red, y que eventualmente use estas características con el objetivo de detectar anomalías o ataques.

Capítulo 5

Evaluación de los errores

5.1 Introducción

En el capítulo precedente se completa la fase de la extracción de los parámetros de la distribución α-estable. De la comparación de los parámetros que se puede ver en las figuras 4.3, 4.4 y 4.5 ya se observa que los valores son bastante parecidos. A primera vista el parámetro β es el que más se diferencia, pero esta diferencia no altera la distribución cuando α es igual a 2. En este capítulo se se describen las pruebas efectuadas y los índices de error calculados para obtener una evaluación cuantitativa de la diferencia entre los parámetros calculados para las dos series temporales, el nivel en que estas últimas se parecen entre si y cómo cada serie temporal se parece al modelo extraído a partir de la otra. Se han efectuado pruebas basadas en el test de bondad de ajuste de Kolmogorov-Smirnov que evalúa la máxima diferencia entre las funciones de densidad acumulativa.
Para la evaluación de los errores se han considerado las ventanas en un periodo temporal de 28 días, totalizando un conjunto de 1344 ventanas de media hora. La elección de un periodo tan largo se justifica para obtener unos resultados mucho más fiables. Hemos encontrado en [11] que los resultados de los cálculos empiezan a convergir según pasan los días y se puede ver como este plazo corresponde a un número de días suficientes para que los resultados converjan.
Además, siempre con el objetivo de que la evaluación de los errores sea lo más fiable posible, no se han calculado los errores para ventanas cuyas muestras son todos ceros o que contienen un bajo número de muestras distintas a cero, puesto que no añadirían información útil.
Los detalles de las pruebas y los resúmenes de los resultados se tratan en las siguientes secciones.
De esta manera en la próxima sección se describe el test de bondad de ajuste de Kolmogorov-Smirnov, con el resumen de los resultados en forma de gráficas y tablas para todas las pruebas efectuadas. A continuación, el capítulo se concluye con un resumen general de las pruebas efectuadas, así como con una discusión de los resultados obtenidos.

5.2 Test de bondad de ajuste de Kolmogorov-Smirnov

Como se ha comentado anteriormente las pruebas que se van a describir están basadas en el Test de bondad de ajuste de Kolmogorov-Smirnov. Este test mide la diferencia máxima entre la función de probabilidad acumulativa de las ventanas. A partir de las muestras y de los parámetros de ambas series temporales el test se ha utilizado para comparar la diferencia entre:

- las distribuciones de las muestras de SNMP y NetFlow;

- las muestras y las distribuciones calculadas a partir de los relativos parámetros (aunque estas dos pruebas miden la fiabilidad del algoritmo utilizado para evaluar los parámetros);

- las muestras y las distribuciones calculada a partir de los parámetros de la otra serie temporal;

- las distribuciones calculadas a partir de los parámetros de las dos series.

En práctica en esta sección se evalúan todas las posibles combinaciones resultantes de las cuatro distribuciones disponibles. Hemos descartado el cálculo del error para ventanas cuyas muestras son todos ceros o que contuvieran un bajo número de muestras distintas a cero porque no garantizaban un resultado fiable y este es el motivo por el se han excluido 13, 63 y 71 ventanas en total para las pruebas que utilizan los parámetros de SNMP, NetFlow y los dos respectivamente.
Dicho esto, se puede pasar directamente a explicar los resultados obtenidos de la primera prueba donde se ha calculado la diferencia máxima entre las funciones de distribución acumulativa de las muestras de las series temporales de SNMP y de NetFlow. La figura 5.1 enseña esta diferencia a lo largo de las 1344 ventanas consideradas y la tabla 5.1 resume el porcentaje de las ventanas que se encuentran por debajo del nivel de error indicado.

