TÉCNICAS DE ESTIMACIÓN DE BUFFER, CENTRADAS EN LAS REDES DE ACCESO, PARA LA TRANSMISIÓN DE FLUJOS IP EN TIEMPO REAL

Tesis Doctoral

Luis Enrique Sequeira Villarreal

Dirigida por Dr. Julián Fernández Navajas

Programa de Doctorado en Tecnologías de la Información y Comunicaciones en Redes Móviles

Communications Networks and Information Technologies for e-Health and Quality of Experience (CeNITEQ)
Instituto de Investigación en Ingeniería de Aragón (I3A)
UNIVERSIDAD DE ZARAGOZA

Zaragoza, mayo 2015

Técnicas de estimación de buffer, centradas en las redes de acceso, para la transmisión de flujos IP en tiempo real

Autor: Luis Enrique Sequeira Villarreal
Director: Dr. Julián Fernández Navajas

La siguiente página de Internet contiene información actualizada de esta disertación y temas relacionados:
`http://sequeira.telecomsharing.com`
Texto impreso en Zaragoza
Primera edición, mayo 2015

A quienes, aún a la distancia, mantienen
el amor, la sensibilidad
e infunden aliento

Maria Luisa, Guiselle y Mónica

Resumen

Esta tesis doctoral presenta una serie de estudios que analizan la respuesta de la transmisión de flujos IP en tiempo real, en escenarios de redes de acceso, en los cuales dichos flujos convergen en un enlace de salida, compitiendo por alcanzar un determinado nivel de calidad de servicio. La concurrencia de este tipo de flujos puede generar ráfagas de paquetes, que en determinadas circunstancias pueden comprometer la capacidad que tienen los *buffer* para absorber paquetes en períodos de congestión.

En estos estudios se ha identificado que el comportamiento del *buffer* en los dispositivos de acceso, es crítico en el planeamiento de una red. Por este motivo, se propone un conjunto de métodos para descubrir y describir características de redes, por medio de la identificación y el modelado de *buffer* presentes e influyentes en un camino de red. Esto se lleva a cabo mediante la estimación del tamaño de los *buffer* y otros parámetros relacionados, como umbrales presentes o las tasas de salida. La caracterización del comportamiento y los parámetros de los *buffer* son útiles ya que estos aspectos dan más información de un enlace que mediante la utilización de las técnicas tradicionales de estimación de ancho de banda disponible.

Los métodos propuestos se pueden clasificar como: de acceso físico o de acceso remoto. El método exacto es el de acceso físico y se utiliza para comparar la exactitud de los otros métodos. El método de acceso remoto es útil en entornos donde no se puede tener acceso físico a los dispositivos que restringen un enlace (que es el caso más común por ejemplo en Internet). Además, se ha propuesto un método que

permite detectar diversos *buffer* que se encuentran en congestión en un camino de red, lo cual permite obtener más información útil de las mediciones.

Todos los métodos propuestos han sido validados mediante implementaciones reales con dispositivos comerciales bien conocidos y de uso común en diversas redes. Dichos dispositivos se han estudiado en escenarios controlados de laboratorio en redes cableadas e inalámbricas. Los resultados muestran que los métodos planteados permiten descubrir y describir el comportamiento y los parámetros del modelo de *buffer* propuesto con altos niveles de exactitud.

Por último, se presenta un análisis de las características de los *buffer* (especialmente su tamaño y la pérdida de paquetes) en los dispositivos de acceso. En particular se estudia cómo estas características pueden afectar a la calidad de las aplicaciones multimedia cuando éstas generan tráfico a ráfagas y sus posibles efectos en el tráfico de otras aplicaciones que comparten un enlace en común.

Abstract

This dissertation presents a series of studies examining the transmission response of real-time IP flows, in access network scenarios, in which these flows converge on an outgoing link, competing to reach a certain level of quality service. The occurrence of this type of flows can generate bursts of packets, which in certain circumstances, may compromise the ability of the buffer to absorb packets during congestion periods.

These studies have identified that the buffer behaviour on the access devices is critical on network planning. For this reason, a set of methods for discovering and describing characteristics of networks is proposed, by means of identification and modeling of present and influential buffer on a network path. This is done by estimating the buffer size and other related parameters such as its exsistent thresholds or the output rates. The characterization of the buffer behaviour and its parameters are useful since these aspects give more information of a link, that using traditional available bandwidth estimation techniques.

The proposed methods can be classified as: physical access or remote access. The accurate method is the physical access and it is used to compare the accuracy of the other methods. The remote access method is useful in environments where we cannot have physical access to the devices that restrict a link (which is the most common case such as the Internet). Furthermore, it is proposed a method for detecting various congested buffers on a network path, this permits to obtain more useful measurement information.

All the methods have been validated by means of real implementations with well known and commonly used commercial devices on diverse networks. These devices have been studied in controlled laboratory scenarios in wired and wireless networks. The results show that the proposed methods can discover and describe the buffer model behaviour and its parameters with high accuracy levels.

Finally, an analysis of the buffer characteristics (specially its size and packet loss) in the access devices is presented. In particular, we study how these features can affect the quality of multimedia applications when they generate bursty traffic and their possible effects on the traffic of other applications that share the same link.

Agradecimientos

El desarrollo de una tesis doctoral conlleva una serie de dificultades y retos que se deben superar; lo cual ha sido posible gracias a la participación de diversas personas e instituciones que han colaborado y facilitado los medios para que un trabajo de tal envergadura llegue a concluirse. Por ello, es para mí un deber utilizar este espacio para expresarles mis agradecimientos.

A Julián Fernández por su guía en el desarrollo de esta tesis y en mi formación como investigador. A José Ruiz, José Saldaña, Luis Casadesus, Idelkys Quintana, José Ramón Gállego y María Canales por su colaboración, revisiones y comentarios durante el desarrollo de esta tesis.

A la *Fundación Carolina*, la *Universidad de Zaragoza* y el grupo de investigación *Communications Networks and Information Technologies for e-Health and Quality of Experience* (CeNITEQ) por proveer los medios y el financiamiento necesario para la elaboración de este trabajo.

Muchas gracias,

Luis Sequeira

mayo 2015

Contribuciones científicas

Las siguientes publicaciones científicas se derivaron durante el proceso de esta investigación. La metodología y los resultados presentados en esta tesis están principalmente basados en ellas. Todas las publicaciones han pasado por una revisión por pares. Orden por año de publicación.

Publicaciones en revistas internacionales (Indexadas en JCR)

Luis Sequeira, Julián Fernandez-Navajas, Jose Saldana, ***"The Effect of the Buffer Size in QoS for Multimedia and bursty Traffic: When an Upgrade Becomes a Downgrade"***, KSII Transactions on Internet and Information Systems, Vol. 8, Issue 9, September 2014.

Luis Sequeira, Julián Fernandez-Navajas, Jose Saldana, José Ramón Gállego, María Canales, ***"Describing the Access Network by Means of Router Buffer Modeling: a New Methodology"***, The Scientific World Journal, Vol. 2014.

Carlos Fernandez, Jose Saldana, Julián Fernandez-Navajas, Luis Sequeira, Luis Casadesus, ***"Video conferences through the Internet: How to Survive in a Hostile Environment"***, The Scientific World Journal, Vol. 2014.

Publicaciones en congresos internacionales

Luis Sequeira, Julián Fernández-Navajas, Jose Saldana, ***"Characterization of Real Internet Paths by Means of Packet Loss Analysis"***, The Eighth International Conference on Digital Society ICDS 2014, Barcelona, Spain, March 2014, pp 80-85. ISBN 978-1-61208-324-7.

Idelkys Quintana-Ramirez, Jose Saldana, José Ruiz-Mas, Luis Sequeira, Julián Fernández-Navajas, Luis Casadesus, ***"Optimization of P2P-TV Traffic by Means of Header Compression and Multiplexing"***, SoftCOM 2013, Split, Croatia, September 18-20, 2013. ISBN 978-953-290-041-5.

Luis Sequeira, Julián Fernández-Navajas, Jose Saldana, ***"Characterization of the Buffers in Real Internet Paths"***, Proc. International Symposium on Performance

Evaluation of Computer and Telecommunication Systems SPECTS 2013, Toronto, Canada, July 2013, pp 666-671. ISBN 1-56555-352-7.

Luis Sequeira, Julián Fernández-Navajas, Luis Casadesus, Jose Saldana, Idelkys Quintana, José Ruiz-Mas, ***"The Influence of the Buffer in Packet Loss for Competing Multimedia and Bursty Traffic"***, Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS 2013, Toronto, Canada, July 2013, pp 645-652. ISBN 1-56555-352-7.

Luis Casadesus, Julián Fernández-Navajas, Luis Sequeira, Idelkys Quintana, Jose Saldana, José Ruiz-Mas, ***"IPTV Quality assessment system"***, IFIP/ACM LANC 2012 7th Latin America Networking Conference 2012, pp. 52-58. ISBN 978-1-4503-1750-4.

Luis Sequeira, Idelkys Quintana, Jose Saldana, Luis Casadesus, Julián Fernández-Navajas, José Ruiz-Mas, ***"The Utility of Characterizing the Buffer of Network Devices in order to Improve Real-time Interactive Services"***, IFIP/ACM LANC 2012 7th Latin America Networking Conference 2012, pp. 19-27. ISBN 978-1-4503-1750-4.

Jose Saldana, Mirko Suznjevic, Luis Sequeira, Julián Fernández-Navajas, Maja Matijasevic, José Ruiz-Mas, ***"The Effect of TCP Variants on the Coexistence of MMORPG and Best-Effort Traffic"***, IEEE ICCCN 2012, 8th International Workshop on Networking Issues in Multimedia Entertainment (NIME'12), Munich, Germany, July 30, 2012. ISBN 978-1-4673-1543-2.

Jose Saldana, Luis Sequeira, Julián Fernández-Navajas, José Ruiz-Mas, ***"Traffic Optimization for TCP-based Massive Multiplayer Online Games"***, Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS 2012, July 8-11, 2012, Genoa, Italy. ISBN 978-1-4673-2235-5.

Luis Sequeira, Julián Fernández-Navajas, Jose Saldana, Luis Casadesus, José Ruiz-Mas, ***"Empirically Characterizing the Buffer Behaviour of Real Devices"***, Proc.

International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS 2012, July 8-11, 2012, Genoa, Italy. ISBN 978-1-4673-2235-5.

Publicaciones en congresos nacionales

Idelkys Quintana, Jose Saldana, José Ruiz Mas, Luis Sequeira, Julián Fernández Navajas, Luis Casadesus, ***"Optimización del Tráfico P2P-TV mediante el uso de Técnicas de Compresión y Multiplexión"***, Jornadas de Ingeniería Telemática JITEL 2013, pp 345-350, Granada, Spain, October 28-30, 2013. ISBN 978-84-616-5597-7.

Luis Sequeira, Julián Fernández Navajas, Jose Saldana, Luis A. Casadesus Pazos. ***"Caracterización de Tecnologías y Dispositivos de Red: Comportamiento de los Buffer"***, Actas del XXVII Simposium Nacional de la Union Científica Internacional de Radio (URSI 2012). Elche (Spain). Sept. 2012. ISBN 978-84-695-4327-6.

Idelkys Quintana, Jose Saldana, José Ruiz Mas, Julián Fernández Navajas, Luis A. Casadesus Pazos, Luis Sequeira. ***"Influencia del Buffer del Router en la Distribución de Video P2P-TV"***, Actas del XXVII Simposium Nacional de la Union Científica Internacional de Radio (URSI 2012). Elche (Spain). Sept. 2012. ISBN 978-84-695-4327-6.

Elisa Santos, Julián Fernández Navajas, Luis Sequeira, Luis Casadesus. ***"Herramienta para Automatizar la Characterización de Entornos de Red: Análisis y Medidas de Calidad"***, Actas del XXVII Simposium Nacional de la Union Científica Internacional de Radio (URSI 2012). Elche (Spain). Sept. 2012. ISBN 978-84-695-4327-6.

Jose Saldana, Julián Fernández-Navajas, José Ruiz Mas, Luis Sequeira, Luis Casadesus, ***"Comparison of Multiplexing Policies for FPS Games in terms of Subjective Quality"***, Proc. II Workshop on Multimedia Data Coding and Transmission 2012, Jornadas Sarteco. Elche (Spain). Sept. 2012. ISBN 978-84-695-4472-3.

Índice general

Índice de figuras

Índice de tablas

Acrónimos

ABETT *Available Bandwidth Estimations Techniques and Tools*

ADSL *Asymmetric Digital Subscriber Line*

AQM *Active Queue Management*

ARED *Adaptive Random Early Detection*

AVL *Adaptative Video Layering*

BDP *Bandwidth Delay Product*

CAC *Call Admission Control*

CBQ *Class Based Queueing*

CBR *Constant Bit Rate*

CeNITEQ *Communications Networks and Information Technologies for e-Health and Quality of Experience*

CSV *Comma-Separated Values*

DSL *Digital Subscriber Line*

ETG *End-to-end Traffic Generator*

FIFO *First In First Out*

FPS *First Person Shooter*

FTP *File Transfer Protocol*

HTTP *Hypertext Transfer Protocol*

IAX *Inter-Asterisk eXchange protocol*

IETF *Internet Engineering Task Force*

IP *Internet Protocol*

IPTV *Internet Protocol Television*

ISA *Integrated Service Architecture*

ISP *Internet Service Provider*

ITU *International Telecommunication Union*

IXP *Internet eXchange Point*

JPEG *Joint Photographic Experts Group*

LAN *Local Area Network*

MOS *Mean Opinion Score*

MPEG-TS *Moving Picture Experts Group - Transport Stream*

MTU *Maximum Transfer Unit*

NAT *Network Address Translation*

Netem *Network Emulator*

NS-2 *Network Simulator 2*

NTSC *National Television System Committee*

OWD *One-Way Delay*

P2P *Peer-to-Peer*

P2P-TV *Peer-to-Peer Television*

PAL *Phase Alternating Line*

PGM *Probe Gap Model*

PPTD *Packet Pair/Train Dispersion*

PQ *Priority Queueing*

PRM *Probe Rate Model*

PYMES *Pequeñas y Medianas Empresas*

QoE *Quality of Experience*

QoS *Quality of Service*

RED *Random Early Detection*

RTP *Real-time Transport Protocol*

RTT *Round-Trip Time*

SIP *Session Initiation Protocol*

SLA *Service Level Agreement*

SLoPS *Self-Loading Periodic Streams*

SUT *System Under Test*

SVC *Scalable Video Coding*

TC *Traffic Class*

TC *Traffic Control*

TCP *Transmission Control Protocol*

ToIP *Telephony over IP*

TOPP *Trains Of Packet Pairs*

ToS *Type of Service*

TS *Transport Stream*

UDP *User Datagram Protocol*

VoIP *Voice over Internet Protocol*

VPS *Variable Packet Size*

WFQ *Weighted Fair Queuing*

WRED *Weighted Random Early Detection*

El comienzo es la parte más importante
de la obra.

Platón

CAPÍTULO
1

Introducción

El amplio crecimiento del número de usuarios de los nuevos servicios multimedia en Internet (por ejemplo: VoIP (*Voice over Internet Protocol*), *streaming*, videovigilancia, videoconferencia, juegos en línea, etc.) genera una cantidad significativa de nuevo tráfico en la red [HYC04, FS08]. Además, la expectativa de crecimiento para las aplicaciones multimedia, indica que la tendencia de uso se incrementará en los próximos años.

Los usuarios de dichos servicios, cada vez demandan mejores experiencias en el uso de las aplicaciones multimedia. Muchos de estos servicios derivan de aplicaciones en tiempo real desarrolladas sobre redes específicas como redes de conmutación de circuitos y tienen estrictos requerimientos de calidad. Sin embargo, las diversas tecnologías de acceso a Internet son bastante heterogéneas, dificultando en muchos casos, la provisión de servicios que satisfagan las expectativas de los usuarios. Este es el principal motivo por el cual es necesario tener en cuenta la Calidad de Servicio (QoS) que estos servicios ofrecen para sus aplicaciones; especialmente cuando las tecnologías de acceso deben soportar aplicaciones y servicios multimedia en tiempo real.

1.1 Introducción

En este ámbito, la presente tesis se centra en el comportamiento del tráfico que tiene un impacto considerable en el consumo de recursos en las redes de acceso. Cada servicio genera un tráfico con características muy particulares, el cual varía

en función de la naturaleza y tamaño de la información. A continuación se comentan ciertos aspectos, tanto de los servicios como de las redes, que intervienen en la calidad que se puede obtener en determinadas condiciones.

1.1.1 El rol del *buffer* en las redes de acceso

En general, los *buffer* son utilizados como mecanismos de regulación de tráfico en los dispositivos de red. Desde el punto de vista de la planificación de red, el tamaño del *buffer* de los nodos de red es un parámetro importante de diseño, ya que existe una relación entre dicho tamaño y la utilización del enlace. Cuando el *buffer* está lleno y la cantidad de memoria es grande, generará un incremento significativo en la latencia, a este fenómeno se le conoce como *bufferbloat*. Por otro lado, si la cantidad de memoria es muy pequeña, se incrementará la pérdida de paquetes en los nodos durante los períodos de congestión. Como consecuencia, la influencia del *buffer* debería ser considerada cuando se trata de mejorar la utilización del enlace. En los últimos años se han publicado muchos estudios en relación al dimensionado de *buffer* [VST09], pero están especialmente enfocados a los *router* del núcleo de la red y para flujos TCP (*Transmission Control Protocol*), trabajan sobre diferentes estructuras de colas y no contemplan en detalle el comportamiento diferente de los *buffer*.

Por otro lado, el crecimiento en la demanda de datos y las complejas arquitecturas de red que se presentan hoy en día, producen que ciertos puntos en la red, fuera de la red troncal, se conviertan en cuellos de botella. Esto sucede principalmente en las redes de acceso, ya que las capacidades son menores que en las redes de transporte; aunque estos puntos críticos de congestión también pueden presentarse en redes de altas prestaciones. En estos puntos, generalmente en el *router* de acceso la principal causa de pérdida de paquetes es el descarte en las colas. Es por esto, que la implementación del *buffer* en los nodos de red y sus políticas de gestión, son de gran importancia para asegurar la entrega del tráfico de las diferentes aplicaciones y servicios.

En un entorno residencial o de PYMES (Pequeñas y Medianas Empresas), los efectos del comportamiento del tráfico o de los *buffer* de los *router* pueden ser

más pronunciados, debido a las modestas infraestructuras de acceso con las que estos cuentan. Así, las características de diseño del *buffer* del nodo de la red y las políticas de gestión que éste implemente, tienen una gran importancia a la hora de asegurar la entrega correcta del tráfico de diferentes aplicaciones y servicios, por lo que, sería útil incluir los parámetros y el comportamiento del *buffer* en la estimación de la capacidad del enlace.

Tradicionalmente, el ancho de banda disponible, el retardo y *jitter* entre dos dispositivos finales de red, se han utilizado como parámetros que dan una idea general de la QoS que se podría tener en un determinado enlace. Pero hoy en día, se sabe que estos parámetros se pueden ver afectados por el comportamiento de los *buffer* que se encuentran entre los equipos terminales [SFNRM$^+$12c, SFNRM$^+$11]. Dicho comportamiento está determinado principalmente por el tamaño y las políticas de gestión de los *buffer* (es decir, la manera en que se llena y se vacía el *buffer*). De tal manera, que la pérdida de paquetes puede ser causada por los *buffer*, cuyo comportamiento a su vez, también podrían modificar ciertos parámetros de QoS.

Es bien conocido que los *router* del núcleo hacen un uso extensivo de diversas técnicas para la gestión de colas de manera activa o AQM (*Active Queue Management*), las cuales son capaces de mantener la longitud de la cola más pequeña que las tradicionales *drop-tail*, lo cual previene el *bufferbloat* y reduce la latencia. En esta área hay algoritmos muy conocidos como RED (*Random Early Detection*) y algunos derivados ARED (*Adaptive Random Early Detection*) o WRED (*Weighted Random Early Detection*), pero estos algoritmos requieren de un ajuste cuidadoso de sus parámetros con el fin de proveer un buen rendimiento [FJ93]. También, existen algoritmos de planificación de la QoS como WFQ (*Weighted Fair Queuing*), el cual es una técnica de planificación de paquetes de datos, que permite establecer estadísticamente, una serie de prioridades a flujos multiplexados.

Sin embargo, estas soluciones presentadas en la mayor parte de los estudios de investigación se aplican sobre estructuras de colas y no son aplicables a las redes de acceso que normalmente utilizan *router* de gama media y baja, que no suelen implementar técnicas avanzadas de gestión de tráfico, e incluso, en la mayoría de los equipos sólo hay un *buffer* tipo FIFO (*First In First Out*) [SS10].

Por otra parte, es cierto que lo más ampliamente estudiado ha sido el rendimiento de TCP y que una gran cantidad de variantes se han desplegado (por ejemplo, SACK, New Reno, Vegas, etc.) con el fin de mejorar determinadas características adaptándose a las diferentes situaciones de la red. No obstante, muchas aplicaciones multimedia y servicios en tiempo real transportan su información sobre UDP (*User Datagram Protocol*), de tal manera, que las aplicaciones tienen que ser capaces de descubrir el comportamiento de la red para poder optimizar el tráfico.

Muchos servicios y aplicaciones multimedia que se transportan sobre UDP (por ejemplo, videoconferencia, video *streaming*, VoIP, entre otros) utilizan herramientas que permiten la estimación del ancho de banda disponible, conocidas como ABETT (*Available Bandwidth Estimations Techniques and Tools*) [ANFN$^+$11, TCR11, GL10], para mejorar la utilización del enlace y algunos parámetros de QoS. Todas estas herramientas tienen dos cosas en común: se enfocan en las estimaciones de los enlaces que conforman el núcleo de la red y no tienen en cuenta el comportamiento del *buffer* y sus parámetros.

1.1.2 Algunas características a tener en cuenta de los servicios en tiempo real

Es habitual que algunos servicios generen tráfico a ráfagas, como por ejemplo los sistemas de videovigilancia, videoconferencia, *streaming* de video, IPTV y otros servicios interactivos. Este comportamiento se presenta cuando se debe enviar una gran cantidad de información (*frames* de video o imágenes) en un tiempo muy corto. Dichas ráfagas pueden incluir diferentes números de paquetes, y eventualmente, podría congestionar ciertos dispositivos de red cuando la cantidad de paquetes transmitidos, es significativa con respecto al tamaño del *buffer*. En dicha situación, existen aplicaciones que son desarrolladas utilizando herramientas de suavizado del tráfico para proveer cierta QoS y una mejor experiencia al usuario, y al mismo tiempo no ser tan perjudicial para la red, pero incrementando el consumo de recursos del dispositivo debido al procesamiento.

También se debe considerar que algunos de estos servicios y aplicaciones de Internet, generan paquetes cuyos tamaños pueden variar desde unas pocas decenas de $bytes$ (como ocurre en el caso de VoIP) hasta otros que utilizan tamaños mayores (por ejemplo, videoconferencia o videovigilancia). Sin embargo, el tamaño del *buffer* y el ancho de banda disponible, pueden estar dimensionados para que dichos parámetros se mantengan estables, creando problemas de congestión en enlaces de acceso sensibles.

1.2 Metas y contribuciones

La meta y contribución principal de la presente tesis doctoral es la caracterización de los parámetros técnicos y funcionales del *buffer* de los nodos de red en los escenarios que se han mencionado con anterioridad (redes de acceso, redes de PYMES, etc.), que se convierte en un aspecto crítico a la hora de realizar el desarrollo de aplicaciones o la planificación de una red o cuando se quiere proveer ciertos niveles de QoS. Incluso, si se determina el tamaño y el comportamiento del *buffer*, se podrá mejorar la utilización del enlace aplicando técnicas que modifiquen el tamaño de las ráfagas y de los paquetes, como pueden ser la multiplexión de varios paquetes pequeños en uno más grande, por un lado, o la fragmentación y suavizado de tráfico, por otro.

Para estas situaciones se propone desarrollar una herramienta capaz de descubrir algunas características de los *buffer* y su comportamiento, teniendo en cuenta que el tipo de acceso que se tiene a los sistemas a medir usualmente es remoto. Sin embargo, también se propone un método para los casos en que se tiene acceso físico debido a que este método es exacto y permite validar los resultados de los métodos con acceso remoto. Las mediciones que dicha herramienta pueda realizar con acceso remoto tienen más relevancia, ya que es la situación más común para las aplicaciones o servicios a través de Internet.

Esta tesis presenta una metodología para describir el comportamiento de los *buffer* en los nodos de la red y sus parámetros (por ejemplo, tamaño, límites, tasas de entrada y salida y otros parámetros), en el contexto de las redes de acceso. En particular se describen técnicas de estimación para el ancho de banda, tamaño y

comportamiento, ya que estos aspectos dan más información útil, que solamente utilizando técnicas de estimación ABETT.

