

UNIVERSIDAD AUTÓNOMA DE MADRID  
ESCUELA POLITÉCNICA SUPERIOR



Trabajo Fin de Máster

**Análisis de técnicas de medida  
del ancho de banda: influencia  
de la carga del sistema y de la red  
sobre la precisión de la medida**

Autor:  
Pedro María Santiago del Río

Directores:  
Dr. Javier Aracil Rico  
Dr. Antonio Cuevas González

Madrid, 2010



TRABAJO FIN DE MÁSTER: Análisis de técnicas de medida del ancho de banda:  
influencia de la carga del sistema y de la red  
sobre la precisión de la medida

AUTOR: Pedro María Santiago del Río

DIRECTORES: Dr. Javier Aracil Rico  
Dr. Antonio Cuevas González

El tribunal para la defensa de este trabajo fin de máster está compuesto por:

PRESIDENTE: Dr. Javier Aracil Rico

VOCALES: Dr. Antonio Cuevas González

Dr. Jorge E. López de Vergara Méndez

Dr. Sergio López Buedo



# Índice general

<b>Índice General</b>	<b>III</b>
Índice General . . . . .	III
Índice de Figuras . . . . .	V
Índice de Tablas . . . . .	VII
<b>Agradecimientos</b>	<b>IX</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	3
<b>2. Estado del arte</b>	<b>5</b>
2.1. Métodos de medida de ancho de banda . . . . .	5
2.1.1. Métodos basados en trenes de paquetes . . . . .	5
2.1.2. Métodos basados en transferencias de ficheros . . . . .	8
2.2. Métodos de análisis estadístico aplicados a <i>Networking</i> . . . . .	11
2.2.1. Análisis de la varianza (ANOVA) . . . . .	11
2.2.2. Regresión lineal múltiple (MLR) . . . . .	13
<b>3. Análisis de la influencia de las aplicaciones concurrentes</b>	<b>17</b>
3.1. Análisis experimental . . . . .	17
3.1.1. Definición de los parámetros de carga . . . . .	18
3.1.2. Entorno de pruebas . . . . .	18
3.1.3. Métodos basados en transferencias de ficheros . . . . .	20
3.1.4. Métodos basados en trenes de paquetes . . . . .	25
3.2. Campaña de medidas en un entorno real . . . . .	27
3.3. Conclusiones . . . . .	30
<b>4. Modelo multinomial para rechazar medidas incorrectas</b>	<b>31</b>
4.1. Distribución del mínimo multinomial . . . . .	32
4.1.1. Planteamiento del problema . . . . .	32
4.1.2. Definición del modelo . . . . .	35
4.1.3. Cálculo de la probabilidad de error . . . . .	35
4.1.4. Algoritmo Adaptativo de la longitud del tren de medida . . . . .	38
4.2. Validación del modelo . . . . .	39
4.2.1. Simulaciones . . . . .	41
4.2.2. Experimentos . . . . .	41
4.2.3. Campaña de medidas en entorno real . . . . .	43
4.3. Conclusiones . . . . .	44

<b>5. Conclusiones</b>	<b>45</b>
5.1. Resumen y Conclusiones . . . . .	45
5.2. Trabajo Futuro . . . . .	46
<b>Bibliografía</b>	<b>49</b>

# Índice de figuras

1.1. El medidor de ancho de banda comparte con las aplicaciones concurrentes los distintos recursos del ordenador. . . . .	2
2.1. Dispersión del par de paquetes. . . . .	6
2.2. Dispersión del par de paquetes cuando hay tráfico cruzado. . . . .	7
2.3. Dispersión del tren de paquetes cuando hay tráfico cruzado. . . . .	8
2.4. Adaptación del ancho de banda al tráfico cruzado. . . . .	10
2.5. Degradación de la tasa de transferencia de un flujo TCP. . . . .	11
3.1. Topología de pruebas. . . . .	19
3.2. Gráficos de dispersión <i>mem</i> vs. <i>cpu_frac</i> vs. Ancho de banda medido, para el caso de capacidad del enlace de 6 Mbps. . . . .	20
3.3. Gráficos de dispersión <i>mem</i> vs. <i>cpu_frac</i> vs. Ancho de banda medido, para el caso de capacidad del enlace de 100 Mbps. . . . .	22
3.4. Modelo de regresión lineal simplificado, ajustado para el caso de Windows. . . . .	24
3.5. Modelo de regresión lineal simplificado, ajustado para el caso de Linux. . . . .	24
3.6. Gráfico de dispersión de los parámetros de carga frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 6 Mbps). . . . .	25
3.7. Gráfico de dispersión de los parámetros de carga frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 100 Mbps). . . . .	26
3.8. Gráfico de cajas de los factores de carga y sistema operativo, frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 6 Mbps). . . . .	27
3.9. Gráfico de cajas de los factores de carga y sistema operativo, frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 100 Mbps). . . . .	29
3.10. Gráficos de dispersión de los factores de carga frente al porcentaje de ancho de banda efectivo en la campaña de medidas. . . . .	30
4.1. Medida afectada por el tráfico interferente. . . . .	32
4.2. Medida correcta a pesar del tráfico interferente. . . . .	33
4.3. Caso real frente a caso peor. . . . .	34
4.4. Diagrama del proceso estocástico de reparto de paquetes en los huecos. . . . .	36
4.5. Distribución de la probabilidad de error respecto al número de paquetes interferentes. . . . .	38
4.6. Número máximo de paquetes interferentes para que la medida sea considerada correcta ( $p_{error} < 5\%$ ) en función de la longitud del tren. . . . .	39
4.7. Longitud mínima del tren de medida en función de la intensidad de tráfico interferente. . . . .	40
4.8. Probabilidades de error estimadas mediante simulación. . . . .	42
4.9. Gráfico de dispersión del número de paquetes interferentes frente al porcentaje de ancho de banda efectivo en la campaña de medidas. . . . .	43





# Índice de tablas

2.1. Resumen de datos para el caso ANOVA de un solo factor. . . . .	11
3.1. Resumen del equipamiento usado para las pruebas. . . . .	19
3.2. Tabla de Análisis de la Varianza (caso de enlace residencial de 6 Mbps con transferencia de fichero). . . . .	21
3.3. Coeficientes de regresión para el modelo considerando carga de CPU y memoria. . . . .	23
3.4. Coeficientes de regresión para el modelo lineal considerando la carga de CPU sólo. . . . .	23
3.5. Tabla de Análisis de la Varianza (caso de enlace residencial de 6 Mbps para el caso de Trenes de Paquetes). . . . .	28
3.6. Tabla de Análisis de la Varianza (caso de enlace de alta velocidad de 100 Mbps). . . . .	28
3.7. Resumen de la campaña con usuarios reales . . . . .	28
3.8. Tabla de Análisis de la Varianza (caso de la campaña de medidas). . . . .	30
4.1. Resumen de las pruebas con tráfico cruzado UDP. $270 \leq m \leq 300, 93 \leq N \leq 100$ . . . . .	43



# Agradecimientos

En primer lugar, me gustaría agradecer a mis padres y a mi hermano su apoyo, su ayuda, su paciencia y su comprensión durante estos años de trabajo. Sin vosotros, no sería ni el científico ni la persona que soy. Gracias, de corazón.

Un especial agradecimiento a mi tutor Javier Aracil porque sin su ayuda, su colaboración, su consejo y su confianza en mí, este trabajo no hubiera sido posible. Quisiera también agradecer a mi co-tutor, Antonio Cuevas, por sus útiles comentarios, sugerencias e ideas, que han ayudado a mejorar la calidad de este trabajo.

Del mismo modo, me gustaría agradecer a los que son mis compañeros de laboratorio y grupo: Javier Ramos, Víctor López, José Luis García, Bas Huiszoon, Felipe Mata, Diego Sánchez, Jaime Garnica, Gustavo Sutter, Jorge López de Vergara, Sergio López-Buedo, Paco Gómez, Luis de Pedro... o lo han sido durante estos años: José Alberto Hernández, Alfredo Salvador y Walter Fuertes. Sin esos cafés que hemos compartido, sin esas charlas futbolísticas, sin esas ligas del comunio... el trabajo hubiera sido más duro. También agradecer a mis compañeros y amigos, en especial a Sergio López, por aguantarme durante tantos años (y tantas entregas de prácticas). A todos aquellos que me haya olvidado mencionar, a todos los que siempre confiaron en mí, y a los que nunca lo hicieron, también va por vosotros. Gracias a todos ;).

Por último, hay que destacar que este trabajo ha sido parcialmente financiado por el Ministerio de Educación del Gobierno de España mediante el programa de becas de Formación de Profesorado Universitario (FPU).



# Capítulo 1

## Introducción

### 1.1. Motivación

En la actualidad, el número de usuarios conectados a Internet está creciendo continua y rápidamente. Además, las capacidades de las redes de comunicaciones y, en particular de las de acceso, son cada vez mayores tanto en ancho de banda como en retardos, e irán en aumento gracias a la introducción de tecnologías ópticas en la llamada última milla (*Fiber To The Home*, FTTH).

Esto está provocando la aparición de una diversidad de servicios en tiempo real con fuertes restricciones de Calidad de Servicio (QoS), como telefonía sobre IP (VoIP), juegos on-line, vídeo bajo demanda (VoD), tele-conferencia, televisión en alta definición/3D (IPTV,HDTV,3DTV), etc.

Uno de los parámetros de QoS que más preocupa a los usuarios es el ancho de banda de su conexión. Por tanto, les interesa cerciorarse de que el ancho de banda recibido se corresponde con el contratado. Cabe destacar que en el pasado año 2009 se recibieron en la Oficina Española de Usuarios de las Telecomunicaciones<sup>1</sup> más de 240 mil quejas, muchas de las cuales se referían a la degradación del servicio de acceso a Internet.

En este sentido, en España existe una orden ministerial (Orden ITC/912/2006, de 29 marzo) basada en la guía de recomendación de la European Telecommunications Standards Institute [ETS05], que establece los procedimientos para la medida del ancho de banda de una conexión a Internet, y que es de obligado cumplimiento para las operadoras.

Tanto para el cumplimiento de la normativa como para mejorar el servicio a sus usuarios, las operadoras de red están interesadas en poder medir con precisión los parámetros de QoS y, en particular, el ancho de banda. Sin embargo, la respuesta a las quejas de los clientes es una tarea costosa para las operadoras, no sólo por la compensación económica que debe ser pagada en caso de degradación del servicio, sino también por los procedimientos de inspección de las conexiones de los usuarios para el diagnóstico de la avería o degradación.

Para simplificar y abaratar estos procedimientos, las compañías ponen a disposición de sus usuarios “velocímetros” (como por ejemplo <http://www.telefonica.es/testvelocidad>) que puedan ejecutarse desde los propios ordenadores de los usuarios. Aunque claramente estos velocímetros son soluciones de bajo coste, presentan una serie de problemas a tener en cuenta. La medida se ejecuta en una plataforma compartida (el ordenador del usuario) y, por tanto, la medida

---

<sup>1</sup><http://www.usuariostelego.es>

puede estar “contaminada” por factores como la carga del ordenador (e.g. alto consumo de CPU y memoria) o por la carga de la red (e.g. carga del enlace de acceso a medir por otras descargas del usuario), véase la Fig. 1.1.

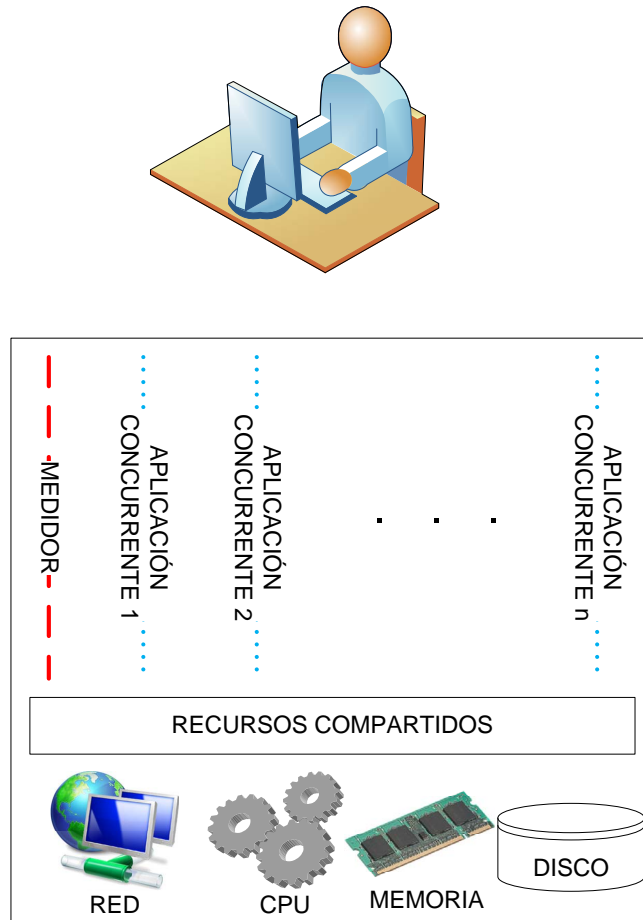


Figura 1.1: El medidor de ancho de banda comparte con las aplicaciones concurrentes los distintos recursos del ordenador.

Para ponernos en contexto, veámoslo con un ejemplo: un usuario que tiene contratado un enlace ADSL 6Mb percibe que la velocidad de la línea es muy inferior a la contratada. Por ello, se queja a su operadora y ésta quiere comprobar el enlace del cliente. Para ello, el usuario se descarga el software de medida y realiza la medida desde su propio ordenador. El resultado de la medida es que el ancho de banda del enlace es de, por ejemplo, 2Mbps (una tercera parte de lo contratado). Pero, ¿puede estar segura la operadora de que el problema de la baja velocidad percibida se debe a un problema del enlace?

Por un lado, si el usuario tiene ejecutándose concurrentemente a la medida otras aplicaciones

(como un reproductor multimedia, el antivirus o firewall, etc), puede que la degradación percibida se deba a la carga introducida por las aplicaciones concurrentes y no a causa de un problema de la línea.

Además, por otro lado, el usuario puede tener conexiones y descargas activas durante la medida. Estas descargas pueden ser explícitas (compartición de ficheros P2P, llamadas Skype, etc) o implícitas (actualizaciones del sistema operativo, software malicioso, etc). Si esto ocurre, la degradación de la velocidad del enlace puede no deberse a ningún problema del operador, sino a la propia carga del enlace por parte del usuario.

Por tanto, es necesario estudiar la influencia de estos factores “contaminantes” como la carga de la CPU, la carga de memoria y la carga de la red, en las medidas de ancho de banda realizadas desde un ordenador. Pero, no sólo es necesario determinar si la medida de ancho de banda se ve afectada por la variación de estos parámetros sino también saber cómo se ve afectada. Esto es, poder decir si una medida es fiable (con cierto nivel de confianza) dados los niveles de carga de CPU y memoria, y la cantidad de tráfico interferente que se ha producido.

El estudio de la influencia se debe hacer desde un punto de vista analítico, utilizando métodos probabilísticos de modelado [Cor07, Ada97] y técnicas estadísticas como el análisis de la varianza (ANOVA) o la regresión lineal múltiple (MLR) [Peñ02].

## 1.2. Objetivos

El trabajo se enfoca en dos tareas fundamentales de la solución al problema, a saber:

- Determinar mediante métodos estadísticos como ANOVA o Regresión Lineal, qué factores del ordenador (carga CPU, carga memoria, sistema operativo, arquitectura del ordenador) influyen en el resultado de la medida de ancho de banda. Las técnicas de medida de ancho de banda sobre las cuales se llevará a cabo el análisis de la influencia de los factores de carga son, principalmente, transferencia de fichero [ETS05] y trenes/pares de paquetes [Pax96, DRM04]
- Definir un modelo probabilístico que nos ayude a determinar la fiabilidad de la medida (probabilidad de que la medida sea correcta) dado el número de paquetes interferentes.

El resto del trabajo se estructura como sigue: en el capítulo 2 se presentará un estado del arte sobre el tema. Por un lado, en la sección 2.1 se describirán las principales técnicas de medida de ancho de banda; por otro lado, en la sección 2.2 se hará un repaso sobre las principales técnicas de probabilidad y estadística que posteriormente serán usadas. El capítulo 3 analiza la influencia de los factores de carga del sistema (CPU, memoria, sistema operativo) sobre las dos técnicas más usadas de medida de ancho de banda, a saber, las basadas en transferencias de fichero y las basadas en trenes y pares de paquetes. En el capítulo 4 definiremos el modelo probabilístico multinomial que nos permitirá rechazar, con cierto nivel de significación, aquellas medidas contaminadas por el tráfico interferente. La validación de los modelos encontrados en los capítulos 3 y 4 se lleva a cabo tanto con simulaciones como con medidas en entornos experimentales y medidas en entornos reales. Por último, en el capítulo 5 resumiremos las principales conclusiones y contribuciones de este trabajo, y ubicaremos las posibles líneas de trabajo futuro.





# Capítulo 2

## Estado del arte

En este capítulo se hará un repaso del estado del arte que nos sirva como punto de partida para el análisis del problema llevado a cabo en este trabajo.

Por un lado, describiremos los métodos de medida de ancho de banda y, en particular, nos centraremos en las dos familias de técnicas que en capítulos posteriores serán analizadas, a saber: las basadas en trenes de paquetes y las basadas en transferencias de ficheros.

Por otro lado, se describirán una serie de técnicas probabilísticas y estadísticas ampliamente usadas en el ámbito del modelado de redes de comunicaciones, como son el análisis de la varianza y la regresión lineal múltiple, que tendremos a nuestra disposición a modo de “caja de herramientas” para ser usadas en capítulos posteriores.

### 2.1. Métodos de medida de ancho de banda

#### 2.1.1. Métodos basados en trenes de paquetes

El primer gran grupo de técnicas de medida de ancho de banda que vamos a describir es el de las basadas en pares (o trenes) de paquetes [Jac88, Kes91, Pax96, DRM04].