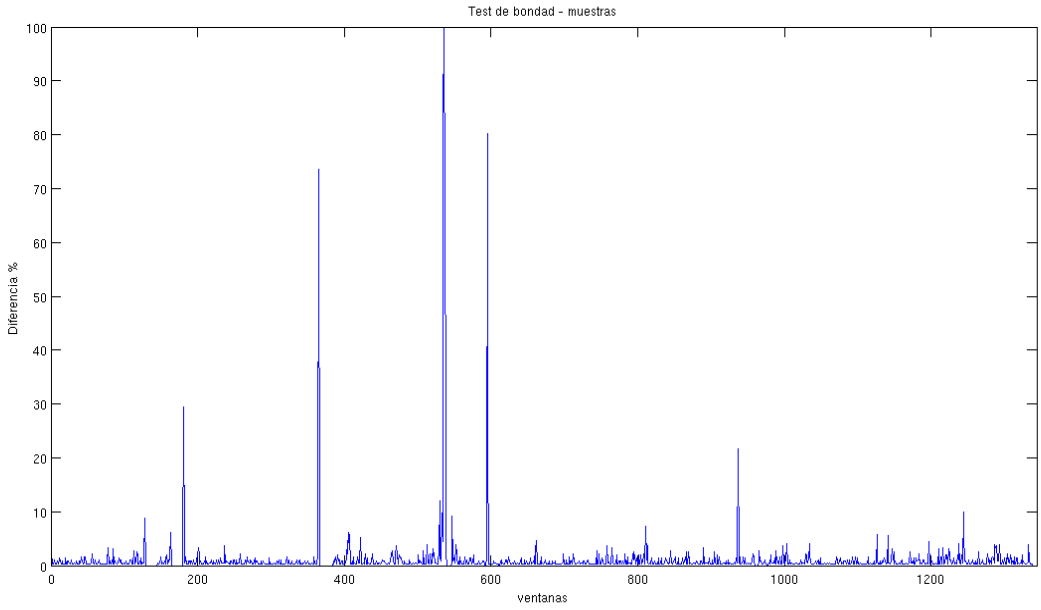

Figura 5.1 – Comparación entre las distribuciones de las muestras SNMP y Netflow

Error [%]	Ventanas	Ventanas [%]	Error [%]	Ventanas	Ventanas [%]
90	1272	99,92	20	1264	99,29
70	1269	99,69	10	1262	99,14
50	1268	99,61	5	1251	98,27
40	1268	99,61	2	1175	92,3
30	1268	99,61	1	980	76,98

Tabla 5.1 – Resumen de las ventanas cuya diferencia está por debajo del umbral de error

Como se puede ver en la tabla, las distribuciones de las dos series después de las primeras fases de elaboración presentan una diferencia muy baja. De hecho más del 98% de las ventanas se diferencian menos del 5%, y más del 92% de las ventanas presentan un error inferior al 2%.
Un nivel de error mayor lo encontramos en la siguiente prueba, en la cual se comparan las distribuciones de las muestras con las de los parámetros extraídos, es decir, el nivel de bondad del algoritmo que ha evaluado los parámetros. En la figura 5.2 se encuentra una comparación entre las diferencias calculadas para las dos series, mientras que en las tablas 5.2 y 5.3, al igual que antes, hay un resumen de las ventanas de SNMP y de NetFlow que no pasan el umbral de error.

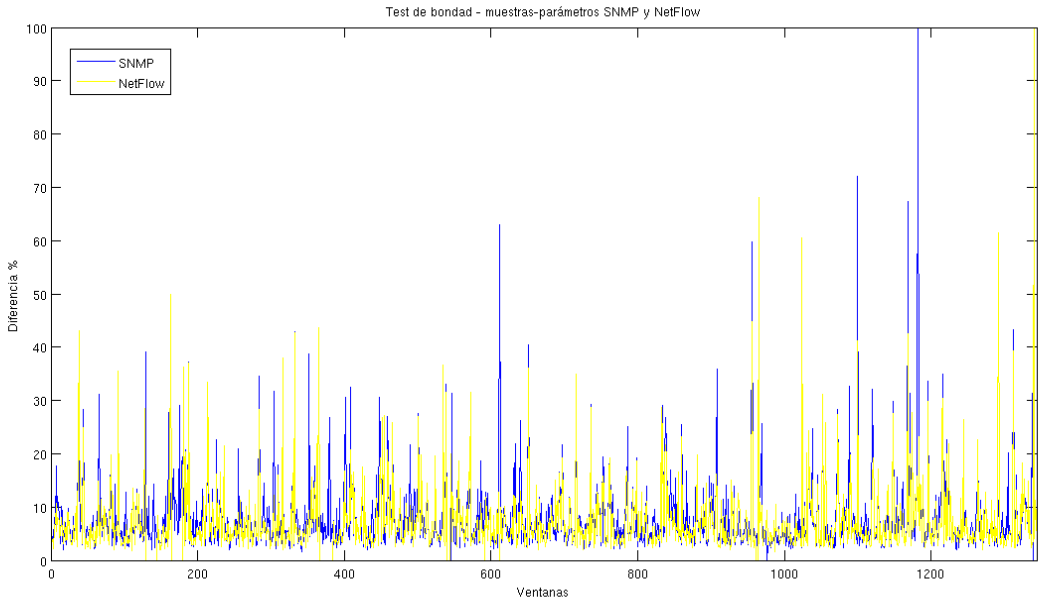

Figura 5.2 – Comparación entre las distribuciones de las muestras y las de los relativos parámetros de SNMP y Netflow

Error [%]	Ventanas	Ventanas [%]	Error [%]	Ventanas	Ventanas [%]
90	1330	99,92	30	1302	97,82
70	1329	99,85	20	1262	94,82
50	1326	99,62	10	1062	79,79
40	1323	99,4	5	555	41,7

Tabla 5.2 – Resumen de las ventanas de SNMP cuya diferencia está por debajo del umbral de error