1.2.1 Objetivos

1. Definir casos de uso de flujos IP (*Internet Protocol*) en tiempo real para diferentes escenarios de red comúnmente utilizados por empresas, modelos de negocios y grupos de usuarios, con la finalidad de identificar sus tecnologías (accesos, *router*, *buffer*, entre otras) y las transmisiones de flujos IP utilizadas.

2. Realizar una clasificación de los diferentes *router* de acceso mediante el estudio del comportamiento ante la presencia de distintos tipos de flujos IP en tiempo real, en particular se profundizará el estudio en los *buffer* contenidos en dichos equipos, con la finalidad de determinar las características técnicas y funcionales de estos dispositivos.

3. Dada una situación en la que se desconocen las características propias de los *router* existentes, se realizarán medidas en distintos escenarios de red, que permitan identificar estas características técnicas y funcionales de los *router* en el acceso, en particular, se analizarán aspectos relacionados con los *buffer* con la finalidad de predecir el comportamiento de éstos frente a flujos IP en tiempo real.

4. Analizar la respuesta en la transmisión de los flujos IP en tiempo real cuando éstos comparten un enlace con servicios que generan tráfico a ráfagas y sus repercusiones en la QoS. Además, valorar el aumento de la capacidad de la red interna como posible solución bajo estas condiciones.

1.3 Estructura de la tesis

Esta tesis se organiza en cuatro partes, la parte I comprende el estado del arte de los temas tratados en el presente trabajo, donde se realiza un estudio relacionado al comportamiento del tráfico de las aplicaciones multimedia. Además, se describen

algunos métodos de medición de la QoS en los que se tienen en cuenta medidas tanto objetivas como subjetivas, así como técnicas de estimación del ancho de banda y medidas de disponibilidad. En esta parte, se analizan los métodos de dimensionado de *buffer* y aspectos relacionados a las disciplinas de colas. Además, se analiza el caso del posible desbordamiento de los *buffer* cuando el enlace tiene un bajo nivel de utilización y la influencia que dichos *buffer* tienen en la QoS de diferentes servicios.

La parte II, inicialmente presenta una metodología general para la detección de *buffer* proponiendo un modelo de *buffer* y un procedimiento para su detección. En dicha parte se propone un método para casos en los que hay acceso físico al sistema a medir, otro método para la detección remota y un método para la estimación de diversos *buffer* concatenados en un camino de red. Al final de esta parte se realiza un análisis de casos en escenarios reales y con dispositivos comerciales ampliamente utilizados en redes cableadas e inalámbricas, en los que se validan los métodos propuestos.

La parte III muestra un análisis del efecto del tráfico a ráfagas en la QoS en función del tamaño del *buffer*. Inicialmente se analiza un escenario en el que convergen flujos de un mismo tipo de tráfico en un enlace con suficiente capacidad y se varía el tamaño de los *buffer*. Luego, se estudia el efecto de incrementar la capacidad de la red interna cuando el enlace de acceso a Internet se mantiene con la misma capacidad. Además, se analiza otro escenario en el que convergen diversos flujos de aplicaciones multimedia y se valora la calidad mediante estimadores objetivos y subjetivos.

Por último, en la parte IV se describen las conclusiones de la presente tesis y se proponen una serie de líneas para futuras investigaciones.

Parte I

Estado del arte

El comportamiento es un espejo en el que cada uno muestra su imagen.

Johann Wolfgang Goethe

CAPÍTULO

2

Servicios multimedia y su comportamiento

En la actualidad los usuarios de Internet demandan una gran variedad de servicios multimedia, muchos de ellos con estrictos requerimientos de tiempo real. En este ámbito, destacan diferentes clases de aplicaciones, por ejemplo, la telefonía por Internet, que debido a la reducción de costes ha alcanzado un gran auge. Por otro lado, destacan los sistemas de televigilancia con un alto grado de aceptación a nivel gubernamental, empresarial e incluso en el ámbito residencial. También, los sistemas de televisión y servicios interactivos forman parte de este tipo de aplicaciones y servicios. En este capítulo se describen algunas de las principales aplicaciones y el comportamiento de éstas, en cuanto al tráfico generado en la red.

2.1 VoIP

El desarrollo de tecnologías de VoIP ha tenido una gran aceptación por parte de empresas que buscan una reducción de costes para sus comunicaciones de voz; principalmente PYMES [SMFN+11b, RMN+10]. VoIP permite la transmisión de voz por medio de una red IP, basándose en la digitalización de las señales de voz por medio de un *codec*. Además, VoIP hace uso de diversos tipos de técnicas para la señalización de la llamada, no habiendo un protocolo único definido en este ámbito. Uno de los protocolos más utilizados para este fin es SIP (*Session Initiation Protocol*) [RSC+02], también, se encuentran implementaciones normalizadas con

H.323 o propietarias que se hacen públicas como IAX (*Inter-Asterisk eXchange protocol*), y por último otras que no se hacen públicas como las de Skype.

Centrándose en SIP, éste es uno de los protocolos con mayor impacto en la implementación de ToIP (*Telephony over IP*) [SAV+09, RMN+10]. Dicho protocolo, se encarga de la señalización extremo a extremo de la comunicación y realiza los procedimientos necesarios para el establecimiento de la llamada, la modificación y la canalización de la comunicación [SMFN+11a]. Por otro lado, para la transmisión de datos en tiempo real, generalmente se hace uso del protocolo RTP (*Real-time Transport Protocol*), dicho protocolo se encarga del control de la transmisión en las sesiones de aplicaciones multimedia y utiliza como protocolo de transporte UDP. La Figura 2.1, muestra la estructura de paquetización de VoIP entre dos equipos terminales.

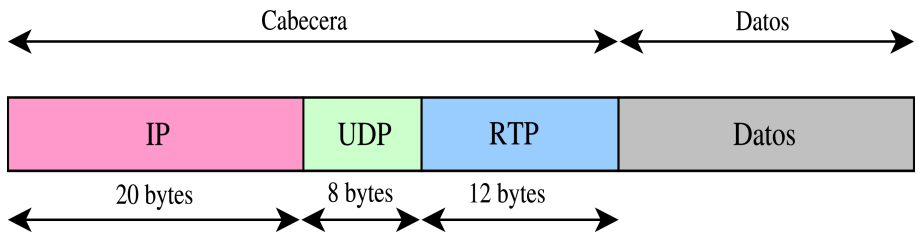

Figura 2.1: Descripción de un paquete VoIP transmitido entre dos estaciones.

El tráfico de este tipo de servicio tiene una distribución uniforme (no se presentan ráfagas de paquetes), cada equipo terminal realiza el envío de paquetes cada cierto tiempo, el cual está determinado por la paquetización y el *codec* utilizado. Por ejemplo, al capturar una traza de este tipo de tráfico, en la cual los equipos terminales se configuraron con el *codec* $G.729$ y con una cantidad de 2 muestras de voz por paquete, se observó que el tiempo medio de envío de paquetes es de $20\,ms$ con una desviación estándar de $0{,}62\,ms$. Cada conexión de este flujo tienen un consumo de ancho de banda a nivel IP $BW = (60 \times 8)/20 \times 10^{-3} = 24\ Kbps$.

En este tipo de servicio la pérdida de paquetes y el retardo son parámetros importantes que determinan la QoS. En términos generales se dice que se ha per-

dido una trama VoIP cuando ésta no llega a tiempo para ser reproducida. Por este motivo, también tiene una gran influencia el *jitter* que tenga la red.

También, hay algunos estudios relacionados al impacto del comportamiento de los usuarios en la estabilidad de la red [BLT06], mostrando que los flujos VoIP no solo consumen menos ancho de banda que los flujos TCP, sino que también, son muy sensibles a la congestión cuando la red está altamente cargada. El estudio demuestra la importancia del comportamiento del usuario (por ejemplo, a la hora de iniciar, cerrar o reiniciar una sesión en determinadas aplicaciones) y el efecto que esto puede tener en la distribución de los recursos de la red y el control de congestión. De esta manera se sugiere que el comportamiento del usuario y el diseño de la aplicación, van a jugar un papel cada vez más importante en el análisis de la infraestructura de red.

2.2 Videovigilancia (Cámaras de video sobre IP)

Las cámaras IP han tenido un impacto importante en los mecanismos de seguridad a nivel empresarial y residencial, este tipo de equipos permite emitir video (utilizando técnicas de compresión de imagen) a través de Internet utilizando TCP/IP. Dentro de sus funciones y que influyen en su comportamiento, se encuentran, la activación mediante movimiento o sensores, control remoto y gestión a través de HTTP (*Hypertext Transfer Protocol*). El formato de imagen más usual es JPEG (*Joint Photographic Experts Group*) con soporte para diferentes niveles de compresión. De manera muy general y para obtener alta calidad, se puede decir que una cámara de este tipo captura imágenes, las convierte a un formato JPEG y las transmite a razón de 25/30 por segundo (PAL, *Phase Alternating Line*/NTSC, *National Television System Committee*). Puede trabajar sin problemas sobre una red con ancho de banda de 10 $Mbps$ o 100 $Mbps$ [AB02].

El comportamiento del flujo de datos difiere en función de la configuración que permita el fabricante para este tipo de dispositivos. Uno de los factores con mayor relevancia es el nivel de compresión que se defina para la imagen, ya que éste define su tamaño (en $bytes$) y afectará de forma directa a las características

del flujo de paquetes en la red, y también influye en la calidad percibida por el usuario.

Un intento de modelar el tráfico de una cámara consiste en considerar que una cámara transmite en cada instante imágenes que tienen un tamaño diferente, estas imágenes son enviadas a la red mediante un determinado número de paquetes en forma de ráfaga, el último paquete de cada ráfaga, tendrá un tamaño menor que los demás, mientras que el resto serán de $1500\ bytes$, esto se aprecia en la Tabla 2.1. Por lo tanto, los parámetros básicos del modelo serán el número de imágenes por segundo y el número de paquetes por imagen.

Resolución	Nivel de compresión	Cantidad de paquetes
$704 \times 576\ pixeles$	$50\ Kbytes$	25
	$16\ Kbytes$	10
$352 \times 288\ pixeles$	$13\ Kbytes$	9
	$4\ Kbytes$	3

Tabla 2.1: Cantidad de paquetes por ráfaga en función de la compresión para una cámara IP AXIS 2120.

En la actualidad la mayoría de estas cámaras tienen sensores para determinar el movimiento y esto tiene un efecto en el tráfico de la red. La cantidad de paquetes que una cámara de este tipo envía a la red también depende del movimiento percibido por la cámara, en estos casos se observa que cuando hay mayor movimiento la cantidad de paquetes, lógicamente aumenta.

2.3 Videoconferencia

Los sistemas de videoconferencia se han extendido ampliamente gracias a las mejoras en las técnicas de compresión de imágenes y al aumento del ancho de banda de las DSL (*Digital Subscriber Line*). En la actualidad, existen múltiples aplicaciones que permiten este tipo de servicio incluso en dispositivos con recursos limitados (por ejemplo, dispositivos móviles). La arquitectura de este tipo de servicios puede dividirse en modelos centralizados (cliente-servidor) o P2P (*Peer-to-Peer*).

Las arquitecturas centralizadas son una solución interesante para dispositivos portátiles como teléfonos inteligentes o tabletas, con una limitada capacidad de procesamiento y energía, ya que permite la reducción de estos aspectos en los clientes, mientras concentra el procesamiento en un nodo central con una alta capacidad. Algunos ejemplos de este tipo de sistemas son: Vidyo y Google plus hangout.

En un sistema P2P cada nodo actúa simultáneamente como cliente y servidor, permitiendo el intercambio directo de información entre los *peer* interconectados. Este tipo de redes, aprovechan el ancho de banda de los usuarios por medio de la conectividad entre ellos mismos, y obtienen mejor rendimiento que con algunos métodos centralizados convencionales, cuando una cantidad de servidores es relativamente pequeña. Además es una técnica interesante para afrontar los problemas de escalabilidad de las redes centralizadas.

Al igual que en la videovigilancia, dentro de los aspectos fundamentales a la hora de modelar el tráfico del servicio se encuentra la codificación y las tecnologías de compresión utilizadas. La codificación tradicional se base en que existen diferentes *codec* que consiguen mayor calidad pero enviando más información. Algunos sistemas de videoconferencia utilizan tecnologías de compresión SVC (*Scalable Video Coding*), que son codificadores de video escalables y diseñados para incluir una mayor flexibilidad a los sistemas multimedia. La gran diferencia con los *codec* tradicionales es que, incluyen AVL (*Adaptative Video Layering*), generando un tráfico de salida en múltiples capas donde cada capa aumenta la calidad del video recibido por el usuario. Este enfoque escalable es adecuado para usuarios con ciertas restricciones de ancho de banda o con accesos con problemas de congestión, ya que en este tipo de entornos, podría recibir una cantidad de capas menor, manteniendo el servicio a pesar de tener una calidad relativamente inferior [TP13]. También, un usuario con mejores prestaciones puede recibir más capas, mejorando la calidad en función de los recursos que dispone. Cada capa emplea predicción con compensación de movimiento e intra-predicción [SR12]. La ventaja del sistema frente a los tradicionales, es que puede adaptarse a las condiciones de la red mediante el filtrado de capas en lugar de tener que cambiar de *codec*. Las

capas pueden ir en paquetes diferenciados, por lo que el filtrado solamente consiste en filtrar determinado tipo de paquetes.

En cualquier caso, la cantidad de paquetes que este tipo de aplicaciones envía a la red depende del tamaño del *frame*, y éste a su vez, depende del modelo de codificación utilizado y el movimiento del video. En [VDRK08], se puede observar una comparación de la variabilidad del tráfico de diversas secuencias de *Silence of the Lambs* y *Star Wars IV* para tres tipos de suavizado de tráfico. Por este motivo, resulta a menudo inviable establecer un modelo de tráfico, siendo preferible la utilización de trazas de tráfico real.

2.3.1 Ejemplos de videoconferencia

Uno de los principales ejemplos de la videoconferencia (que por cierto, sigue una arquitectura P2P) es *Skype*. En [BS06], se analizan las principales funciones de Skype como *login*, NAT (*Network Address Translation*), el establecimiento de la llamada y el *codec*. Algunos estudios se centran en caracterizar ciertas capas de la arquitectura, el comportamiento del protocolo P2P y el tráfico de voz. En [BMM+08], los autores comentan que esta aplicación reacciona diferente ante la pérdida de trayectoria y la congestión de la red, además, que Skype inunda la red con paquetes pequeños de señalización con la finalidad de mantener el servicio de forma eficaz, sin embargo, esta práctica puede resultar costosa desde el punto de vista de algunos dispositivos de red.

Otros estudios [CHHL06, HHCW10], analizan la calidad de las llamadas de voz y la QoE (*Quality of Experience*), proponiendo un modelo para cuantificar el nivel de satisfacción de los usuarios. Algunos [CMP08] han propuesto mecanismos para el control de la congetión en el tráfico de VoIP de Skype.

Con respecto a la capacidad de respuesta de las llamadas de video de Skype, los autores de [DMP11] midieron las variaciones del ancho de banda y llegaron a la conclusión de que el tiempo de respuesta de Skype es grande, cuando el ancho de banda se incrementa. Sin embargo, este estudio sólo tiene en cuenta el comportamiento transitorio de Skype, y no midió sistemáticamente su comportamiento

estacionario cuando éste es alcanzado. Además, los autores de [ZXH+13] caracterizaron los sistemas de control de velocidad y la calidad de las video llamadas de Skype, mostrando que Skype es robusto cuando las pérdidas de paquetes y los retardos de propagación son leves y puede utilizar de manera eficiente el ancho de banda de red disponible.

Otro ejemplo de este tipo de sistemas es Vidyo, el cual es una alternativa propietaria cuya ventaja frente a otras soluciones de videoconferencia es la utilización de AVL. En [FSFN+14], se presenta un estudio que muestra una comparativa de la adaptación del tráfico a la red, para Skype y Vidyo, cuando cambian ciertos parámetros de la red. Las pruebas presentadas se realizaron en un entorno controlado de laboratorio en el cual se varía el ancho de banda, el retardo y la pérdida de paquetes. Los resultados muestran que Vidyo es capaz de detectar rápidamente variaciones en la red y puede adaptar su tráfico según corresponda. Ante los cambios en la red, Vidyo reacciona variando el ancho de banda generado, el tamaño de los paquetes y el tiempo entre paquetes.

2.4 Algunos otros servicios

A continuación, se comentan ciertos servicios multimedia que se encuentran con un buen nivel de aceptación entre los usuarios de Internet, y que, por sus características de interactividad o de tiempo real, poseen importantes requerimientos temporales. Dichos servicios no son objeto de estudio en el presente trabajo, sin embargo, muestran otros escenarios en los cuales pueden ser aplicados los métodos que se proponen en capítulos posteriores.

2.4.1 Video *streaming* y TV *streaming*

El crecimiento a nivel mundial en el acceso a Internet por parte de los usuarios ha generado el desarrollo de diversas aplicaciones y nuevos modelos de negocio dentro de los que se encuentran la radio y la televisión por Internet (por mencionar algunos) [MBM+10] y [MA06]. Este tipo de servicios basa su funcionamiento en la transmisión *streaming*. El *streaming* consiste en la distribución de audio o video

por Internet, esta palabra hace referencia a una transmisión en forma continua, sin interrupciones y sin la necesidad de descargas previas.

El comportamiento de este tipo de tráfico está relacionado con la configuración del proveedor del servicio, la selección de protocolos para el transporte, el control y la compresión. En este ámbito, es común para aplicaciones en tiempo real, el uso de UDP para el transporte y RTP para el control de sesión en tiempo real, además, en cuanto a la compresión muchos servicios utilizan MPEG-TS (*MPEG Transport Stream*). También, existen estudios que han caracterizado y medido el impacto en el tráfico de la red para aplicaciones de video *streaming* [LGL08].

El tráfico *streaming* no tiene un comportamiento uniforme, presenta ráfagas que tienen grandes diferencias en cuanto a los tiempos entre los inicios entre ellas. Las ráfagas producidas durante la transmisión presentan variaciones en la duración de las mismas, sin embargo, si se utiliza TS (*Transport Stream*) los paquetes tienen un tamaño máximo entorno a $1370\,bytes$. Las ráfagas se caracterizan por mantener un período de inactividad antes del inicio de la próxima ráfaga. En general, el tamaño de los paquetes es el mismo en todos los casos, ya que cada *stream* elemental tiene un tamaño fijo de $188\,bytes$ [Rec12] y los paquetes IP deben contener múltiplos de éstos, por lo tanto el mayor tamaño posible si consideramos un MTU (*Maximum Transfer Unit*) de $1500\ bytes$ sería de $1370\,bytes$. Sin embargo, al igual que en los sistemas de videovigilancia, resulta inviable establecer un modelo de tráfico, dadas las características del tráfico, siendo mejor utilizar trazas de tráfico real.

En cuanto a la percepción de los usuarios, mientras los servicios de video *streaming* no tienen unos requerimientos temporales excesivos y pueden sufrir retardos sin un deterioro de la calidad percibida, el TV *streaming* es un servicio más sensible a los retardos debido a sus características, como se ve por ejemplo en los programas en vivo.

2.4.2 **P2P-TV**

IPTV (*Internet Protocol Television*) es un servicio interactivo en tiempo real que tiene un impacto importante en el tráfico de la red, ya que los requisitos de ancho

de banda, derivados del envío de flujos unicast a cada usuario, es bastante caro [LGL08]. Los sistemas P2P-TV (*Peer-to-Peer Television*) son una técnica de difusión de contenidos usando una arquitectura *peer-to-peer*. Estos sistemas son una solución práctica a los problemas de escalabilidad de las redes IPTV, debido a que el consumo de recursos de ancho de banda es menor, y por lo tanto disminuye su coste [KS08].

Sin embargo, los sistemas P2P-TV tienen un comportamiento particular que se debe tener en cuenta ya que generan una gran cantidad de paquetes pequeños. En [SF07], se caracterizó el tráfico de una de las aplicaciones más populares en sistemas P2P-TV, en dicho estudio, los autores afirman que las aplicaciones de este tipo, generan su propio patrón de tráfico, pero tienen en común que éste se compone de una combinación de paquetes pequeños y paquetes de gran tamaño, además, que los de menor tamaño corresponden a paquetes de señalización, mientras que los grandes a paquetes de video.

Una aplicación de este tipo de servicios es SopCast, la cual utiliza UDP como protocolo de transporte. Esta aplicación tiene un *overhead* bastante alto, ya que, cerca del $60\,\%$ de los paquetes que generan son de señalización y solo el $40\,\%$ corresponde a datos de video [FLK$^+$08]. El comportamiento del tráfico de un *peer* se caracterizó en [QRSRM$^+$13], descubriendo que un cliente SopCast envía un paquete UDP de confirmación por cada paquete de video recibido, generando una cantidad considerable de paquetes pequeños en el enlace ascendente del cliente, además, el flujo de paquetes de confirmación influye negativamente en el tráfico de video que dicho *peer* envía hacia otros, ya que ambos flujos competirán en el enlace de subida.

De la misma manera que en los casos anteriores, es recomendable utilizar trazas de tráfico real, en vez de un modelo de tráfico.

La calidad nunca es un accidente; siempre es el resultado de un esfuerzo de la inteligencia.

John Ruskin

CAPÍTULO

3

Calidad de servicio

La redes IP fueron diseñadas en un contexto donde las aplicaciones eran relativamente tolerantes a los retardos, a las posibles pérdidas de paquetes, y a los enlaces de modesta capacidad y con baja demanda de tráfico [Sta04]. Sin embargo, en los últimos años, estas redes se han desplegado ampliamente por todo el mundo, dando paso a una gran cantidad de nuevos tipos de servicios, así como a un aumento importante en la demanda del tráfico por parte de los usuarios.

Muchos de los servicios que se utilizan en la actualidad, son en tiempo real o tienen requerimientos de cierta interactividad. En general estos servicios son muy sensibles a los retardos, ya que en algunos casos, puede que no tenga sentido procesar un paquete de datos si el retardo es muy grande, como sucede en los servicios interactivos (por ejemplo, en video juegos *online*).

La congestión es otro factor que puede comprometer la calidad de un servicio, debido a que los *buffer* de los nodos de la red pueden descartar paquetes que no pueden procesar, y por lo tanto, generar pérdida de paquetes, degradando la calidad de servicios como la voz.

Para hacer frente a esta situación no basta con incrementar la capacidad de una red. En muchos casos se hace necesario la implementación de mecanismos que gestionen el tráfico y controlen la congestión. Para satisfacer estas necesidades, la IETF (*Internet Engineering Task Force*) está desarrollado un conjunto de estándares bajo el marco general de ISA (*Integrated Service Architecture*) [BCS94]. Los servicios integrados o también llamados *IntServ* pretenden gestionar los recursos

necesarios para garantizar la QoS, realizando una reserva extremo a extremo de recursos en los elementos que conforman la red.

La IETF también ha desarrollado otra serie de estándares denominados servicios diferenciados o *DiffServ* que proporcionan un método que busca garantizar la calidad de servicio [BBC+98] de una manera más simple y de bajo coste. El modelo de servicios diferenciados analiza flujos de datos en vez de reservas de recursos. Para realizar un tratamiento diferenciado de la calidad de servicio, los paquetes IP son etiquetados utilizando el campo ToS (*Type of Service*) de la cabecera IPv4 [Pos81a] o TC (*Traffic Class*) en IPv6 [DH98]. Esto significa que la negociación será realizada para todo el tráfico de una red, ya sea un ISP (*Internet Service Provider*) o una empresa. A dichas negociaciones se les llama SLA (*Service Level Agreement*). Este tipo de acuerdos especifican qué clases de tráfico serán provistos y qué garantías se darán a cada flujo de tráfico.

Sin embargo, independientemente de la manera de gestionar la QoS, los parámetros de red suelen ser los mismos y el efecto que éstos provocan varía en función del tipo de servicio ya que algunos toleran en cierta medida la pérdida de paquetes y otros son muy sensibles al retardo y el *jitter*.

3.1 Parámetros objetivos de QoS

La percepción que tienen los usuarios de la QoS de un servicio en tiempo real (por ejemplo, la voz o el video), está relacionada con ciertos parámetros objetivos de la red, como lo son el retardo, el *jitter* y la pérdida de paquetes.

3.1.1 Retardo

El retardo es un parámetro crítico para ciertas aplicaciones y servicios en tiempo real; es también un factor de importancia en el diseño de las redes y en ocasiones es utilizado como una medida del rendimiento de una red. Puede definirse como el tiempo que necesita un dato para viajar a través de la red desde un nodo a otro. El retardo depende de muchos factores; entre los cuales se pueden mencionar [Par05]:

- El sistema de codificación.
- La paquetización.

- La codificación del canal.
- El retardo del paquete en los *buffer*.
- El retardo de propagación.

En redes de paquetes es común utilizar el término OWD (*One-Way Delay*) para referirse al retardo en un sentido de la comunicación (de fuente a destino), además se utiliza RTT (*Round-Trip Time*) para designar al tiempo de ida y vuelta de un paquete.