La Fig. 2.1 ilustra el principio de funcionamiento básico de estos métodos. Al enviar dos paquetes entre dos puntos de la red, el tiempo que transcurre entre la llegada del último bit del primer paquete y la llegada del último bit del segundo paquete, viene determinado por la velocidad de transmisión del enlace cuello de botella (es decir, del enlace con menor velocidad de transmisión de todo el camino). En el primer tramo los paquetes se separan tanto como el tiempo que tarda en transmitirse el segundo paquete (la longitud del paquete entre el ancho de banda). En el segundo tramo, cuyo ancho de banda es menor, los paquetes se separan más. En el tercer tramo, aunque el ancho de banda sea de nuevo mayor, los paquetes siguen estando dispersos tanto tiempo como lo estaban tras el segundo tramo. Por tanto, la dispersión entre dos paquetes viene dada por el tiempo en que tarda en transmitirse el segundo paquete por el enlace cuello de botella.

#### Notación

- $BW \equiv$  ancho de banda del enlace cuello de botella entre dos puntos de la red.
- $t_i \equiv$  tiempo de llegada del último bit del  $i$ -ésimo paquete.

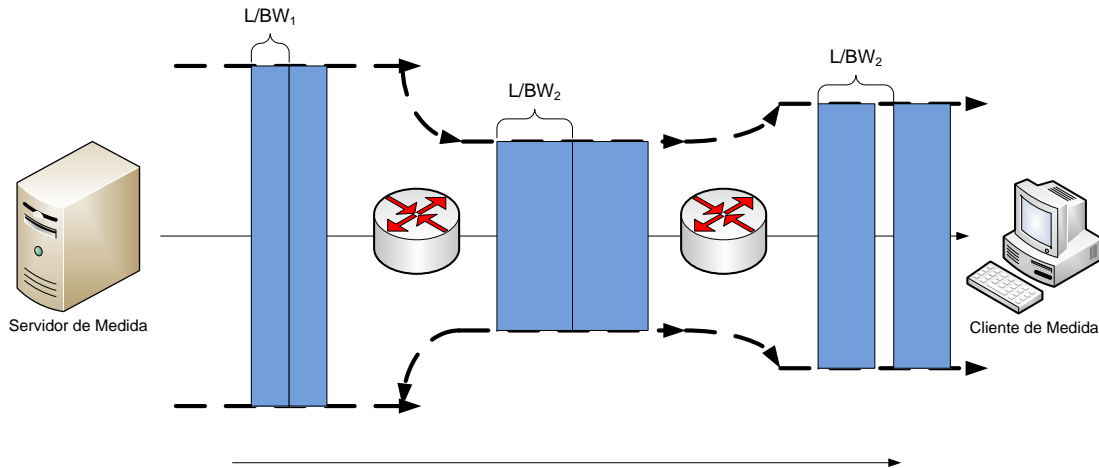


Figura 2.1: Dispersión del par de paquetes.

- $L \equiv$  longitud del paquete
- $\Delta_i \equiv t_{i+1} - t_i$  tiempo entre llegadas (en inglés *interarrival*) entre el paquete  $i$  y el paquete  $i + 1$ .
- $N$  número de paquetes del tren. En el caso de par de paquetes,  $N = 2$ .

Aprovechando el fenómeno de dispersión entre dos paquetes, podemos estimar el ancho de banda de un enlace del siguiente modo:

- Desde un extremo, enviamos dos paquetes, de tamaño  $L$ , a la máxima velocidad.
- Si los paquetes son enviados de manera consecutiva, el interarrival nos proporciona el ancho de banda del enlace cuello de botella, como el cociente del tamaño del paquete y el tiempo de transmisión.

Así, el ancho de banda del enlace cuello de botella, viene dado por la siguiente fórmula:

$$BW = \frac{L}{t_2 - t_1} = \frac{L}{\Delta_1}$$

Las principales ventajas de este método son varias, a saber:

- Rapidez. La duración de la medida es igual al tiempo de transmisión de dos paquetes. Por ejemplo, si medimos un único enlace cuyo ancho de banda es de 10 Mbps con paquetes de longitud 1500 Bytes, la medida tarda  $2 \times \frac{1500 \times 8b}{10 \times 10^6 b/s} = 2.4$  ms.

- No intrusividad. Es decir, los paquetes de medida, al ser sólo dos, no interfieren en el rendimiento de las conexiones existentes en los enlaces que se pretenden medir. Esto es muy útil, ya que no es necesario interrumpir ni degradar el funcionamiento del enlace para poder medir su ancho de banda.

Sin embargo, este método tiene una gran desventaja relacionada con el tráfico cruzado (i.e. el tráfico perteneciente a otras aplicaciones que se cuele entre los dos paquetes de prueba). En el caso de que al menos un paquete perteneciente a otra aplicación se colara entre los dos paquetes de la medida, el interarrival,  $\Delta$ , aumentaría y, por tanto, el ancho de banda estimado sería inferior al real, tal y como podemos observar en la Fig. 2.2.

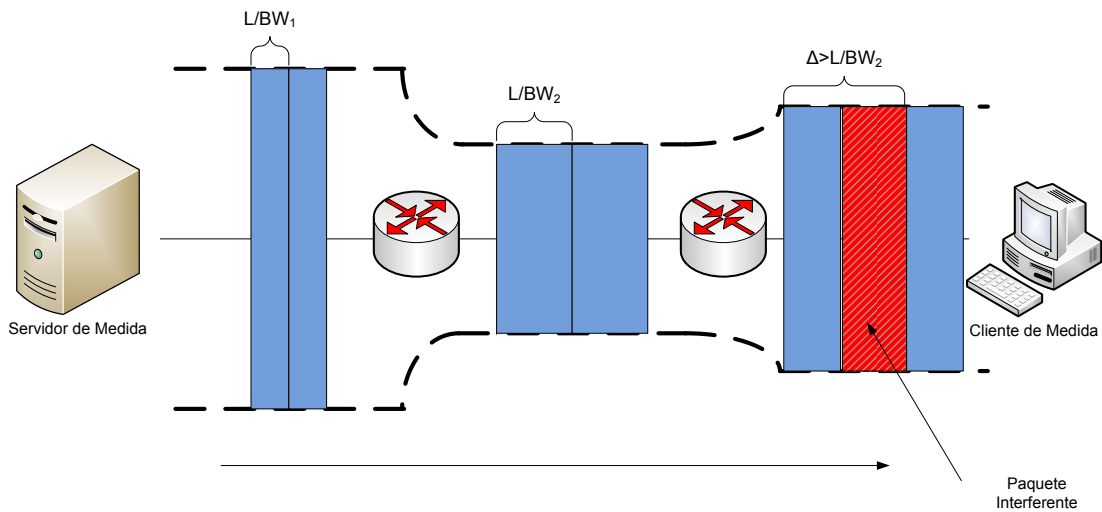


Figura 2.2: Dispersión del par de paquetes cuando hay tráfico cruzado.

Para disminuir esta sensibilidad al tráfico interferente es necesario aumentar el número de paquetes de prueba. Esta mejora de la técnica es lo que se conoce como tren de paquetes. Su funcionamiento es el siguiente:

- En lugar de enviar dos paquetes, enviamos un tren de  $N$  paquetes, de tamaño  $L$ , a la máxima velocidad.
- Desde el receptor, se calculan todos los tiempos entre llegadas de paquetes consecutivos del tren, a saber:  $\Delta_1, \Delta_2, \dots, \Delta_{N-1}$  con  $\Delta_i = t_{i+1} - t_i$
- Se estima el ancho de banda usando el mínimo de los  $\Delta_i$ :

$$BW = \frac{L}{\min_{i=1, \dots, N-1} \Delta_i}$$

De este modo, la medida sólo se verá afectada por el tráfico interferente cuando haya paquetes de otras aplicaciones entre cada uno de los  $N - 1$  pares de paquetes del tren.

Por ejemplo, en la Fig. 2.3 podemos observar como un tren de tres paquetes no se ve afectado por un único paquete interferente. El paquete interferente aumenta el interarrival entre el paquete 1 y el paquete 2. Sin embargo, no afecta al interarrival entre el paquete 2 y paquete 3, con el cual podríamos obtener el ancho de banda del enlace de cuello de botella correctamente.

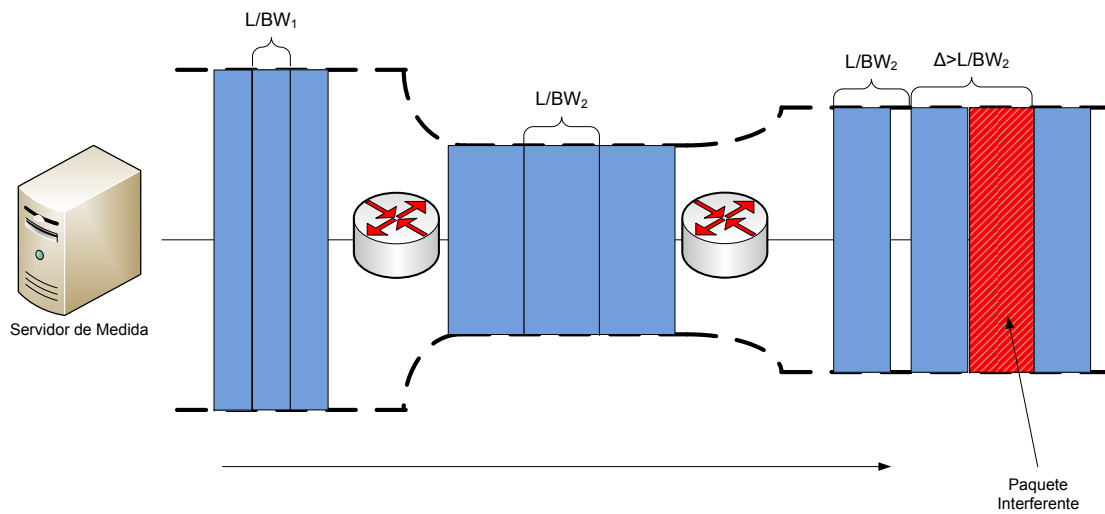


Figura 2.3: Dispersión del tren de paquetes cuando hay tráfico cruzado.

Hay que tener en cuenta que la longitud del tren no se puede hacer arbitrariamente grande, ya que el tráfico de la medida sería demasiado intrusivo, llegando a deteriorar el servicio de la línea que se quiere medir.

### 2.1.2. Métodos basados en transferencias de ficheros

La idea del método es sencilla: medir cuánto tarda la transferencia de un fichero de tamaño conocido. De este modo, estimamos el ancho de banda del enlace como el cociente entre el tamaño del fichero y el tiempo que tarda en ser transmitido.

Dependiendo del nivel en el que sea medido (nivel físico, nivel de red, nivel de transporte o nivel de aplicación), nos proveerá el ancho de banda percibido en cada uno de esos niveles. Por ejemplo, si el enlace a medir es un ADSL 6 Mbps (a nivel físico) pero nosotros medimos el tiempo de descarga de un fichero a través de la web, la velocidad estimada será a nivel de aplicación y obtendremos un ancho de banda estimado menor de 6 Mbps (debido a las cabeceras de los protocolos inferiores: HTTP, TCP, IP, Ethernet).

El método de referencia basado en esta técnica es el descrito en la guía de recomendación ETSI [ETS05], que sirve como referencia para la Orden del Ministerio de Industria, Turismo y Comercio ITC/912/2006, de 29 de marzo, que regula las condiciones relativas a la calidad de servicio en la prestación de los servicios de comunicaciones electrónicas.

El método es el siguiente:

- Se genera un fichero de contenido aleatorio de tamaño 8 veces el ancho de banda teórico del enlace a medir. Por ejemplo, si queremos medir un enlace de 1Mbps, el fichero tiene que tener  $8 \times 10^6$  bits =  $10^6$  Bytes.
- Se transfiere el fichero generado desde un servidor de referencia hasta el ordenador conectado al enlace que se quiere medir. Se mide el tiempo de la transferencia, y se estima el ancho de banda como:

$$BW = \frac{S}{t_{\text{fin descarga}} - t_{\text{inicio descarga}}} [bps]$$

donde  $S$  es el tamaño del fichero en bits,  $t_{\text{inicio descarga}}$  es el instante de tiempo (en microsegundos) en el que el equipo cliente introduce en la red toda la información necesaria para requerir la descarga del fichero de pruebas y  $t_{\text{fin descarga}}$  es el instante de tiempo (en microsegundos) en el que el equipo cliente recibe el último bit del fichero solicitado.

Merece la pena mencionar algunos matices necesarios para el buen funcionamiento del método, a saber:

- Es necesario usar algún algoritmo de resumen, como SHA-1 o MD5, que permita verificar la integridad y corrección del fichero. El tiempo consumido en esta verificación no debe ser contado como parte del tiempo de descarga.
- El contenido del fichero debe ser aleatorio para evitar el uso de algoritmos de compresión y optimización por parte del servidor web desde el que se hace la descarga, que reducirían el tamaño del fichero.
- La orden ministerial obliga a la utilización de HTTP como protocolo para llevar a cabo la transferencia del fichero.

Esta técnica, a pesar de ser la referencia de la ETSI y formar parte de la legislación española que regula la calidad de servicio que las operadoras deben garantizar a sus clientes, tiene una serie de claros inconvenientes, a saber:

- Tiempos más altos de medida: En el caso de que el enlace funcione a la velocidad esperada, la transmisión del fichero tarda en torno a 8 segundos y, en caso de degradación de la calidad, este tiempo aumenta. Hay que recordar que una medida con la técnica de pares de paquetes de un enlace de 1 Mbps dura del orden de los milisegundos.
- Intrusividad: Durante todo el tiempo que dura la medida, la descarga satura el enlace, lo que provoca la degradación de las posibles conexiones existentes.
- Alta sensibilidad al tráfico interferente: Al tratarse de una transferencia TCP (en particular HTTP), el throughput de la conexión se adapta al ancho de banda disponible del enlace. Así,

el ancho de banda del enlace será compartido por todas las conexiones TCP concurrentes, reduciendo el ancho de banda medido. La Fig. 2.4 muestra de manera simplificada el comportamiento para los casos en los que hay ninguna, una y dos conexiones concurrentes (a, b y c, respectivamente).

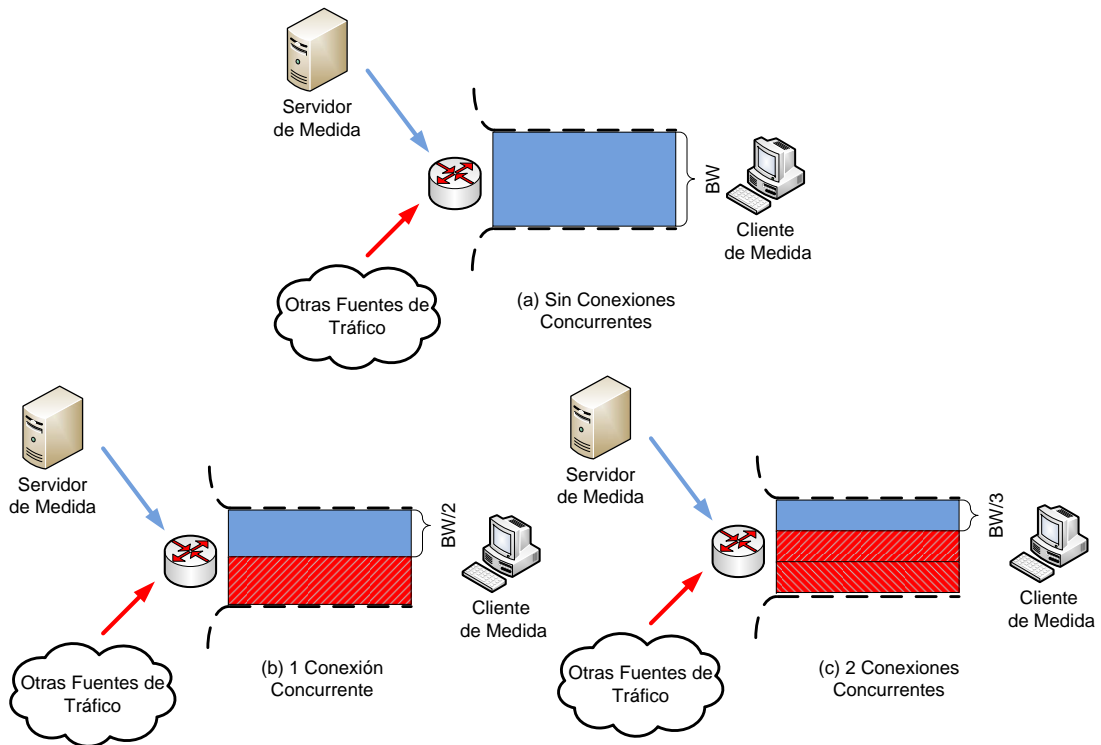
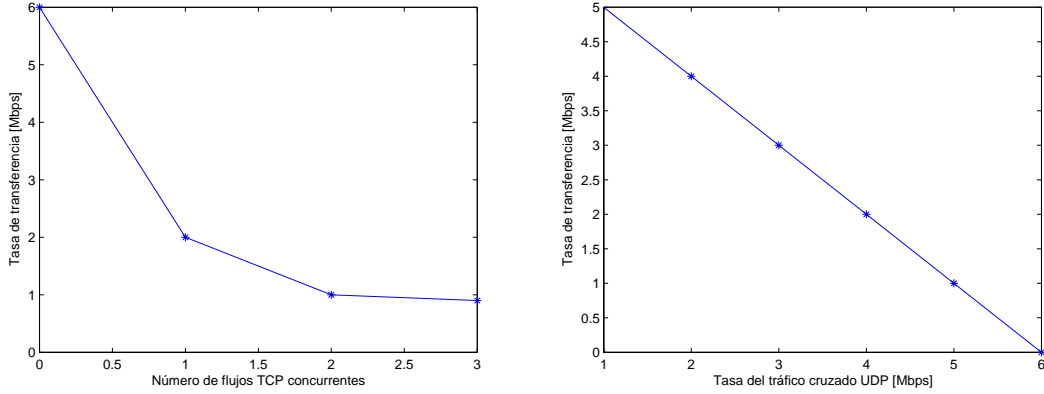


Figura 2.4: Adaptación del ancho de banda al tráfico cruzado.