Error [%]	Ventanas	Ventanas [%]	Error [%]	Ventanas	Ventanas [%]
90	1280	99,92	30	1257	98,13
70	1280	99,92	20	1224	95,55
50	1277	99,69	10	1053	82,2
40	1270	99,14	5	565	44,11

Tabla 5.3 – Resumen de las ventanas de NetFlow cuya diferencia está por debajo del umbral de error

Como se puede observar a simple vista el error es bastante mayor en este caso con respecto al anterior. Comparando los dos protocolos entre sí se observa como ambas series presentan un valor de error bastante parecido, como demostración del hecho de que esta medida resulta también afectada por el error del algoritmo de evaluación de los parámetros.
Se puede esperar un resultado con la misma tendencia de las pruebas efectuadas entre la distribución de las muestras de una serie temporal y la distribución de los parámetros de la otra. Los resultados de esta pruebas se ilustran en la figura 5.3, y al igual que antes, usando el mismo criterio, se resumen en las tablas 5.4 y 5.5.

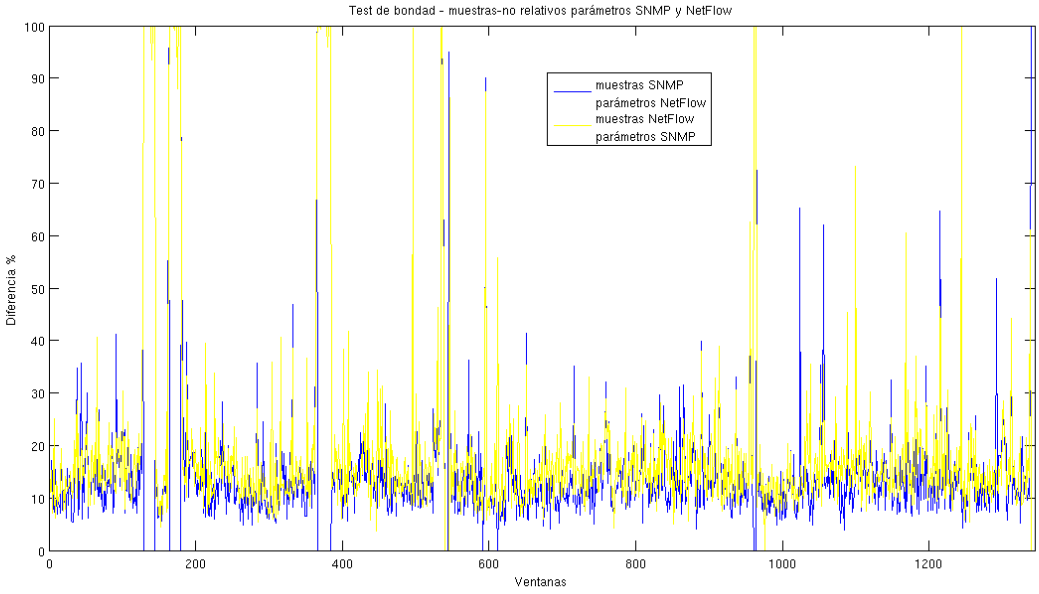

Figura 5.3 – Comparación entre las distribuciones de las muestras y las de los parámetros no relativos de SNMP y Netflow

Error [%]	Ventanas	Ventanas [%]	Error [%]	Ventanas	Ventanas [%]
90	1268	99,61	30	1224	96,15
70	1264	99,29	20	1130	88,77
50	1258	98,82	10	387	30,4
40	1250	98,19	5	11	0,86

Tabla 5.4 – Resumen de las ventanas de SNMP con parámetros de NetFlow cuya diferencia está por debajo del umbral de error

Error [%]	Ventanas	Ventanas [%]	Error [%]	Ventanas	Ventanas [%]
90	1271	99,84	30	1220	95,84
70	1264	99,29	20	1068	83,9
50	1258	98,82	10	179	14,06
40	1250	98,19	5	4	0,31

Tabla 5.5 – Resumen de las ventanas de NetFlow con parámetros de SNMP cuya diferencia está por debajo del umbral de error

Ya se podría esperar que los valores de estos resultados fueran mayores que los de las primeras pruebas, dado que mezclando las distribuciones, lo más normal es que se sumen los errores entre las series y que igualmente se sumen los errores entre estas y los modelos. Sin embargo, de la figura y de las tablas extraemos que incluso son bastante mayores que los obtenidos en la prueba donde las muestras se comparaban con las distribuciones relativas a los parámetros de la otra serie temporal.
Finalmente y ante todas las posibles combinaciones a prueba, faltaría la de la diferencia entre las distribuciones calculadas a partir de los parámetros extraídos. Sin duda las pruebas hechas hasta ahora tienen una importancia relativa permitiendo evaluar cuantitativamente el grado de error inicial entre las series, o el introducido por la evaluación de los parámetros. No obstante,la próxima prueba es de extrema importancia convirtiéndose en el objetivo principal de este trabajo. En el capítulo precedente ya se ha visto que excepto el parámetro β, que es el que indica la dirección de la asimetría de la distribución, los otros tres son bastantes parecidos. En las figuras que representaban la comparación sólo se consideraba un día, pero ahora se puede ver si a lo largo de 4 semanas la diferencia sigue siendo baja. En la figura 5.4 y en la tabla 5.6 se dibuja la variación de la diferencia entre las distribuciones con los parámetros calculados de las series temporales y se anota el resumen de estos en la forma utilizada hasta ahora.