El retardo puede tener un impacto importante en la calidad percibida por un usuario, ya que en niveles altos puede ocasionar problemas de comunicación para servicios interactivos. Por ejemplo, si un teléfono IP o *gateway* VoIP se conecta a través de una línea de baja velocidad donde el retardo puede ser significativo, se puede percibir eco o incluso problemas de interacción en la comunicación.

3.1.2 *Jitter*

Usualmente, los servicios en tiempo real requieren que los tiempos de llegada entre los paquetes sean constantes, de tal manera que puedan ser reproducidos por la aplicación en el tiempo correspondiente, como sucede con el tráfico de VoIP o videoconferencia. Sin embargo, las redes introducen retardos a los paquetes que difieren en su magnitud. Esta fluctuación de la magnitud del retardo se denomina *jitter* y se define como la variación máxima de retardo que experimentan los paquetes en una sola sesión [Sta04]. Uno de los principales factores que causan el *jitter* es la variación del retardo en los *buffer* de los nodos de la red.

Para mitigar el efecto del *jitter*, algunas aplicaciones introducen un *buffer*, también llamado *de-jitter buffer* [NMNA06], el cual tiene la función de retardar ligeramente los paquetes para poder entregarlos a una velocidad constante al software que genera la señal de salida, por ejemplo audio o video.

El *jitter* es un factor crítico para algunas aplicaciones en tiempo real, ya que cuanto mayor sea la variación del retardo que éstas permitan, más grande será el retardo real en la entrega de los datos, y por lo tanto, mayor será el tamaño del *buffer* de *de-jitter* que se necesitará en la recepción.

3.1.3 Pérdida de paquetes

La pérdida de paquetes es uno de los principales problemas a los que se enfrentan las redes de comunicaciones. Se dice que hay pérdida de paquetes cuando un paquete no ha podido llegar a su destino. Este problema se puede presentar por diversos motivos, algunos de ellos relacionados con la degradación de la señal en el medio de comunicación, problemas en el *hardware* o *driver*. También se da el caso en el que los paquetes son descartados por políticas específicas de gestión de tráfico, con la intención de mantener cierto rendimiento de la red. Un ejemplo de esto es el descarte de paquetes en los *buffer* de los nodos de la red, el cual puede darse en períodos de congestión. La pérdida de paquetes también puede darse por el *buffer* de *de-jitter*, ya que, cuando un paquete llega demasiado tarde para ser reproducido por la aplicación es desechado y para efectos prácticos es como si no hubiera llegado.

El protocolo TCP posee mecanismos para solicitar la retransmisión de un paquete que no ha llegado a su destino [Pos81b], pero ésto conlleva un aumento del retardo en la red y no todos los servicios pueden ser capaces de tolerarlo. Además, es necesario tener en cuenta que una gran parte de los servicios en tiempo real utilizan UDP como protocolo de transporte.

3.2 Medidas objetivas y subjetivas de QoS

En las redes de telecomunicaciones, la calidad es uno de los aspectos de medición que tiene un alto nivel de importancia. Por lo tanto, la capacidad de una monitorización continua de la calidad es una prioridad para mantener la satisfacción de los usuarios de un determinado servicio.

El método más fiable de obtener una medida veraz de la percepción de un usuario con respecto a la calidad de un servicio es desarrollar adecuadamente un test basándose en ciertos criterios de satisfacción y aplicarlo a los usuarios para obtener una medida subjetiva de la calidad percibida por él [MP09]. No obstante, este tipo de medición es lenta y su coste puede llegar a ser muy elevado, haciéndolo inadecuado para la monitorización en tiempo real.

Como alternativa, existen diversos modelos de medidas objetivas de la calidad, que proporcionan una evaluación automática de los sistemas de comunicación sin la necesidad de la intervención de los usuarios. Estas medias objetivas utilizan modelos matemáticos para determinar los niveles de calidad y pueden ser fácilmente computarizados. Por lo general, se basan en la medida de los parámetros de QoS presentados en el apartado anterior (retardo, *jitter* y pérdidas). Debido a la naturaleza heterogénea del tráfico de los distintos servicio que transporta una red, se puede decir, que es necesario un modelo diferente por cada tipo de servicio o aplicación.

A continuación se comentan dos ejemplos de servicios que requieren modelos diferentes, explicando cómo se realizan las medidas de los mismos.

3.2.1 Medidas de calidad en voz

En términos de voz, la calidad se refiere a la claridad con que una persona percibe la voz en una comunicación. La medición de la calidad de voz resulta, a menudo, muy útil en la evaluación de la gestión de los servicios de una red telefónica.

3.2.1.1 El método fiable

El MOS (*Mean Opinion Score*) es una medida subjetiva que se utiliza cuando es necesario valorar los efectos subjetivos en la calidad de la voz, por ejemplo, cuando se incluye algún nuevo equipo de transmisión o se realizan modificaciones en las características de la transmisión de una red telefónica. Los métodos para obtener evaluaciones subjetivas de los sistemas y componentes de transmisión se encuentran estandarizados por la ITU (*International Telecommunication Union*). El MOS está definido en la recomendación P. 800 [Rec96], en la cual cada percepción de los usuarios se clasifica en una escala subjetiva como se muestra en la Figura 3.1.

3.2.1.2 La alternativa

El *E-model* ofrece una alternativa al MOS y consiste en otra recomendación de la ITU-T [Rec14], la cual se usa como instrumento para estimar un posible nivel de calidad en función de distintos parámetros de QoS medidos de forma objetiva

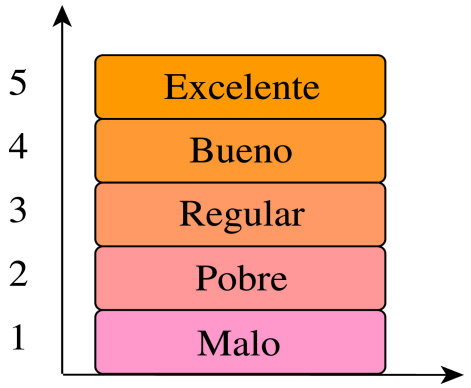

Figura 3.1: Escala de calidad según ITU-T.

(retardo, pérdidas, etc.), proporcionando un medio para estimar el MOS [CR01]. En general, se puede decir que el *E-model* consiste en medir el MOS en un entorno controlando los parámetros de QoS, de tal forma que sepamos qué es lo que diría un usuario medio acerca de la calidad percibida en determinadas condiciones.

3.2.2 Medida de calidad en juegos *online*

En [WKvVA06], los autores propusieron un método, para medir la calidad extremo a extremo, que permite cuantificar la calidad percibida de juegos *online* interactivos. La metodología se expone utilizando un juego llamado *Quake IV*, el cual es del tipo FPS (*First Person Shooter*) y con gran aceptación a nivel mundial.

El método se denomina *G-model* (por su similitud con el *E-model*) y se llevó a cabo mediante una serie de experimentos subjetivos para cuantificar el impacto de los parámetros de la red, en la calidad percibida por los usuarios. Las pruebas se realizaron en una red *Gigabit* con un servidor y 6 clientes con excelentes prestaciones de *hardware* para gráficos con el fin de minimizar los posibles errores. Además, entre los enlaces de los clientes al servidor se introducen retardo, *jitter* y pérdida de paquetes mediante Netem (*Network Emulator*), el cual forma parte del kernel de Linux en las distribuciones actuales. Las pruebas realizadas demuestran que el *G-model* permite predecir un MOS o calificación de calidad basando en valores medidos de retardo y *jitter* con una correlación muy alta ($R = 0{,}98$) con los

datos subjetivos.

3.3 Disponibilidad

En términos generales, la disponibilidad se refiere a cuánto tiempo un dispositivo o sistema está operativo respecto del tiempo total que se hubiese deseado que funcionase. Por otro lado, es necesario relacionar el ancho de banda con la calidad obtenida en un determinado servicio y para ello se utiliza el concepto de disponibilidad. Se puede decir que cuanto mayor sea el ancho de banda más disponibilidad se puede tener.

En las redes de paquetes, el término ancho de banda a menudo se utiliza para caracterizar la cantidad de datos que una red puede transferir por unidad de tiempo. La estimación del ancho de banda es un parámetro de interés cuando se desea optimizar el rendimiento de transporte de extremo a extremo, por ejemplo, en el enrutamiento de una red o la distribución de contenidos en sistemas P2P. Además, esta estimación es importante para el soporte de la ingeniería de tráfico y la planificación de la capacidad de la red.

Existen varias métricas relacionadas con el ancho de banda (capacidad y ancho de banda disponible). En la actualidad, existen herramientas de estimación de ancho de banda que emplean diversas estrategias para medir estos parámetros. A lo largo de este capítulo se presentan algunas de estas técnicas, así como, herramientas de medición de los parámetros mencionados.

3.3.1 Métricas relacionadas con el ancho de banda

3.3.1.1 Capacidad

La capacidad de un enlace se define como la cantidad máxima de información en $bits$ que se puede enviar en un segundo. Es muy común que en un enlace (a nivel de capa 2) sea posible transmitir a una tasa de bit constante, la cual está limitada por la tecnología de la red, que marca el ancho de banda permitido en el medio de propagación y también por las limitaciones del *hardware* en los dispositivos transmisores y receptores. Por ejemplo, esta tasa es de $10\ Mbps$ para Ethernet $10BaseT$, de $1{,}544\ Mbps$ para un $T1$ o de $2{,}048\ Mbps$ para un $E1$. Sin embargo,

para enlaces inalámbricos esto no es cierto, ya que algunas tecnologías de capa 2 no trabajan con tasas de transmisión constante [iee12], como sucede en los sistemas inalámbricos 802.11, los cuales conmutan la tasa en función de las características del medio y la tasa de error.

A nivel IP, la capacidad que se pueden alcanzar es inferior debido al proceso de encapsulado. Para explicar este fenómeno, se supondrá un enlace con una capacidad a nivel de capa 2, C_{L_2}, y un encabezado de capa 2, H_{L_2}, entonces el tiempo t_{L_3} necesario para transmitir un paquete IP de tamaño L_{L_3} es:

$$t_{L_3} = \frac{L_{L_3} + H_{L_3} + H_{L_2}}{C_{L_2}} \tag{3.1}$$

Por lo tanto, la capacidad en la capa 3, C_{L_3}, es:

$$\begin{aligned} C_{L_3} &= \frac{L_{L_3} + H_{L_3}}{t_{L_3}} \\ &= \frac{L_{L_3} + H_{L_3}}{\frac{L_{L_3} + H_{L_3} + H_{L_2}}{C_{L_2}}} \\ &= C_{L_2} \frac{L_{L_3} + H_{L_3}}{L_{L_3} + H_{L_3} + H_{L_2}} \\ &= C_{L_2} \frac{1}{1 + \frac{H_{L_2}}{L_{L_3} + H_{L_3}}} \end{aligned} \tag{3.2}$$

Nótese que la capacidad IP descrita en la ecuación 3.2 depende de la relación entre los tamaños del encabezado (de capa 2) y el paquete IP (con su respectivo encabezado de capa 3). Como el uso del protocolo IP está tan generalizado y con la finalidad de uniformizar el término independientemente de la tecnología, para efectos del presente trabajo, se va a definir la capacidad de extremo a extremo como la máxima tasa de transferencia posible medida a nivel IP. Desde el punto de vista de un camino de red, la capacidad está limitada por el enlace con la mínima capacidad en dicho camino, a este enlace se le conoce como *narrow link*.

Ejemplo (La capacidad)**.**

Si un paquete de 150 $bytes$ (incluyendo la cabecera IP) se transmite entre dos nodos adyacentes en un enlace Ethernet $10BaseT$ con una C_{L_2} de 10 $Mbps$, el cual tiene un encabezado, H_{L_2}, de 38 $bytes$, la capacidad IP, C_{L_3}, es de 7,97 $Mbps$, mientras que si el paquete tiene un tamaño de 1500 $bytes$, la capacidad sería de 9,75 $Mbps$.

3.3.1.2 **Ancho de banda disponible**

El ancho de banda disponible es un término relacionado al ancho de banda que no se utiliza o que queda libre en un enlace durante un determinado período. Como se mencionó en el apartado anterior, la capacidad de un enlace depende de las características de la capa de transmisión de una determinada tecnología y el medio de propagación. Sin embargo, el ancho de banda disponible está ligado a la carga de tráfico y su comportamiento en un determinado enlace, y usualmente es una métrica que varía en función del tiempo [PDMC03] y el comportamiento de las aplicaciones que comparten el enlace.

Esta métrica puede ser influenciada por diversos factores, entre los que se destacan, el tráfico compuesto por la combinación de aplicaciones que utilizan TCP y UDP cuando éstas comparten un mismo enlace. La implementación de los mecanismos de control de congestión que se incluyen en la recomendación del RFC 3782 [MA01], incluida cada variante de TCP (como, Tahoe, Reno, New Reno [FHG04], SACK [MMFR96], etc.) que permite alcanzar un nivel diferente de *throughput*. Por otro lado, aspectos como el tamaño de las tramas, el comportamiento y tamaño de los *buffer* en los extremos de la red, la capacidad y la carga del enlace, así como el número de conexiones que compiten en un mismo enlace también influyen para el cálculo de ancho de banda disponible.

Por este motivo, las aplicaciones con estrictos requerimientos de QoS usualmente deben adaptarse a las variaciones del ancho de banda disponible, y por lo tanto, necesitan medirlo con relativa rapidez ya que puede variar drásticamente a lo largo del día en función de la carga de la red.

3.3.2 Técnicas de estimación del ancho de banda

En [Bel92] y [Jac97] se proponen las primeras herramientas para la estimación VPS (*Variable Packet Size*). Además, se encuentra una gran cantidad de técnicas para la estimación del ancho de banda como: PPTD (*Packet Pair/Train Dispersion*) [Bol93, CC96], SLoPS (*Self-Loading Periodic Streams*) [JD03] y TOPP (*Trains Of Packet Pairs*) [MBG00, MBG]. Pero en la actualidad, estas técnicas son conocidas como ABBET y la mayoría de ellas se clasifican en dos grandes tendencias: PGM (*Probe Gap Model*) y PRM (*Probe Rate Model*).

3.3.2.1 *Probe Gap Model (PGM)*

Este tipo de métodos se caracterizan por ser rápidos y fáciles de implementar. Utilizan el muestreo de paquetes para observar la dispersión de los tiempos entre ellos y así estimar un ancho de banda disponible. Este tipo de técnicas tiene la desventaja que los resultados no son muy precisos en entornos con múltiples saltos [LDS06], además, asumen que sólo existe un cuello de botella de extremo a extremo.

En [SKK03], se presenta una herramienta simple y ligera para la medición del ancho de banda disponible, sin embargo los autores afirman que necesita ser mejorada en relación con la precisión a la hora de realizar las estimaciones y que no se pueden realizar mediciones en el mismo equipo donde se está ejecutando.

3.3.2.2 *Probe Rate Model (PRM)*

Los PRM difieren de los PGM en que son herramientas intrusivas, hacen uso de sondas de prueba que inducen un estado de congestión en la red para poder realizar medidas o estimaciones [RRB+03]. Las estimaciones se realizan enviando tráfico de prueba, si este se envía a una tasa menor que el ancho de banda disponible, la tasa de prueba corresponde a la tasa de salida en el otro extremo de la red, por el contrario si la tasa de prueba es mayor, los paquetes se encolan en los *buffer* intermedios generando retardos y tasas de salida menores. Estos métodos presentan mayor precisión que los PGM pero el tiempo necesario para la estimación y la intrusión son sus principales desventajas.

En [GRT10], se presenta una herramienta denominada Assolo, la cual funciona basada en los principios comentados anteriormente. Su principal ventaja es que se ejecuta en un sistema operativo en tiempo real, lo cual aumenta la estabilidad de las estimaciones y permite probar varias tasas con un solo flujo de paquetes.

3.3.3 **Medidas de disponibilidad**

En el ámbito empresarial, muchas veces es necesario medir o estimar la cantidad de conexiones que se pueden establecer sin pérdidas de datos, en un enlace con cierto ancho de banda disponible y para un servicio determinado. Esto es equivalente a medir o calcular el número de servidores disponibles en un determinado sistema, con lo que puede calcularse la probabilidad de bloqueo del sistema y utilizarse como parámetro de calidad relacionado con la disponibilidad del servicio. Con esta información las empresas pueden estimar el efecto de incorporar un nuevo servicio en la red teniendo en cuenta el número de servidores necesarios para dicho servicio y los efectos producidos por su tráfico correspondiente, o bien, valorar la posibilidad de un incremento del ancho de banda en un enlace, para poder disponer de un número de servidores tal que la disponibilidad del servicio sea aceptable. En estos casos, es necesario estimar cuánto tráfico requiere la disponibilidad de un nuevo servidor. Dicha estimación se realiza en función de la tecnología de la red, ya que depende de aspectos como el protocolo de acceso al medio, el tamaño de la trama y los encabezados utilizados por un determinado servicio.

Se puede poner como ejemplo la telefonía IP que es una de las crecientes soluciones utilizadas para mitigar los costes de la telefonía tradicional y donde la disponibilidad es fundamental. Sin embargo, en muchos casos las soluciones libres o propietarias carecen de mecanismos adecuados para proporcionar la QoS necesaria. Esto sucede porque no siempre se puede disponer de los servidores necesarios dado que a mayor número de servidores, mayor tráfico en la red, factor que puede disminuir la calidad. En [RMN+10], los autores estiman diferentes parámetros de calidad en la implementación de un sistema CAC (*Call Admission Control*) en un entorno virtualizado. Los resultados sugieren que el aumento en el número de llamadas de voz repercute negativamente en la pérdida de paquetes de otras

aplicaciones que comparten la red.

En [SMFN+11a], se presenta un esquema de multiplexión de paquetes VoIP para diferentes políticas de *buffer*. La multiplexión consiste en incluir en un mismo paquete, los paquetes de diferentes flujos de llamadas IP. Los autores afirman que dicho esquema reduce el ancho de banda, lo que permite más servidores, pero introduce nuevos retardos debido a la retención y al procesamiento en ambos extremos de la comunicación lo que disminuye la calidad.

En general, un análisis similar puede realizarse para el resto de servicios analizados en la presente tesis, dado que las probabilidades de bloqueo y por lo tanto la disponibilidad del servicio es un parámetro que puede utilizarse indistintamente en diferentes servicios.

Cada uno tiene el máximo de memoria para lo que le interesa y el mínimo para lo que no le interesa.

Arthur Schopenhauer

CAPÍTULO

4

Buffer

Internet puede definirse como un conjunto descentralizado de redes de comunicación, por esto, la arquitectura es bastante heterogénea. Los diferentes nodos en la red difieren en cuanto a su capacidad de procesamiento, memoria y ancho de banda. Además, las velocidades de entrada y salida de un *router* pueden tener grandes diferencias dependiendo de la tecnología o los accesos utilizados, por ejemplo, las redes Ethernet tienen tasas de 10, 100 y 1000 $Mbps$, una red WiFi, puede tener una velocidad desde 2 hasta 54 $Mbps$ para 802.11g o hasta 300 $Mbps$ para 802.11n. Por otro lado, las tecnologías asimétricas de acceso como cable módem y ADSL (*Asymmetric Digital Subscriber Line*) presentan diferencias en las tasas de subida y bajada, por lo que la relación entre las velocidades de entrada y salida de los *router* también depende de la dirección del flujo de información. Dicha relación de velocidades, también se presenta en la interconexión de grandes ISP o en IXP (*Internet eXchange Point*) con tasas mayores, incluso en redes móviles donde los recursos de este tipo son todavía más escasos cuando la cantidad de usuarios de una celda es muy grande.

Estas diferencias entre las velocidad de entrada y de salida producen cuellos de botella donde puede ocurrir la pérdida de paquetes. Los *router* utilizan *buffer* para reducir las pérdidas de paquetes absorbiéndolos cuando éstos no pueden ser reenviados en ese preciso instante, también, se utilizan como instrumentos que ayudan a mantener los enlaces con un alto grado de utilización en casos de congestión.

4.1 Dimensionado

Desde 1994, en [VS94] se propuso la denominada *rule of thumb* o también llamada BDP (*Bandwidth Delay Product*), la cual fue aceptada por muchos investigadores durante varios años, con el fin de determinar el tamaño de los *buffer* en los nodos de una red. Esta regla, se describe en la ecuación 4.1, la cual define el tamaño del *buffer*, B, como el producto del ancho de banda del enlace, C, por el retardo de ida y vuelta, RTT. La ecuación 4.1 se obtuvo utilizando 8 flujos TCP en un enlace de $40\ Mbps$, que en la actualidad no son datos representativos del tráfico en una red. Por este motivo, hoy en día no resulta un método factible debido al aumento de la cantidad de memoria necesaria con anchos de banda más grandes, por ejemplo, con una capacidad de $40\ Gbps$, y un RTT de $250\ ms$, se obtendría un tamaño del *buffer* de $1{,}25\ Gbytes$ que es un tamaño muy grande, además, no se tuvo en cuenta el caso de flujos con RTT diferentes.

$$B = C \times RTT \tag{4.1}$$

En 2004, esta regla fue puesta en duda, por el llamado *Stanford model* [AKM04] o también denominado *small buffer* [VST09], que reduce el tamaño del *buffer*, dividiéndolo por la raíz cuadrada del número N de flujos TCP, como se muestra en la ecuación 4.2. Esto se debe a que la ausencia de sincronización entre los flujos permite realizar una aproximación. Este nuevo modelo se realiza bajo el supuesto de que la duración de los flujos es larga y el número de flujos es lo suficientemente grande como para considerarlos asíncronos e independientes. Usando esta aproximación, un *router* que gestione $10{,}000$ flujos solamente necesitaría $12{,}5\ Mbytes$ de tamaño de *buffer*.

$$B = \frac{C \times RTT}{\sqrt{N}} \tag{4.2}$$

Debido al modelo propuesto por [AKM04] se generó una serie de investigaciones en este ámbito. En [EGG$^+$05] se propuso la utilización de *buffer* todavía más pequeños, denominados *tiny buffer*, que consideran que un tamaño de entre 20 y 50 paquetes (que equivale a algunas decenas de $Kbytes$) es suficiente como

para alcanzar una utilización del enlace de entre el $80\,\%$ y el $90\,\%$. Ésto, basado en el hecho que los flujos no están sincronizados y el tráfico no presenta ráfagas. Sin embargo, muchos de los flujos IP en tiempo real comúnmente tienen un comportamiento de ráfagas como por ejemplo el *streaming* de video, ésto deja un elemento de incertidumbre en cuanto a los modelos de dimensionado de *buffer*.

Por otro lado, son poco los trabajos que consideran servicios de tiempo real, probablemente por el hecho de que gran parte del tráfico de Internet es TCP, pero hoy en día, los servicios interactivos y aplicaciones multimedia tienen una demanda cada vez más grande [MBM+10]. En [VS08] y [VSR09] se ha considerado un tráfico combinado de TCP y UDP utilizando *buffer* pequeños, descubriendo una región anómala, en la que las pérdidas de paquetes de UDP crecen con el aumento del tamaño del *buffer* mientras que el *throughput* de TCP se mantiene.

En [DD06] se presentó una simulación mediante NS-2 (*Network Simulator 2*), basada en una topología en árbol con 18 nodos y enlaces con una capacidad de $50\ Mbps$, que muestra las variaciones de la pérdida de paquetes en función del tamaño de los *buffer* para diferentes políticas de tráfico con la finalidad de mejorar el *Stanford model*.

En general, un *buffer* es un espacio de memoria en el cual se pueden almacenar un determinado número de paquetes. Entonces, la cantidad de paquetes que se pueden almacenar en un *buffer* depende del tamaño, tanto de la memoria como del paquete. Por este motivo, algunos investigadores y fabricantes definen los *buffer* en términos de $bytes$, mientras que otros, lo hacen en paquetes como se puede observar en [SBGW08].

4.2 Disciplinas de gestión de colas

A medida que un sistema se congestiona, el retardo del servicio en el sistema aumenta, en estos casos, la probabilidad de tener un deterioro en la calidad, puede llegar a ser inaceptable para ciertos servicios con estrictos requerimientos de QoS. Por esta razón, la relación entre la congestión y el retardo es esencial para el diseño de algoritmos de control de congestión eficaces [Ada13]. Las disciplinas de

gestión de colas son herramientas que administran flujos de datos mediante determinadas políticas. Los sistemas operativos actuales, tanto de *host* como de *router*, implementan diversas técnicas para la gestión de los *buffer* de las interfaces de red, estas técnicas pueden ser clasificadas como disciplinas de colas basadas en clases CBQ (*Class Based Queueing*) o colas de prioridad PQ (*Priority Queueing*).

Los *buffer* de tipo *drop-tail* son un ejemplo muy utilizado de colas PQ. Estos elementos encolan un paquete o $byte$ si la cantidad de paquetes o $bytes$ es menor que el tamaño máximo del *buffer*, de lo contrario el paquete o $byte$ es descartado. Un ejemplo de este tipo son los *buffer* FIFO, en los cuales, el orden en que se almacena la información está asociado al orden de llegada de los datos, es decir, el primer dato en llegar, será el primero en ser transmitido en el momento que el *router* tenga la capacidad de hacerlo. El tamaño de este tipo de cola puede ser definido en número de paquetes o en $bytes$.