El comportamiento de flujos TCP en paralelo ha sido ampliamente estudiado en la literatura [HAN02, ABTV06, LQDB05]. El mecanismo para evitar la congestión que implementa TCP provoca que la tasa de transferencia del flujo se vea afectada por las pérdidas de la siguiente medida [PFTK00]:

$$B = \frac{MSS}{RTT} \sqrt{\frac{k}{p}}$$

donde  $MSS$  es el tamaño máximo de segmento,  $RTT$  es el retardo ida y vuelta,  $p$  es la probabilidad de error y  $k$  es una constante que depende de la implementación concreta de TCP. Al aumentar el tráfico interferente, aumentan las pérdidas y, por tanto, se degrada la tasa de transferencia del flujo TCP. Las Fig. 2.5a y 2.5b muestran un ejemplo de degradación de la tasa de transferencia que sufre un flujo TCP por la presencia de otros flujos TCP concurrentes y por la presencia de tráfico cruzado UDP, respectivamente.



(a) Degradación de la tasa de transferencia de un flujo TCP por conexiones TCP concurrentes. (b) Degradación de la tasa de transferencia de un flujo TCP por tráfico cruzado UDP.

Figura 2.5: Degradación de la tasa de transferencia de un flujo TCP.

## 2.2. Métodos de análisis estadístico aplicados a *Networking*

En esta sección veremos dos métodos estadísticos ampliamente usados en el ámbito de las redes de comunicaciones, como son el análisis de la varianza (ANOVA) y la regresión lineal múltiple (MLR). Estos métodos nos permiten estudiar si existe dependencia entre unas variables explicativas o factores y una variable respuesta, y de qué manera influyen estas variables explicativas en la variable respuesta. Estas herramientas nos serán útiles en el siguiente capítulo, para estudiar si existe influencia de los factores de carga en el resultado de la medida del ancho de banda y de qué manera estos factores influyen en la medida.

### 2.2.1. Análisis de la varianza (ANOVA)

El objetivo de los métodos de análisis de la varianza (ANOVA) es estudiar si existe relación entre una variable cuantitativa, la variable respuesta, y una variable cualitativa (o cuantitativa discreta) llamada variable factor. Los valores en los que se presenta el factor se llaman niveles.

La tabla 2.1 muestra un resumen de como se presentan los datos para el caso de un solo factor. Denotemos por  $N := \sum_{i=1}^k n_i$  el número total de muestras. Si  $n_i = n = \frac{N}{k}$ ,  $\forall i = 1, \dots, k$ , se dice que el diseño es equilibrado.

Tabla 2.1: Resumen de datos para el caso ANOVA de un solo factor.

Nivel	Respuesta	Medias	Totales
1	$y_{11}, y_{12}, \dots, y_{1n_1}$	$\bar{y}_{1\bullet} := \frac{1}{N} \sum_{i=1}^{n_1} y_{1i}$	$y_{1\bullet} := \sum_{i=1}^{n_1} y_{1i}$
$\vdots$	$\vdots$	$\vdots$	
$k$	$y_{k1}, y_{k2}, \dots, y_{kn_k}$	$\bar{y}_{k\bullet}$	$y_{k\bullet}$

Para alcanzar su objetivo, descomponemos la variabilidad observada en los datos en variabilidades debidas a distintas causas y las comparamos. Se distinguen dos fuentes de variabilidad, a

saber:

- **Intergrupos:** variabilidad debida al factor
- **Intragrupos:** variabilidad debida a otras causa

En primer lugar, veamos el modelo más sencillo: modelo unifactorial.

$$y_{ij} = \theta_i + \epsilon_{ij} \quad i = 1, \dots, k; \quad k = 1, \dots, n_i$$

$\theta_i$  es el parámetro que representa el valor medio de la respuesta para el nivel  $i$  y  $\epsilon_{ij}$  representa los errores aleatorios debidos a causas distintas al factor (error experimental).

Las hipótesis del modelo son las siguientes:

1. **Indenpendencia**
2. **Normalidad** de los errores. Esta es una hipótesis usual para los errores de medida que generalmente se justifica a través del Teorema Central del Límite.
3. **Homocedasticidad:** La varianza  $\sigma^2$  no depende del nivel.

Con estas hipótesis, tenemos  $\epsilon_{ij} \equiv \mathcal{N}(0, \sigma)$  independientes. Por tanto, el modelo unifactorial es equivalente a:

$$\begin{aligned} y_{11}, \dots, y_{1n_1} &\stackrel{iid}{\sim} \mathcal{N}(\theta_1, \sigma) \\ &\vdots \\ y_{k1}, \dots, y_{kn_k} &\stackrel{iid}{\sim} \mathcal{N}(\theta_k, \sigma) \end{aligned}$$

$k$  muestras normales independientes con igual varianza pero  $k$  medias posiblemente distintas.

La variabilidad total observada en la variable respuesta puede medirse mediante:

$$\sum_{i=1}^k \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_{\bullet\bullet})^2$$

que verifica:

$$\underbrace{\sum_{i=1}^k \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_{\bullet\bullet})^2}_{SCT} = \underbrace{\sum_{i=1}^k n_i (y_{i\bullet} - \bar{y}_{\bullet\bullet})^2}_{SCTR} + \underbrace{\sum_{i=1}^k \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_{i\bullet})^2}_{SCE}$$

donde

- $SCT$  es la suma de cuadrados total
- $SCTR$  es la suma de cuadrados debida a los tratamientos. Mide la variabilidad entre niveles debida al factor.
- $SCE$  es la suma de cuadrados de los errores. Mide la variabilidad dentro de los grupos.



Si el factor no influye en la respuesta, entonces  $\theta_1 = \dots = \theta_k$ . Por tanto, es importante contrastar esta hipótesis, es decir,

$$\begin{cases} H_0 : & \theta_1 = \dots = \theta_k \\ H_1 : & \theta_i \neq \theta_j \text{ para algún par } i, j \end{cases} \quad (2.2.1)$$

La base del contraste de  $H_0$  es el cociente  $\frac{SCTR}{SCE}$ . Si  $H_0$  es cierta, bajo las hipótesis habituales,

$$F = \frac{\frac{SCTR}{k-1}}{\frac{SCE}{N-k}} \stackrel{H_0}{\equiv} F_{k-1, N-k}$$

Por tanto la región crítica del test es

$$R = \{F > F_{k-1, N-k, \alpha}\}$$

Veamos ahora, el modelo bifactorial aditivo.

En este caso, hay dos factores que pueden intervenir en la respuesta, con  $A$  y  $B$  niveles, respectivamente. A cada combinación de niveles se llama tratamiento. El segundo factor, llamado factor bloque, se introduce en el modelo con el objetivo de reducir la SCE y facilitar la posibilidad de encontrar diferencias significativas respecto al otro factor. En este caso no vamos a considerar interacciones entre los factores, es decir, el efecto de un tratamiento es la suma de los efectos de los niveles que lo componen.

El modelo es el siguiente:

$$y_{ij} = \alpha_i + \beta_j + \epsilon_{ij}$$

donde  $\epsilon_{ij} \stackrel{iid}{\sim} \mathcal{N}(0, \sigma)$ .

Siguiendo un razonamiento análogo al caso anterior, para contrastar las hipótesis nulas  $H_{0A} : \alpha_1 = \dots = \alpha_A$  y  $H_{0B} : \beta_1 = \dots = \beta_B$  tenemos las siguientes regiones críticas:

$$\begin{aligned} R &= \{F_A > F_{(A-1), (A-1)(B-1), \alpha}\} \\ R &= \{F_B > F_{(B-1), (A-1)(B-1), \alpha}\} \end{aligned}$$

Cabe destacar que la complejidad y variedad de los modelos puede aumentar, tanto añadiendo nuevos factores al modelo como añadiendo interacciones entre los factores. No obstante, el razonamiento seguido para obtener las regiones críticas de los tests de igualdad de medias es similar. Para más detalle, véase, por ejemplo, [Peñ02]. Por tal razón, no se detallan en esta sección más tipos de modelos ANOVA.

### 2.2.2. Regresión lineal múltiple (MLR)

Los modelos de regresión lineal son una herramienta estadística que nos permite construir una simple fórmula (lineal) para predecir el valor de una variable respuesta a partir del valor dado de un conjunto de variables explicativas o regresoras. También nos sirven para ver si existe una relación lineal entre una variable respuesta y un conjunto de variables explicativas.

En primer lugar, veamos el caso más sencillo: sólo hay una variable explicativa,  $x$ , cuyos valores

$x_i$  están relacionados con los de la variable respuesta,  $y$ , por

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

donde los residuos,  $\epsilon_i$ , son variables independientes idénticamente distribuidas (iid) con distribución normal  $\mathcal{N}(0, \sigma)$ . Así, las respuestas,  $y_i$ , son independientes con distribuciones  $\mathcal{N}(\beta_0 + \beta_1 x_i, \sigma)$ . En particular, se supone que los residuos (y las  $y_i$ ) son homocedásticos (es decir, todos los  $\epsilon_i$  tienen la misma varianza). Esta es una hipótesis crucial que, en la práctica, algunas veces se cumple de forma sólo aproximada. No obstante, el tratamiento estadístico habitual es razonablemente “robusto” frente a desviaciones de esta hipótesis. Las hipótesis del modelo son:

- **Linealidad** en los parámetros.
- **Independencia:** el modelo no sirve para ajustar series temporales, es decir, medidas de la misma variable en distintos instantes de tiempo.
- **Normalidad:** el error es la suma de efectos independientes de poca importancia individual. Esta es una hipótesis usual para los errores de medida que generalmente se justifica a través del Teorema Central del Límite.
- **Media cero:** los errores tienen media nula. Esta hipótesis no se cumple cuando en la muestra aparecen valores atípicos (*outliers*).
- **Homocedasticidad:**  $Var(\epsilon_i) = \sigma^2$  (no depende de  $x_i$ ).

Con estas hipótesis, en términos de las variables respuesta, el modelo se escribe

$$y_i \stackrel{iid}{\equiv} \mathcal{N}(\beta_0 + \beta_1 x_i, \sigma)$$

Nuestro objetivo es una recta  $\hat{\beta}_0 + \hat{\beta}_1 x$  que se “ajuste bien” a los datos experimentales  $(x_1, y_1), \dots, (x_n, y_n)$ . Los procedimientos más usuales para obtener los estimadores son el método de mínimos cuadrados y el de máxima verosimilitud. Es bien sabido que ambos coinciden en el caso de errores normales. Los correspondientes estimadores son:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.2.2)$$

$$\hat{\beta}_0 = \bar{y} - \bar{x} \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.2.3)$$

Para el caso de más de una variable explicativa, el modelo de regresión lineal múltiple para  $k$  variables regresoras es el siguiente:

$$y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_k x_{ik} \epsilon_i \quad (2.2.4)$$

$$\epsilon_i \stackrel{iid}{\sim} \mathcal{N}(0, \sigma) \quad (2.2.5)$$

Expresándolo en forma vectorial:

$$x_i \equiv (1, x_{i1}, \dots, x_{ik})^t, \text{ vector de var. regresoras para la obs. } i \quad (2.2.6)$$

$$X^t \equiv (x_1 \dots x_n), \text{ matriz de diseño} \quad (2.2.7)$$

$$Y \equiv (y_1 \dots y_n)^t, \text{ vector de var. respuesta} \quad (2.2.8)$$

$$\epsilon \equiv (\epsilon_1 \dots \epsilon_n), \text{ vector de errores} \quad (2.2.9)$$

$$\beta \equiv (\beta_0, \beta_1, \dots, \beta_k)^t \quad (2.2.10)$$

$$Y = X\beta + \epsilon \quad (2.2.11)$$

donde  $\epsilon_i \stackrel{iid}{\sim} \mathcal{N}(t, \sigma)$ .

Los parámetros  $\beta$  deben ser estimados a partir del conjunto de datos observados como sigue:

$$\hat{\beta} = (X'X)^{-1}X'y$$



## Capítulo 3

# Análisis de la influencia de las aplicaciones concurrentes

Este capítulo trata sobre el primero de los objetivos marcados para este Trabajo Fin de Máster, a saber: definir un modelo estadístico que nos ayude a determinar cuál es el impacto sobre la precisión de la medida de las condiciones de carga (CPU y memoria) del ordenador sobre el que se ejecuta la medida.

Cabe destacar que en este capítulo estudiaremos la influencia de los factores de carga sobre los dos tipos de técnicas de medida de ancho de banda descritos en la sección 2.1, esto es: métodos basados en trenes de paquetes y métodos basados en transferencias de ficheros. Además de la carga del ordenador, tanto de CPU como de memoria, incluiremos en el estudio la influencia de otro factor como el sistema operativo sobre la calidad de la medida de ancho de banda.

Como vimos en el capítulo 1, es importante (sobre todo, para abaratar costes) que los medidores de ancho de banda sean ejecutados en los ordenadores de los propios usuarios. Sin embargo, esto provoca que el proceso de medida comparta los recursos del PC (CPU, memoria, red, disco duro, sistema operativo, etc) con otros procesos y aplicaciones que se ejecutan de manera concurrente al de medida (actualizaciones del sistema operativo, antivirus y cortafuegos, reproductores multimedia, gestores de descargas, etc). Por ejemplo, estamos interesados en responder a la pregunta, “*Si en el PC donde se ha realizado la medida, bajo un sistema Windows XP de 32 bits, la CPU está cargada al 80% y la memoria al 75% ¿es correcta esa medida (con un cierto nivel de confianza)?*”. Así, nosotros buscamos un modelo que nos diga si es correcta la medida obtenida en un PC con cierto sistema operativo, dados los niveles de carga de CPU y memoria durante la medida.

### 3.1. Análisis experimental

En esta sección, definiremos los modelos estadísticos que mejor ajusten a los datos experimentales obtenidos en una batería de pruebas.

### 3.1.1. Definición de los parámetros de carga

En primer lugar, necesitamos expresar los factores de carga de CPU y memoria. Para ello, definimos los siguientes parámetros:

- Carga de CPU,  $cpu\_frac$ :

$$cpu\_frac = \frac{t_{proc\_fin} - t_{proc\_inicio}}{t_{total\_fin} - t_{total\_inicio}}$$

donde  $t_{proc\_fin}$  y  $t_{proc\_inicio}$  son el tiempo de proceso acumulado por el medidor de ancho de banda al final y al inicio de la medida, respectivamente. Análogamente,  $t_{total\_fin}$  y  $t_{total\_inicio}$  es el tiempo de proceso acumulado por todas las aplicaciones (el medidor inclusive) al final y al inicio de la medida, respectivamente. Por lo tanto,  $cpu\_frac$  representa la fracción de tiempo que el proceso de medida está ejecutándose en el procesador. De este modo, cuando el ordenador está muy cargado con muchas aplicaciones concurrentes, la aplicación de medida estará menos tiempo en el procesador (y  $cpu\_frac$  tendrá valores cercanos a 0). Por otro lado, cuando el ordenador no esté cargado, con pocas o ninguna aplicación concurrente, la aplicación de medida estará más tiempo en el procesador (y  $cpu\_frac$  tendrá valores cercanos a 1).

- Carga de memoria,  $mem$ ,

$$mem = \frac{mem_{libre}}{mem_{total}}$$

donde  $mem_{libre}$  y  $mem_{total}$  son la cantidad de memoria libre antes de la medida y la cantidad de memoria total del sistema, respectivamente. Por lo tanto,  $mem$  representa la fracción de memoria libre en el sistema.

Además, en los experimentos que siguen se va a considerar otro factor adicional: el sistema operativo (se usan dos de los sistemas operativos más populares, a saber, Windows y Linux).

### 3.1.2. Entorno de pruebas

Los siguientes experimentos se han llevado a cabo en una red de área local. La Fig. 3.1 muestra la topología de pruebas.

El enlace de cuello de botella (enlace 3) emula el enlace de acceso doméstico. Esta emulación es llevada a cabo mediante un ordenador con Linux con dos interfaces y la utilidad `tc`<sup>1</sup>. Se han repetido las pruebas para 2 velocidades distintas, a saber: 6 Mbps (la cual es la velocidad de acceso típica para usuarios residenciales y PyMES en España) y 100 Mbps (la cual pronto estará disponible en los hogares españoles y, por ejemplo, ya está extendida en más de 4 millones de hogares en Japón [CFEK06]).

El switch representa la “red troncal” (que asumimos opera a 100 Mbps). Además, para los experimentos de tráfico cruzado que van a ser descritos en el siguiente capítulo, se incluye en la topología un PC como fuente de tráfico interferente que emula tráfico auto-inducido desde el ordenador cliente (por ejemplo, la descarga de un fichero o la reproducción de un vídeo de *YouTube*).

<sup>1</sup><http://linux.die.net/man/8/tc>

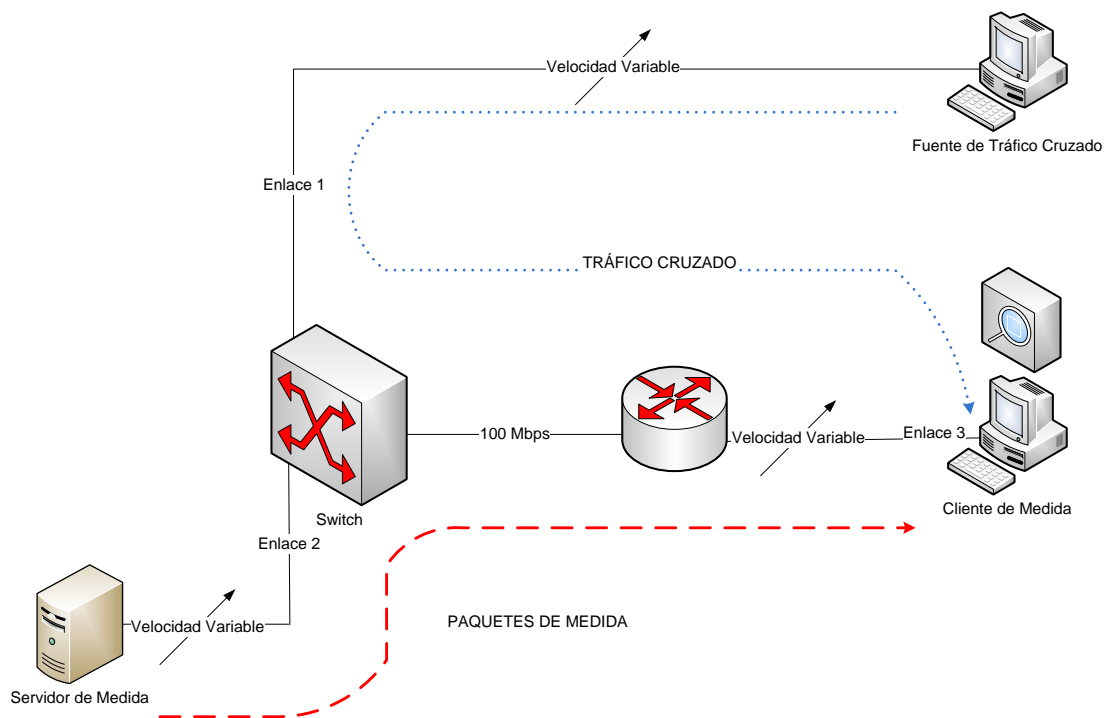


Figura 3.1: Topología de pruebas.