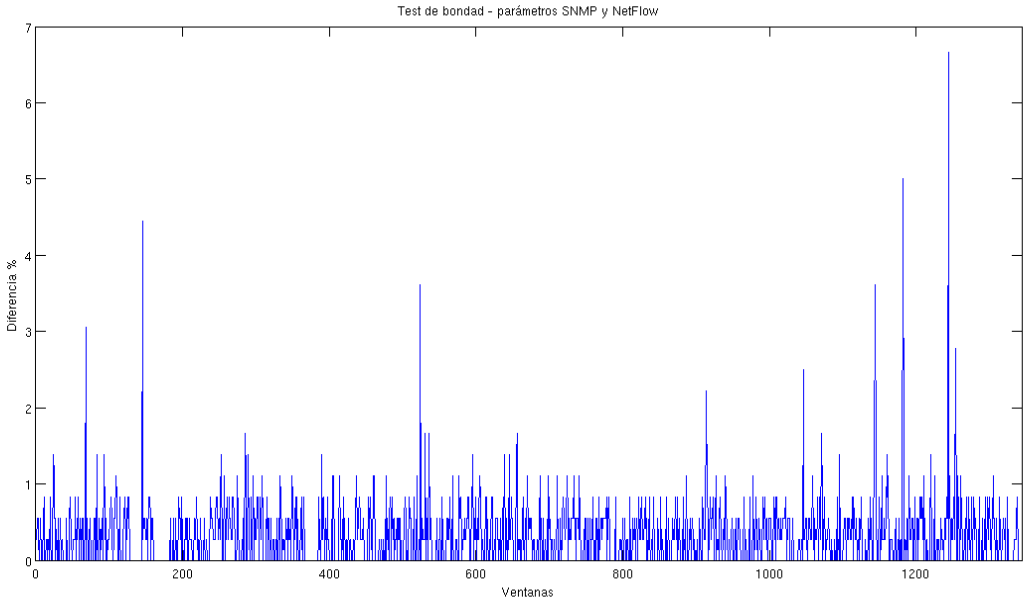

Figura 5.4 – Comparación entre las distribuciones de los parámetros de SNMP y Netflow

Error [%]	Ventanas	Ventanas [%]	Error [%]	Ventanas	Ventanas [%]
10	1273	100	3	1267	99,53
6	1272	99,92	2	1264	99,29
5	1271	99,84	1	1202	94,42
4	1270	99,76	0,5	806	63,32

Tabla 5.6 – Resumen de las ventanas de los parámetros de SNMP y NetFlow cuya diferencia está por debajo del umbral de error

Como se puede ver de los resultados para este tipo de comparación, dado que estas distribuciones están calculadas directamente a partir de los parámetros, no se presentan ni muestras iguales a cero ni muestras sin sentido. Esto provoca que la máxima diferencia entre sus funciones acumulativas baje.
En la siguiente sección se va a resumir lo que se ha visto a lo largo del capítulo discutiendo los resultados obtenidos en todas las pruebas efectuadas.

5.3 Conclusiones

Como se ha podido ver a lo largo de este capítulo se ha efectuado el cálculo de los errores para evaluar la diferencia entre todas las distribuciones incluidas, tanto a partir de las muestras, como de los parámetros. Teniendo 4 distribuciones, los posibles errores que se pueden calcular son 6. Para calcular los errores, entre varias alternativas [4] se ha elegido la máxima diferencia entre la función de distribución de probabilidad acumulativa. Este es el motivo por el que se ha efectuado el cálculo para la diferencia entre las distribuciones de las muestras, entre muestras y relativos parámetros, entre muestras y parámetros de la otra serie temporal y para finalizar, entre las distribuciones de los parámetros, que es justo el objetivo de este TfdM. La evaluación se ha extendido a un periodo de 1344 ventanas que abarcan 28 días completos, siendo este un número suficiente de días para asegurar la convergencia de los resultados. Como quedó dicho en párrafos precedentes, el cálculo del error no ha sido efectuado para ventanas cuyas muestras son todos ceros o que contengan un bajo número de muestras distintas a cero pues no garantizarían un resultado fiable. Realmente para estas ventanas no se encuentran disponibles ni siquiera los parámetros ya que, al igual que las muestras, no aportan nueva información. Estas ventanas en las cuales se ha omitido el cálculo del error son 71 cuando se usan los parámetros de ambas series temporales, 13 con los de SNMP y 63 con los de NetFlow.