Un *buffer* PQ puede tener diversas colas donde los paquetes se van almacenando, cada cola tiene una prioridad diferente en función de determinadas políticas, las colas de menor prioridad podrán enviar paquetes solo si, las colas de mayor prioridad están vacías. Un ejemplo de este tipo de *buffer* es FIFO-fast, el cual es un tipo de cola que puede ser configurada en cualquier interfaz de red en los sistemas operativos Linux [Hub12]. Este *buffer* se compone de tres colas FIFO con distintas prioridades, donde la máxima prioridad la tiene la cola 0 y la mínima prioridad la cola 2. Los paquetes encolados en la 0 serán los primeros en ser procesados, los de la 1 serán procesados cuando no haya paquetes en la 0 y los paquetes asignados a la 2 serán procesados cuando no haya paquetes en las colas 0 ni 1. La asignación de un determinado paquete a una cola específica se realiza por medio del campo TOS (*Type of Service*) de la cabecera IP.

Las colas de tipo *drop-tail* tienden a penalizar los tráficos a ráfagas y a causar sincronización global en flujos TCP debido a que los nodos reducirán, al mismo tiempo, la tasa de transmisión cuando se presenta la pérdida de paquetes. A pesar de esto, los *drop-tail* FIFO abarcan una gran parte de las colas utilizadas en Internet debido a que son muy fáciles de implementar [RRQ04], sin embargo, agravan las limitaciones de los esquemas de control de congestión de los terminales, como sucede en TCP.

También, existen disciplinas de colas activas AQM que suelen evitar este tipo de problemas, ya que descartan o marcan los paquetes para ser descartados probabilísticamente, antes de que la cola esté llena. La primera propuesta completa de AQM fue RED [FJ93], el cual fue desarrollado para TCP mediante el reemplazo de la colas *drop-tail*. Los principales objetivos de RED fueron detectar la congestión cuando ésta se está iniciando, lograr una equidad entre los flujos a ráfagas que tienen comportamientos diferentes, controlar la latencia, la sincronización global [MGT00], reducir al mínimo la pérdida de paquetes y proporcionar altos niveles de utilización del enlace.

Existen una gran cantidad de implementaciones diferentes de RED [RRQ03], pero en general, se puede decir que se comporta como un *buffer* FIFO cuando la cantidad de paquetes es menor a cierto umbral, por lo tanto, si el *buffer* está casi vacío o por debajo de dicho umbral, se aceptan todos los paquetes entrantes. Cuando el tamaño de la cola crece por encima del umbral, la probabilidad de que un paquete sea descartado también crece. Cuando la ocupación del *buffer* supera el umbral, los paquetes son descartados o marcados probabilísticamente. Cuando el *buffer* está lleno, la probabilidad ha alcanzado el valor de 1 y todos los paquetes entrantes se eliminan.

El principal problema de esta técnica es la dificultad del ajuste óptimo de sus parámetros para un adecuado funcionamiento [HMTG01b, HMTG01a] ya que es muy sensible a las condiciones de la red. Otro problema, es que utiliza la longitud de la cola como una medida de su rendimiento y como un indicador de congestión, produciendo un deterioro en el *throughput* y el retardo con el aumento del tráfico [RC04], debido a que se obtendrá una alta tasa de pérdidas y un retardo grande cuando hay congestión.

Por otra parte, cuando los *buffer* de los nodos de la red se encuentran llenos, las redes de conmutación de paquetes pueden causar valores muy elevados de latencia y el *jitter*, deteriorando el rendimiento global de la red. A este fenómeno se le conoce como *bufferbloat* y su efecto es más pronunciado cuando los *buffer* son más grandes. Sin embargo, existen diversas técnicas de AQM o variantes de RED, las cuales son capaces de mantener la longitud de la cola más pequeña, incluso algoritmos de planificación de la QoS, que previenen el *bufferbloat* y reducen la

latencia. Sin embargo, estas implementaciones de *buffer* requieren mayor procesamiento y consumo de recursos para identificar tipos de tráfico, realizar mediciones de parámetros como RTT o contar flujos, además, más memoria para gestionar diferentes colas. Por esto, en los *router* de acceso no es común que se implementen este tipo de técnicas.

Se podría seguir citando una gran cantidad de estudios relacionados a técnicas de gestión de tráfico o algoritmos de planificación de colas, sin embargo, estos temas no son objeto de estudio en la presente tesis. Este tipo de técnicas ayudan a mantener cierto nivel de QoS cuando la utilización del enlace es alta y usualmente se implementan en *router* de gama alta debido al consumo de recursos. El presente estudio se enfoca en los problemas de congestión cuando la utilización de los enlaces es media, valorando la influencia de los *buffer* FIFO (sencillos de implementar y presentes en la mayoría de equipos comerciales de acceso de gama media-baja) en la QoS para servicios específicos.

4.3 El rol del *buffer* en la QoS

En la actualidad existe un gran número de usuarios de servicios multimedia que generan una cantidad de tráfico significativa en Internet [HYC04, FS08] y la expectativa de crecimiento en el uso de aplicaciones multimedia indica que esta tendencia se incrementaría en los próximos años. Los servicios multimedia como la videoconferencia, la videovigilancia, la P2P-TV o los juegos *online* alcanzan niveles de tráfico considerables para los ISP [SKR07]. Esta carga de tráfico se puede considerar relativamente alta y se combina con el hecho de que este tipo de tráfico presenta características de comportamiento diferentes en comparación con otros servicios como WWW, *e-mail* o FTP.

Al mismo tiempo el tráfico generado por cada servicio depende de la naturaleza de la información que se transporta y de su tamaño. Como se ha comentado en el capítulo 2, algunas de las aplicaciones multimedia generan tráfico a ráfagas cuando mucha información tiene que ser transmitida en un tiempo muy corto. Estas ráfagas pueden congestionar los dispositivos de red si la cantidad de paquetes es significativa con respecto al tamaño del *buffer* de los dispositivos. Por otro

lado, algunas aplicaciones trabajan para generar tráfico alisado [VG13], con el objetivo de proveer un cierto nivel de QoS y una mejor experiencia al usuario sin ser perjudicial para la red, pero con el coste de un incremento en la capacidad de procesamiento.

El tamaño de los paquetes generados por estas aplicaciones puede variar entre los diferentes servicios de Internet, mientras algunos generan paquetes de tamaños pequeños que alcanzan unas pocas decenas de $bytes$ (por ejemplo VoIP) otros usan paquetes de mayor tamaño (por ejemplo videoconferencia). Sin embargo, el tamaño del *buffer* y el ancho de banda disponible para soportar dichos servicios se mantienen en los mismos valores, por esto algunos enlaces de acceso podrían presentar problemas de congestión.

4.3.1 El desbordamiento del *buffer* con baja utilización del enlace

Los *buffer* pueden entrar en congestión por dos motivos principales: cuando la tasa de entrada es mayor a la tasa de salida, es decir el ancho de banda se agota y la utilización del enlace es alta, o bien, por problemas de dimensionado en la implementación de los dispositivos de red. Como se puede observar en la Figura 4.1, el tiempo que se requiere para congestionar un *buffer* está relacionado a la tasa de llenado, la cual está dada por la relación de las tasas de entrada y salida [SFNSC12]. Se puede definir R_{in} y R_{out} como las tasas de entrada y salida respectivamente, además, se define R_{fill} como la tasa en la cual el *buffer* se llena cuando R_{in} es más grande que R_{out} ($R_{fill} = R_{in} - R_{out}$).

Entonces, cuando un tráfico a ráfagas es generado en la red, la tasa de llenado del *buffer* es muy alta, y en esos momentos, se puede dar una pérdida de paquetes, esto puede producirse incluso cuando la utilización del enlace es media o baja. Este fenómeno se puede presentar porque la longitud de la ráfaga es cercana al tamaño del *buffer*, ya que éste puede entrar en congestionamiento más fácilmente, también puede darse, cuando la longitud de la ráfaga es mayor que el tamaño del *buffer*, en cuyo caso, la pérdida de paquetes será mucho más probable.

Es cierto que muchas aplicaciones trabajan para generar un tráfico suavizado, pero el tráfico global de Internet tiene un comportamiento a ráfagas en todas

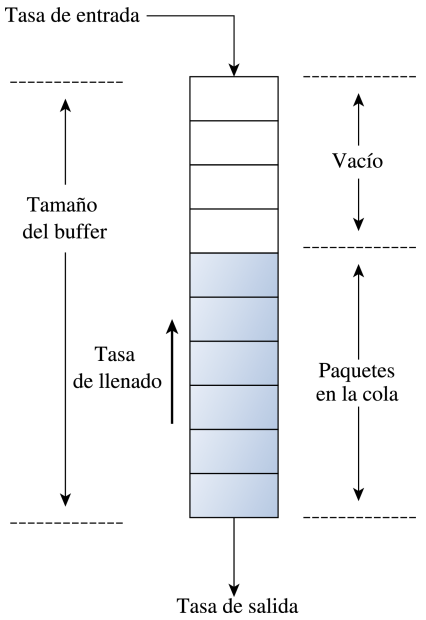

Figura 4.1: Principales características de los *buffer*.

las escalas [JD05]. Para estos casos, sería útil estudiar la generación de tráfico, y analizar cómo afecta el suavizado de tráfico de ciertas aplicaciones.

En algunos escenarios, cuando se presentan problemas de congestión de la red, una práctica común puede ser aumentar el ancho de banda en la red local. Por esta razón, muchas compañías cambian sus dispositivos de red interna (por ejemplo, cambiando de una velocidad menor a una mayor) tratando de resolver los problemas de congestión. Pero, si R_{out} se mantiene en el mismo valor y R_{in} se cambia a una tasa mayor, la tasa de llenado del *buffer* será mayor (R_{fill}) en la nueva red. Por esta razón, en casos de tráficos a ráfagas el *buffer* se congestionará más rápidamente. Así, en ciertos casos, este aumento de la velocidad en la red interna puede producir una respuesta peor de la red, de tal manera, que esta mejora se convierte en un fracaso.

En definitiva, la relación entre las velocidades de la red local y el acceso a Internet, y la relación entre el tamaño del *buffer* y la longitud de la ráfaga son, de hecho, parámetros importantes que no pueden ser descuidados.

Por otro lado, la generación de tráfico a ráfagas por parte de las aplicaciones no es el único motivo por el cual un *buffer* puede producir una pérdida de paquetes. Usualmente, el tráfico de las aplicaciones comparte un enlace con otros tipos de tráfico de servicios con comportamientos diferentes en cuanto a la generación de sus paquetes. Dichos flujos de datos pueden ser generados por el mismo *host*, o bien, por la convergencia de flujos de diversos equipos hacia un enlace en común. Esta combinación de los flujos de tráfico que comparten un mismo enlace, puede producir ráfagas que repercutan más drásticamente en la tasa de llenado de los *buffer* de la red. Además, dicha ráfaga puede llegar a contener una cantidad de paquetes que supere el tamaño de ciertos *buffer*, lo cual repercutiría negativamente en la QoS de ciertas aplicaciones más susceptibles.

4.3.2 Influencia del *buffer* en diferentes servicios

Existen muchas publicaciones científicas relacionadas con la influencia del *buffer* en diferentes servicios y aplicaciones que muestran cómo la QoS es afectada por el comportamiento del *buffer*, el cual está principalmente definido por su tamaño y sus políticas de gestión. En estos casos, el conocer las características técnicas y funcionales de estos dispositivos se convierte en un aspecto fundamental. Este conocimiento puede ser útil para diversas aplicaciones y servicios con la finalidad de decidir y gestionar la forma en que el tráfico es generado. Además, se pueden aplicar ciertas técnicas de gestión de paquetes como por ejemplo, multiplexar un cierto número de paquetes pequeños dentro de uno más grande, o por el contrario, aplicar la fragmentación o incluso suavizar el tráfico, de acuerdo a cada escenario [SFNRM$^+$12a].

La manera en que se estudia la influencia del *buffer*, para el tráfico multimedia, es determinando las características de QoS, basado en parámetros bien conocidos de la red (por ejemplo, *jitter*, pérdida de paquetes, etc.). También se usan evaluaciones de la calidad subjetiva para determinar la percepción de los usuarios para ciertos servicios. El E-Model de la ITU [Rec14, CR01], presenta un procedimiento con el objetivo de calcular el MOS, el cual es útil en el planeamiento de transmisión de red. Otros autores [WKvVA06], han desarrollado un modelo similar para

juegos *online* con base en el retardo y el *jitter* y en general se puede afirmar que existe un modelo para cada tipo de comunicación.

La influencia del *buffer* en VoIP se ha estudiado en [SFNRM$^+$12a], donde se probaron tres políticas de *buffer* diferentes (*buffer* dedicado, grande y limitado en tiempo) con dos técnicas de multiplexión. Donde, cada política del *buffer* del *router* causó un comportamiento diferente en la pérdida de paquetes y también modificó la calidad de la voz, la cual se midió por medio del *R-Factor* [CR01]. En el mismo artículo, se estudió un método de multiplexión para flujos de VoIP, en el cual se obtuvo una reducción del ancho de banda con el aumento del tamaño de los paquetes, lo que influye en la pérdida de paquetes dependiendo de la implementación del *buffer* y su tamaño. En este caso, el tráfico nativo de VoIP mostró un buen comportamiento cuando se usaron *buffer* pequeños y medidos en $bytes$, ya que en estos casos, los paquetes pequeños tienen menos probabilidad de ser descartados que los grandes

En [SFNRM$^+$12b], los autores presentaron un estudio de simulación, de la influencia de un método de multiplexión en los parámetros que definen la calidad subjetiva de los juegos *online* (principalmente retardo, *jitter* y pérdida de paquetes). Los resultados muestran que los *buffer* pequeños, presentan mejores características para mantener el retardo y el *jitter* en valores adecuados, pero a costa de incrementar la pérdida de paquetes. Además, los *buffer* cuyos tamaños se miden en paquetes también incrementan la pérdida de paquetes.

Muchos dispositivos de las redes de acceso están diseñados para la transferencia de datos a granel [SFNRM$^+$12d], como los servicios de correo, web o FTP (*File Transfer Protocol*). Sin embargo, otras aplicaciones (por ejemplo, *streaming* de video P2P, juegos *online*, etc.) generan altas tasas de paquetes pequeños, en estos casos, los *router* podrían experimentar problemas para gestionar todos los paquetes. Por lo tanto, la capacidad de procesamiento se puede convertir en un cuello de botella si no pueden gestionar demasiados paquetes por segundo [FCFW02], ya que en estos casos, lo que sucede es que la tasa de salida disminuye.

La generación de altas tasas de paquetes pequeños también ha sido observada en aplicaciones P2P-TV [VGN$^+$12], dichas aplicaciones además generan tráfico de video. En los casos, en los que un tráfico mixto de paquetes pequeños y grandes

atraviesa un *buffer* medido en paquetes, los paquetes de video pueden verse penalizados por los paquetes pequeños ya que ambos tendrán la misma probabilidad de ser descartados, y como consecuencia, el comportamiento del *peer* no será la esperada en una estructura P2P.

Parte II

Metodología para la detección de los *buffer* y análisis de casos

Algunos de los contenidos de la parte II se han publicado en:

***"Empirically Characterizing the Buffer Behaviour of Real Devices"**, Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS, Julio 2012, ISBN 978-1-4673-2235-5, cuyos autores son: Luis Sequeira, Julián Fernández-Navajas, Jose Saldana, Luis Casadesus y José Ruiz-Mas.*

***"The Utility of Characterizing the Buffer of Network Devices in order to Improve Real-time Interactive Services"**, IFIP/ACM 7th Latin America Networking Conference LANC, Octubre 2012, ISBN 978-1-4503-1750-4, cuyos autores son: Luis Sequeira, Idelkys Quintana, Jose Saldana, Luis Casadesus, Julián Fernández-Navajas y José Ruiz-Mas.*

***"Characterization of the Buffers in Real Internet Paths"**, Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS, Julio 2013, ISBN 1-56555-352-7, cuyos autores son: Luis Sequeira, Julián Fernández-Navajas y Jose Saldana.*

***"Describing the Access Network by Means of Router Buffer Modeling: a New Methodology"**, The Scientific World Journal, Vol. 2014, cuyos autores son: Luis Sequeira, Julián Fernández-Navajas, Jose Saldana, José Ramón Gállego y María Canales.*

***"Characterization of Real Internet Paths by Means of Packet Loss Analysis"**, The Eighth International Conference on Digital Society ICDS, Marzo 2014, ISBN 978-1-61208-324-7, cuyos autores son: Luis Sequeira, Julián Fernández-Navajas y Jose Saldana.*

Metodología histórica, como yo la veo, es un producto del sentido común aplicado a las circunstancias.

Samuel Eliot Morison

CAPÍTULO
5

Metodología para detectar un *buffer*

Como se ha mencionado anteriormente, los ISP tienen que garantizar un alto rendimiento de la red con cierto grado de QoS, especialmente cuando las redes de acceso deben soportar aplicaciones y servicios en tiempo real. Además, el tráfico de una comunicación extremo a extremo atraviesa un número variable de dispositivos de red a lo largo de su recorrido, los cuales pueden modificar la tasa, el retardo y la pérdida de paquetes. Los caminos de red son desconocidos para la mayoría de las aplicaciones y los servicios, que en el mejor de los casos, sólo miden el ancho de banda disponible para limitar el tráfico generado y rara vez para adaptarlo, por ejemplo, por medio del alisado. Así, el conocimiento de ciertas características de los dispositivos intermedios o del comportamiento de los *buffer* es muy útil para mejorar la utilización del enlace mediante la variación de los parámetros de la comunicación.

Por lo tanto, es común que ciertos puntos críticos de la red se conviertan en cuellos de botella debido a que la tasa de entrada es mayor que la tasa de salida de un dispositivo, o bien, los paquetes no pueden ser procesados en ese mismo instante por lo que se descartan o son procesados más lentamente. Por este motivo, la caracterización de los parámetros técnicos y funcionales de los cuellos de botella se convierte en un conocimiento fundamental cuando se trata de proveer ciertos niveles de QoS. Este conocimiento puede ser útil para aplicaciones y servicios con el fin de decidir correctamente la manera en que el tráfico es generado. Con esta información se podrían aplicar algunas técnicas para mejorar la utilización de los enlaces, como por ejemplo, alisar el tráfico de salida a la red, multiplexar cierto

número de paquetes pequeños dentro de un paquete más grande, o bien, fragmentar y todo ello en función de determinados comportamientos de los *buffer* presentes en la comunicación.

El objetivo principal de la metodología presentada en este capítulo es proponer un procedimiento que permita descubrir y describir características de redes, que tiene como fin encontrar el modelo de los *buffer* presentes e influyentes (que permita, por ejemplo, describir su comportamiento, tamaño, límites, tasas de entrada y salida, etc.). Lo cual es útil para afinar correctamente el tráfico de las aplicaciones.

5.1 Metodología general

5.1.1 Modelo de *buffer* propuesto

Tradicionalmente, el ancho de banda disponible, la pérdida de paquetes, el retardo y el *jitter* entre dos dispositivos terminales, se han utilizado como parámetros de QoS que dan una idea de la calidad que se puede esperar de un determinado enlace. Dicho modelo se muestra en la Figura 5.1 y se fundamenta en la detección de puntos críticos que limitan los parámetros. Éstos puntos críticos de la red pueden ser determinados mediante muchas de las técnicas de estimación de ancho de banda disponible que se han expuesto en el capítulo anterior.

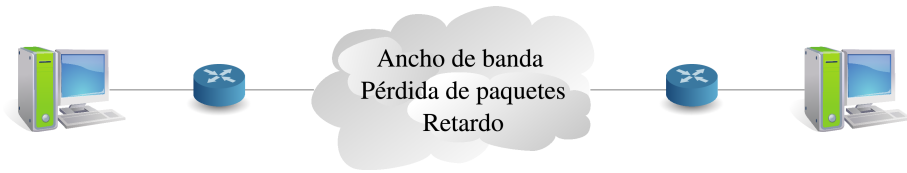

Figura 5.1: Modelo tradicional de un camino de red.

Sin embargo, las técnicas de estimación disponibles no proporcionan toda la información que es útil. Hoy en día, se sabe que dichos parámetros de QoS son comúnmente afectados por el comportamiento de los *buffer* en los cuellos de botella de una red [CR01, WKvVA06]. Por este motivo, se propone una caracterización de los enlaces incluyendo un modelo de *buffer* como se muestra en la Figura 5.2.

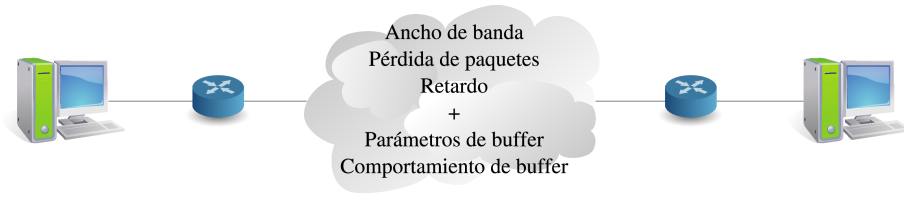

Figura 5.2: Modelo propuesto para un camino de red.

Definición (Modelo propuesto)**.**

Modelo: Describir un camino de red en función de los parámetros y el comportamiento de los *buffer*. Estos factores definen de una manera más completa un determinado enlace y no excluye al modelo tradicional, sino que, lo amplía.

Aunque lo único que se propone es un modelo de *buffer*, se sabe que habitualmente se corresponde con la presencia de un *buffer* real, por ello, en cuanto al modelo del *buffer*, en esta tesis sólo se consideran *buffer* tipo FIFO, ya que debido a su fácil implementación y bajo coste, son muy comunes en dispositivos comerciales de baja gama, los cuales son más susceptibles de convertirse en cuellos de botella en las redes de acceso.

El comportamiento de dicho *buffer* se puede caracterizar de la siguiente manera: una vez que el *buffer* se ha llenado completamente, no se admiten más paquetes o $bytes$, hasta que cierta cantidad de memoria está disponible. Por lo tanto, para un *buffer* de este tipo se pueden definir dos límites uno superior y otro inferior. Cuando el límite superior se alcanza debido a que el *buffer* se encuentra completamente lleno, no se aceptan más paquetes o $bytes$ hasta que, por efecto del vaciado, la ocupación del *buffer* corresponda al límite inferior, como se puede observar en la Figura 5.3. En algunos casos, la diferencia entre el límite superior e inferior puede ser tan pequeño como de un paquete, pero en otros, esta diferencia puede llegar a decenas de paquetes.

Este modo de actuar del *buffer* genera un descarte de paquetes en ráfagas cuando está lleno. La cantidad de paquetes descartados en cada ráfaga depende de la

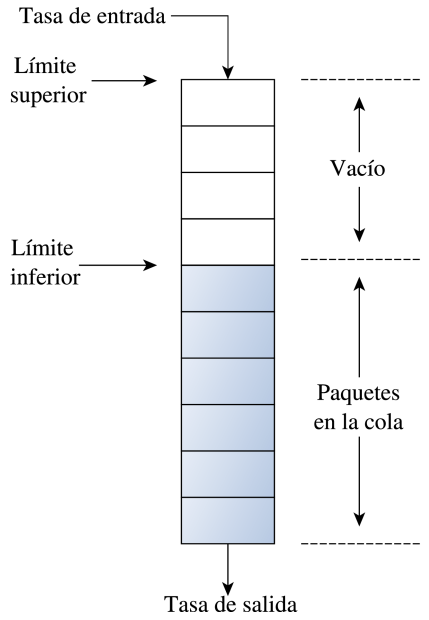

Figura 5.3: Modelo para un *buffer* tipo FIFO.

relación entre las tasas de entrada y salida del *buffer*. Cuando dichas tasas son muy similares, la cantidad de paquetes en la ráfaga disminuye y puede llegar a ser de solo un paquete.

En resumen, estos parámetros que se han presentando son los responsables de modificar el ancho de banda, el retardo, el *jitter* o las pérdidas y definen de una forma más completa un determinado enlace, ampliando así el modelo tradicional.

5.1.2 Procedimiento de detección

El esquema de pruebas para obtener los parámetros del modelo propuesto se observa en la Figura 5.4. En dicha figura existe un SUT (*System Under Test*), el cual puede ser un solo dispositivo o toda una red. Dicho procedimiento o prueba está basada en el envío de una ráfaga de paquetes UDP desde la máquina fuente hasta la destino, y donde todos los paquetes tienen el mismo tamaño. El objetivo principal es producir un desbordamiento del *buffer* que actúa como punto crítico en el SUT. Para facilitar el procedimiento, todos los paquetes que se transmiten

son identificados con un número de secuencia incluido en el *payload*. Finalmente, se analiza el comportamiento del tráfico de prueba para determinar los parámetros del modelo del camino de red.