Para generar la carga de CPU, se ejecutan en el ordenador cliente programas, escritos en C y Java, que hacen operaciones con números en coma flotante y procesamiento de ficheros. Por otro lado, para generar carga de memoria, se ejecutan otros programas, C y Java, que reservan una cantidad de memoria y la mantienen reservada durante las pruebas. Estos programas nos sirven para emular las distintas aplicaciones concurrentes.

La configuración de equipos usada para las pruebas es resumida en la tabla 3.1.

Tabla 3.1: Resumen del equipamiento usado para las pruebas.

Ordenador	Procesador	Sistema Operativo
Cliente de Medida	Pentium DualCore 1.80GHz	Linux Windows Vista
Servidor de Medida	Intel Xeon QuadCore 2.33GHz	Linux
Fuente de Tráfico	AMD Geode LX700 433MHz	Linux

### 3.1.3. Métodos basados en transferencias de ficheros

El primer experimento estudia los efectos de la carga de CPU y memoria en la calidad de la medida en el escenario de métodos de transferencia de fichero. Para ello, llevamos a cabo una serie de medida de ancho de banda, usando la técnica de transferencia de fichero, en el cliente de medidas con diferentes niveles de carga de CPU y memoria. Las pruebas se hicieron para ancho de banda a medir de 6 Mbps y 100 Mbps, sobre el cliente con Windows.

La Fig. 3.2 muestra la matriz de gráficos de dispersión para el ancho de banda medido y los parámetros de carga de CPU y memoria. Esta figura representa el grado de correlación entre cada par de variables. Por ejemplo, la entrada (3 : 1) de la matriz de gráficos de dispersión (tercera fila y primera columna) muestra la medida de ancho de banda como función del parámetro *cpu\_frac*. En este caso, cuando la velocidad del enlace a medir es de 6 Mbps, se puede observar que no hay correlación entre la carga (de CPU o memoria) y el ancho de banda medido.

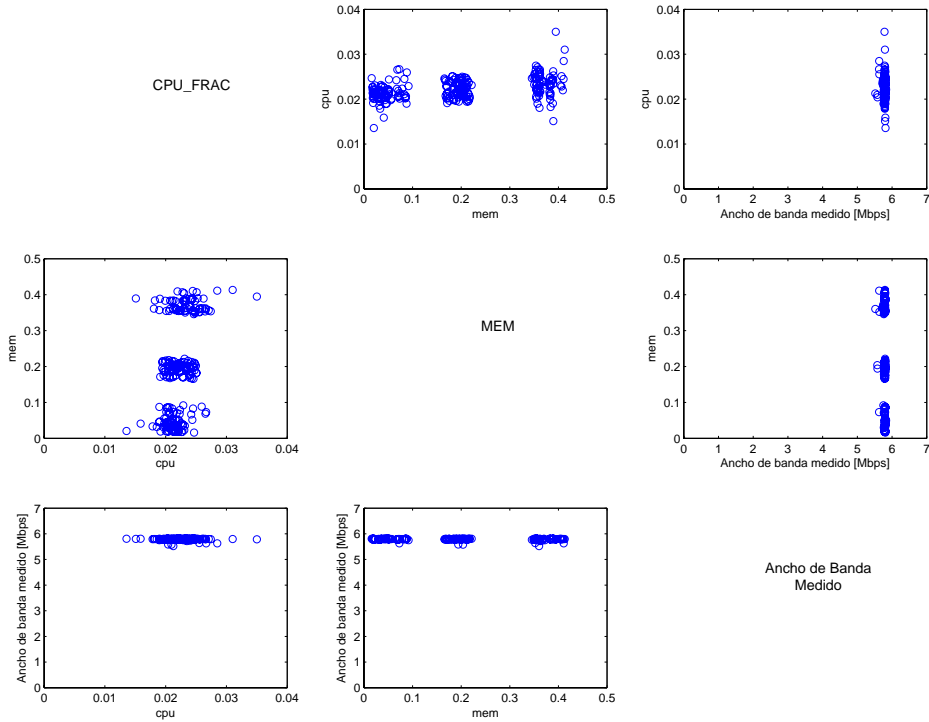


Figura 3.2: Gráficos de dispersión *mem* vs. *cpu\_frac* vs. Ancho de banda medido, para el caso de capacidad del enlace de 6 Mbps.

Para comprobar de una manera más rigurosa la influencia entre carga y ancho de banda medido, llevamos a cabo un test ANOVA bifactorial aditivo. Esto es, suponemos el siguiente modelo:

$$Y_{ijk} = \alpha_i + \beta_j + \epsilon_{ijk}$$

donde:



Tabla 3.2: Tabla de Análisis de la Varianza (caso de enlace residencial de 6 Mbps con transferencia de fichero).

Factor	Grados de libertad	Suma de Cuadrados	p-valor
mem	2	0.001344	0.3145
cpu_frac	2	0.000193	0.8463
Residuos	235	0.271769	

- $\alpha_i$  es el valor medio para el nivel  $i$ -ésimo de carga de CPU. Se definen tres niveles, a saber: baja (el 33% de medidas con *cpu\_frac* mayor), alta (el 33% de medidas con *cpu\_frac* menor) y media (el resto de medidas).
- $\beta_j$  es el valor medio para el nivel  $j$ -ésimo de carga de memoria. Se definen tres niveles, a saber: baja (el 33% de medidas con *mem* mayor), alta (el 33% de medidas con *mem* menor) y media (el resto de medidas).
- $\epsilon_{ijk}$  es el error experimental de la repetición  $k$ -ésima del experimento con nivel de carga de CPU  $i$ -ésimo y nivel de carga de memoria  $j$ -ésimo.

Las hipótesis nulas del test son:

$$H_0^{(1)} : \quad \alpha_{baja} = \alpha_{media} = \alpha_{alta} \quad (3.1.1)$$

$$H_0^{(2)} : \quad \beta_{baja} = \beta_{media} = \beta_{alta} \quad (3.1.2)$$

La tabla 3.2 muestra los resultados del test ANOVA para el caso del enlace residencial de 6 Mbps con el método de transferencia de fichero. A la vista de los resultados, no podemos rechazar ninguna de las hipótesis nulas, para el nivel de significación habitual del 5%. Por lo tanto, podemos concluir que ninguno de los dos factores es estadísticamente significativo (al nivel habitual del 5%) para el caso de velocidades de enlaces domésticos.

Ahora repetimos el experimento para un enlace de 100 Mbps, obteniendo resultados muy diferentes. La Fig. 3.3 muestra la matriz de gráficos de dispersión entre los parámetros de carga y el ancho de banda medido, para este caso. Se observa que hay una alta correlación lineal entre la carga de CPU y el ancho de banda medido. Sin embargo, se aprecia un comportamiento similar al anterior para la carga de memoria (esto es, no se observa correlación entre la carga de memoria y el ancho de banda medido).

Ya que para bajas velocidades hemos observado que no se produce deterioro de la medida con la carga de CPU y memoria, a partir de ahora, nos centramos en el caso de 100 Mbps.

Para modelar los efectos que causan la carga de CPU y memoria sobre la medida, utilizamos un modelo de regresión lineal múltiple como el siguiente:

$$y = \beta_0 + \beta_1 \cdot \text{cpu\_frac} + \beta_2 \cdot \text{mem} + \epsilon$$

donde  $y$  es el ancho de banda medido y  $\epsilon \sim \mathcal{N}(0, \sigma^2)$  representa el error de medida (con valor medio igual a cero y varianza igual a  $\sigma^2$ ). Hay que destacar que los valores de  $\beta_1$  y  $\beta_2$  representan

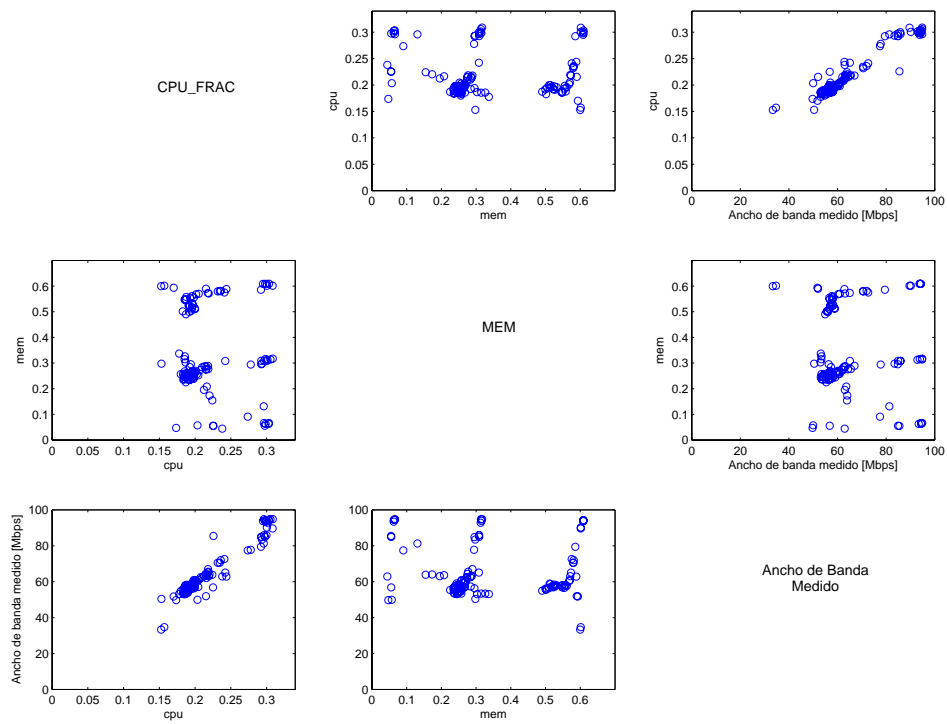


Figura 3.3: Gráficos de dispersión *mem* vs. *cpu\_frac* vs. Ancho de banda medido, para el caso de capacidad del enlace de 100 Mbps.

Tabla 3.3: Coeficientes de regresión para el modelo considerando carga de CPU y memoria.

	Coeficiente [Mbps]	Error Estándar	p-valor
$\hat{\beta}_0$	-3.4789	2.0154	0.087
$\hat{\beta}_1$	311.3522	8.3068	$< 10^{-16}$
$\hat{\beta}_2$	-0.4605	2.1346	0.830
$R^2$	0.9233		

Tabla 3.4: Coeficientes de regresión para el modelo lineal considerando la carga de CPU sólo.

	Coeficiente [Mbps]	Error Estándar	p-valor
$\hat{\beta}$	295.29	1.61	$< 10^{-16}$
$R^2$	0.9965		

la influencia sobre el ancho de banda medido del parámetro  $cpu\_frac$  y del parámetro  $mem$ , respectivamente.

La tabla 3.3 muestra los coeficientes de regresión obtenidos,  $\hat{\beta}_i$ , para cada variable explicativa, la desviación típica estimada y el p-valor para el test cuya hipótesis nula es  $\beta_i = 0$  con  $i = 0, 1, 2$ . Además, la última fila de la tabla nos proporciona el valor  $R^2$ , el cual representa la cantidad de varianza explicada por el modelo lineal.

Los resultados muestran que la carga de memoria no es un factor significativo en el resultado de la medida (no podemos rechazar la hipótesis  $\beta_2 = 0$ ) mientras que la carga de CPU influye en el ancho de banda medido (rechazamos la hipótesis  $\beta_1 = 0$  con p-valor cercano a cero), como esperábamos. Además, cabe destacar que el coeficiente  $\beta_0$  no es significativo (esto es, no podemos rechazar la hipótesis  $\beta_0 = 0$  a un nivel de significación típico  $\alpha = 0.05$ ). Este hecho es coherente en el sentido de que *si el ordenador no da tiempo de proceso a la aplicación de medida (i.e.  $cpu\_frac = 0$ ), el ancho de banda medido es cero*.

Con los resultados anteriores, es más aconsejable un modelo lineal más sencillo que sólo considere la carga de CPU (sin tener en cuenta el parámetro  $mem$  ni el término independiente,  $\beta_0$ ). El modelo ajustado es el siguiente:

$$y = \beta \cdot cpu\_frac + \epsilon$$

donde  $y$  es el ancho de banda medido y  $\epsilon \sim \mathcal{N}(0, \sigma^2)$  representa el error de medida. Nótese que el valor del parámetro  $\beta$  (en Mbps) representa la pendiente de la recta, que pasa por el origen, que mejor ajusta los datos.

La Tabla 3.4 muestra los resultados obtenidos tras la aplicación de la regresión lineal con el modelo simplificado. Este modelo simplificado explica el 99.65 % de la varianza del ancho de banda medido. Cabe destacar que el modelo simplificado explica una cantidad mayor de la varianza del ancho de banda medido que el modelo previo más complejo (el cual explica el 92.33 % de la varianza).

La Fig. 3.4 muestra la regresión lineal estimada para el ancho de banda medido como función

del parámetro de carga  $cpu\_frac$ . Hay que destacar que si, por ejemplo, la fracción de tiempo de proceso para la aplicación de medida,  $cpu\_frac$ , está por debajo del 20%, se producen subestimaciones del ancho de banda de hasta el 70%.

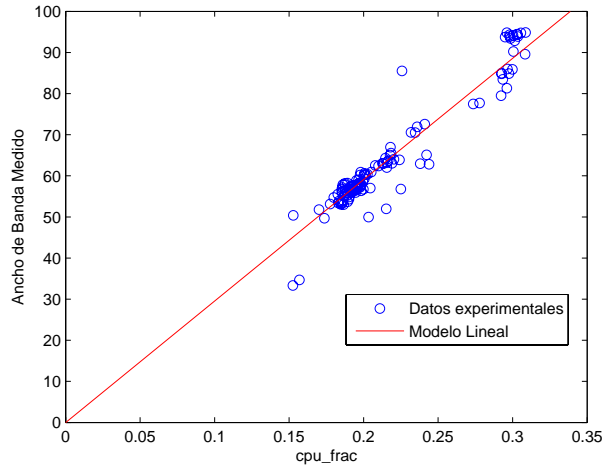


Figura 3.4: Modelo de regresión lineal simplificado, ajustado para el caso de Windows.

Hay que recordar que los experimentos anteriores se han llevado a cabo sobre el cliente de medida con sistema operativo Windows. Para el caso de Linux se obtuvieron resultados similares, es decir, alta correlación lineal entre el parámetro de carga de CPU y el ancho de banda medido (con ligeras diferencias en la pendiente) y ausencia de correlación entre la carga de memoria y el ancho de banda medido. La Fig. 3.5 muestra los resultados para el caso de Linux. Se observa que las escalas de CPU son distintas para los casos de Windows y Linux (lo cual, puede ser debido a los distintos algoritmos de planificación del procesador que tienen los sistemas operativos).

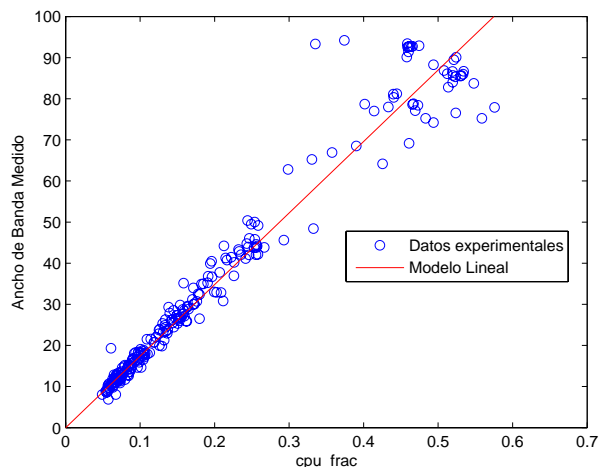


Figura 3.5: Modelo de regresión lineal simplificado, ajustado para el caso de Linux.

A la vista de los resultados, podemos concluir que los métodos de medida basados en transferencias de ficheros son muy sensibles a la carga de la CPU cuando miden enlaces de alta velocidad

(independientemente del sistema operativo sobre el que se ejecute el medidor y la carga de memoria del ordenador), mientras que tampoco se ven afectados por la carga de CPU cuando miden enlaces de baja velocidad (hasta decenas de Mbps). Por tanto, no son buenos candidatos para medir el ancho de banda en plataformas compartidas o de propósito general como los ordenadores de los usuarios, donde se ejecutan otras aplicaciones de manera concurrente.

### 3.1.4. Métodos basados en trenes de paquetes

En los experimentos que se detallan en esta sección, analizamos los efectos de la carga de CPU y memoria sobre el método de medida basado en trenes de paquetes. De aquí en adelante, salvo que se diga lo contrario, la longitud del tren usada en los experimentos es  $N = 100$  paquetes.

Para ello, hacemos medidas en la topología de pruebas pero esta vez usando el método de medida basado en trenes de paquetes, ejecutando la medida sobre todos los tipos de ordenadores cliente descritos en la tabla 3.1. En primer lugar, configuramos la velocidad del enlace de cuello de botella a 6 Mbps (caso de enlace residencial). Las Fig. 3.6a y 3.6b muestran los gráficos de dispersión del ancho de banda medido en función de la carga de CPU (izquierda) y carga de memoria (derecha), respectivamente. Podemos observar, que al igual que en el caso del método de transferencia de fichero, no se observa correlación entre los factores de carga y el ancho de banda medido, para este caso de baja velocidad.