Entrando más en detalle, el error de la primera prueba está calculado para cuantificar la diferencia entre las series temporales de SMNP y NetFlow después de las fases de elaboración previa de las señales. De esta forma se puede demostrar que estas fases, que durante la elaboración del mismo trabajo han resultado necesarias, han devuelto los resultados esperados bajando significativamente el valor del error. De hecho como se observa en los resultados un 98% de las ventanas presentan un error inferior al 5% y un poco más del 92% de las ventanas consiguen diferir menos de un 2%.

En la segunda prueba se calcula el error entre las distribuciones de las muestras y las de los relativos parámetros. Este error, como he comentado anteriormente, se ve afectado por el error introducido por el algoritmo de evaluación de los parámetros y por eso se puede esperar un valor más alto. De los resultados calculados se puede ver como efectivamente el valor del error en general ha subido, pero comparando los dos protocolos, estos presentan unas diferencia mínima con NetFlow que asegura un nivel de error un poco más bajo.

En la tercera prueba se encuentra la comparación entre las distribuciones de las muestras de una serie temporal con las de de los parámetros de la otra. Como en el algoritmo que evalúa el test de bondad el umbral que elige cuando descartar las ventanas está fijado para los parámetros (que sólo se han calculado para ventanas con muestras significativas) y por eso se puede esperar un nivel de error un poco más alto con respecto al caso precedente. De todas formas, teniendo en cuenta que las muestras proceden de un protocolo y los parámetros del otro, se encuentran ventanas en las cuales las muestras no tienen sentido aunque existan los parámetros de la otra serie temporal. En este caso el valor del error sube bastante. Para corregir el porcentaje de ventanas que presentan un valor de error inferior al umbral en este cálculo, se ha restado al total de las ventanas el número total de las ventanas descartadas (71), como en el primer caso.

Finalmente en la última prueba se pasa a evaluar la diferencia entre las funciones de densidad acumulativa calculadas a partir de los parámetros extraídos. En práctica esta prueba permite dar una evaluación cuantitativa de la diferencia entre los parámetros de las distribuciones α-estables, que es exactamente el objetivo del trabajo. Mirando las figuras que ilustran la comparación en el capítulo precedente ya se podía vislumbrar que la diferencia entre los parámetros extraídos de la serie temporal de SNMP y de NetFlow efectivamente no eran muy distintos, pero es gracias a esta prueba que se puede dar un valor numérico a esta diferencia. Por lo que aparece en los resultados casi la totalidad de las ventanas presentan una diferencia menor de un 5%, mientras que poco más

de un 94% de las ventanas tienen un error inferior al 1%.
A continuación en el capítulo siguiente se va a resumir todo lo que hemos visto a lo largo de este TfdM con las discusiones relativas y una propuestas de mejoras y trabajos futuros.

Capítulo 6

Conclusiones

6.1 Resumen de contribuciones

Este trabajo fin de Máster tiene como objetivos la evaluación y la comparación de los parámetros de una distribución α-estable a partir de dos registros: un registro con los contadores del trafico agregado de la red de SNMP y un registro con los flujos del tráfico de la misma red generado por NetFlow. Los routers que proporcionan la información se encuentran en la red de la universidad de Valladolid. Sabiendo que para modelar las características del tráfico de red, se puede utilizar un modelo estadístico, en el segundo capítulo del trabajo se tratan varias distribuciones de probabilidad. Después de una introducción que describe unas de las primeras distribuciones utilizadas en este campo, la de Poisson y la normal, se pasa a examinar la que dentro de la gran variedad de modelos disponibles en literatura, compensa los limites de estas primeras distribuciones y además es capaz de tener en cuenta la característica de alta variabilidad del tráfico: la distribución α-estable. Por todas estas características, elegir este modelo asegura un buen nivel de ajuste a pesar del protocolo utilizado para obtener los datos. A continuación, en el capítulo 3 se comparan los registros y se describen las primeras operaciones necesarias para el desarrollo de este TfdM. Los registros, guardados en distintos ficheros , almacenan tanto el tráfico de bajada como el de subida. A pesar de que proceden de la misma red, los dos protocolos, SNMP y NetFlow, proporcionan información muy diferente. El primero facilita la información del tráfico a partir del 4 de Junio del 2007 hasta el 30 de Julio del 2008 en dos contadores incrementales, uno para el tiempo y el otro para los bytes, mientras que el segundo empieza el 1 de Septiembre del 2007 y se concluye el 31 de Diciembre del 2008 proporcionando no sólo la información temporal y de tráfico a través de campos donde guarda la hora de inicio y fin de la transmisión junto al número de paquetes y bytes transmitidos, sino que consigue también otro tipo de información que podría resultar muy valiosa para la gestión y la seguridad en la red. Dada la gran cantidad de datos generados, el router muestrea el tráfico creando registros NetFlow que tienen en consideración solo 1 paquete de cada 100. Debido a la distinta manera de almacenar los datos, la primera operación que se ha efectuado ha sido la de transformar esta información en dos series temporales, una por cada protocolo, uniformando el periodo de muestreo y expresando el tráfico como velocidad de red. Los contadores de SNMP se generan a partir de peticiones que se envían al router con un intervalo de 5 segundos. Sin embargo las peticiones pasan por la red y eso hace que este periodo no sea constante para cada muestra. Este jitter se encuentras reflejado en el valor de la velocidad calculada. En consecuencia, durante el calculo de la serie se ha compensado este error introducido. En el registro de NetFlow, la operación para crear la serie temporal ha consistido en calcular la longitud de cada flujo, y suponiendo la velocidad constante, dividiendo los bytes y los paquetes entre los relativos intervalos de tiempo de forma proporcional. De esta forma se han creado las series temporales. La falta de una referencia temporal absoluta en las serie de SNMP hace posible una comparación entre las series sólo después de una sincronización. Para hacer eso se ha dividido el fichero de SNMP en dos partes de forma que la segunda empieza por la tarde del 31 de Agosto.