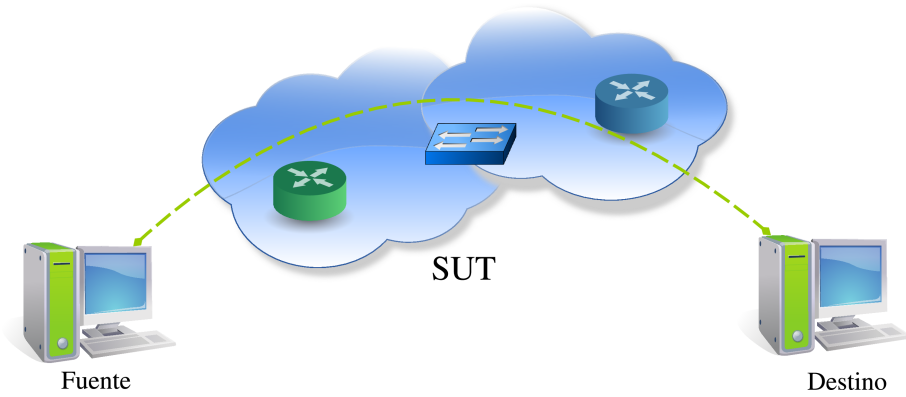

Figura 5.4: Topología general utilizada para las pruebas.

Definición (Procedimiento)**.**

Tráfico: Enviar una ráfaga de paquetes UDP del mismo tamaño desde la fuente hasta el destino.

Captura: Obtener el tráfico en ambos extremos de la red. A la entrada y la salida del SUT.

Análisis: Obtener los parámetros del modelo de *buffer*.

5.1.3 Tipos de métodos

Los métodos propuestos en esta tesis difieren en función del tipo de acceso que se tenga al SUT y se han clasificado de la siguiente manera:

Definición (Tipos de métodos)**.**

Acceso físico: Se puede utilizar cuando se tiene acceso físico a los dispositivos a medir. Es el método más exacto, además se utiliza como punto de comparación con otros resultados. Es utilizado para medir un solo *buffer* a la vez.

Acceso remoto: Este método es útil para entornos en los cuales se necesita realizar medidas en los casos donde no hay acceso físico. Además, puede ser utilizado para estimar varios *buffer* concatenados.

Estos métodos requieren especial atención en cuanto al control de flujo, ya que dicho sistema previene la congestión induciendo al transmisor a estados de pausa [iee11] hasta alcanzar tasas de transmisión uniformes en los tramos de un camino de red. Por este motivo es conveniente desactivar los mecanismos de control de flujo para obtener una mayor precisión, o bien, analizar el tráfico hasta que se produzca la primera pérdida de paquetes. Además, se ha observado que el tamaño del *buffer* puede variar si se cambia el estado del control de flujo, ya que los *buffer* están para soportar situaciones de congestión y con el control de flujo activado éstas pueden no producirse.

5.2 Métodos con acceso físico

En esta sección se analizan los detalles de los métodos para determinar el modelo de un determinado *buffer* cuando se puede garantizar el acceso físico a dichos dispositivos.

5.2.1 Medición de la ocupación y el tamaño del *buffer*

Esta medida se realiza cuando se puede garantizar un acceso físico al SUT de tal forma que se incluyen dos *hub* y un *sniffer* como se muestra en la Figura 5.5. Esto permitirá la captura del tráfico de prueba en ambos extremos de la red, una a la entrada del SUT y otra a la salida. De esta manera, se puede determinar el instante

en que cada paquete entra y sale del SUT. Se hará la aproximación de que este tiempo es provocado por la existencia de un *buffer* en el SUT.

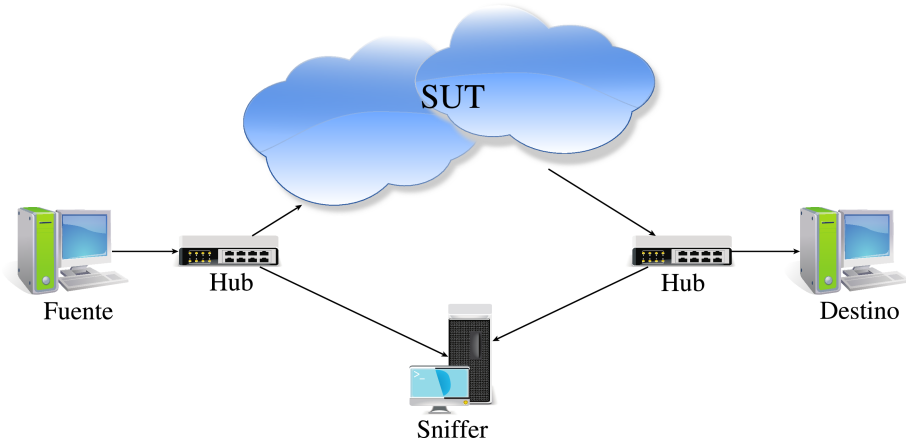

Figura 5.5: Topología utilizada para determinar la ocupación del *buffer* con acceso físico.

Para obtener el modelo de un *buffer* (comportamiento, tamaño, límites y tasas de entrada y salida) se propone utilizar dos metodologías diferentes. Ambos métodos están basados en la transmisión de una ráfaga de paquetes que congestione el sistema en estudio y en las modificaciones en la estructura temporal que se puedan presentar por la existencia del *buffer*, como se muestra en la Figura 5.6.

5.2.1.1 **Método 1**

Este método consiste en contar el número de paquetes en la cola en el momento que un paquete sale del *buffer*, sabiendo el momento en que ha entrado. Con la información obtenida de las capturas de tráfico, se puede determinar la cantidad de paquetes que han entrado al *buffer* entre los tiempos en que un paquete entró y salió del *buffer*, para esto se procede de la siguiente manera: para cada uno de los paquetes contenidos en la captura de salida, se debe buscar en la captura de entrada, obteniendo el tiempo en que ha entrado al *buffer*. En la captura de salida se cuenta el número de paquetes entre los tiempos de llegada y salida del paquete

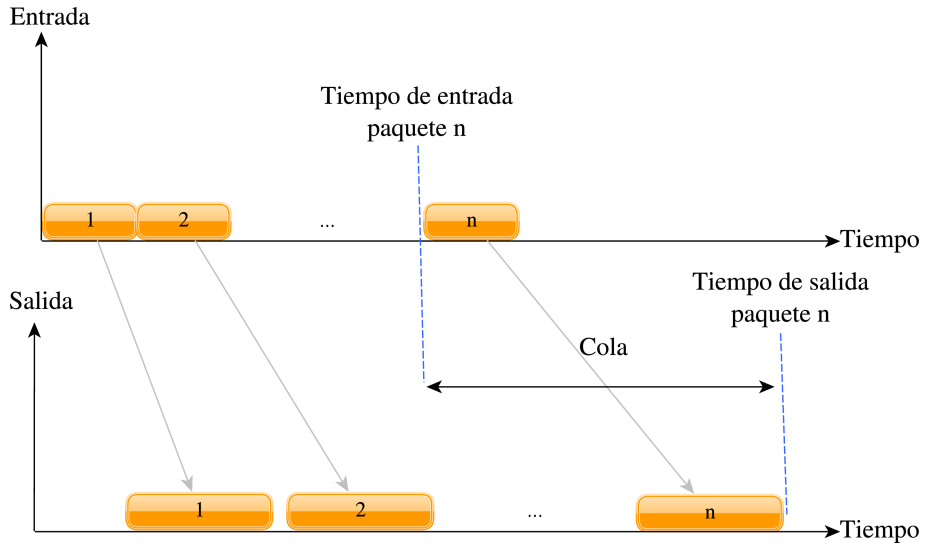

Figura 5.6: Relación temporal de las capturas de entrada y salida del tráfico de prueba y el tamaño del *buffer*.

en el *buffer*, que resultan ser los paquetes que hay en el *buffer* cuando entró el paquete escogido (ver Figura 5.6) y por lo tanto nos indica el tamaño del *buffer*.

Una desventaja de este método es que presenta ciertos inconvenientes en cuanto al consumo de recursos computacionales necesarios para procesar la información, los cuales pueden ser un problema serio si las capturas son grandes o se requieren los resultados en el menor tiempo posible.

5.2.1.2 **Método 2**

En este caso, se calcula la ocupación del *buffer* mediante la tasa de salida, el tamaño de los paquetes y los tiempos de entrada y salida de cada paquete. La intención de este método es determinar la ocupación o el tamaño del *buffer* por sus efectos en la relación temporal de las trazas de entrada y salida a dicho dispositivo.

Este método se basa en la premisa de que la tasa de salida puede ser obtenida mediante la captura del tráfico de prueba en el equipo destino y de que todos los paquetes tienen el mismo tamaño. Se debe tener en cuenta que dicha tasa depende del tipo de tecnología de la red que se utilice (LAN, WiFi, etc.). Para este caso se sigue un procedimiento similar al método anterior: se obtienen los tiempos de

entrada y salida para cada paquete (ver Figura 5.6) y se determina el tamaño del *buffer* como la relación del total de $bits$ transmitidos con respecto al tamaño del paquete, mediante la ecuación 5.1.

$$Buffer_{size} = \frac{R_{output} \times (t_{output} - t_{input})}{packet_{size}} \tag{5.1}$$

Es necesario aclarar que la tasa de salida puede variar durante una misma prueba, este fenómeno se percibe principalmente en entornos como WiFi, siendo menos pronunciado en redes como Ethernet. Por este motivo, es conveniente realizar los cálculos de la tasa de salida en diversas ocasiones a lo largo de la captura realizada y estimar el correspondiente tamaño del *buffer* para cada tasa, ya que en muchos casos la tasa media de salida podría ser un valor que no represente el valor real de toda una transmisión.

5.2.1.3 Medición del tamaño del *buffer*

Los dos métodos anteriores permiten obtener la ocupación del *buffer* basado en la cantidad de paquetes en la cola entre los tiempos de llegada y salida al *buffer* de cada paquete. El tamaño máximo del *buffer* puede ser determinado con el mismo procedimiento, ya que dicho parámetro corresponde a la ocupación máxima del *buffer*.

En la Figura 5.7 se muestra las trazas de entrada y salida de un *buffer* con diferentes velocidades de entrada y salida. Donde n es la cantidad de paquetes de prueba enviados desde la fuente hasta el destino, por otro lado, i es el último paquete recibido antes de la primera pérdida.

De las trazas se puede obtener el tiempo en que el el paquete i se encuentra en la entrada del *buffer* y el tiempo cuando éste sale. Esta información es equivalente a saber el tiempo en que el paquete i está en la última posición del *buffer* y en la primera, y por lo tanto, los métodos propuestos anteriormente darían como resultado el tamaño del *buffer*. El resultado de esta medición es una cantidad determinada de $bytes$ o de paquetes.

A continuación, es necesario repetir la prueba con un tamaño distinto de paquete, de tal forma que, si las pruebas con diferentes tamaños de paquetes, generan

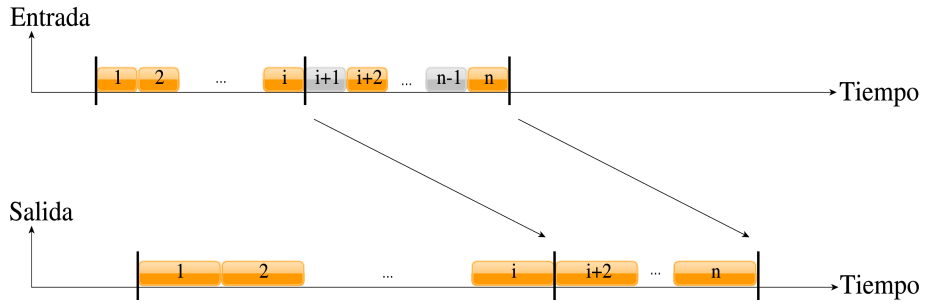

Figura 5.7: Trazas de entrada y salida de un *buffer* en congestión.

el mismo número de paquetes como valor de la ocupación o del tamaño del *buffer*, entonces la ocupación o el tamaño del *buffer* es en número de paquetes. Si las pruebas con diferentes tamaños de paquetes, genera valores diferentes de paquetes de la ocupación o del tamaño del *buffer*, entonces la ocupación o el tamaño del *buffer* es en número de $bytes$.

5.3 Método con acceso remoto

Existen diversos motivos por los cuales se hace necesario realizar una estimación de forma remota, principalmente en situaciones en las que no se tiene acceso al dispositivo a medir, que es el caso habitual cuando se quiere realizar estimaciones por un camino de red a través de Internet.

Por otro lado, las aplicaciones que quieren ajustar su tráfico en función de las variaciones del ancho de banda disponible, van a realizar sus mediciones de manera remota y desatendida. Por estos motivos, en esta sección se describen los detalles necesarios para poder obtener el modelo del *buffer* de forma remota.

Desde el punto de vista de la estimación, la diferencia principal es que en una estimación remota solamente se puede tener acceso a la captura en el *host* de destino, de la cual se debe obtener toda la información posible. El método propuesto para la estimación remota está basado en el modelo y el procedimiento descrito en la metodología general. Sin embargo, la forma de calcular la ocupación o el tamaño del *buffer* difiere a la utilizada cuando se garantiza el acceso físico. Se de-

be aclarar que los métodos con acceso físico proveen resultados más precisos, por esto se utilizan para comparar los resultados obtenidos con el método con acceso remoto.

5.3.1 Estimación de la ocupación y el tamaño del *buffer*

La premisa fundamental de este método es que se puede obtener la tasa de entrada al *buffer* con la traza capturada en el *host* de destino basándose en el porcentaje de pérdidas obtenidas. La tasa de salida del *buffer* se obtiene de manera inmediata de la traza de salida. En la Figura 5.8 se representa una relación temporal de cada paquete, para las trazas de entrada y salida del *buffer*. En dicha figura, T_{fill} es el tiempo necesario para que el *buffer* se llene completamente antes que ocurra la primera pérdida de paquetes, mientras que T_{buffer} es la suma de dos tiempos: el tiempo en el cual el *buffer* se llena (T_{fill}), más el tiempo que necesita el último paquete (aceptado en el *buffer*) para a travesar el *buffer* (T_{empty}), como se observa en la ecuación 5.2.

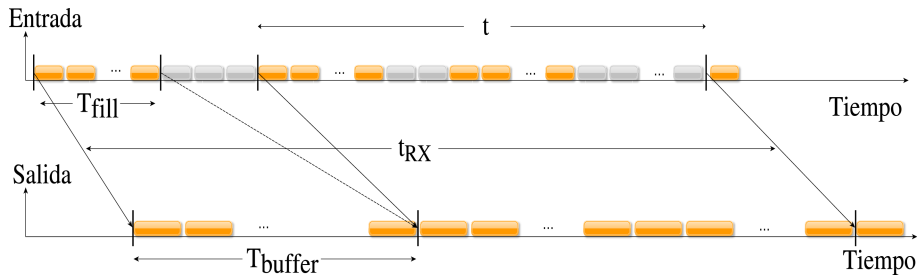

Figura 5.8: Relación temporal de las trazas de entrada y salida a un *buffer*.

$$T_{buffer} = T_{fill} + T_{empty} \tag{5.2}$$

Además, sean las tasas de entrada y salida del *buffer* R_{in} y R_{out} respectivamente y R_{fill} la tasa a la cual el *buffer* se llena cuando R_{in} es mayor que R_{out} ($R_{fill} = R_{in} - R_{out}$). Por último, el tamaño del *buffer* es L_{buffer} medido en $bytes$.

Entonces, se sabe que un paquete tarda $L_{buffer}/tasa$ para atravesar el *buffer* cuando está lleno y por lo tanto se puede obtener una expresión para T_{buffer} como se ve en la ecuación 5.3 y despejar para obtener el tamaño del *buffer*, como se puede ver en la ecuación 5.4.

$$T_{buffer} = \frac{L_{buffer}}{R_{fill}} + \frac{L_{buffer}}{R_{out}} \tag{5.3}$$

$$L_{buffer} = \frac{T_{buffer}}{\frac{1}{R_{in}-R_{out}} + \frac{1}{R_{out}}} \tag{5.4}$$

La tasa de salida se puede obtener con facilidad porque la captura remota incluye los n paquetes recibidos en los t_{RX} segundos que duró la transmisión y el tamaño de los paquetes es conocido. Para determinar la tasa de entrada, se conocen los n paquetes recibidos ya que se encuentran en la captura, y debido al número de secuencia que contiene cada paquete, se puede determinar los m paquetes perdidos por la congestión producida en el *buffer*. De esta manera, con la información de la traza en el *host* de destino se pueden estimar las tasas de salida y entrada, mediante las expresiones 5.5 y 5.6 respectivamente.

$$R_{out} = \frac{n}{t_{RX}} \times tamaño_{paquete} \tag{5.5}$$

$$R_{in} = \frac{n+m}{t_{RX}} \times tamaño_{paquete} \tag{5.6}$$

El principal problema que se presenta cuando se determina el modelo del *buffer* mediante acceso remoto es determinar de manera correcta la tasa de entrada, esto se debe a que solamente se tiene la captura del tráfico en el receptor y por lo tanto dicha tasa debe ser estimada. Por otro lado, se debe de estar seguro que las tasas de entrada y salida son estables durante el período en el cual se realizan las mediciones ya que la precisión depende de eso. Con respecto a esto, se han realizado diversas pruebas y se puede concluir que el período menos indicado para determinar las tasas, se encuentra entre el tiempo que se recibe el primer paquete y el momento en el cual el *buffer* descarta el primer paquete. Esto se debe a que en ese período no hay pérdida de paquetes porque el *buffer* se encuentra absorbiendo

la congestión. Por lo tanto, las mediciones o estimaciones de las tasas de entrada y salida del *buffer* se deben realizar en el intervalo t que se muestra en la Figura 5.8.

Además, se debe tener en cuenta, como se ha mencionado anteriormente (en la sección 5.2.1.2 sobre el método 2 para acceso físico), que la tasa de salida puede variar durante una misma prueba y que este efecto es dependiente de la tecnología. Por lo tanto, es recomendable que las estimaciones de las tasas y el tamaño del *buffer* se realicen en diversas ocasiones durante una misma prueba en períodos que puedan definirse como estables en cuanto a la tasa de salida.

La ciencia avanza a pasos, no a saltos.

Thomas Babington Macaulay

CAPÍTULO

6

Metodología para detectar *buffer* concatenados

En el capítulo 5 se ha visto cómo un camino de red puede ser modelado por medio de un solo *buffer* y sus parámetros relacionados. Este modelo es bastante acertado debido a que a menudo, solo un *buffer* está implicado en el enlace que se convierte en punto crítico, o que durante los períodos de congestión es el único que tiene un efecto continuado en el deterioro de la QoS.

Sin embargo, existen ocasiones en las que en los caminos de red se puede encontrar más de un dispositivo susceptible de convertirse en punto crítico. Por este motivo, en el presente capítulo se describe un nuevo método que permite determinar más de un *buffer* en congestión con sus respectivos parámetros; lo cual permite ampliar el modelo propuesto y obtener más información útil como resultado de las mediciones.

Para ello, se tendrá en cuenta varios factores como el comportamiento del tráfico de las aplicaciones, el tamaño de las tramas, la relación de las tasas de entrada y salida o el comportamiento de los *buffer*.

6.1 Análisis previo y ejemplo

Con el objetivo de explicar la metodología de trabajo se analizará el caso de dos *buffer* concatenados que tienen comportamientos diferentes, en cuanto a la diferencia entre los límites superior e inferior de cada *buffer*. Se debe recordar que hay casos donde la diferencia entre los límites es grande (algunas decenas de paquetes)

y el *buffer* tarda un tiempo en aceptar un nuevo paquete y por lo tanto los paquetes se descartan en ráfagas. Sin embargo, en otros casos la diferencia es de solamente un paquete y el *buffer* descarta paquetes de forma aislada.

Para este análisis se utilizará un escenario como el que se muestra en la Figura 6.1, donde el *buffer* 1 tiene una gran diferencia entre sus límites superior e inferior, mientras que para el *buffer* 2 dicha diferencia es de solamente un paquete. Esta selección se realiza de esta manera porque el comportamiento de la pérdida de paquetes depende de dichos parámetros y esta información es necesaria para poder obtener un modelo correcto del *buffer*. Se debe tener en cuenta que ambos *buffer* se llenarán únicamente cuando $R_1 > R_2 > R_3$.

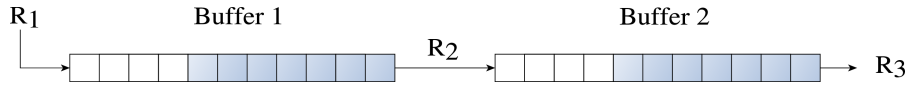

Figura 6.1: Dos *buffer* concatenados.

La Figura 6.2 representa la relación temporal del flujo de paquetes a través de los dos *buffer* mencionados, teniendo en cuenta la diferencia de tasas a las entradas y salidas de cada *buffer* y el posible comportamiento de la pérdida de paquetes en cada uno de ellos. En el primer eje se muestran los paquetes que llegan a la entrada del *buffer* 1 con una tasa R_1, además se han marcado los posible paquetes que dicho *buffer* descartará. En el segundo eje se representan los paquetes que salen del *buffer* 1 y entran al *buffer* 2 con una tasa R_2, de la misma manera se marcan los paquetes a descartar. El tercer y último eje corresponde a los paquetes que logran atravesar el *buffer* 2 y salen de éste a una tasa R_3. Como se puede ver en dicha figura, el *buffer* 1 descarta paquetes a ráfagas mientras que en el *buffer* 2, cuando un paquete sale, otro puede entrar al *buffer*; de este modo sólo se descarta un paquete de forma aislada.

Utilizando el mismo procedimiento para la detección de *buffer* con acceso remoto descrito en el capítulo 5, pero modificando las ecuaciones 5.4, 5.5 y 5.6 en función de cada *buffer*, sus tasas de entrada y salida, así como la pérdida de paque-

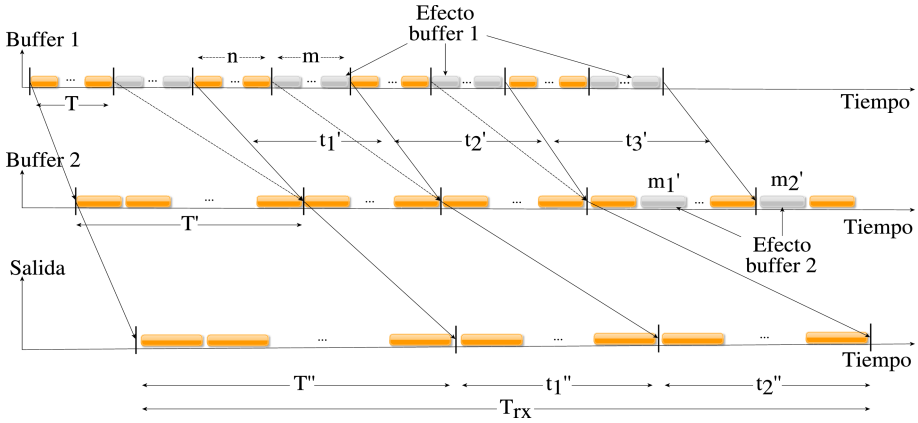

Figura 6.2: Relación temporal de los paquetes a través de dos *buffer* concatenados.

tes asociada a cada *buffer*, se pueden obtener las expresiones que se resumen en la Tabla 6.1.

Tasa	Tamaño del buffer
$R_3 = \frac{n_{rx}}{T_{rx}} \times packet_{size}$	$L_{Buffer1} = \frac{T'}{\frac{1}{R_1 - R_2} + \frac{1}{R_2}}$
$R_2 = \frac{n_{rx} + m'}{t''_n} \times packet_{size}$	$L_{Buffer2} = \frac{T''}{\frac{1}{R_2 - R_3} + \frac{1}{R_3}}$
$R_1 = \frac{n_{rx} + m + m'}{t'_n} \times packet_{size}$	

Tabla 6.1: Ecuaciones para estimar los parámetros de dos *buffer* concatenados.

Estas ecuaciones describen un caso muy particular y con condiciones ideales. La exactitud de los resultados en una situación real depende de la correcta estimación de cada uno de los parámetros de los cuales dependen (tasas de entrada y salida, pérdidas asociadas a cada *buffer*, etc.). En general, se convierte en un proceso más complejo a la hora de procesar los datos que incluso puede llegar a ser inviable, ya que se debe inferir en qué *buffer* se han producido las pérdidas y

se debe estimar una cantidad mayor de tasas de entrada y salida en función de la cantidad de los *buffer*.

6.1.1 Ejemplo de validación real controlada

Con la finalidad de comprobar las conclusiones acerca de la complejidad e inviabilidad del método expuestas previamente sobre un escenario ideal, se implementó el escenario controlado de laboratorio que se muestra en la Figura 6.3. Para este escenario, se utilizaron los dos tipos de comportamientos de *buffer*: el *Buffer* 1 que descarta paquetes en ráfagas y el *Buffer* 2 que descarta un paquete a la vez. El *Buffer* 1 corresponde a un *switch* $3COM$ 4500 mientras que el *Buffer* 2 se ha implementado mediante TC (*Traffic Control*) en un *host* Linux con kernel $2.6.38 - 7$, procesador $Intel® \ Core^{TM} \ i3 \ CPU \ 2{,}4 \ GHz$, tarjeta de red a 100 $Mbps$. El comportamiento de cada uno de estos equipos se comprobó previamente mediante el análisis que se describió en el capítulo 5. Además, se ha tenido en cuenta que la relación de las tasas sea la adecuada para producir un punto crítico ($R_1 > R_2 > R_3$), configurando el escenario para que las tasas máximas para R_2 y R_3 sean 10 y 6 $Mbps$ respectivamente.