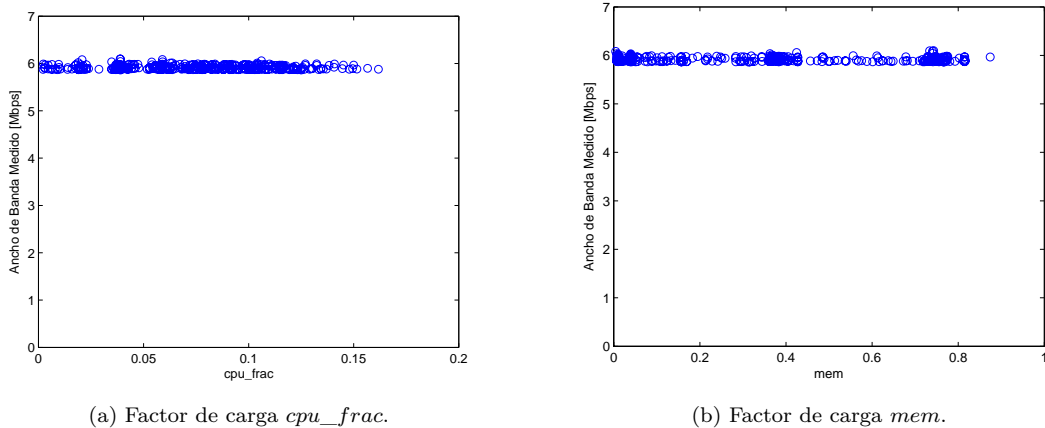


Figura 3.6: Gráfico de dispersión de los parámetros de carga frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 6 Mbps).

Ahora, repetimos el experimento para la configuración del enlace de cuello de botella a 100 Mbps (caso de enlace de alta velocidad). Las Fig. 3.7a y 3.7b muestran los gráficos de dispersión del ancho de banda medido en función de los parámetros de carga de CPU (izquierda) y carga de memoria (derecha), respectivamente. A diferencia del método de transferencia de fichero, el método basado en trenes de paquetes no se ve afectado por la carga del ordenador, ni siquiera en el caso de altas velocidades de enlace (hasta centenas de Mbps).

Para analizar de una manera más profunda la influencia de los 3 factores considerados (CPU, memoria y sistema operativo) y probar estadísticamente la independencia entre éstos y el ancho

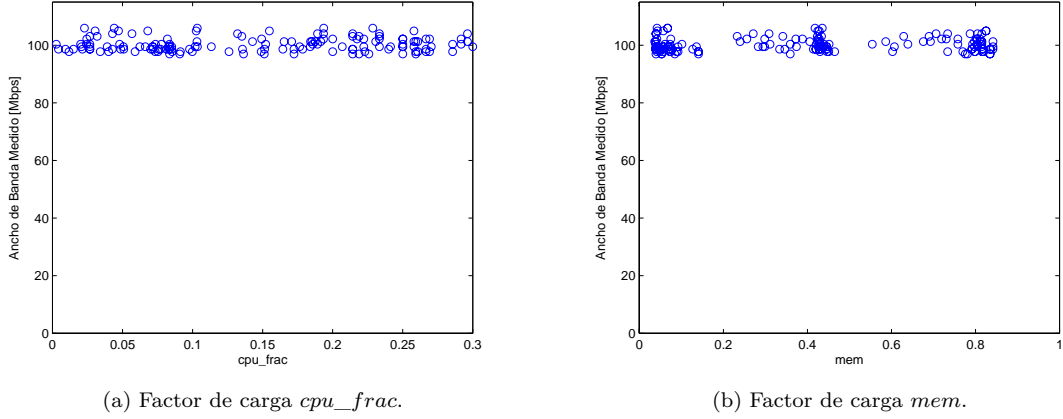


Figura 3.7: Gráfico de dispersión de los parámetros de carga frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 100 Mbps).

de banda medido, se lleva a cabo un test ANOVA trifactorial aditivo (sin interacciones). Esto es,

$$Y_{ijkl} = \alpha_i + \beta_j + \gamma_k + \epsilon_{ijkl}$$

donde:

- $\alpha_i$  es el valor medio para el nivel  $i$ -ésimo de carga de CPU. Se definen tres niveles, a saber: baja (el 33 % de medidas con  $cpu\_frac$  mayor), alta (el 33 % de medidas con  $cpu\_frac$  menor) y media (el resto de medidas).
- $\beta_j$  es el valor medio para el nivel  $j$ -ésimo de carga de memoria. Se definen tres niveles, a saber: baja (el 33 % de medidas con  $mem$  mayor), alta (el 33 % de medidas con  $mem$  menor) y media (el resto de medidas).
- $\gamma_k$  es el valor medio para el nivel  $k$ -ésimo de sistema operativo. Se definen dos niveles, a saber: Windows y Linux.
- $\epsilon_{ijkl}$  es el error experimental de la repetición  $l$ -ésima del experimento con nivel de carga de CPU  $i$ -ésimo, nivel de carga de memoria  $j$ -ésimo y sistema operativo  $k$ .

Las Fig. 3.8a- 3.8c muestran el diagrama de cajas y bigotes (boxplot) para cada uno de los factores para el caso del enlace residencial de 6Mbps. Este gráfico muestra un resumen de la muestra para cada nivel: mínimo, primer cuartil, mediana, tercer cuartil y máximo. Podemos observar que tanto las medias como las dispersiones son (visualmente) similares para los distintos niveles de cada factor.

La tabla 3.5 muestra los resultados del test ANOVA para el caso del enlace residencial de 6 Mbps con el método de trenes de paquetes. A la vista de los resultados, no podemos rechazar las hipótesis de  $\alpha_{baja} = \alpha_{media} = \alpha_{alta}$ ,  $\beta_{baja} = \beta_{media} = \beta_{alta}$  y  $\gamma_{windows} = \gamma_{linux}$ , para el nivel de significación habitual del 5%. Por lo tanto, podemos concluir que ninguno de los tres factores es estadísticamente significativo (al nivel habitual del 5%) para el caso de velocidades de enlaces domésticos. También hemos llevado a cabo el Análisis de la Varianza para el caso de enlaces de

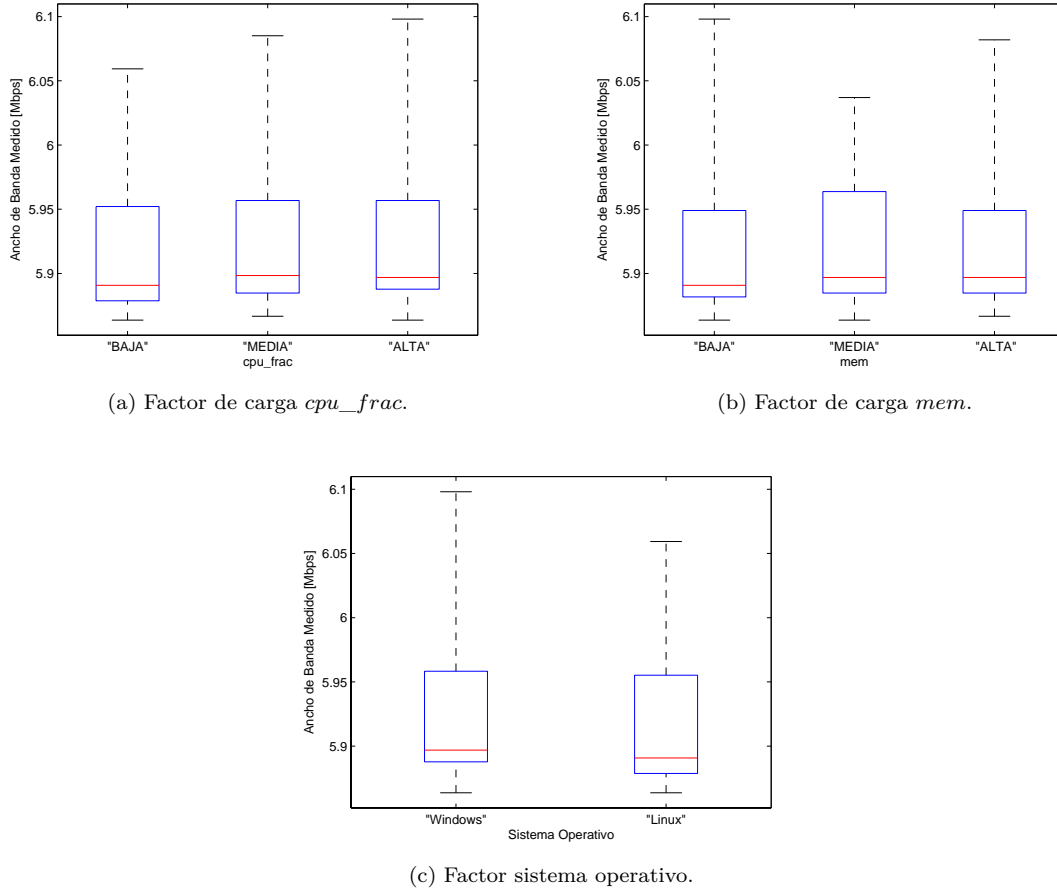


Figura 3.8: Gráfico de cajas de los factores de carga y sistema operativo, frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 6 Mbps).

alta velocidad de 100 Mbps. Las Fig. 3.9a- 3.9c y la tabla 3.6 muestran los gráficos de cajas y la tabla ANOVA, respectivamente, para el caso de 100 Mbps. Se observa un comportamiento muy similar al caso de enlace residencial (no hay influencia de ningún factor sobre la precisión de la medida de ancho de banda).

Los resultados mostrados en esta sección nos permiten concluir que no hay evidencia estadística de que tanto la carga de CPU y memoria como el sistema operativo de la plataforma sobre la que se ejecutan, influyan sobre los métodos de medida basados en trenes de paquetes.

## 3.2. Campaña de medidas en un entorno real

En esta sección, presentamos las medidas que hemos obtenido mediante una campaña de medidas con usuarios domésticos. Para llevar a cabo estas medidas se ha desarrollado una herramienta basada en Java que ha sido publicada en <http://danu.ii.uam.es:8080/QoSPollPro>. Esta herramienta es capaz de medir el ancho de banda de acceso tanto mediante el método de transferencia de fichero como mediante el método de trenes de paquetes. Además, toma los valores de los parámetros de carga (CPU y memoria), el sistema operativo y el tráfico interferente (auto-

Tabla 3.5: Tabla de Análisis de la Varianza (caso de enlace residencial de 6 Mbps para el caso de Trenes de Paquetes).

Factor	Grados de libertad	Suma de Cuadrados	p-valor
Sistema Operativo	1	0.00448	0.1362
mem	2	0.00241	0.5497
cpu_frac	2	0.00036	0.9138
Residuos	414	0.83107	

Tabla 3.6: Tabla de Análisis de la Varianza (caso de enlace de alta velocidad de 100 Mbps).

Factor	Grados de libertad	Suma de Cuadrados	p-valor
Sistema Operativo	1	13.90	0.1062
mem	2	14.87	0.2468
cpu_frac	2	16.28	0.2164
Residuos	162	853.21	

inducido).

Cabe destacar que sólo nos centraremos en las medidas obtenidas con el método de trenes de paquetes, ya que, como hemos visto en la sección anterior, este método mejora al de transferencia de fichero.

En esta campaña hemos obtenido 79 medidas de ancho de banda de bajada de cuatro operadoras españolas, desde diferentes localizaciones del país, de dos servicios diferentes, a saber: velocidad de bajada de 3 Mbps y de 6 Mbps. La tabla 3.7 resume la distribución, por operadora y servicio, de la muestra obtenida en la campaña.

En primer lugar, veamos de manera visual la influencia de la carga de CPU y memoria al ancho de banda medido en la muestra de la campaña. Las Fig. 3.10a y 3.10b muestran el ratio de ancho de banda medido entre ancho de banda contratado frente a los parámetros de carga de CPU, *cpu\_frac*, y memoria, *mem*, respectivamente. Se observa que ninguno de los factores de carga influye y que la variabilidad no cambia con los diferentes valores de *cpu\_frac* y *mem*.

Cabe destacar que en el eje Y se muestra el ratio de ancho de banda medido entre ancho de banda contratado para que, de este modo, sea posible comparar todas las medidas de la campaña,

Tabla 3.7: Resumen de la campaña con usuarios reales

	Operadora 1	Operadora 2	Operadora 3	Operadora 4	Total
<b>3 Mbps</b>	16	0	0	0	16
<b>6 Mbps</b>	0	58	4	1	63
<b>Total</b>	16	58	4	1	79



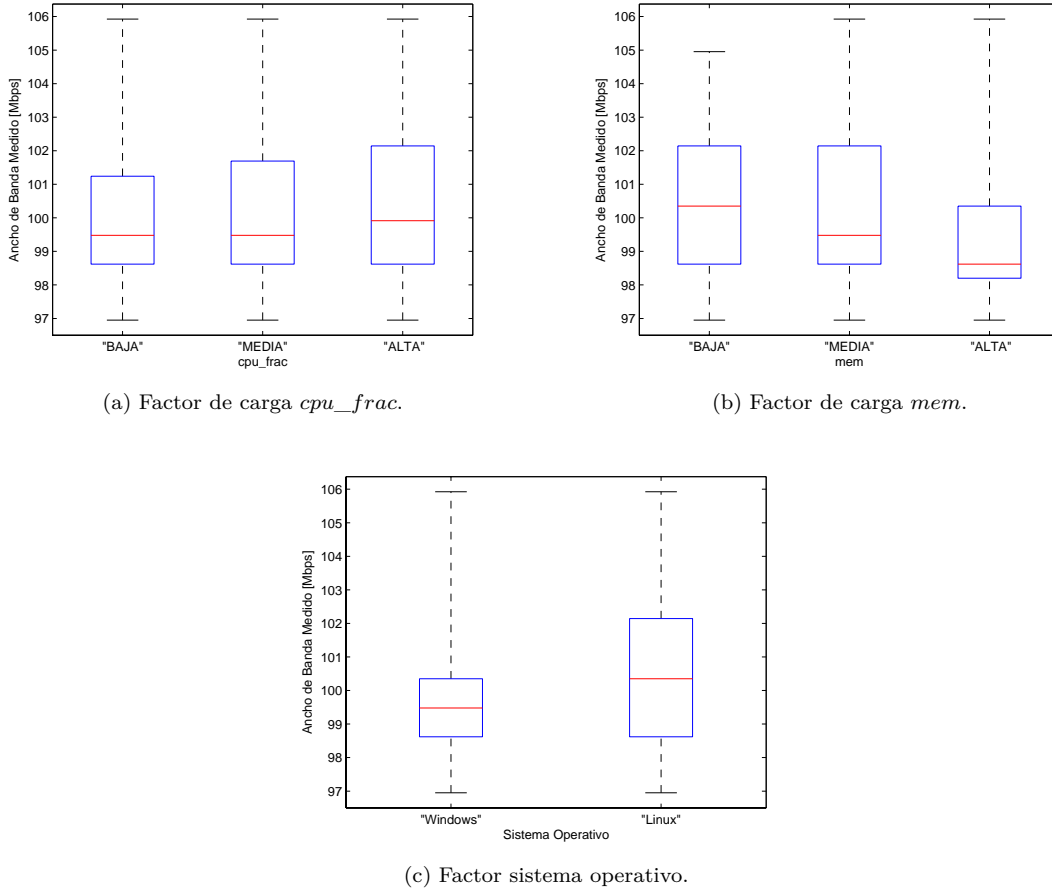


Figura 3.9: Gráfico de cajas de los factores de carga y sistema operativo, frente al ancho de banda medido para el escenario de trenes de paquetes (Enlace de 100 Mbps).

independientemente del valor nominal del ancho de banda del enlace.

Para estudiar de manera más rigurosa la influencia de los factores de carga sobre la medida, realizamos un test ANOVA bifactorial aditivo, similar a los que se han mostrado en la sección anterior. La tabla 3.8 muestra los resultados del test. Como podíamos esperar, ninguno de los dos factores es significativo al nivel habitual del 5%. Por tanto, al igual que en los experimentos del laboratorio, en las medidas en el entorno real no se ha encontrado suficiente evidencia estadística de que la carga de CPU y memoria del PC del usuario influyen sobre los métodos de medida de ancho de banda basados en trenes de paquetes.

Cabe destacar que el ratio ancho de banda medido entre ancho de banda teórico es, en algunos casos, inferior al obtenido en los experimentos. Ya que hemos comprobado que los factores de carga no son factores significativos (y el tráfico interferente no es el suficiente en ningún caso para deteriorar la medida), podemos decir que, en estos casos, hay otros factores que influyen (como, por ejemplo, la operadora de red), o que, en realidad, hay una degradación del servicio en el enlace. Por ejemplo, hay una medida en el nivel de CPU bajo ( $cpu\_frac \approx 0.08$ ) y nivel de memoria medio ( $mem \approx 0.3$ ) con una degradación de la velocidad de la línea del 20%.

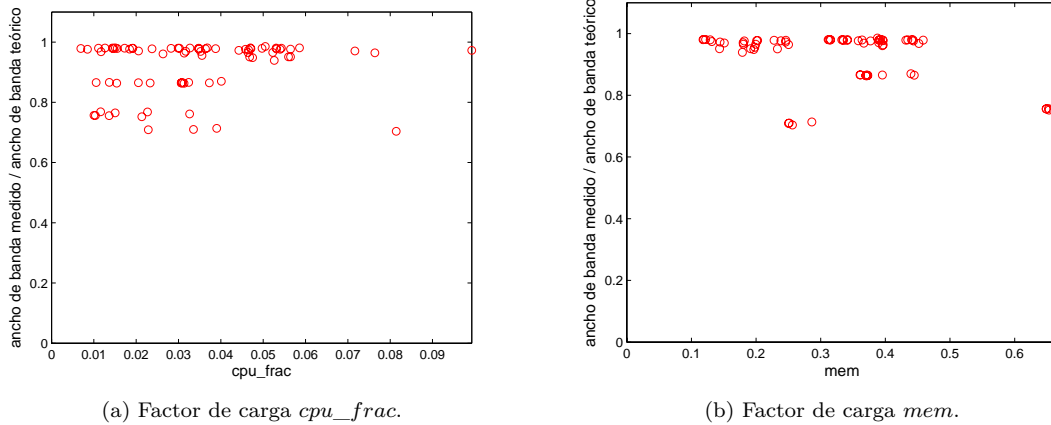


Figura 3.10: Gráficos de dispersión de los factores de carga frente al porcentaje de ancho de banda efectivo en la campaña de medidas.

Tabla 3.8: Tabla de Análisis de la Varianza (caso de la campaña de medidas).