De esta segunda parte se ha aislado más o menos un día y medio y se ha comparado con el primer día de la serie de NetFlow (el 1 de Septiembre). La sincronización se ha efectuado en dos fases: en una primera se ha comparado la posición de los máximos relativos y alrededor de esta diferencia, en la segunda fase se ha encontrado el valor del desplazamiento que minimiza el error cuadrático medio. La comparación entre las dos series ha sido útil para ver que la serie de NetFlow presenta una variabilidad temporal más alta respecto a la de SNMP. Vista esta diferencia se ha calculado la densidad de potencia de las series en el dominio de las frecuencias. De la comparación del dominio transformado se ha visto que la alta variabilidad depende de 6 componentes espectrales de alta frecuencia. Lo más destacable es que las componentes de ruido se concentran alrededor de 6 frecuencias fijas y que según los cálculos serían las que se encuentran en correspondencia con los índices 1441, 2881, 4320, 5760, 7200 y 8640 (1/12, 1/6, 1/4, 1/3, 5/12 y 1/2 de la frecuencia de muestreo respectivamente) del vector de densidad de potencia. Además se ha comprobado que el ruido se encuentra con las mismas características a lo largo de todos los días del periodo considerado y no se limita sólo al día considerado en el ejemplo. Los resultados que se encuentran en el dominio de las frecuencias se reflejan en el dominio del tiempo, donde se puede comprobar que cada minuto (12 muestras) la tasa de flujos que el router está almacenando en el fichero baja, probablemente como consecuencia de alguna tarea periódica configurada en el mismo router. A continuación se han efectuado otras pruebas para ver si estas componentes pueden de alguna forma depender de la limitación temporal de la serie temporal o de efectos de aliasing, aplicando una ventana de Hamming o cambiando la frecuencia de muestreo. Ninguna de estas pruebas ha resultado exitosa y por eso se ha pensado que el ruido pueda depender de la configuración de la red y se ha comprobado si en routers de otras ciudades se encuentra la misma característica. Efectivamente en cada una de las otras 14 ciudades consideradas se han encontrado las mismas componentes de ruido exactamente en las mismas posiciones. Conociendo la posición de las componentes espectrales del ruido se ha podido proyectar el filtro, o más bien la cascada de filtros, ya que con sólo un filtro no es posible eliminar selectivamente el ruido dejando la señal inalterada. Como se ha elegido una distribución de tipo α-estable para modelar la tasa de bytes por unidad de tiempo para aprovechar de su característica de alta variabilidad, un filtro paso bajo aunque eliminase muy bien el ruido, eliminaría también componentes de la señal que permiten un mejor ajuste del modelo. Por eso se ha preferido una cascada de filtros Notch que, bien diseñados, tienen la ventaja de filtrar sólo el ruido. Después de una elaboración previa (y poco costosa a nivel computacional) para homogeneizar la información contenida en los registros obtenidos con ambos protocolos, en el capítulo 4, se han calculados los parámetros de la distribución a través de un algoritmo del paquete STBL_CODE [10] de Matlab. en concreto la función “stblfit”, basada en el método de Koutrouvelis. Con esta fase se completa el primer objetivo de este TfdM. En el mismo capítulo se ilustra la comparación de los parámetros a lo largo de un día entero. Esta comparación ya permite una evaluación cualitativa del parecido entre los parámetros. En el capítulo 5 se ha efectuado el cálculo de los errores para evaluar la diferencia entre las todas las distribuciones calculadas tanto a partir de las muestras, como de los parámetros. La evaluación se ha extendido a un periodo de 28 días completos (1344 ventanas). Teniendo 4 distribuciones, los posibles errores que se pueden calcular son 6. Para calcular los errores se ha elegido el test de bondad de Kolmogorov-Smirnov. Se ha efectuado el calculo para la diferencia entre las distribuciones de las muestras, entre muestras y relativos parámetros, muestras y parámetros de la otra serie temporal y por último entre las distribuciones de los parámetros, que es precisamente el segundo objetivo de este TfdM. Además se resumen todos los resultados obtenidos del test de bondad a través de figuras y tablas.