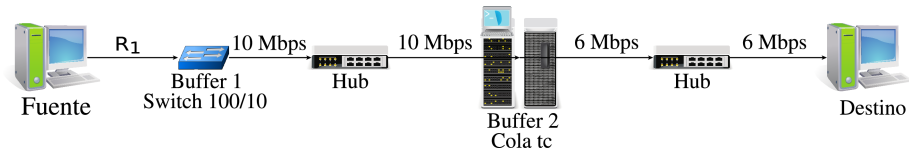

Figura 6.3: Escenario de pruebas para dos *buffer* concatenados.

La prueba se repite para tres tráficos diferentes de entrada (R_1 con valores de 20 $Mbps$, 30 $Mbps$ y 40 $Mbps$) y para un tamaño de paquete de 1500 $bytes$. La Figura 6.4 muestra los resultados de las pruebas, aplicando las ecuaciones correspondientes, para un tráfico de entrada de 30 $Mbps$, en este caso la velocidad de llenado del *buffer* 1 es de 20 $Mbps$ y la del *buffer* 2 es de 4 $Mbps$. En dicha figura se puede observar que durante los primeros 2000 μs el *buffer* del *switch* es el único que entra en congestión en varias ocasiones, mientra que el *Buffer* 2 no ha

perdido ningún paquete en ese mismo período, al mismo tiempo se puede observar que el tamaño del *Buffer* 1 es de 115 paquetes.

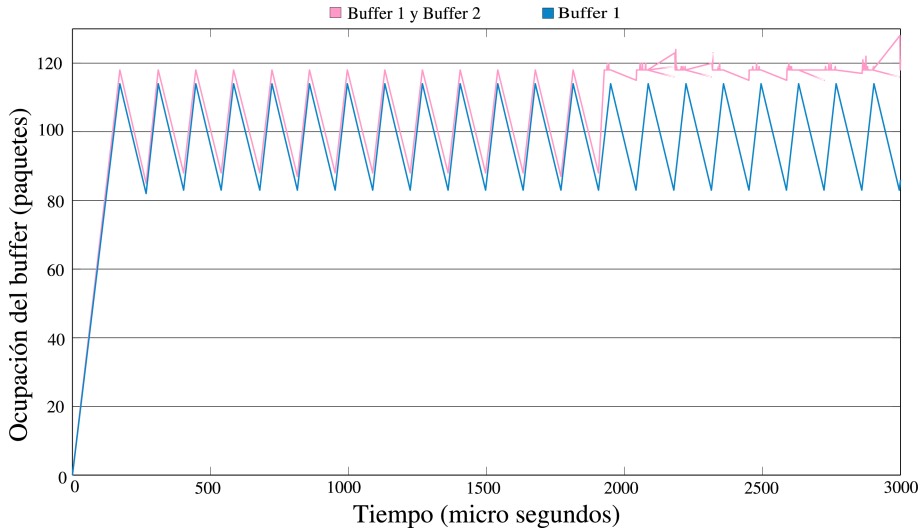

Figura 6.4: Estimación de la ocupación de dos *buffer* concatenados.

Sin embargo, después de $t = 2000\ \mu s$, el *Buffer* 2 se llena y el efecto de la pérdida de paquetes de cada uno de los dos *buffer* se solapa. Desde ese momento, el método de análisis utilizado en el capítulo 5 no permite estimar las ocupaciones de los dos dispositivos, ya que no es posible obtener una estimación precisa de la tasa de entrada del *Buffer* 2. En la Tabla 6.2 se presenta un resumen de los resultados del tamaño del *buffer* 1 (ya que es el único que ha podido obtener) para todos los valores de tasa utilizados en las pruebas.

Por este motivo es necesario desarrollar un nuevo método de análisis y estimación de la ocupación para los casos en los que se requiere detectar múltiples *buffer* en un camino de red.

6.2 Nuevo procedimiento

En esta sección se propone un nuevo método que permita descubrir características de la red por medio de determinar los parámetros del modelo de los *buffer* incluso

Dispositivo	Tasa	Tamaño	
	(Mbps)	LI	LS
Switch	20	85	115
	30	85	115
	40	85	115

Tabla 6.2: Tamaño del *buffer* de un *switch* $3COM$ 4500 (LI: Límite inferior y LS: Límite superior) para diferentes tasas de prueba y con tamaño de paquetes de 1500 $bytes$.

cuando diversos dispositivos están concatenados. Con esta información las aplicaciones tienen ventajas significativas para adaptar el tráfico que generan evitando la degradación de la QoS.

Este nuevo método está basado en la metodología general descrita en el capítulo 5, se utiliza un escenario igual al descrito en la Figura 5.4 y el procedimiento también consiste en el envío de una ráfaga de paquetes UDP que desborda los *buffer* desde un *host* fuente hasta el destino. Todos los paquetes son identificados por medio de un número de secuencia que se incluye en el *payload*. La diferencia está en que se ha mejorado la técnica de detección de los *buffer* por medio del uso de cuatro pasos:

- Análisis de los patrones de pérdida de paquetes.
- Determinar tasas.
- Inferir ubicaciones.
- Estimar tamaño de los *buffer*.

A modo de ejemplo y para explicar mejor este nuevo método se ha reconstruido la relación temporal de los paquetes a través de cada uno de los *buffer* como se observa en la Figura 6.5. Además, se ha seleccionado el mismo ejemplo de validación real controlada visto en el apartado anterior y donde el comportamiento de los dos *buffer* concatenados es: el *Buffer* 1 descarta ráfagas de paquetes mientras el *Buffer* 2 descarta un paquete de forma aislada. Es necesario aclarar que este comportamiento no necesariamente debe ser así y que se ha seleccionado este

comportamiento a modo de ejemplo ya que es un comportamiento bastante común y se presenta en múltiples equipos comerciales.

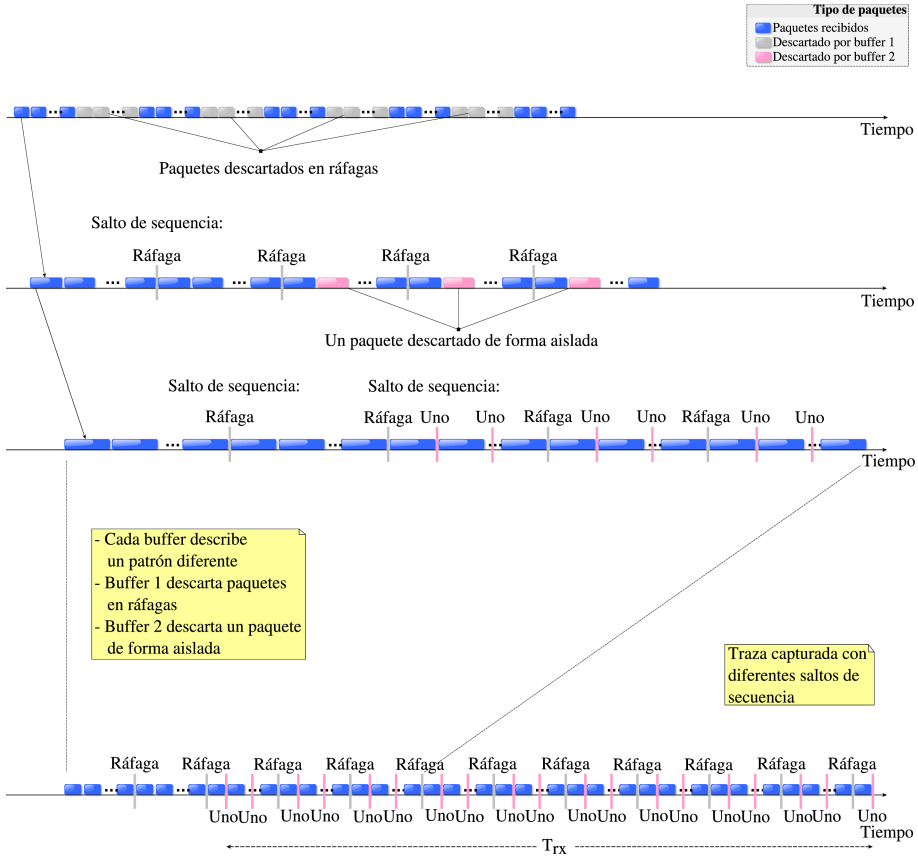

Figura 6.5: Relación de pérdida de paquetes a través de dos *buffer* concatenados.

Este procedimiento permite determinar los mismos parámetros de los *buffer* que los obtenidos mediante los procedimientos que se describieron el capítulo 5 en el caso de que sólo haya un punto crítico en el camino de transmisión pero con la diferencia que ahora se pueden detectar diversos *buffer* en el camino de red, por medio de un análisis de los patrones de las pérdidas de paquetes.

Se debe tener en cuenta que se asume que la estimación es remota, y por lo tanto, para este análisis solamente se tiene acceso a la información en el *host* de

destino, ya que si se tuviera acceso físico al SUT se podría aplicar el método correspondiente descrito en el capítulo 5.

6.2.1 **Análisis de patrones de pérdidas**

Este análisis consiste en observar los diferentes patrones de pérdidas que definen el comportamiento de cada *buffer* y determinar la cantidad de paquetes perdidos por cada patrón. Para esto se utiliza un mapa de pérdidas de paquetes, el cual consiste en un gráfico que muestra la dispersión de la pérdida de paquetes, es decir el grado de la variabilidad de la distribución de dichas pérdidas. Dicho gráfico se puede obtener si se coloca en el eje "*x*", el tiempo en que ocurre cada pérdida (o la correspondiente numeración secuencial del evento) y en el eje "*y*", el valor correspondiente del tamaño de la ráfaga de paquetes perdidos, como se observa en la Figura 6.6 donde se muestra el efecto de dos *buffer*.

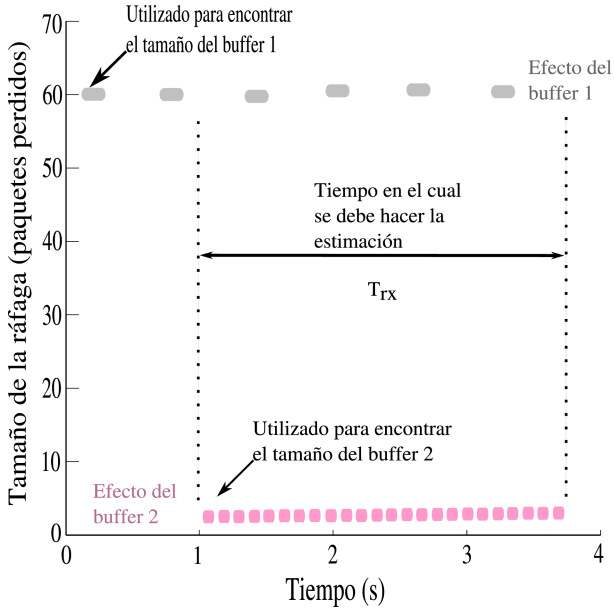

Figura 6.6: Mapa de pérdida de paquetes para dos *buffer* concatenados.

Del mapa de pérdidas se puede obtener la cantidad de *buffer* que se encuentran

en congestión y la tasa de pérdidas de cada uno de ellos. Por ejemplo, en la Figura 6.6 se observan dos patrones diferentes que corresponden a dos *buffer*. La pérdida de paquetes para el primero es de 60 paquetes por ráfaga mientras que el segundo descarta 1 paquete por ráfaga. Además, se puede determinar el número de paquetes perdidos para el *Buffer* 1 (N_1) y el *Buffer* 2 (N_2) en un período determinado (T_{rx}).

Este análisis es útil para determinar la cantidad de pérdidas en cada intervalo en el cual cada *buffer* descarta paquetes y para observar los patrones que definen el modelo del *buffer*. Además, se debe tener en cuenta que la información contenida en el mapa de pérdidas es equivalente a la ocupación de cada *buffer* bajo las mismas condiciones, ya que el instante en el que se presenta cada pérdida corresponde con el punto de máxima ocupación de cada *buffer*.

6.2.2 Determinar tasas

Las tasas de salida y entrada se puede determinar por medio de la traza en el *host* de destino (ver Figura 6.5), ya que se conocen los paquetes que han llegado y se pueden contar los paquetes perdidos porque se conocen los números de secuencia de los paquetes que faltan. Con esta información de la traza en el *host* de destino, se puede calcular la tasa de salida (R_3 en la Figura 6.5) con la ecuación 6.1 debido a que se conocen todos lo paquetes que han llegado, N, el tiempo de llegada de cada uno y la longitud de los paquetes, P_L. La tasa de entrada (R_1 en la Figura 6.5) se puede estimar con los paquetes que han llegado y todos los paquetes perdidos mediante la ecuación 6.2. Es necesario aclarar que todas las tasas se deben estimar durante el período en el que se observan todos los patrones de pérdidas de paquetes T_{rx}, ver Figura 6.5 y 6.6.

$$R_3 = \frac{P_L}{T_{rx}} \times N \tag{6.1}$$

$$R_1 = \frac{P_L}{T_{rx}} \times (N + N_1 + N_2) \tag{6.2}$$

Como se puede ver en Figura 6.5, los paquetes que atraviesan el *Buffer* 1 se propagan a una tasa intermedia (R_2 en dicha figura). Esta cantidad de paquetes es la suma de los paquetes que han llegado al *host* de destino más los paquetes que

ha descartado el *Buffer* 2 (N_2). La tasa intermedia se debe calcular como el total de $bits$ que atraviesan el *buffer* 1 dividido entre el tiempo correspondiente. Sin embargo, a pesar que se sabe que existen dos patrones de pérdidas de paquetes en la Figura 6.6, no es posible determinar, por el momento, a qué *buffer* corresponde cada uno de ellos y por lo tanto se tienen dos posibles valores de R_2.

Por este motivo, se realiza el cálculo de la tasa intermedia para los dos posibles valores de pérdida de paquetes mediante la ecuación 6.3. Con la finalidad de obtener el valor correcto de dicha tasa se debe realizar una nueva prueba, en la cual, el valor de la tasa de generación de paquetes debe estar entre los valores obtenidos mediante la ecuación 6.3 para que en función de la posición de los *buffer* se congestione uno de ellos o los dos. Si los resultados de esta nueva prueba muestran (por medio de un mapa de pérdidas) que los dos *buffer* se encuentran en congestión, entonces el valor correcto de R_2 será el más bajo. Si los resultados de esta nueva prueba muestran que solo un *buffer* se encuentra en congestión, el valor de R_2 será el más alto.

$$R_2 = \begin{cases} \dfrac{P_L}{T_{rx}} \times (N + N_1) \\ \\ \dfrac{P_L}{T_{rx}} \times (N + N_2) \end{cases} \tag{6.3}$$

Por otro lado, al realizar las estimaciones de las tasas se debe tener en cuenta las posibles variaciones que éstas puedan sufrir en una misma prueba, como se ha comentado en el capítulo anterior. En este sentido, se sugiere que las estimaciones de dichas tasas se realicen de manera periódica para cada tráfico de prueba. Para los casos en que las pérdidas se producen a ráfagas, el instante en el cual el *buffer* descarta paquetes puede ser considerado como un evento interesante para realizar dichos cálculos. Sin embargo para los casos en que las pérdidas se producen de manera continua, intercalando tramas transmitidas con pérdidas, habrá que definir una ventana de cálculo que habrá que ir desplazando.

6.2.3 Inferir ubicaciones

La relación de las tasas de entrada y salida de un *buffer* define la tasa de pérdida de paquetes que éste va a tener. De la misma manera, cuando se sabe la tasa de pérdida de cada *buffer* (debido a las relaciones $R_1 - R_2$ y $R_2 - R_3$, ver Figura 6.5) se puede comparar dicha tasa con la relación de las tasas de entrada y salida de cada *buffer* (R_1, R_2 y R_3) para inferir la ubicación de cada dispositivo. Si R_2 es el correspondiente a la ecuación donde aparece N_1 el *buffer* que provoca estas pérdidas está en segundo lugar. En caso contrario será el *buffer* que provoca las pérdidas N_2 el que se sitúa en segundo lugar.

6.2.4 Estimar tamaño del *buffer*

Con los procedimientos anteriores se obtiene información del comportamiento de la pérdida de paquetes, las tasas de entrada y salida de cada *buffer* y sus respectivas ubicaciones. Con esta información se puede determinar cual de los *buffer* es el más restrictivo debido a su mayor tasa de llenado, además, se sabe el instante en que cada *buffer* se congestiona y pierde paquetes.

El tamaño de cada *buffer* se puede determinar si se selecciona un valor adecuado de la tasa de entrada que congestione solamente uno de los *buffer*. En este caso, los métodos analizados en el capítulo 5 se pueden aplicar ya que solamente habría un *buffer* en congestión a la vez. Por ejemplo, en el caso que se ha estado analizando en la Figura 6.5, se sabe que el *buffer* 1 es el más restrictivo ya que tiene la tasa de llenado más alta que el *buffer* 2 ($R_2 - R_1 > R_3 - R_2$) y por lo tanto es el primero en perder paquetes. Entonces se puede seleccionar una tasa de entrada (R_1) que congestione rápidamente el *buffer* 1 ($R_1 > R_2$), donde la ráfaga de paquetes de prueba sea lo suficientemente pequeña para no congestionar el *buffer* 2. Para congestionar solamente el *buffer* 2 la tasa de prueba debe tener un valor que cumpla con $R_2 > R_1 > R_3$.

Otra forma de estimar el tamaño de un *buffer* es por medio de la latencia de un paquete en el *buffer* cuando está lleno, para esto es necesario saber las tasas de entrada y salida de dicho *buffer*. Para este caso se debe utilizar el último paquete recibido antes de la primera pérdida de paquetes como se muestra en la Figura

6.7, debido a que dicho paquete es el primero que llena el *buffer* por completo. El número de secuencia (SN) de este paquete n corresponde a la cantidad de paquetes enviados en un tiempo T. El $n-ésimo$ paquete entra al $Buffer_m$ en un tiempo: $T_m = n * P_L/R_m$ y el tiempo de salida es $T_{m+1} = n * P_L/R_{m+1}$. Entonces, la latencia del paquetes es $T_{m+1} - T_m$, así el tamaño del *buffer*, en paquetes, del $m-ésimo$ *buffer* se puede estimar utilizando la ecuación 6.4, la cual solamente depende de la relación de las tasas de entrada y salida y de los n paquetes que llegaron antes de la primera pérdida de paquetes de cada *buffer*.

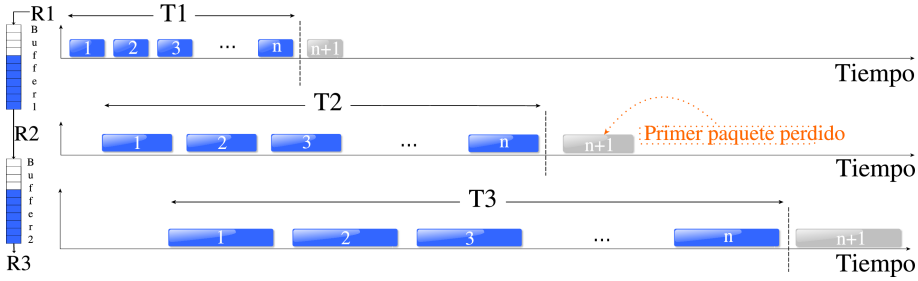

Figura 6.7: Estimación del tamaño del *buffer* desde el último paquete recibido antes de la primera pérdida de paquetes.

La cantidad de paquetes que llegan a cierto *buffer* depende de qué *buffer* descarta paquetes primero y de la ubicación física de cada *buffer*. De la Figura 6.5 se sabe que el *Buffer* 1 está físicamente antes que el *Buffer* 2, entonces, los paquetes descartados por el *Buffer* 1 nunca llegarán al *Buffer* 2. De la Figura 6.6 se sabe que el *Buffer* 1 es el primero en descartar paquetes, de esta manera se puede deducir que $n_1 = SN_1$ y $n_2 = SN_2 - N_1$. El mismo análisis se puede aplicar cuando el *Buffer* 2 está antes que el *Buffer* 1. En dicho caso los paquetes perdidos por el *Buffer* 2 podrán atravesar el *Buffer* 1, por lo que $n_1 = SN_1$ y $n_2 = SN_2$.

$$\begin{aligned} L_{Buffer_m} &= (T_{m+1} - T_m) \times \frac{R_{m+1}}{P_L} \\ &= n_m \times \left(1 - \frac{R_{m+1}}{R_m}\right) \end{aligned} \tag{6.4}$$

Para determinar si el *buffer* está medido en $bytes$ o paquetes se puede seguir el mismo procedimiento descrito en el capítulo 5, en el cual se repite la prueba cambiando el tamaño de los paquetes y se comparan los resultados: si el tamaño del *buffer* es el mismo para todas las pruebas, el *buffer* está medido en número de paquetes, de lo contrario en $bytes$.

6.3 Ejemplo teórico

A continuación se analiza un ejemplo con el procedimiento que se describió anteriormente. El escenario en cuestión se muestra en la Figura 6.8, en la cual dos equipos terminales se comunican a través de un camino de red desconocido. Durante la prueba se envía desde el *host* fuente hasta el de destino, un tráfico de prueba que corresponde a una ráfaga de 160 paquetes con un tamaño de 1500 $bytes$. Una captura de tráfico se realiza únicamente en el equipo de destino, en la cual se reciben 60 paquetes.

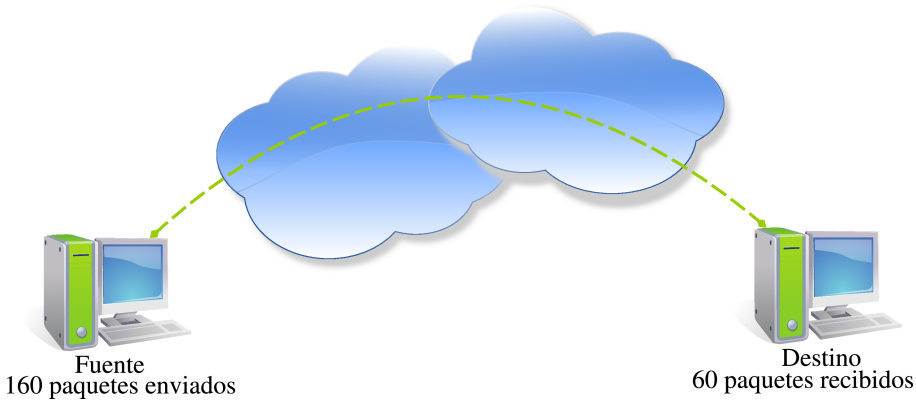

Figura 6.8: Topología de referencia para el ejemplo.

Ejemplo (Dos *buffer* concatenados)**.**

Dos equipos se comunican a través de un camino de red desconocido. Se envía un tráfico de prueba de 160 paquetes para descubrir el modelo de los *buffer* en dicho camino.

Existen dos formas de realizar un mapa de pérdidas: la primera consiste en representar en el eje "x", el tiempo en el que se produjeron las pérdidas, la segunda en representar en dicho eje el número de secuencia de los paquetes asociados a las mismas pérdidas. En ambos casos la información que se obtiene es la misma. Con la finalidad de mostrar ambas situaciones, en este ejemplo se realizaran ambos gráficos para el caso del primer mapa, y posteriormente, se utilizará el número de secuencia para el resto del ejemplo.

El mapa de pérdida de paquetes se muestra en las Figuras 6.9 y 6.10 en las cuales se observan dos patrones de pérdida bien definidos. El $Patrón\ A$ tiene una tasa de pérdida de 2 paquetes cada vez mientras que para el $Patrón\ B$ es de 20 paquetes. Los resultados sugieren que existen dos *buffer* (de comportamiento diferente) que se han congestionado durante la prueba en el camino de red.

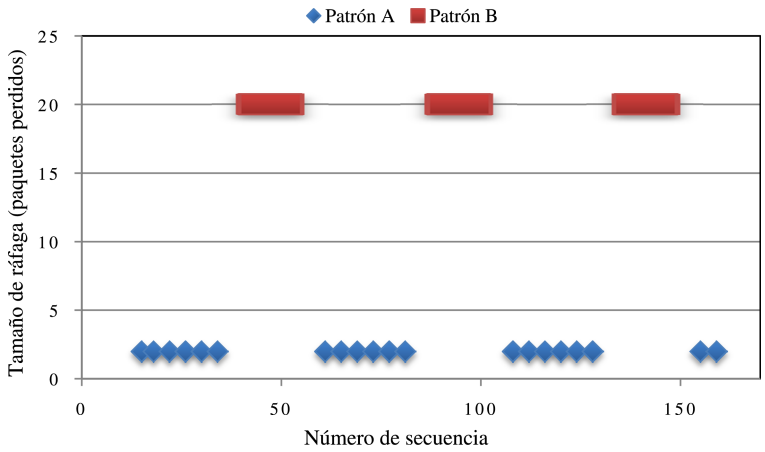

Figura 6.9: Mapa de pérdida de paquetes para el ejemplo con dos patrones diferentes en función del número de secuencia.