Factor	Grados de libertad	Suma de Cuadrados	p-valor
mem	2	0.02820	0.1436
cpu_frac	2	0.02699	0.1558
Residuos	74	0.52363	

### 3.3. Conclusiones

Por todo lo visto en este capítulo, podemos concluir que:

- El factor carga de memoria, que mide la cantidad de memoria libre en el ordenador de medida, no es un factor significativo en el resultado del ancho de banda medido, en ninguno de los métodos de medida de ancho de banda usados.
- El factor carga de CPU, que mide la cantidad de tiempo que el procesador dedica al proceso de medida, no es un factor significativo en el resultado del ancho de banda medido, para el método de medida basado en trenes de paquetes.
- El factor carga de CPU sí influye en el resultado del ancho de banda medido (con una dependencia lineal), para el método de medida basado en transferencias de ficheros en el caso de medir enlaces de alta velocidad. Por el contrario, la carga de CPU no es un factor significativo en el resultado de la medida en el caso de medir enlaces de baja velocidad.
- El sistema operativo sobre el que se ejecuta la aplicación de medida no es un factor significativo en el resultado del ancho de banda medido.

## Capítulo 4

# Modelo multinomial para rechazar medidas incorrectas

Este capítulo trata sobre el segundo de los objetivos marcados para este Trabajo Fin de Máster, a saber: definir un modelo probabilístico que nos ayude a determinar la fiabilidad de la medida (probabilidad de que la medida sea correcta) dado el número de paquetes interferentes.

Cabe destacar que en este capítulo sólo nos vamos a centrar en el método de medida basado en trenes de paquetes, ya que, como hemos visto en el capítulo 3, los métodos de medida basados en transferencia de ficheros se ven afectados y degradados por el factor carga de CPU (y, por ende, quedan descartados como buenas técnicas de medida), mientras que los métodos basados en trenes de paquetes son inmunes a este tipo de factores.

Como ya hemos dicho, el objetivo de este capítulo es buscar un modelo tal que nos permita rechazar o aceptar medidas dependiendo del número de paquetes interferentes. Por ejemplo, queremos ser capaces de responder a la pregunta “*si se han colado 200 paquetes interferentes entre los 100 paquetes del tren de la medida, ¿es correcta esa medida (con un cierto nivel de confianza)?*”. Esto es, nosotros buscamos un modelo que nos diga la probabilidad de que haya al menos un paquete interferente en todos los huecos entre dos paquetes consecutivos del tren de medidas, para, de este modo, poder decir que la medida es fiable o no (dependiendo si esta probabilidad de error es menor o mayor a la fiabilidad exigida).

Merece la pena recordar que, en este contexto, cuando nos referimos a paquetes interferentes o tráfico cruzado, queremos decir el tráfico que ha sido originado en (destinado a) el ordenador donde se ejecuta la medida y no el que ha sido originado en (destinado a) otros ordenadores de la misma red local. Esto es, no nos referimos al tráfico cruzado en routers intermedios, que ha sido ampliamente estudiado en las técnicas de medida basadas en trenes o pares de paquetes [SK03, CSWS08, RRB<sup>+</sup>03, JD02]. Normalmente, se supone que el dispositivo de medida se dedica exclusivamente a tareas de medida. Sin embargo, es más útil considerar que el dispositivo de medida es un ordenador de propósito general, que está compartido con otras aplicaciones y que, probablemente, sufra de tráfico cruzado auto-inducido. Las fuentes de este tráfico auto-inducido son otras aplicaciones que se ejecutan en el propio ordenador, concurrentemente a la medida, como, por ejemplo, programas de compartición de ficheros P2P, gestores de descargas, antivirus y

firewalls, actualizaciones, etc.

## 4.1. Distribución del mínimo multinomial

### 4.1.1. Planteamiento del problema

Claramente, el tráfico interferente influye en la precisión de la medida. En la Fig. 4.1 podemos observar cómo se ve afectada una medida por el tráfico interferente hacia el ordenador. En este ejemplo, se envía un tren de 3 paquetes y entre cada uno de ellos, se cuela un paquete perteneciente a otras aplicaciones concurrentes, que, para simplificar, supondremos del mismo tamaño que los paquetes de nuestro tren. Este hecho provoca que el tiempo entre llegadas de cualquier par de paquetes del tren aumente al doble y, por tanto, el ancho de banda medido se reduce a la mitad del ancho de banda real del enlace.

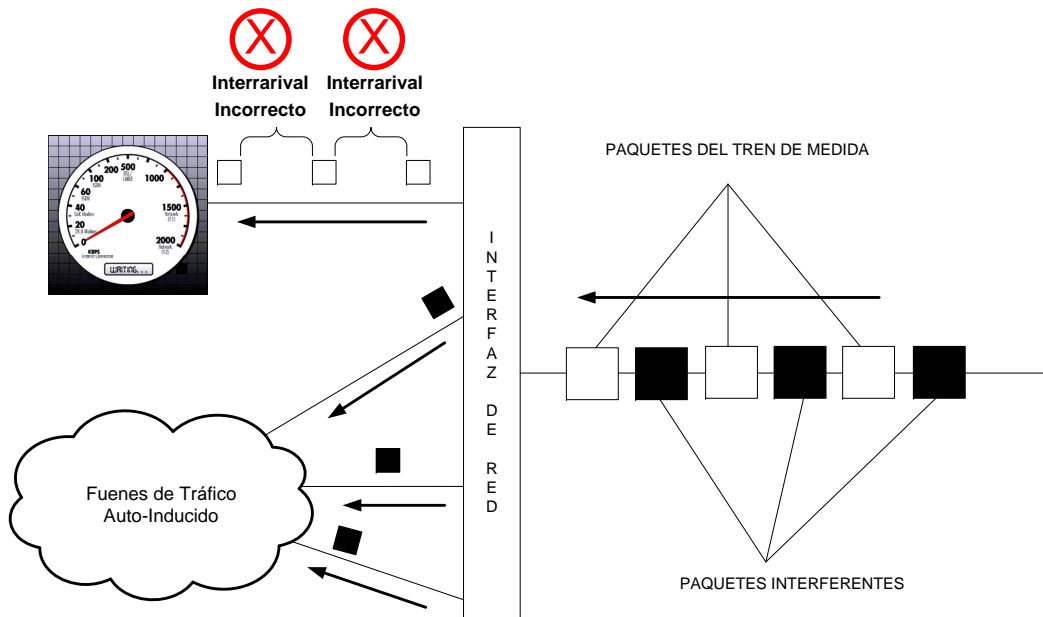


Figura 4.1: Medida afectada por el tráfico interferente.

Así, para saber si la medida es correcta necesitamos saber si, al menos, dos paquetes del tren han llegado de manera consecutiva o, lo que es lo mismo, sin paquetes interferentes entre ellos. El tiempo entre la llegada de estos dos paquetes nos sirve para estimar el ancho de banda del enlace,

con la fórmula siguiente:

$$BW = \frac{l}{\Delta t} \quad (4.1.1)$$

donde  $l$  es la longitud de los paquetes de prueba  
 $\Delta t$  es el tiempo entre llegadas de los paquetes

(4.1.2)

La figura 4.2 ilustra este comportamiento. Como en el caso anterior, se envía un tren de 3 paquetes, pero a diferencia de aquel, hay dos paquetes del tren de medida que llegan consecutivamente y, por tanto, podemos estimar de manera correcta el ancho de banda a partir del tiempo entre las llegadas de esos dos paquetes.

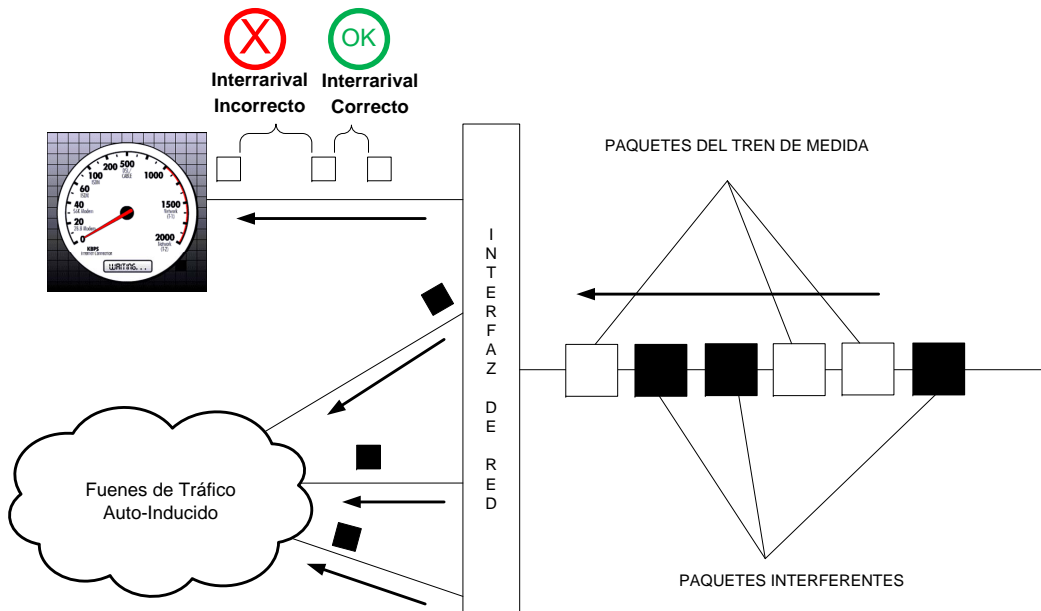


Figura 4.2: Medida correcta a pesar del tráfico interferente.

Por lo tanto, es una tarea crucial saber si los paquetes del tren de medida han llegado de manera consecutiva. Una primera (e ineficaz) aproximación nos lleva a capturar todos los paquetes que están entrando por la interfaz de red del ordenador y ver si han llegado de forma consecutiva, al menos, dos paquetes del tren. Para conocer el número de paquetes interferentes, la aplicación medidora consulta un contador de paquetes que mantiene el sistema operativo. Sin embargo, estos contadores no son actualizados en intervalos pequeños de tiempo y, por tanto, no es posible saber cuántos paquetes interferentes se han colado entre cada dos paquetes del tren de medida. Sólo es posible saber cuántos paquetes interferentes se han colado en total en el tren de medidas.

Por lo tanto, buscamos una técnica que nos permita marcar una medida como incorrecta, con el único conocimiento del número total de paquetes interferentes durante todo el tren de medida. Con este fin, se ha desarrollado un modelo analítico que nos da una cota conservadora del número máximo de paquetes interferentes permitidos para considerar una medida como válida, con cierta confianza.

Sea  $N$  el número total de paquetes del tren de medida y  $m$  el número total de paquetes interferentes al tren de medidas. Nuestro objetivo es calcular la probabilidad de que al menos 2 paquetes del tren lleguen de manera consecutiva, esto es, la probabilidad de que no haya paquetes interferentes entre, al menos, dos paquetes del tren. Por simplificación, hacemos la siguiente hipótesis: cada paquete interferente (de los  $m$  totales) cae con igual probabilidad en cualquiera de los  $N - 1$  huecos entre paquetes del tren. Es evidente, que un paquete interferente no puede ser transmitido en un hueco entre paquetes que sucedió antes de la transmisión real del paquete. Por lo tanto, esta suposición es de caso peor, esto es, la probabilidad real de descarte de una medida es menor que la probabilidad dada por nuestro modelo. La Fig. 4.3 ilustra este hecho.

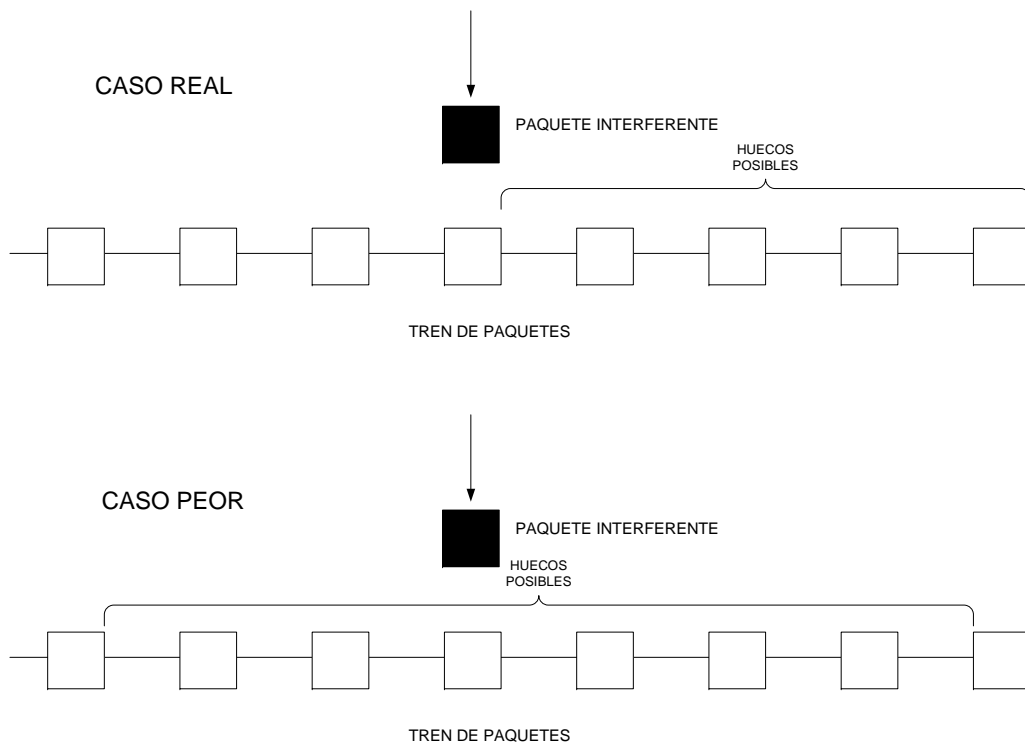


Figura 4.3: Caso real frente a caso peor.

Cabe destacar que el objetivo es rechazar con muy alta probabilidad las medidas que estén contaminadas por el tráfico interferente, aunque para ello rechazamos medidas que eran correctas. Esto es, el objetivo es que las medidas que cataloguemos como correctas sean correctas con mucha confianza.

### 4.1.2. Definición del modelo

#### Notación

- $N \equiv$  número total de paquetes del tren de medida. Nótese que el número total de huecos entre paquetes son  $N - 1$ .
- $m \equiv$  número total de paquetes interferentes dentro del tren de medida.
- $p_i \equiv$  probabilidad de que un paquete interferente dado caiga en el hueco  $i$ -ésimo, para  $i = 1, \dots, N - 1$ .
- $n_i \equiv$  número de paquetes interferente que han caído en el hueco  $i$ -ésimo. Trivialmente se tiene que  $\sum_{i=1}^{N-1} n_i = m$ .

#### Hipótesis del modelo

- La distribución de un paquete interferente dado en los  $N - 1$  es uniforme. Esto es, la probabilidad  $p_i \equiv p = \frac{1}{N-1}$ . Esta hipótesis es coherente con el proceso de llegadas de paquetes de Poisson.
- La distribución de un paquete interferente en los  $N - 1$  huecos es independiente de la distribución de los otros  $m - 1$  paquetes.

Con las hipótesis anteriores, el reparto aleatorio de los  $m$  paquetes interferentes en los  $N - 1$  huecos entre los paquetes del tren, sigue una distribución multinomial. Por lo tanto, la probabilidad de ocurrencia de cierta distribución  $(n_1, \dots, n_{N-1})$  viene dado por:

$$P(n_1, \dots, n_m) = \prod_{k=1}^{m-1} \binom{m - n_{k-1}}{n_k - n_{k-1}} \left( \frac{1}{N-1} \right)^{n_k - n_{k-1}} \left( \frac{N-2}{N-1} \right)^{m - n_k} \quad (4.1.3)$$

Nosotros estamos interesados en hallar la probabilidad de que la medida sea errónea, esto es, la probabilidad de que ningún hueco contenga menos de 1 paquete:

$$P \left( \min_{i=1, \dots, N-1} n_i \geq 1 \right) \quad (4.1.4)$$

Cabe destacar que en todo el análisis anterior el número de paquetes interferentes,  $m$ , está prefijado. Esto es, las probabilidades obtenidas están condicionadas al valor de  $m$ . Sin embargo, el análisis podría ser ampliado para considerar un enfoque en el que el valor de  $m$  también sea una variable aleatoria. Esto puede venir motivado por la necesidad de medir enlaces de muy alta velocidad ( $\geq 1Gbps$ ). En este contexto, la duración de la medida es muy inferior y, por lo tanto, no es posible saber la cantidad exacta de paquetes interferentes a la medida, sino, únicamente, una estimación de la misma.

### 4.1.3. Cálculo de la probabilidad de error

Obtener la probabilidad exacta de la mínima cantidad de paquetes que cae en un hueco cuando repartimos  $m$  paquetes en  $N - 1$  huecos no es una tarea fácil. Merece la pena destacar que la cantidad total de posibles distribuciones de los paquetes en los huecos se incrementa rápidamente

tanto con el número de paquetes como con el número de huecos. Por ejemplo, hay 10 posibilidades de repartir 5 paquetes en 2 huecos, mientras que hay 12 mil millones de posibilidades de repartir 50 paquetes en 10 huecos. Notar que, en el caso de una medida real, tenemos un orden de magnitud más (la longitud de un tren está en torno a 100 paquetes), con el consecuente aumento del número de posibilidades. En [Cor07] proponen un modo rápido para calcular la distribución del mínimo multinomial que buscamos.