En la sección siguiente se van a extraer las conclusiones más destacables de este trabajo y unas consideraciones antes de concluir con una propuestas de mejoras y trabajos futuros.

6.2 Conclusiones

En la sección precedente se ha resumido como se ha ido desarrollando este trabajo de los registros de los dos protocolos y las operaciones que han sido necesarias hasta la evaluación de los parámetros y el calculo de los errores. Esta sección concluye la sección previa con el resumen de los resultados de los errores que se han calculado y con unas consideraciones que se pueden extraer de todo lo que se ha ido exponiendo.

El test de bondad de ajuste, como hemos visto anteriormente, se ha efectuado para todas las posibles combinaciones que se obtiene a partir de las 4 distribuciones (2 de las muestras y 2 de los parámetros). Así que se ha podido observar que comparando la distribución de las muestras de SNMP con la de NetFlow, es decir las series temporales después de la elaboración, estas presentan una diferencia bastante baja. De hecho más del 98% de las ventanas se diferencian en menos del 5%, y en más del 92% de las ventanas se presenta un error inferior al 2%. Lo más destacable de los resultados obtenidos es percibir como, tras realizar adecuadamente una sincronización y un filtrado, con los registros de NetFlow se pueden obtener valores similares a los obtenidos con SNMP con un margen de error muy pequeño, además dándose el caso de que la tasa de muestreo de paquetes no tiene prácticamente influencia en los parámetros estimados. Claramente ha sido posible obtener estos resultados gracias al filtrado ya que la serie temporal de NetFlow presenta un alto valor de ruido. Además se ha comprobado que este ruido se encuentra también en las series temporales de otras ciudades españolas. Eso permite concluir que a pesar de que la causa del ruido sea desconocida (solo se puede decir que a ciencia cierta no depende ni de la limitación temporal de las series, ni de efectos de aliasing), todas las series se pueden filtrar con el mismo filtro, o mejor dicho, con el mismo banco de filtros independientemente de la ciudad de procedencia.

En la segunda parte de las pruebas se han ido comparando las distribuciones de las muestras, primero con las de los parámetros del mismo protocolo y después con las de los parámetros del otro. Estas pruebas evalúan el grado de error introducido por el algoritmo utilizado. De hecho en ambas pruebas los errores son bastante mayores con respecto a los de la primera prueba, mientras que comparando los errores calculados entre sí para los dos protocolos, estos resultan ser muy parecidos.

Finalmente se efectúa la prueba que permite evaluar directamente el objetivo de este TfdM: la comparación entre los parámetros de las distribuciones. En este test la tendencia precedente se invierte cuando la diferencia baja notablemente. En los resultados destaca como casi la totalidad de las ventanas presentan una diferencia menor de un 5%, mientras que poco más de un 94% de las ventanas tienen un error inferior al 1%.

Estos resultados demuestran que es posible extraer los parámetros de una distribución α-estable a partir de una serie temporal de NetFlow en lugar que de una de SNMP, y que estos parámetros no presentan mucha diferencia con respecto a los otros.

La ventaja principal de este resultado es, por tanto, la posibilidad de llegar a resultados similares con una simple configuración inicial del router y sin tener que cargar el router ni la red con peticiones SNMP periódicas (cada 5 segundos) para trabajar en la misma escala de tiempos y con aproximadamente el 1% de los flujos disponibles por efecto del muestreo. Además utilizando toda la información disponible en los flujos de NetFlow, se podría construir una herramienta muy efectiva para la gestión, y sobre todo, la seguridad de la red.

En la siguiente sección se concluye este trabajo proponiendo una serie de posibles mejoras y de trabajos futuros que se podrían elaborar a partir del mismo.

6.3 Trabajos futuros

En los capítulos anteriores ha sido posible seguir el desarrollo de este proyecto de principio a fin. A pesar de todo el trabajo realizado y de lo completo que pueda resultar, como en todos los trabajos de investigación, siempre es posible aportar mejoras o utilizarlo como base para futuros desarrollos. Esta última sección se ha reservado para este tipo de propuestas. A continuación se van a ver unos ejemplos que justifican las nuevas aportaciones o el nuevo trabajo.

Posibles mejoras:

- Cambiar el banco de filtros para filtrar la serie temporal de NetFlow.

Sabiendo que el ruido se encuentra en las series de todas las ciudades y utilizando esta información, se podría intentar proyectar un nuevo filtro (o cascada de filtros) variando un poco el ancho de banda para que se adapte perfectamente a la densidad de potencia del ruido. Hay que tener en cuenta que la última armónica de este ruido se encuentra justo a la mitad de la frecuencia de muestreo (que se gestiona difícilmente), y que el número de armónicas depende del periodo de muestreo.