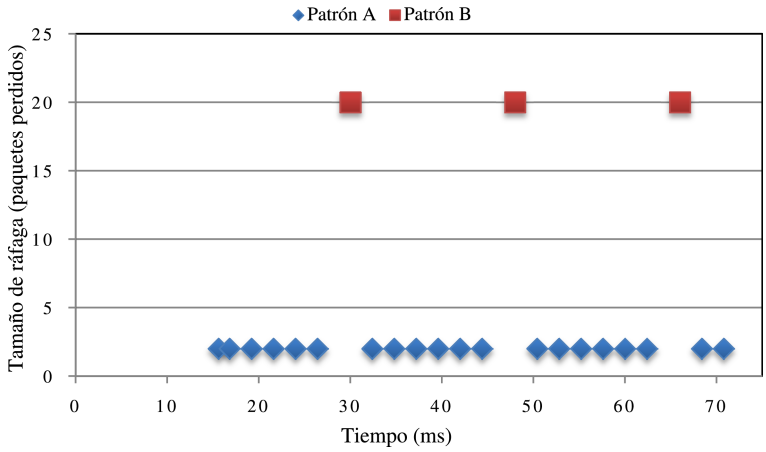

Figura 6.10: Mapa de pérdida de paquetes para el ejemplo con dos patrones diferentes en función del tiempo.

De la traza en el *host* de destino se puede determinar que el tamaño de los paquetes es 1500 $bytes$, $T_{rx} = 72\ ms$, los paquetes que llegaron son 60 y por lo tanto los perdidos son 100. Con esto se pueden determinar las tasas de salida y entrada al SUT por medio de las ecuaciones 6.1 y 6.2 respectivamente. En este caso teórico, se ha considerado un comportamiento ideal del tráfico en el cual no se presentan variaciones en la tasa, por lo que solamente se calcula un valor para cada tasa.

$$R_{out} = \frac{P_L}{T_{rx}} \times N = \frac{1500 \times 8}{0{,}072} \times 60 = 10\ Mbps$$

$$R_{in} = \frac{P_L}{T_{rx}} \times (N + N_A + N_B) = \frac{1500 \times 8}{0{,}072} \times (60 + 100) = 26{,}67\ Mbps$$

Por otro lado, al contar la cantidad de paquetes perdidos por cada patrón se observa que el $Patrón\ A$ tiene un total de 40 paquetes perdidos mientras que en el $Patrón\ B$ dicha cantidad es de 60, entonces la tasa intermedia puede tener los siguientes dos valores, según la ecuación 6.3.

$$R_A = \frac{P_L}{T_{rx}} \times (N + N_A) = \frac{1500 \times 8}{0{,}072} \times (60 + 40) = 16{,}67\ Mbps$$

$$R_B = \frac{P_L}{T_{rx}} \times (N + N_B) = \frac{1500 \times 8}{0{,}072} \times (60 + 60) = 20\ Mbps$$

Para determinar cual de las tasas intermedias (R_A o R_B) es la correcta se realiza una nueva prueba con una tasa de generación de paquetes de $18\ Mbps$, la cual se encuentra entre los dos posibles valores. El mapa de pérdidas correspondiente a dicha prueba se muestra en la Figura 6.11, la cual muestra solamente un patrón de pérdidas de paquetes, por lo tanto, el valor correcto de la tasa intermedia es $R_B = 20\ Mbps$.

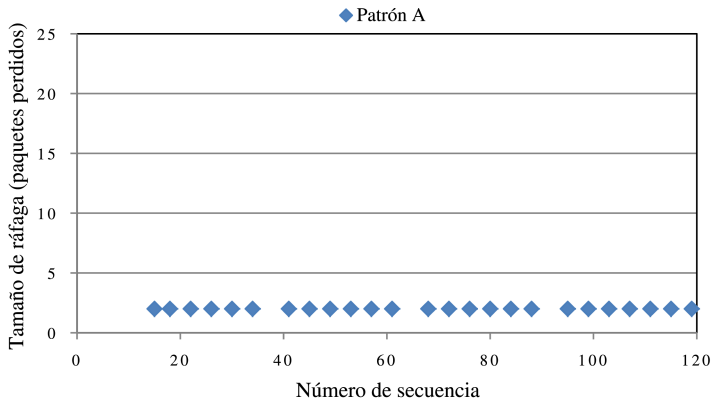

Figura 6.11: Mapa de pérdida de paquetes para el ejemplo con una tasa de prueba de $18\ Mbps$.

La Figura 6.12 resume la información que se ha obtenido hasta el momento. Se sabe que existen dos *buffer* que se congestionan en el camino de red y que $R_{in} > R_B > R_{out}$. A la vez, la relación entre las tasas de entrada y salida de cada *buffer* define la tasa de llenado que éste va a experimentar, la cual tiene una influencia directa en la tasa de pérdida de paquetes. Por este motivo, se puede inferir que el *Buffer* 2 debe descartar más paquetes que el *Buffer* 1, y por lo tanto el $Patrón\ A$ corresponde al *Buffer* 1 y el $Patrón\ B$ al *Buffer* 2. Entonces, se puede concluir que el *Buffer* 1 tuvo una tasa de pérdida de paquetes de 2 por vez con una pérdida total de 40 paquetes de los 160 que llegaron a su entrada, y que, el *Buffer* 2

descartaba ráfagas de 20 paquetes con una pérdida total de 60 paquetes de los 120 que llegaron a su entrada, como se observa en la Figura 6.12.

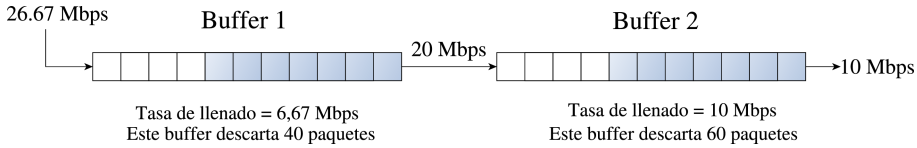

Figura 6.12: Asignación de las ubicaciones y las tasas de entrada y salida de los *buffer* para el ejemplo.

Con el fin de estimar el tamaño de los *buffer* se debe determinar el último paquete recibido antes de la primera pérdida para cada uno de los *buffer*. De la traza se observa que dichos paquetes son el número 40 y el 70 para el *Buffer* 1 y el *Buffer* 2 respectivamente, con esta información se estima el tamaño de cada *buffer* mediante la ecuación 6.4.

$$L_{Buffer_1} = n_1 \times \left(1 - \frac{R_B}{R_{in}}\right) = 40 \times \left(1 - \frac{20}{26{,}67}\right) = 10\ paquetes$$

$$L_{Buffer_2} = n_2 \times \left(1 - \frac{R_{out}}{R_B}\right) = 70 \times \left(1 - \frac{10}{20}\right) = 35\ paquetes$$

6.4 *N-buffer*

Es necesario dejar claro que no se puede estar seguro del número de dispositivos que se encuentran en un determinado camino de red, tampoco de su orden o su comportamiento. Los análisis que se describieron en este capítulo se presentaron a modo de ejemplo, pero la metodología en sentido general se puede aplicar a cualquier comportamiento de los dispositivos de red. Además, se debe tener en cuenta que lo más importante es describir un modelo de *buffer* del dispositivo más restrictivo ya que éste es el que tiene un efecto pronunciado en la degradación de la QoS.

Sin embargo, es posible determinar más de dos *buffer* mediante el uso de la metodología propuesta en este capítulo. El mapa de pérdidas es la principal herramienta para determinar el número de *buffer* en congestión, pero hay que tener cuidado a la hora de determinar dicho número ya que un determinado patrón puede ser confundido con el efecto de dos o más *buffer* que descartan paquetes al mismo tiempo como se muestra en la Figura 6.13.

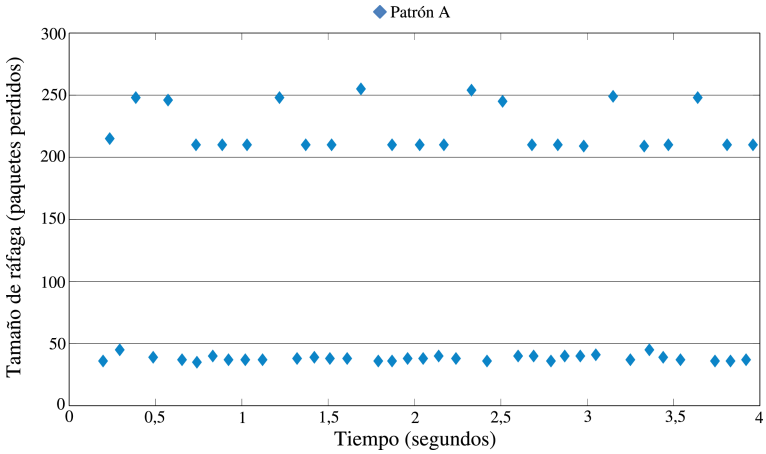

Figura 6.13: Ejemplo de dos *buffer* concatenados que descartan paquetes al mismo tiempo.

Al observar dicha figura se pueden determinar tres patrones diferentes, el primero de ellos con una tasa alrededor de los 40 paquetes, el segundo con una tasa de 210 paquetes y el tercero con una tasa de 250 paquetes. El tercer patrón tiene dos características que lo excluyen de corresponder al efecto de un tercer *buffer*: la primera es que el valor de la tasa (250 paquetes) corresponde a la suma de los otros dos patrones y la segunda es que en los instantes que hay pérdidas con un valor de 250 paquetes, los otros dos patrones presentan discontinuidades en cuanto a la frecuencia en que se producían las pérdidas en cada uno de los *buffer*.

El presente trabajo se limita a estimar como máximo dos *buffer* concatenados por las razones que se expusieron anteriormente, dejando abierta una nueva línea de investigación relacionada a determinar múltiples *buffer*.

La lógica de validación nos permite movernos entre los dos límites del dogmatismo y el escepticismo.

Paul Ricoeur

CAPÍTULO

7

Análisis de casos

En este capítulo se analizan los detalles de los métodos para determinar el modelo de un determinado *buffer*, mediante una serie de casos de estudio que demuestran y validan los métodos que se describieron en los capítulos 5 y 6. Los casos que se presentan son implementaciones reales con dispositivos comerciales bien conocidos y que se encuentran comúnmente en diversas redes. Para el desarrollo de las pruebas se plantean escenarios controlados de laboratorio en los cuales se analizan redes cableadas e inalámbricas.

Para realizar las pruebas se utilizan dos sondas en los extremos de la red que consisten en equipos IP que transmiten y reciben ráfagas de paquetes mediante un generador de tráfico, dichos equipos pueden establecer una comunicación entre ellos y realizar el envío del tráfico de manera desatendida. Las sondas son *host* Linux con kernel $2.6.38 - 7$, procesador $Intel® \ Core^{TM} \ i3 \ CPU \ 2{,}4 \ GHz$ y tarjeta de red a $100 \ Mbps$. El generador de tráfico que se incluyó en las sondas es una aplicación desarrollada en el CeNITEQ (Communications Networks and Information Technologies for e-Health and Quality of Experience) de la Universidad de Zaragoza, el cual se denomina ETG (*End-to-end Traffic Generator*) [PNM+11]. Esta herramienta es capaz de generar comunicaciones extremo a extremo entre dos *host* y enviar múltiples flujos UDP en modo *one-way* o *round trip*, además gestiona las capturas de tráfico en ambos extremos y realiza otras funciones adicionales.

7.1 Acceso físico

Con el fin de validar la metodología para el análisis de *buffer* cuando se tiene acceso físico se han propuesto dos escenarios reales diferentes, el primero de ellos es una red cableada (Ethernet) y el segundo un sistema inalámbrico (WiFi). Dichos escenarios se han seleccionado debido a sus condiciones de transmisión, esto permite analizar un escenario bastante estable (Ethernet) y otro muy variable (WiFi).

Para el procesamiento de los datos, se han desarrollado diversos *script* utilizando el *shell* de Linux para automatizar tareas: procesar las capturas realizadas, determinar las tasas de entrada y salida, la ocupación, los límites y el tamaño del *buffer*. Para ésto se han utilizado los procedimientos y las ecuaciones correspondientes al método de acceso físico que se describieron en el capítulo 5, además de las sugerencias relacionadas a la estimación del ancho de banda. En este aspecto, para el caso del método 2, los *script* se han elaborado para que las estimaciones de las tasas de entrada y salida del *buffer* se realicen de manera periódica cada vez que se detecta una pérdida de paquetes, y por lo tanto, es posible obtener una estimación del tamaño del *buffer* en ese mismo instante. Dichos *scritp* generan datos en ficheros con formato CSV (*Comma-Separated Values*) que luego son manipulados por software de procesamiento matemático para la correspondiente presentación de los resultados.

7.1.1 Escenario ethernet

Para este caso, se implementa un entorno controlado de laboratorio que permite determinar el modelo del *buffer* de un *switch* $3COM\ 4500$, dicho escenario se muestra en la Figura 7.1. En dicha figura se observa que el *switch* tiene una tasa de entrada de $100\ Mbps$ y una salida de $10\ Mbps$, lo que provoca la situación de congestión necesaria para realizar el análisis. Los *hub* se incluyen con la finalidad de permitir la captura del tráfico por medio de un *sniffer* a la entrada y la salida del *buffer*.

Inicialmente, se han realizado una serie de pruebas preliminares para determinar la tasa de prueba adecuada para realizar las medidas y se observa que a tasas cercanas a la capacidad del enlace ($10\ Mbps$) los resultados presentan variaciones

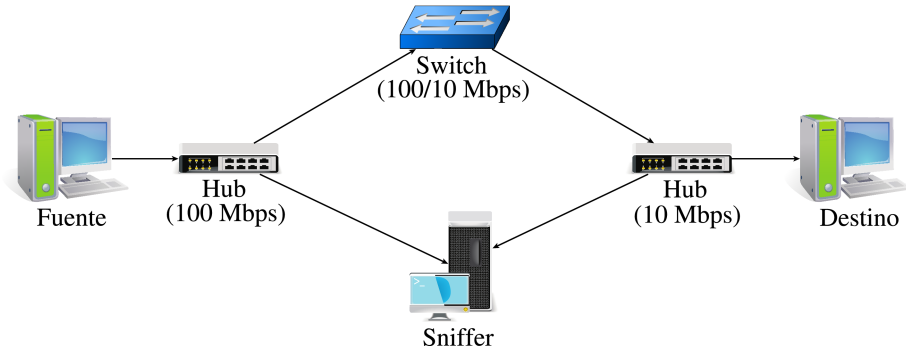

Figura 7.1: Escenario para determinar el modelo del *buffer* de un *switch*.

sensibles. Por esto, se han seleccionado tasas de entrada suficientemente altas que no tengan este problema. Cada prueba se realizó para tres valores diferentes de tasas de entrada (20, 30 y 40 $Mbps$) y se han repetido para tres tamaños de paquetes (1500, 800 y 200 $bytes$) en cada caso.

7.1.1.1 Resultados mediante el método 1

Los resultados muestran que la ocupación del *buffer* tiene un comportamiento similar para los tres tipos de tasa de entrada utilizadas, encontrándose que el tamaño del *buffer* se puede determinar con la misma exactitud y tiene un valor de 115 paquetes. Además, como las pruebas realizadas para los diferentes tamaños de paquetes generan el mismo valor de tamaño de *buffer*, se puede decir que el *buffer* del *switch* establece su capacidad en número de paquetes y no en número de $bytes$.

Dada la similitud de los resultados obtenidos para las diversas pruebas, se han seleccionado para su presentación los correspondientes a la tasa de prueba de 30 $Mbps$ con todos los tamaños de paquetes utilizados, los cuales se muestran en la Figura 7.2. En dicha figura se observa que, como se ha dicho anteriormente, el tamaño es de 115 paquetes para los tres tamaños de paquetes, y además, la diferencia entre los límites superior e inferior es de aproximadamente 30 paquetes.

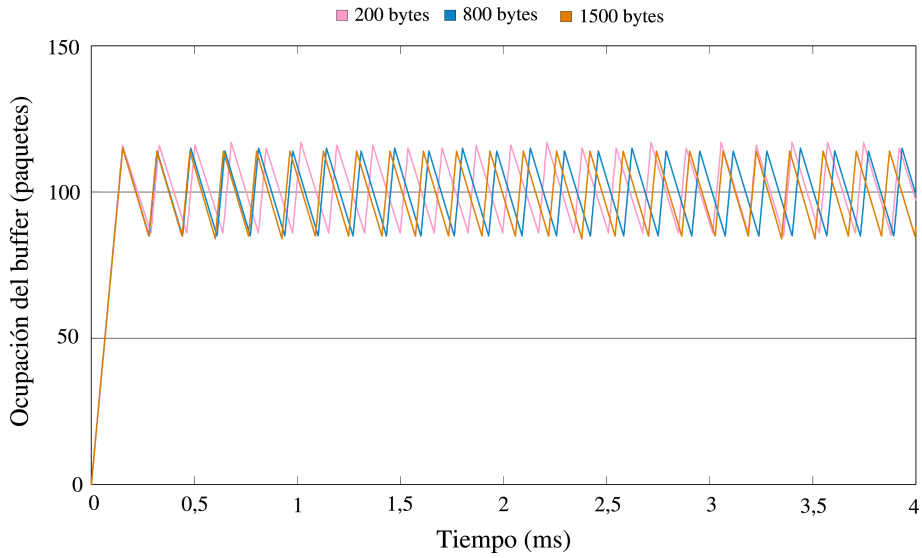

Figura 7.2: Ocupación del *buffer* de un *switch* $3COM$ 4500 para tres flujos con diferentes tamaños de paquete analizados mediante el método 1.

7.1.1.2 Resultados mediante el método 2

En este caso, se han realizado las pruebas para los mismos anchos de banda (20, 30 y 40 $Mbps$) y tamaños de paquetes (1500, 800 y 200 $bytes$) que los utilizados para el método 1, la razón de esto es estimar el tamaño del *buffer* y comparar los resultados con ambos métodos. Como se mencionó en el capítulo 5, el método 2 está basado en que la tasa de salida se puede obtener con trazas capturadas en el *host* de destino, y que, por tanto, la precisión de la estimación del tamaño depende de la precisión con que se estime dicha tasa.

En la Figura 7.3 se observan los resultados correspondientes a una tasa de prueba de 30 $Mbps$ y para los tamaños de paquetes mencionados. Para los flujos con tamaños de paquetes de 1500 $bytes$, con el método 2 se han obtenido para el tamaño del *buffer* y sus límites, valores idénticos que los del método 1. Para tamaños de paquetes de 800 y 200 $bytes$ se observa que éstos datos sufren una pequeña variación (ver Tabla 7.1), que aunque no resulta significativa puede alterar la precisión del método. Esto se debe a que cuanto menor es el tamaño de los paquetes mayor es el error en la estimación de la tasa, y por lo tanto, en la estimación del

tamaño del *buffer*. El principal motivo es que los paquetes con diversos tamaños alcanzan un ancho de banda diferente en una red IP a causa de que obtienen un valor de eficiencia diferente en la transmisión como se mencionó en el capítulo 3.

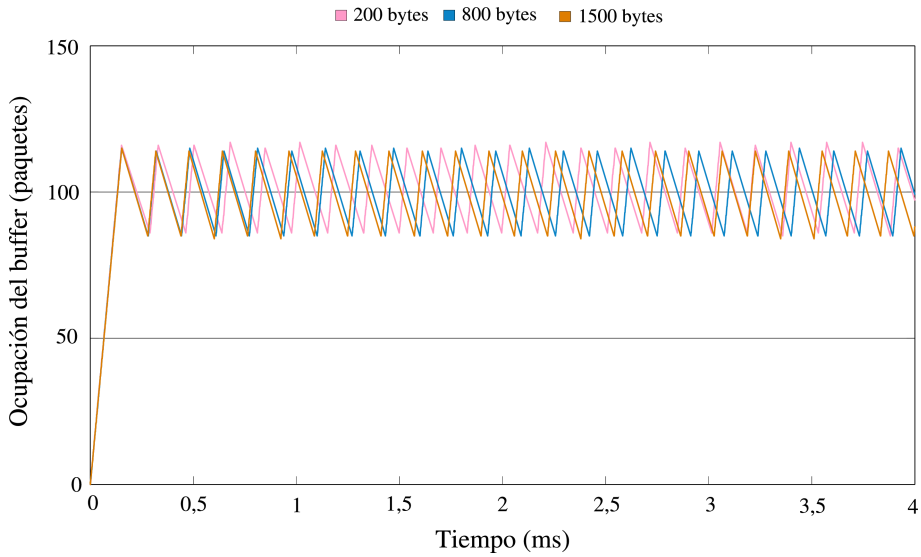

Figura 7.3: Ocupación del *buffer* de un *switch* $3COM$ 4500 para dos flujos diferentes mediante el método 2.

Dispositivo	**Paquete**	**20 *Mbps***		**30 *Mbps***		**40 *Mbps***	
	(*Bytes*)	**LI**	**LS**	**LI**	**LS**	**LI**	**LS**
Switch	200	85	116	86	116	86	116
	800	85	114	85	114	85	115
	1500	85	115	85	115	85	115

Tabla 7.1: Estimación del tamaño del *buffer* del *switch* en número de paquetes, mediante el método 2, para diversos tráficos de prueba y con diferentes tamaños de paquete (LI: Límite inferior y LS: Límite superior).

Por otro lado, los resultados muestran que en este caso las tasas de entrada y salida del *buffer* son muy estables a lo largo de cada prueba y por lo tanto la

estimación del tamaño de dicho dispositivo, lo cual se puede observar al comparar los límites superiores del *buffer* entre sí o bien sus límites inferiores (ver Figura 7.3). Además, los valores obtenidos para el tamaño del *buffer* en la Figura 7.3 y en la Tabla 7.1 comprueban que la capacidad de dicho dispositivo está definida en número de paquetes, debido a que las pruebas con diferentes tamaños de paquetes dan como resultado un valor muy similar.

En la Tabla 7.2 se presenta un resumen comparativo de los resultados para los dos métodos con tres anchos de banda diferentes y para un tamaño de paquete de $1500\ bytes$, ya que como se ha justificado anteriormente, es el más exacto y el que debe utilizarse para las pruebas. Sin embargo, siempre será necesario realizar pruebas con otros valores para determinar si el *buffer* establece su capacidad en paquetes o en $bytes$.

Dispositivo	**Tasa**	**Método 1**		**Método 2**		**Error (%)**	
	(Mbps)	**LI**	**LS**	**LI**	**LS**	**LI**	**LS**
Switch	20	85	115	85	115	0	0
	30	85	115	85	115	0	0
	40	85	115	85	115	0	0

Tabla 7.2: Tamaño del *buffer* del *switch* en número de paquetes para diferentes tasas de entrada (LI: Límite inferior y LS: Límite superior), utilizando paquetes de $1500\ bytes$.

7.1.2 Escenario WiFi

En este caso, también se implementa un entorno controlado de laboratorio que permite determinar el modelo del *buffer* de un punto de acceso WiFi (AP 1). Para estas pruebas se ha seleccionado un dispositivo $Linksys\ WAP54G$ y una tasa de prueba de $40\ Mbps$, ya que con este valor se minimizan las posibles variaciones en los resultados como se mencionó anteriormente. Además, se utilizan tres tamaños de paquetes diferentes (1500, 800 y 200 $bytes$) al igual que se hizo para el caso

anterior. El escenario propuesto para las pruebas se muestra en la Figura 7.4. El punto de congestión se producirá en el enlace de radio entre los dos puntos de acceso, debido a que se configura para diversas capacidades (1, 2, 5,5, 11, 24 y 54 $Mbps$) en cada prueba, lo cual permite alcanzar una tasa que siempre es menor a los 40 $Mbps$ de entrada.

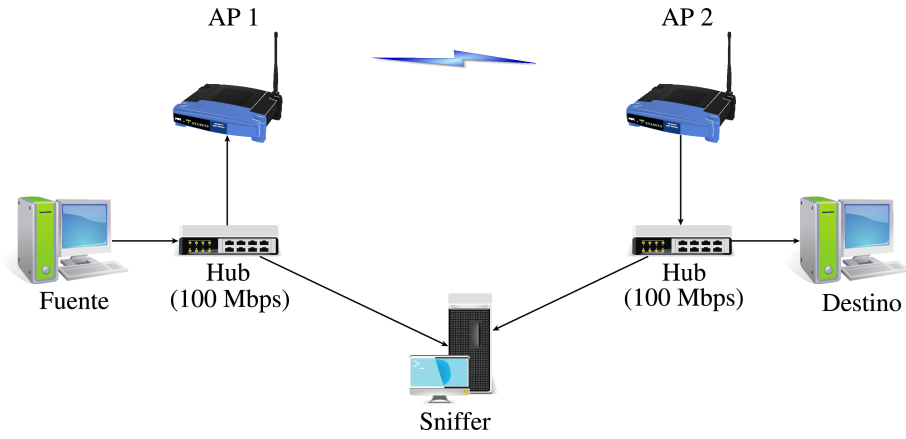

Figura 7.4: Escenario para determinar el modelo del *buffer* de un punto de acceso WiFi.