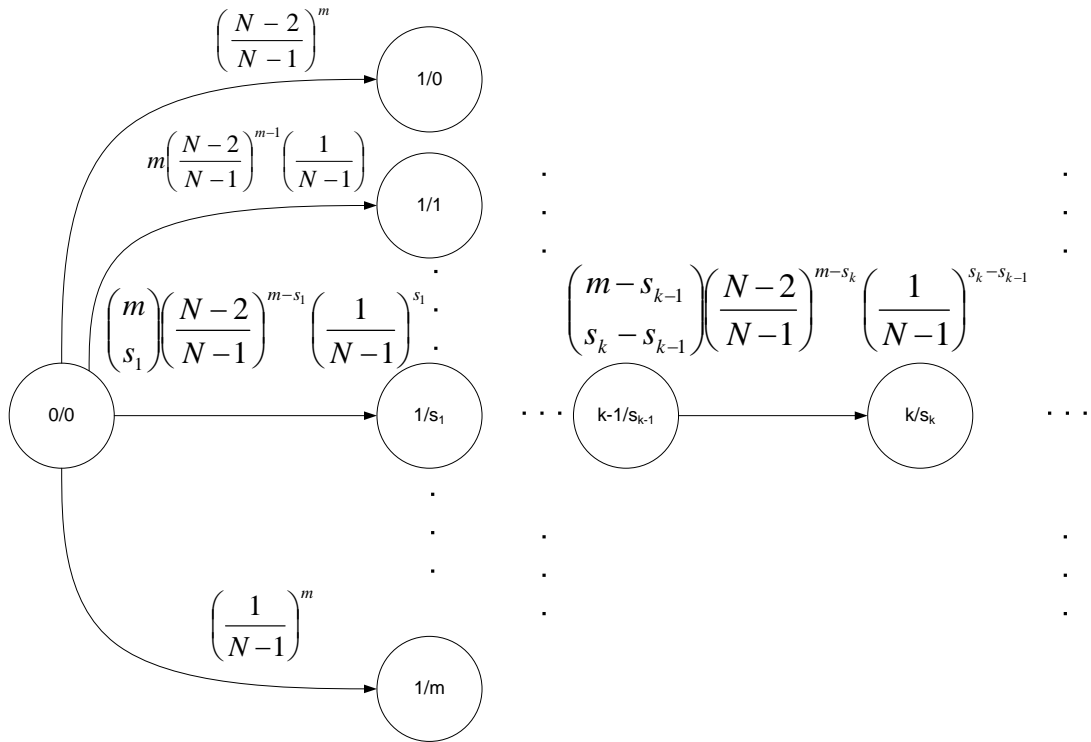


Figura 4.4: Diagrama del proceso estocástico de reparto de paquetes en los huecos.

El problema de la distribución de la multinomial puede ser visto como un proceso estocástico que representa el reparto de paquetes en los huecos. La Fig. 4.4 muestra un diagrama que ilustra este proceso de reparto de paquetes. El diagrama se puede leer como sigue: el estado genérico  $k/s_k$  da el número de paquetes,  $s_k$ , que han llenado los  $k$  primeros huecos, para  $k = 0, \dots, N-1$  y  $s_k = 0, \dots, m-s_{k-1}$ ,  $s_0 = 0$ . Por ejemplo, el estado etiquetado con  $1/1$  nos dice que 1 paquete ha caído en el primer hueco. Además, la figura también muestra las probabilidades de transición, en general, del estado  $k-1/s_{k-1}$  ( $s_{k-1}$  paquetes en los primeros  $k-1$  huecos) al estado  $k/s_k$  ( $s_k$  paquetes en los primeros  $k$  huecos). Claramente,  $s_{k-1} \leq s_k$ . La probabilidad de transición entre el estado  $k-1/s_{k-1}$  al estado  $k/s_k$  es la probabilidad de que  $s_k - s_{k-1}$  paquetes de los  $m - s_{k-1}$  que quedan, caigan en el hueco  $k$ -ésimo. La expresión analítica de esta probabilidad de transición



es la siguiente:

$$P(s_k|s_{k-1}) = \left\{ \begin{array}{ll} \binom{n-s_{k-1}}{s_k-s_{k-1}} \left(\frac{1}{N-1}\right)^{s_k-s_{k-1}} \left(\frac{N-2}{N-1}\right)^{m-s_k} & \text{para } s_k \geq s_{k-1} \\ 0 & \text{en otro caso} \end{array} \right\} \quad (4.1.5)$$

$$k = 1, \dots, N-1$$

$$s_k = 0, \dots, m - s_{k-1} \leq m$$

Por ejemplo, la transición del estado inicial 0/0 al estado 1/1 ocurre con probabilidad

$$m \times \frac{1}{N-1} \times \left(\frac{N-2}{N-1}\right)^{m-1}$$

porque esta transición ocurre cuando 1 de los  $m$  paquetes cae en el primer hueco, y los  $m-1$  restantes no.

Sea  $Q_k$  la matriz estocástica que determina las transiciones entre los estados previos, esto es, aquella cuya entrada  $s_{k-1}, s_k$  contiene la probabilidad mostrada en la ecuación 4.1.5. El producto de las matrices  $Q_1 \cdot \dots \cdot Q_k$  representa la distribución de la convolución  $s_k = n_1 + \dots + n_k$ . Hay que observar que, en  $k = m$ , la distribución de la convolución es la distribución degenerada en  $m$  (i.e. toma el valor  $m$  con probabilidad 1). Con esta notación, el cálculo de las distribuciones multinomiales conjuntas y, en particular, la que nos atañe del mínimo, se reduce a un producto de  $m$  matrices, convenientemente construidas.

Para calcular la probabilidad de que ningún hueco contenga menos de 1 paquete, la probabilidad de transición entre los estados  $a/k-1$  y  $a/k$  debe ser cero para cualquier  $a$  (i.e.  $P(s_k|s_{k-1}) = 0$  siempre que  $s_k = s_{k-1}$ ). O, lo que es lo mismo, tenemos que construir las matrices  $Q_k$  con las entradas  $a, a$  puestas a cero para cualquier  $a$ . Por tanto, el producto  $Q_1 \cdot \dots \cdot Q_m$  nos da  $P(\min_{i=1, \dots, N-1} n_i \geq 1)$  exactamente.

La Fig. 4.5 muestra la probabilidad de error de una medida variando el número de paquetes interferentes,  $m$ , de 0 a 1000, para distintos valores de la longitud del tren de medidas,  $N = \{10, 50, 100, 150\}$ . Claramente, la probabilidad de error se incrementa con el número de paquetes interferentes  $m$ . Además, se observa que la probabilidad de error decrece rápidamente cuando la longitud del tren,  $N$ , se incrementa. Por ejemplo, para  $N = 50$  y  $m = 3 \times N = 150$ , la probabilidad de error es mayor del 5% (0.085) mientras que si  $N = 100$  y  $m = 3 \times N = 300$  la probabilidad es menor del 1% (0.005) e, incluso más, si  $N = 150$  y  $m = 3 \times N = 450$ , la probabilidad de error es más pequeña del 1 por mil (0.0003).

De este modo, cuando hacemos una medida, sabemos de antemano el número de paquetes del tren,  $N$ , (y, por tanto, de huecos,  $N-1$ ) y podemos conocer el número total de paquetes interferentes,  $m$ . Y, gracias al modelo analítico, podemos calcular entonces la probabilidad de que la medida sea incorrecta. En conclusión, podemos responder a la pregunta: “*si se han colado 200 paquetes interferentes entre los 100 paquetes del tren de la medida, ¿cuál es la probabilidad de que esa medida sea incorrecta?*”. También, visto de otro modo, dada la longitud del tren de paquetes,  $N$ , podemos calcular el número de paquetes interferentes,  $m_0$ , tal que, la probabilidad de que una medida, con  $m_0$  paquetes interferentes sea errónea, sea menor o igual que un cierto umbral  $\epsilon$  (típicamente del 5%). Por ejemplo, “*con trenes de 100 paquetes, ¿cuál es el número máximo de paquetes interferentes para considerar la medida correcta con probabilidad mayor del 95%?*”.

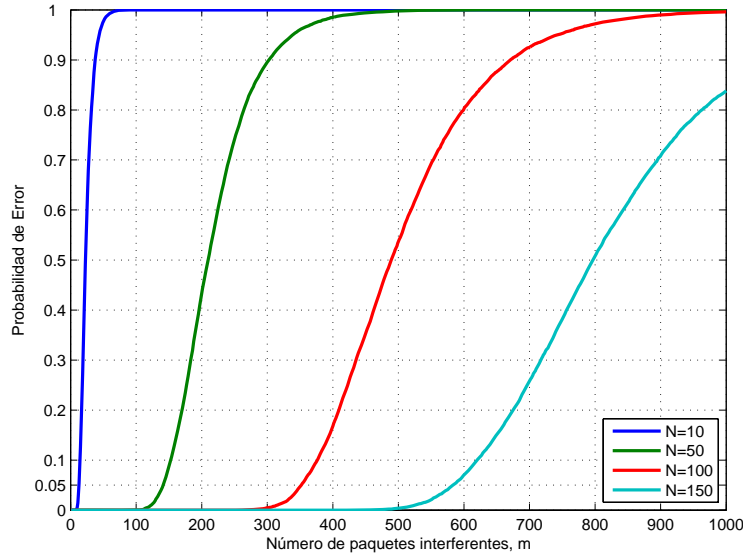


Figura 4.5: Distribución de la probabilidad de error respecto al número de paquetes interferentes.

La Fig. 4.6 nos permite responder a esta pregunta. En el eje  $X$  se muestra la longitud del tren de medida,  $N$ , mientras que en el eje  $Y$  se muestra el número máximo de paquetes interferentes,  $m_{max}$ , para el que la medida de  $N$  paquetes puede ser considerada correcta eligiendo una confianza del 95% (probabilidad de error  $\leq 5\%$ ). Podemos observar que el número máximo de paquetes interferentes crece con la longitud del tren de medida. Por ejemplo, el número máximo de paquetes interferentes para poder aceptar la medida es igual a 141, 352 y 588 para longitudes del tren de paquetes de 50, 100 y 150 paquetes, respectivamente.

#### 4.1.4. Algoritmo Adaptativo de la longitud del tren de medida

Si la probabilidad de error de una medida (dadas la longitud del tren de medida y el número de paquetes interferentes) es mayor que un cierto umbral (digamos, un 5%), la medida queda descartada porque su resultado no es lo suficientemente fiable. Entonces, debemos repetir la medida, pero ¿qué longitud elegimos para el tren de medida? Si las condiciones de tráfico interferente son similares, muy probablemente, la medida siga siendo incorrecta para la misma longitud del tren de medida. Pero, por otro lado, si hacemos que el tren de medida sea demasiado largo la medida será demasiado intrusiva (esto es, el tráfico de pruebas puede afectar al rendimiento de las conexiones activas que van por enlace a medir).

Por lo tanto, es importante adecuar la longitud del tren de medidas a la intensidad del tráfico interferente, de tal manera que sea lo suficientemente largo para conseguir medidas correctas pero lo suficientemente corto para que la medida no sea intrusiva con el tráfico del enlace que se está midiendo. En este sentido, gracias al modelo propuesto, podemos calcular la longitud mínima del tren de medidas,  $N_{min}$ , tal que la probabilidad de error de la medida sea inferior al 5%, dada una intensidad de tráfico interferente. Con intensidad de tráfico nos referimos al ratio entre paquetes interferentes y longitud del tren de medida,  $\frac{m}{N}$ .

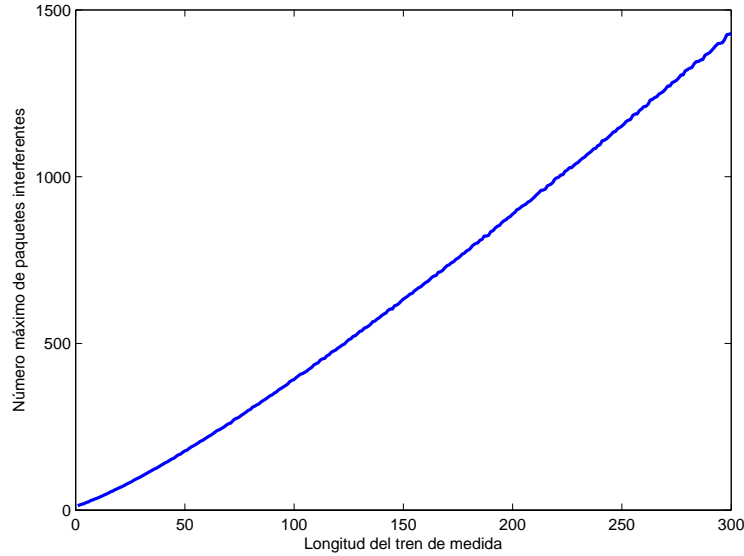


Figura 4.6: Número máximo de paquetes interferentes para que la medida sea considerada correcta ( $p_{error} < 5\%$ ) en función de la longitud del tren.

La Fig. 4.7 muestra la longitud del tren mínima para la cual la medida es correcta (con probabilidad  $\leq 5\%$ ) dada la tasa de tráfico interferente,  $\frac{m}{N}$ . Se observa que para tasas de tráfico interferente de hasta 3 (i.e. se cuelan en total 3 paquetes interferentes por cada paquete del tren de medida) se pueden usar trenes de longitud 100 paquetes. Por otro lado, para tasas de hasta 5 paquetes interferentes por cada paquete de medida, se necesitan trenes de longitud 450. Cabe destacar que para tasas de paquetes interferentes superiores a 5, la longitud del tren de paquetes debe ser demasiado grande para mantener la característica de no intrusividad que se busca.

Ayudándonos del modelo y de las gráficas anteriores, podemos diseñar un algoritmo que sea capaz de adaptar la longitud del tren de medidas a la tasa de tráfico interferente que está sufriendo el enlace a medir. El algoritmo viene descrito en 4.1. Se realiza una primera medida con una longitud de tren inicial,  $N_0$  (por ejemplo, de 100 paquetes). Si la probabilidad de error es menor del umbral  $\epsilon$  (digamos del 5%), entonces la medida es correcta. En caso contrario, se repite la medida con una longitud de tren adecuada a la tasa de tráfico interferente observada en la medida anterior,  $m/N$ . Este proceso se repite hasta que la probabilidad de error sea menor del 5% o hasta que la longitud necesaria sea más grande de un determinado umbral,  $N_{thres}$ .

## 4.2. Validación del modelo

El objetivo de esta sección es darle validez empírica al modelo propuesto en la sección anterior. Para ello, vamos a utilizar simulaciones, resultados experimentales y resultados obtenidos de una campaña de medidas en un entorno real.

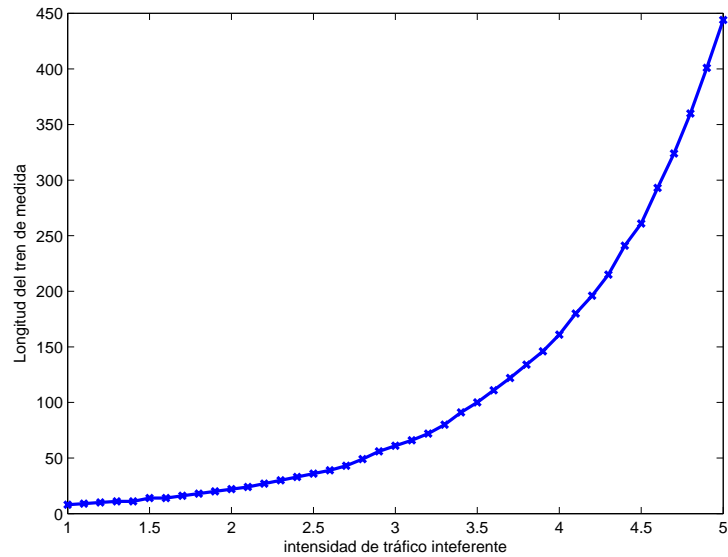


Figura 4.7: Longitud mínima del tren de medida en función de la intensidad de tráfico interferente.

---

**Algoritmo 4.1** Algoritmo adaptativo de la longitud del tren de medida respecto a la intensidad de tráfico interferente

---

```

 $N = N_0$ 
[m,medida]=medir(N);
 $p_{error}$ =calcular_probabilidad(N,m);
mientras  $p_{error} > \epsilon$  hacer
    N=calcular_N_min(m/N);
    si  $N > N_{thres}$  entonces
        print medida demasiado intrusiva
    salir
    fin si
    [m,medida]=medir(N);
     $p_{error}$ =calcular_probabilidad(N,m);
fin mientras
print medida correcta

```

---

### 4.2.1. Simulaciones

En esta primera parte, vamos a comprobar mediante simulaciones, que nuestro modelo nos da una cota superior válida para la probabilidad de rechazo. Esto es, que cuando aceptamos una medida como correcta, la probabilidad de que ésta lo sea es igual o superior al nivel de confianza dado (un 5 % en los casos anteriores).

Para ello, vamos a generar muestras de tiempos entre llegadas de paquetes y, según éstas, contabilizar cuántos paquetes han caído en cada hueco entre paquetes del tren de medidas. Si algún hueco entre paquetes de medida ha quedado vacío, contabilizamos la medida simulada como correcta. En caso contrario, como incorrecta. Repitiendo el experimento podemos estimar la probabilidad de error para una distribución de tiempos entre llegadas del tráfico interferente dada. Para poder recorrer el rango de 0 a 1000 paquetes interferentes, tenemos que ajustar los parámetros de cada distribución. Por ejemplo, si queremos estimar la probabilidad de error cuando hay 300 paquetes interferentes, generamos las muestras con una media de  $1/3$  (notar que las simulaciones se llevan a cabo sólo para el caso de longitud del tren de medida de  $N = 100$  paquetes).

Vamos a usar 2 distribuciones distintas para el tiempo entre llegadas de paquetes. Por un lado, usamos el modelo clásico de tiempo entre llegadas exponencial (que viene producido por un proceso de llegadas de Poisson) y, por otro lado, un modelo de dependencia a largo plazo usando una distribución de cola pesada tipo Pareto [YCCQ02, HP98, LTWW94, Rob01]. En particular, las funciones de distribución usadas son:

- Exponencial

$$F_X(t) = 1 - e^{-\lambda t}$$

$$\text{con } E(X) = \frac{1}{\lambda}, \text{Var}(X) = \frac{1}{\lambda^2}$$

- Pareto

$$F_X(t) = 1 - \left(\frac{t_m}{t}\right)^\alpha$$

$$\text{con } E(X) = \frac{\alpha}{\alpha-1}t_m, \text{Var}(X) = \frac{\alpha}{(\alpha-1)^2(\alpha-2)}$$
 para  $\alpha > 2$  y  $t_m > 0$  el mínimo del soporte de la distribución.

La Fig. 4.8 muestra la probabilidad de error estimada para cada una de las distribuciones y la probabilidad teórica que nos proporciona nuestro modelo. Se puede observar que la probabilidad de error estimada se encuentra, para cada una de las tres distribuciones, por debajo de la probabilidad de descarte que nos da nuestro modelo.

Por lo tanto, con los resultados de las simulaciones, podemos concluir que nuestro modelo nos da una cota superior para la probabilidad de error de la medida. Esto es, si fijamos una significación de, por ejemplo,  $\alpha = 0.05$ , estamos seguros de que la probabilidad de que una medida no descartada por nuestro modelo sea incorrecta es superior al 95 %.