- Solapar parcialmente las ventanas.

De esta forma se podrían tener más ventanas por cada día a paridad de longitud o alternativamente a paridad de ventanas diarias un numero de muestras más elevado. Siempre hay que considerar un tiempo en el cual el tráfico se pueda considerar estacionario.

- Resincronización con periodo más corto.

A la hora de extraer los parámetros la pérdida de sincronización se ha compensado volviendo a sincronizar las trazas cada 12 horas. Eligiendo un tiempo más corto podría mejorar la sincronía mejorando la comparación de los parámetros.

- Ampliar el periodo de tiempo de evaluación de los errores.

Aunque 28 días completos podría ser un tiempo suficientemente largo, ampliando el periodo de evaluación seguramente se consigue dar más robustez a los resultados obtenidos.

Trabajos futuros:

- Utilizar más información almacenada por NetFlow.

En este trabajo solo se ha usado la información relativa a la hora y al numero de paquetes y bytes, pero el registro de NetFlow almacena mucha más información. El uso de toda esta información podría dar al sistema la posibilidad de implementar nuevas tareas, como filtrar el tráfico de un puerto, un protocolo, un servicio en particular, etc.

- Desarrollar un sistema de detección de anomalías o ataque.

Al igual que lo que se ve en [3] usando el protocolo SNMP, se podría pensar en añadir un detector de anomalías o ataques en la red extendiendo lo que se ha visto en este trabajo y a lo mejor introduciendo toda o parte de la información almacenada por el registro de NetFlow.

Estos son sólo algunos ejemplos de mejoras o trabajos futuros que se podrían realizar a partir de este trabajo. Cualquier otra propuesta que no aparece en la lista y que merecería ser citada es bienvenida.

Referencias

[1] Carrie Gates, Michael Collins, Michael Duggan, Andrew Kompanek and Mark Thomas, “More NetFlow Tools: For Performance and Security”. Carnegie Mellon University. Pp. 121-132 of the Proceedings of LISA '04: Eighteenth Systems Administration Conference, (Atlanta, GA: USENIX Association, November, 2004).

[2] D. Harrington, R. Presuhn, B. Wijnen, “An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks”. IETF RFC 3411, Diciembre de 2002

[3] Federico Jesús Simmross Wattenberg. “Detección de anomalías en el tráfico agregado de redes IP basada en inferencia estadística sobre un modelo α–estable de primer orden”, Tesis Doctoral, Universidad de Valladolid. Julio de 2009

[4] Gonzalo R. Arce, Nonlinear Signal Processing. A Statistical Approach, John Wiley and sons, New Jersey, NJ, USA, 2005.

[5] F. Simmross-Wattenberg, J. I. Asensio-Pérez, P. Casaseca-de-la-Higuera, M. Martín-Fernández, I. A. Dimitriadis, C. Alberola-López, “Anomaly detection in network traffic based on statistical inference and α-stable modeling”. IEEE Transactions on Dependable and Secure Computing 8:4 494-509, 2011.

[6] G. Samorodnitsky and M. S. Taqqu, Stable non–Gaussian random processes. Stochastic models with infinite variance. Boca Raton, CA, USA: Chapman & Hall, 1994.

[7] Gaurab Raj Upadhaya. “NetFlow, Flow-tools tutorial”. AfNOG Tutorials 14 May 2006, Nairobi, Kenya

[8] Jesper Schmidt Hansen, Gnu Octave Beginner's Guide, Packt Publishing Ltd, 2011.

[9] José Luis García-Dorado, Jorge E. López de Vergara, Javier Aracil, Víctor López, José Alberto Hernández, Sergio López-Buedo, Luis de Pedro, “Utilidad de los flujos netFlow de RedIRIS para análisis de una red académica, Jornadas Técnicas RedIRIS 2007, Mieres, Asturias, 19-23 de noviembre de 2007. Publicado en el Boletín de RedIRIS, número 82-83, abril de 2008

[10] MathWorks. “Alpha-Stable distributions in MATLAB”. [Fecha de consulta Abril 2013]. Disponible en http://math.bu.edu/people/mveillet/html/alphastablepub.html

[11] José Luis García Dorado, José Alberto Hernández, Javier Aracil, Jorge E. López de Vergara, Francisco J. Montserrat, Esther Robles and Tomás P. de Miguel. “On the Duration and Spatial Characteristics of Internet Traffic Measurement Experiments”, IEEE Communications Magazine, vol. 46, no. 11, pp. 148-155. Nov. 2008.

[12]L. Leon-Garcia y I. Widjaja. Communication networks. Fundamental concepts and key architectures. McGraw–Hill, Boston, MA, EEUU, segunda edicion, 2006.