### 4.2.2. Experimentos

En esta sección, queremos evaluar el rendimiento de nuestro modelo en un entorno experimental. Esto es, hasta qué punto el modelo puede descartar de manera satisfactoria medidas incorrectas, dados la longitud del tren y la cantidad de paquetes interferentes dentro del tren.

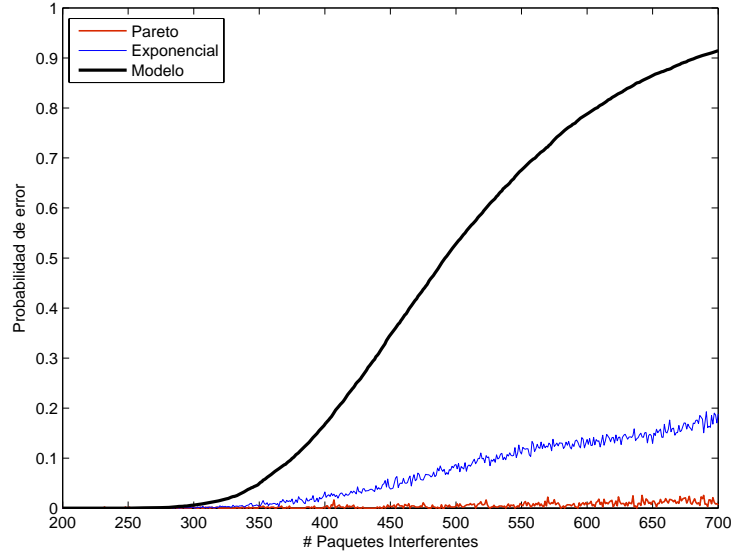


Figura 4.8: Probabilidades de error estimadas mediante simulación.

Para hacer esto, llevamos a cabo más de tres mil medidas y contamos cuántas de ellas son incorrectas, comparándolo con la cantidad teórica que nos dice nuestro modelo. La topología de pruebas usada es la descrita en la Fig. 3.1. Desde la fuente de tráfico cruzado se genera tráfico a distintas tasas y con diferentes tiempo entre llegadas aleatorios. Para la generación de tráfico UDP se usó la herramienta de Linux `hping2`<sup>1</sup>. Para generar retardos entre paquetes de distribuciones aleatorias concretas usamos la herramienta `tc`<sup>2</sup> con su extensión `netem`.

En este experimento, usamos trenes de medida de 100 paquetes. Además, inyectamos tráfico cruzado UDP (con tiempo entre llegadas aleatorio siguiendo una distribución Pareto-Normal [FAM01, ITF04, ZH07]) con una tasa media del triple del ancho de banda del enlace a medir (así, obtenemos en torno a 300 paquetes interferentes por cada tren de medida). Hemos repetido el experimento para tres valores distintos de ancho de banda del enlace, a saber: 900 Kbps, 6 Mbps y 12 Mbps.

La tabla 4.1 muestra los resultados obtenidos.  $\hat{m}$  y  $\hat{N}$  representan el número medio de paquetes interferentes y la longitud del tren, respectivamente. Nótese que los trenes no eran de longitud  $N = 100$  en todos los experimentos debido a la pérdida de paquetes. El error estimado es la proporción de medidas incorrectas, esto es,  $\frac{\#medidas\ incorrectas}{\#total\ de\ medidas}$ . Cabe destacar que la probabilidad de error dada por nuestro modelo es mayor que el ratio de error obtenido en los experimentos (recordar que el modelo es un caso peor).

Por lo tanto, con los resultados experimentales, podemos decir que el modelo nos da una cota superior válida para la probabilidad de error. Así, podemos concluir que nuestro modelo es una buena herramienta para descartar medidas incorrectas “contaminadas” por tráfico cruzado.

<sup>1</sup>[linux.die.net/man/8/hping2](http://linux.die.net/man/8/hping2)

<sup>2</sup>[linux.die.net/man/8/tc](http://linux.die.net/man/8/tc)

Tabla 4.1: Resumen de las pruebas con tráfico cruzado UDP.  $270 \leq m \leq 300, 93 \leq N \leq 100$ .

Ancho de banda	$\bar{m}$	$N$	# Medidas	Error Est.	Error Teo.
900 Kbps	282	96	3601	0.0061	$0.0006 \leq p \leq 0.0187$
6 Mbps	281	96	5021	0.0030	$0.0006 \leq p \leq 0.0187$
12 Mbps	289	96	4332	0.0115	$0.0006 \leq p \leq 0.0187$

### 4.2.3. Campaña de medidas en entorno real

Por último, vamos a ver cómo afectan los paquetes interferentes en las medidas realizadas con usuarios reales. Como se describe el capítulo anterior, se ha llevado a cabo una campaña de más de 75 medidas de usuarios domésticos reales. Todas las medidas se hicieron con una longitud de tren de 100 paquetes.

Hay que recordar que si el número de paquetes interferentes es menor que el número de huecos (en este caso 99) la medida no se ve afectada por el tráfico interferente. Trivialmente, si hay menos huecos que paquetes, entonces alguno de los huecos tiene que estar vacío.

La Fig. 4.9 muestra el ratio de ancho de banda medido entre ancho de banda contratado frente al número de total de paquetes interferentes en la medida. Se puede observar que las medidas con paquetes interferentes suficientes para verse degradadas por el efecto del tráfico interferente (i.e. aquellas con un número de paquetes interferentes  $m > 98$ ), tienen un ratio ancho de banda medido entre ancho de banda contratado cercano a uno. Esto es, ninguna de las medidas se ha visto afectada por el tráfico interferente. Este hecho era esperable, ya que la probabilidad de error dada por nuestro modelo para tal número de paquetes interferentes ( $m < 300$ ) es inferior al 1%.

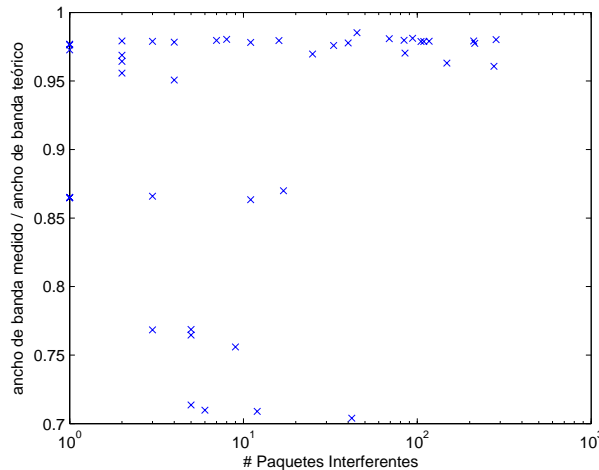


Figura 4.9: Gráfico de dispersión del número de paquetes interferentes frente al porcentaje de ancho de banda efectivo en la campaña de medidas.

Cabe destacar que existen medidas cuyo ancho de banda es inferior al ancho de banda contratado, a pesar de que el tráfico interferente no es lo suficientemente alto. Como dijimos en el capítulo anterior, esto se puede deber a otros factores no contemplados (como la operadora de red) o a que, en realidad, hay una degradación en el servicio.

### 4.3. Conclusiones

Por todo lo visto en este capítulo, podemos concluir que:

- El tráfico interferente auto-inducido influye en el resultado de la medida, siempre que el número de paquetes interferentes sea mayor que la longitud del tren de medida más uno.
- Se propone un modelo probabilístico, basado en la distribución multinomial, que nos permite descartar las medidas potencialmente contaminadas por el tráfico interferente. El descarte de la medida se basa en la longitud del tren y en el número de paquetes interferentes. De este modo, las medidas que no son descartadas, son correctas con una confianza superior al 95 %.
- El modelo propuesto ha sido validado mediante simulaciones (con distintas distribuciones de probabilidad para el tiempo entre llegadas de paquetes). Además, se ha comprobado con experimentos en una red controlada que el modelo nos da una cota de caso peor para el descarte de medidas contaminadas por el tráfico interferente. Por último, también se utilizaron las medidas de la campaña en un entorno real, para la validación del modelo.
- Además, se propone un algoritmo que ajusta la longitud del tren de medidas a la tasa de tráfico interferente.



# Capítulo 5

## Conclusiones

### 5.1. Resumen y Conclusiones

En este trabajo fin de máster se lleva a cabo un análisis de la precisión de las técnicas de medida de ancho de banda ejecutadas en plataformas compartidas (como ordenadores de usuario). La motivación del trabajo es la siguiente:

- Los usuarios cada vez están más preocupados por la calidad de servicio de sus conexiones de banda ancha. *¿Realmente está percibiendo el usuario el ancho de banda que tiene contratado?*
- Las operadoras buscan herramientas para medir el ancho de banda de sus clientes reduciendo al máximo los costes. Para ello, desarrollan medidores que los usuarios descargan y ejecutan en sus propios ordenadores.
- Al ejecutarse en el propio ordenador del usuario, puede haber otras aplicaciones ejecutándose de manera concurrente al medidor. El medidor comparte con estas aplicaciones los recursos del ordenador. Los recursos compartidos son tanto la CPU y memoria, como la propia red.
- Entonces, *¿cómo puede estar segura la operadora de que la degradación del ancho de banda se debe a un problema de la línea realmente? ¿y no a la interferencia que producen las aplicaciones concurrentes en el medidor?*

Por lo tanto, los objetivos que se marcaron para este trabajo fin de máster eran dos:

1. Por un lado, estudiar la influencia de los factores de carga (CPU y memoria) sobre la medida.
2. Por otro lado, estudiar la influencia del tráfico interferente sobre la medida. Además, ser capaces de descartar aquellas medidas que han sido contaminadas por los factores de carga.

En primer lugar, se han descrito los principales métodos de medida de ancho de banda. Por un lado, se han explicado los fundamentos, ventajas e inconvenientes del método de medida basado en transferencia de ficheros. Este método está expuesto en la guía de recomendación de la ETSI [ETS05] y es el usado para regular la verificación de los parámetros de Calidad de Servicio (QoS) por el Ministerio de Industria, Turismo y Comercio del Gobierno de España. Por otro lado,

se han visto los principios de funcionamiento, fortalezas y debilidades de los métodos de medida basados en pares y trenes de paquetes.

En el capítulo 3 se ha analizado la influencia de los factores de carga de CPU y memoria, además del sistema operativo sobre el que se ejecuta la aplicación de medida. El estudio de la influencia de los distintos factores en el resultado de la medida se ha llevado a cabo mediante la metodología estadística de análisis de la varianza (ANOVA). Los resultados obtenidos, tanto en entorno experimental como en entorno real, nos permiten concluir que:

- Para los métodos basados en trenes de paquetes, ninguno de los tres factores influyen de manera significativa en el resultado de la medida.
- Para los métodos basados en transferencias de ficheros, el factor de carga de memoria no es significativo. El factor de carga de CPU no influye de manera significativa cuando el enlace medido es de baja velocidad. Por el contrario, existe una dependencia lineal entre la carga de CPU y el ancho de banda medido, cuando el enlace es de alta velocidad.

En el capítulo 4 se ha analizado la influencia del tráfico interferente sobre la medida, para los métodos basados en trenes de paquetes. Nótese que los métodos basados en transferencias de ficheros quedan descartados en el capítulo 3 como buenas técnicas de medida de ancho de banda. Para estudiar la influencia se propone un modelo probabilístico, basado en la distribución multinomial, que nos permite descartar aquellas medidas que están potencialmente contaminadas por el tráfico interferente. El modelo propuesto ha sido validado mediante simulaciones, experimentos en un entorno controlado y medidas en un entorno real. A la vista de los resultados, podemos concluir que:

- El tráfico interferente auto-inducido no influye en el resultado de la medida, siempre que el número de paquetes interferentes sea menor que la longitud del tren de medida más uno.
- En caso contrario, dadas la longitud del tren de medida y la cantidad total de paquetes interferentes, podemos aceptar únicamente aquellas medidas no influidas por el tráfico interferente con una confianza superior al %95.

## 5.2. Trabajo Futuro

En la campaña con usuarios reales, se obtuvieron medidas degradadas, pero no a causa de los factores contemplados en este trabajo (carga de CPU, memoria, sistema operativo y tráfico interferente). Sería conveniente buscar y analizar las causas de estas degradaciones. Por ejemplo, sería interesante buscar la correlación entre calidad de servicio de la medida y otros factores como la operadora de red que provee el acceso a Internet o la zona geográfica del enlace medido.

Como ya se comentó en el capítulo 4, el modelo multinomial puede ser ampliado a un modelo en el que el número de paquetes interferentes sea también una variable aleatoria. Esto puede venir motivado por la necesidad de medir enlaces de muy alta velocidad ( $\geq 1Gbps$ ). En este contexto, la duración de la medida es muy inferior y, por lo tanto, no es posible saber la cantidad exacta de paquetes interferentes a la medida, sino, únicamente, una estimación de la misma.

Otra tarea para el trabajo futuro es aplicar la misma metodología propuesta (análisis y modelización de la influencia de factores externos) a otros tipos de medidas de parámetros de red y de calidad de servicio.



# Bibliografía

- [ABTV06] E. Altman, D. Barman, B. Tuffin, and M. Vojnovic, *Parallel tcp sockets: Simple model, throughput and validation*, INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings (Barcelona, Spain), apr 2006, pp. 1–12.
- [Ada97] A. Adas, *Traffic models in broadband networks*, Communications Magazine, IEEE **35** (1997), no. 7, 82–89.
- [CFEK06] K. Cho, K. Fukuda, H. Esaki, and A. Kato, *The impact and implications of the growth in residential user-to-user traffic*, SIGCOMM Comput. Commun. Rev. **36** (2006), no. 4, 207–218.
- [Cor07] C. J. Corrado, *The exact joint distribution for the multinomial maximum and minimum and the exact distribution for the multinomial range*, Available at SSRN: <http://ssrn.com/abstract=989082> (2007).
- [CSWS08] L.J. Chen, T. Sun, B.C. Wang, and M.Y. Sanadidi, *Pbprobe: A capacity estimation tool for high speed networks*, Computer Communications **31** (2008), 3883–3893.
- [DRM04] C. Dovrolis, P. Ramanathan, and D. Moore, *Packet-dispersion techniques and a capacity-estimation methodology*, IEEE/ACM Transactions on Networking (TON) **12** (2004), 963–977.
- [ETS05] ETSI, *202 057-4 v.1.1.1: Speech processing, transmission and quality aspects (STQ); user related QoS parameter definitions and measurements; part 4: internet access*.
- [FAM01] K. Fujimoto, S. Ata, and M. Murata, *Statistical analysis of packet delays in the internet and its application to playout control for streaming applications*, IEICE Trans. Commun **E84** (2001), no. 6, 1504–1512.
- [HAN02] T.J. Hacker, B.D. Athey, and B. Noble, *The end-to-end performance effects of parallel tcp sockets on a lossy wide-area network*, Parallel and Distributed Processing Symposium., Proceedings International, IPDPS 2002, Abstracts and CD-ROM (Marriott Marina, Fort Lauderdale, FL, USA), april 2002, pp. 434 – 443.
- [HP98] Z. Harpantidou and M. Paterakis, *Random multiple access of broadcast channels with pareto distributed packet interarrival times*, Personal Communications, IEEE **5** (1998), no. 2, 48–55.

- [ITF04] Y. Ito, S. Tasaka, and Y. Fukuta, *Psychometric analysis of the effect of end-to-end delay on user-level qos in live audio-video transmission*, Communications, 2004 IEEE International Conference on, ICC'04 (Paris, France), vol. 4, jun 2004, pp. 2214–2220.
- [Jac88] V. Jacobson, *Congestion avoidance and control*, SIGCOMM '88: Symposium proceedings on Communications architectures and protocols (Stanford, CA, USA), 1988, pp. 314–329.
- [JD02] M. Jain and C. Dovrolis, *End-to-end available bandwidth: measurement methodology, dynamics, and relation with tcp throughput*, 2002.
- [Kes91] S. Keshav, *A control-theoretic approach to flow control*, SIGCOMM '91: Proceedings of the conference on Communications architecture & protocols (Zurich, Switzerland), 1991, pp. 3–15.
- [LQDB05] D. Lu, Y. Qiao, P.A. Dinda, and F.E. Bustamante, *Modeling and taming parallel tcp on the wide area network*, Parallel and Distributed Processing Symposium, IPDPS 2005. Proceedings. 19th IEEE International (Denver, CO), apr 2005, pp. 68b – 68b.
- [LTWW94] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson, *On the self-similar nature of ethernet traffic (extended version)*, Networking, IEEE/ACM Transactions on **2** (1994), no. 1, 1–15.
- [Pax96] V. Paxson, *Measurements and analysis of end-to-end internet dynamics phd. thesis*, U.C. Berkeley, 1996.
- [Peñ02] D. Peña, *Regresión y diseño de experimentos*, Alianza Editorial, 2002.
- [PFTK00] J. Padhye, V. Firoiu, D.F. Towsley, and J.F. Kurose, *Modeling tcp reno performance: a simple model and its empirical validation*, Networking, IEEE/ACM Transactions on **8** (2000), no. 2, 133–145.
- [Rob01] J.W. Roberts, *Traffic theory and the internet*, Communications Magazine, IEEE **39** (2001), no. 1, 94–99.
- [RRB<sup>+</sup>03] V.J. Ribeiro, R.H. Riedi, R.G. Baraniuk, J. Navratil, and L. Cottrell, *pathchirp: Efficient available bandwidth estimation for network paths*, in Proc. Passive and Active Measurement Workshop (La Jolla, CA, USA), 2003.
- [SK03] J. Strauss and F. Katabi, D. Kaashoek, *A measurement study of available bandwidth estimation tools*, in Proc. Internet Measurement Conference (IMC'03) (Miami Beach, FL, USA ), 2003, pp. 39–44.
- [YCQ02] X. Yu, Y. Chen, and C. Qiao, *A study of traffic statistics of assembled burst traffic in optical burst switched networks*, In Proceedings of Opticomm, 2002, pp. 149–159.
- [ZH07] W. Zhang and J. He, *Modeling end-to-end delay using pareto distribution*, Internet Monitoring and Protection, 2007. ICIMP 2007. Second International Conference on (Silicon Valley, CA, USA), jul 2007.