En este escenario, se presentan grandes variaciones en la tasa de transmisión de la red WiFi siendo mayores cuando el acceso inalámbrico se configura para las tasas de velocidad superiores. Por lo tanto, esto genera un error significativo en los cálculos realizados cuando los puntos de acceso se configuran para las tasas más altas. La razón de esto es que en los puntos de acceso WiFi, la velocidad en una misma prueba va cambiando de valor a lo largo del tiempo desde la velocidad más alta a la más baja dependiendo de las condiciones del canal radio. Para corroborar este hecho, en la Tabla 7.3 se muestran los resultados de diversas mediciones de la tasa de salida para este escenario utilizando paquetes de 1500 $bytes$. De forma adicional se puede observar que dicha tasa también varía con el tamaño de los paquetes a causa de la eficiencia, cuyo efecto es mucho más acentuado que en Ethernet; por lo que también se hace aconsejable realizar pruebas con tamaño máximo de paquetes (1500 $bytes$) en las que hay menos variaciones de tasa de transmisión

inalámbrica. Como se ha mencionado anteriormente, siempre será necesario realizar pruebas con otros valores de tamaños de paquetes para determinar si el *buffer* establece su tamaño en número de paquetes o en $bytes$. Dada la variabilidad de los resultados las conclusiones para este análisis serán menos fiables.

Tasa (Mbps)	**54**	**24**	**11**	**5,5**	**2**	**1**
Min. (Mbps)	10,88	13,7	5,75	2,29	1,24	0,65
Max. (Mbps)	28,36	16,84	6	3,13	1,41	0,65

Tabla 7.3: Variaciones observadas en la tasa de salida del *buffer* cuando el punto de acceso se configura para diferentes tasas, utilizando paquetes de 1500 $bytes$.

7.1.2.1 **Resultados mediante el método 1**

Analizando los resultados obtenidos para todas las pruebas realizadas, se puede decir que en todos los casos el tamaño del *buffer* y sus límites fueron los mismos. En la Tabla 7.4 se presentan los resultados correspondientes al tráfico de paquetes de 1500 $bytes$. Con los resultados obtenidos en las pruebas con los otros valores de tamaños de paquetes, se determina que la capacidad del *buffer* se establece en número de paquetes y tiene un valor de 55 paquetes.

7.1.2.2 **Resultados mediante el método 2**

Los resultados obtenidos mediante el método 2 para los anchos de banda más bajos de WiFi fueron idénticos que los obtenidos aplicando el método 1. Sin embargo, los resultados entre los métodos 1 y 2 difieren cuando se configuran capacidades mayores entre los puntos de acceso porque se generan más variaciones en las tasas de salida.

En la Tabla 7.5 se muestra una comparación de los métodos propuestos en la estimación del tamaño del *buffer*. Dicha tabla presenta los valores correspondientes a las pruebas realizadas con paquetes de 1500 $bytes$, ya que con las pruebas con tamaños de paquetes más pequeños el error es mayor como se aprecia en la Tabla 7.6. Sin embargo, los resultados obtenidos para estos tamaños más pequeños son

Dispositivo	Tasa	Método 1	
	(Mbps)	LI	LS
AP 1	1	30	55
	2	30	55
	5,5	30	55
	11	30	55
	24	30	55
	54	30	55

Tabla 7.4: Tamaño del *buffer* en número de paquetes (LI: Límite inferior y LS: Límite superior) cuando el punto de acceso se configura para diferentes tasas.

bastante claros y permiten concluir que la capacidad del *buffer* se mide en paquetes y no en $bytes$.

Dispositivo	Tasa	Método 1		Método 2		Error (%)	
	(Mbps)	LI	LS	LI	LS	LI	LS
AP 1	1	30	55	30	55	0	0
	2	30	55	30	55	0	0
	5,5	30	55	30	55	0	0
	11	30	55	32	54	6,67	1,81
	24	30	55	33	52	10	5,45
	54	30	55	34	58	13,33	5,45

Tabla 7.5: Tamaño del *buffer* en número de paquetes, de un punto de acceso Wi-Fi, cuando éste se configura para diferentes tasas (LI: Límite inferior y LS: Límite superior).

Dispositivo	Tasa	200 *bytes*		800 *bytes*		1500 *bytes*	
	(Mbps)	LI	LS	LI	LS	LI	LS
AP 1	1	30	55	30	55	30	55
	2	29	56	30	55	30	55
	5,5	28	52	25	52	30	55
	11	28	53	29	55	32	54
	24	25	52	27	51	33	52
	54	24	50	32	57	34	58

Tabla 7.6: Estimación del tamaño del *buffer* de un punto de acceso WiFi en número de paquetes (LI: Límite inferior y LS: Límite superior), mediante el método 2, cuando éste se configura para diferentes tasas y utilizando tráfico de prueba con tres tamaños de paquetes.

7.2 Acceso remoto

Para las pruebas que se presentan a continuación, se han implementado los mismos escenarios controlados de laboratorio que se plantearon en el caso con acceso físico (ver Figura 7.1 y 7.4) con las mismas tasas de entrada de prueba y los mismos tamaños de paquetes, esto con la finalidad de comparar los resultados. La principal diferencia es que en este caso las medidas en vez de obtenerse por medio de una máquina "*sniffer*", se obtienen en el destino de la comunicación.

Para el análisis de los datos mediante el método de acceso remoto se han realizado nuevos *script* para procesar la captura realizada ya que en este caso sólo se tiene la captura en el *host* de destino, siendo diferente la estimación de las tasas de entrada y salida, la ocupación, los límites y el tamaño del *buffer* al presentado en el caso con acceso físico. Para ésto se han utilizado los procedimientos y las ecuaciones respectivas que se describieron en el capítulo 5. En este caso, también se realizan las estimaciones periódicas de las tasas de entrada y salida del *buffer* en los instantes en los cuales se detectan pérdidas de paquetes con el fin de obtener diversas estimaciones del tamaño del *buffer* a lo largo de una misma prueba. Además, se generan los datos correspondientes en ficheros con formato CSV que

luego son manipulados por software de procesamiento matemático para la correspondiente presentación de los resultados.

7.2.1 Escenario ethernet

Como se puede observar en la Tabla 7.7, la precisión de la estimación remota es muy alta, esto se debe a que las tasas de entrada y salida del *buffer* son muy estables. Además, el incremento de la tasa de prueba genera una disminución en el porcentaje de error. En general, los resultados son los mismos que los obtenidos con el método de acceso físico. En dicha tabla se muestran los resultados para las pruebas con tráfico de prueba de paquetes de $1500\ bytes$ ya que son los más precisos.

Dispositivo	**Tasa**	**Método 1**		**Método 2**		**Error (%)**	
	(Mbps)	**LI**	**LS**	**LI**	**LS**	**LI**	**LS**
Switch	20	85	115	84	113	1,1	1,7
	30	85	115	85	116	0	0,8
	40	85	115	85	115	0	0

Tabla 7.7: Tamaño del *buffer* del *switch* en número de paquetes para diferentes tasas de entrada (LI: Límite inferior y LS: Límite superior), mediante el método de estimación para acceso remoto.

Por otro lado, la Tabla 7.8 muestra un resumen de los resultados obtenidos para las pruebas con diferentes tamaños de paquetes según la tasa de prueba utilizada, los cuales han sido útiles para determinar que el *buffer* se mide en número de paquetes y no en $bytes$.

7.2.2 Escenario WiFi

Los resultados de las pruebas muestran que la estimación remota del tamaño del *buffer* es menos precisa que con acceso físico, esto se debe a que las tasas no son estables, principalmente cuando la capacidad de transmisión se configura a

Dispositivo	Paquete	20 *Mbps*		30 *Mbps*		40 *Mbps*	
	(*Bytes*)	LI	LS	LI	LS	LI	LS
Switch	200	88	117	86	116	86	116
	800	82	112	87	116	85	116
	1500	84	113	85	116	85	115

Tabla 7.8: Estimación remota del tamaño del *buffer* del *switch* en número de paquetes, para diversos tráficos de prueba y con diferentes tamaños de paquete (LI: Límite inferior y LS: Límite superior).

los valores más altos. Este comportamiento en entornos WiFi es común, ya que la capacidad cambia en función de las características del canal.

En la Tabla 7.9 se muestra la comparación de los resultados obtenidos para el método con acceso físico y con acceso remoto. Los resultados presentados corresponden a flujos de paquetes con tamaños de $1500\ bytes$ y una tasa de prueba de $40\ Mbps$ ya que son los más precisos. En este caso, algunos resultados difieren de los obtenidos en la Tabla 7.5, ya que como se ha mencionado con anterioridad, las condiciones del canal pueden cambiar de una prueba a otra, y por lo tanto las tasas y la estimación del tamaño del *buffer* también varían. De nuevo, en la Tabla 7.10 se presentan los resultados obtenidos para los otros valores de tamaños de paquetes que permiten determinar que la capacidad del *buffer* se mide en paquetes.

7.3 *Buffer* concatenados

A continuación, se presentan los resultados de validación correspondientes al método de análisis para determinar *buffer* concatenados. En dicho escenario se analiza el caso de dos dispositivos de red concatenados en un enlace entre dos puntos terminales.

Al igual que en los dos casos anteriores se han elaborado una serie de *script* utilizando el *shell* de Linux para automatizar tareas: procesar las capturas realizadas, determinar las tasas de entrada y salida, la ocupación, los límites y el tamaño de cada uno de los *buffer*. En este caso los *script* difieren de los anteriores ya que

Dispositivo	Tasa	Método 1		Método 2		Error (%)	
	(Mbps)	LI	LS	LI	LS	LI	LS
AP 1	1	30	55	30	55	0	0
	2	30	55	30	55	0	0
	5,5	30	55	30	55	0	0
	11	30	55	32	53	6,67	3,63
	24	30	55	33	52	10	5,45
	54	30	55	36	59	20	7,27

Tabla 7.9: Tamaño del *buffer* en número de paquetes, de un punto de acceso Wi-Fi, cuando éste se configura para diferentes tasas (LI: Límite inferior y LS: Límite superior), mediante el método de estimación para acceso remoto.

Dispositivo	Tasa	200 *bytes*		800 *bytes*		1500 *bytes*	
	(Mbps)	LI	LS	LI	LS	LI	LS
AP 1	1	29	56	30	55	30	55
	2	28	56	30	55	30	55
	5,5	29	51	25	53	30	55
	11	26	52	29	54	32	53
	24	25	52	27	51	33	52
	54	23	50	32	57	36	59

Tabla 7.10: Estimación remota del tamaño del *buffer* de un punto de acceso WiFi en número de paquetes (LI: Límite inferior y LS: Límite superior), mediante el método 2, cuando éste se configura para diferentes tasas y utilizando tráfico de prueba con tres tamaños de paquetes.

el método de estimación de los parámetros citados anteriormente es diferente. Para ésto se han utilizado los procedimientos y las ecuaciones respectivas que se describieron en el capítulo 6. También, se han realizado estimaciones periódicas del tamaño del *buffer* y sus tasas de entrada y salida cuando se detecta la pérdida de paquetes. Además, se han elaborado *scritp* que generan datos en ficheros con formato CSV para luego ser manipulados por software de procesamiento matemático para la correspondiente presentación de los resultados.

7.3.1 Escenario propuesto

Para el análisis de casos en los cuales se requiere determinar dispositivos concatenados, se ha implementado un escenario controlado de laboratorio utilizando los equipos estudiados anteriormente, ya que se conocen los parámetros correspondientes a cada *buffer* y se puede comparar la precisión del nuevo método con respecto a los otros. El escenario propuesto se muestra en la Figura 7.5, en el cual se han configurado adecuadamente las capacidades de transmisión del *switch* y los puntos de acceso para crear un punto crítico que cumpla con una relación de velocidades ($R_1 > R_2 > R_3$) que permita observar el efecto de ambos *buffer* en congestión. En este sentido, se ha seleccionado una capacidad máxima a la entrada del *switch* de 100 $Mbps$ mientras que la de salida (entrada al AP 1) se ajusta a 10 $Mbps$ y el enlace entre los dos puntos de acceso se configura a 11 $Mbps$, por lo cual, dicho enlace inalámbrico siempre alcanzará una tasa por debajo de 10 $Mbps$.

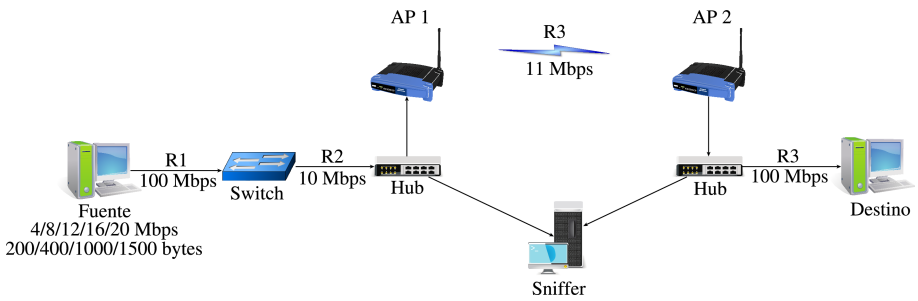

Figura 7.5: Escenario para determinar el modelo de dos *buffer* concatenados.

Siguiendo el mismo procedimiento utilizado en los casos anteriores, se envía una ráfaga de prueba desde el *host* fuente hasta el destino con el objetivo de saturar el SUT. Para obtener un mapa de pérdidas que permita un análisis preciso de los patrones del comportamiento de las pérdidas de cada *buffer* y que sea lo menos intrusivo posible, se han seleccionado pruebas con diferentes tasas de entrada (4, 8, 12, 16 y 20 $Mbps$) y se repiten para diversos tamaños de paquetes (200, 400, 1000 y 1500 $bytes$).

Para estimar el tamaño de los *buffer* se han seleccionado las pruebas realizadas con tasas de 8 y 20 $Mbps$, las cuales se repiten para los tamaños de paquetes: 200, 400, 1000 y 1500 $bytes$, con el objetivo de determinar si la capacidad total de los *buffer* está dada en número de paquetes o en $bytes$. Las pruebas con otras tasas de entrada se utilizan para la estimación de la tasa intermedia, la ubicación de los *buffer* y para ampliar los análisis que se detallan en las siguientes secciones, por tanto, en el tráfico de dichas pruebas sólo se utilizan paquetes con tamaño de 1500 $bytes$.

Se debe aclarar que a pesar de que la tasa de entrada se varía de una prueba a otra, la tasa intermedia o de entrada al punto de acceso AP 1 (que corresponde a la tasa de salida del *switch*) alcanzará como máximo 10 $Mbps$. Esta condición produce que el punto de acceso tenga una tasa de entrada constante para todos los casos en los que la tasa de entrada del *switch* es igual o mayor a 10 $Mbps$.

7.3.2 Análisis mediante mapas de pérdidas

El mapa de pérdidas para las tasas de entrada de 8, 12, 16 y 20 $Mbps$ se muestra en la Figura 7.6. En dicha figura se ha omitido el mapa correspondiente 4 $Mbps$ ya que dicha tasa no es capaz de inducir una congestión en ninguno de los *buffer* en el camino de red. Por otro lado, cuando la tasa de entrada es menor a 10 $Mbps$ (la máxima tasa de salida del *switch*) solamente se observa el efecto de un *buffer* que corresponde al *buffer* del punto de acceso WiFi ya que el *switch* no se puede congestionar a esa tasa.

Los patrones se pueden determinar por medio de las agrupaciones de pérdidas de paquetes que coinciden aproximadamente en un mismo valor, como se ex-

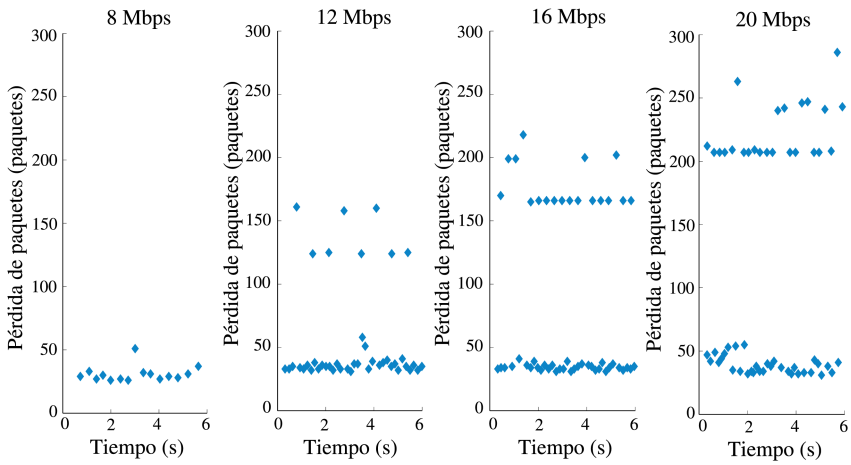

Figura 7.6: Mapa de pérdidas que muestra los diferentes patrones de pérdida para diversos anchos de banda.

plicó en el capítulo 6. En la Figura 7.6 se puede observar que para las tasas de prueba más grandes (las cuales saturan todos los dispositivos en el enlace), se observan tres patrones de pérdidas distintos, lo que inicialmente podría suponerse como tres *buffer* diferentes. Sin embargo, al analizar la cantidad de paquetes en cada ráfaga de pérdidas se comprueba que el valor del tercer patrón coincide con la suma del primero y el segundo. Por lo tanto, se puede afirmar con una probabilidad bastante alta que en el enlace sólo se han detectado dos *buffer* (el del *switch* y el del punto de acceso) y que el tercer patrón corresponde al efecto del instante en el cual los dos *buffer* descartan paquetes al mismo tiempo.

En cuanto a los tres patrones, el primero de ellos se encuentra alrededor de los 45 paquetes, el cual se mantiene en el mismo valor para todas las pruebas ya que corresponde al efecto del *buffer* del punto de acceso y su tasa de entrada se ha establecido a un máximo de 10 $Mbps$ debido al *switch*. El segundo patrón de pérdidas de paquetes se encuentra alrededor de 125, 160 y 205 paquetes para tasas de entrada al SUT de 12, 16 y 20 $Mbps$ respectivamente. En este caso, el incremento de la pérdida de paquetes es coherente con la relación entre las tasas de entrada y salida del *switch*, ya que la tasa de salida se mantiene en 10 $Mbps$ mientras que se incrementa la tasa de entrada, por lo tanto, la pérdida de paquetes

debe crecer. Por último, el tercer patrón presenta un fenómeno interesante ya que el valor de la pérdida de paquetes corresponde a la suma de la pérdida de paquetes del *switch* y el punto de acceso, como se ha mencionado anteriormente.

Por otro lado, cuando se tienen diversos dispositivos en un punto crítico de red es posible analizar el efecto de cada *buffer* de forma aislada si se ajusta la tasa de entrada al SUT de manera que sólo uno de ellos se congestione. La prueba realizada con una tasa de entrada de $8\ Mbps$ es un ejemplo de este tipo de análisis. La Figura 7.7 muestra los resultados de la pérdida de paquetes para diversos tamaños de paquetes cuando sólo el *buffer* del punto de acceso descarta paquetes. En dicha figura se observa que la dispersión entre los resultados es más alta para los paquetes más pequeños, esto se debe a las variaciones en la tasa del enlace WiFi y que el tamaño de los paquetes afecta al comportamiento de la pérdida de paquetes porque modifica el ancho de banda utilizado, como se ha comentado en casos anteriores.

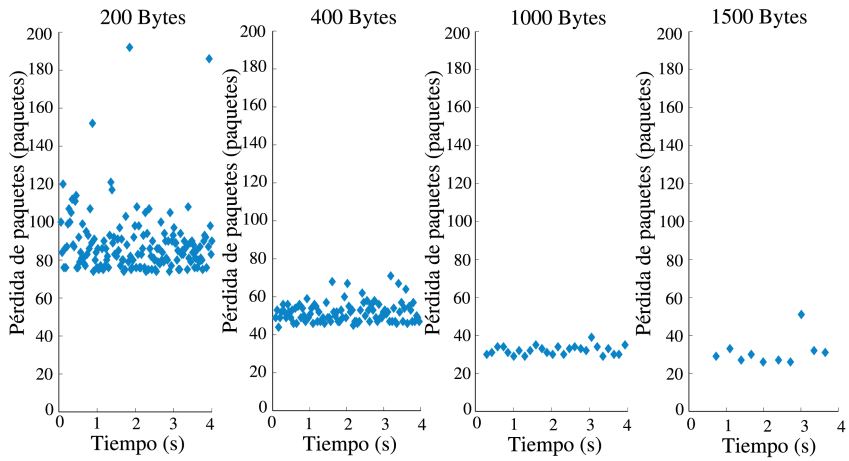

Figura 7.7: Mapa de pérdidas que muestra los diferentes patrones de pérdida para diversos tamaños de paquetes para un ancho de banda de $8\ Mbps$.

Además, en cada prueba realizada para una tasa de $8\ Mbps$ (ver Figura 7.7), la cantidad de paquetes perdidos está aproximadamente en el mismo valor debido a la relación entre el tamaño del *buffer* y las tasas de entrada y salida del *buffer* del punto de acceso. En tecnologías inalámbricas como WiFi, la tasa de salida se

incrementa cuando el tamaño de los paquetes es más grande, entonces el valor de la pérdida de paquetes se ve afectada por el tamaño de los paquetes y decrece cuando los paquetes son más grandes. Además, la pérdida de paquetes tiene menos dispersión cuando los paquetes son más grandes.

Para analizar los patrones de la pérdida de paquetes de ambos dispositivos al mismo tiempo, se debe utilizar una traza que asegure la congestión en los nodos, para este caso se ha realizado una prueba a 20 $Mbps$. La Figura 7.8 muestra los resultados de dicha prueba para diferentes tamaños de paquetes. De nuevo, se observan tres patrones de pérdida de paquetes, los cuales presentan cierta dispersión en cuanto a sus valores debido a las variaciones de la tasa de salida de los *buffer* que dependen del tamaño de los paquetes.

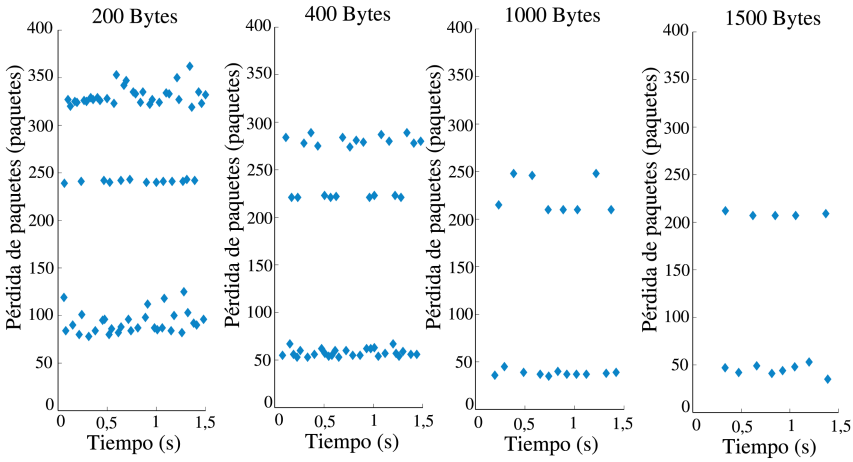

Figura 7.8: Mapa de pérdidas que muestra los diferentes patrones de pérdida para diversos tamaños de paquetes para un ancho de banda de 20 $Mbps$.

7.3.3 Análisis de mapas de pérdidas para estimar tasas

La estimación de las tasas de entrada, intermedia y de salida de los *buffer* (R_1, R_2 y R_3 respectivamente, ver Figura 7.5) se realiza para cada una de las pruebas correspondientes a los tráficos de prueba de 8 y 20 $Mbps$, ya que son las que se utilizarán para estimar los tamaños de los *buffer*. En el primer caso (8 $Mbps$), la

tasa de prueba se encuentra por debajo de la capacidad de salida del *switch* por lo cual no puede congestionarse, produciendo que la tasa de entrada sea igual a la tasa intermedia ($R_1 = R_2$). En este caso, el AP 1 es el único dispositivo en el cual la tasa de entrada es mayor que la de salida ($R_2 > R_3$) y por lo tanto el único *buffer* detectable. En el segundo caso (20 $Mbps$), la tasa de prueba es superior a la capacidad de salida del *switch* y del AP 1 pudiéndose detectar los *buffer* de cada dispositivo.

Para facilitar en la medida de lo posible el procesamiento de los datos y la repetibilidad, se han elaborado *script* que realizan el cálculo de las tasas mencionadas una vez que la prueba ha terminado. Las estimaciones se realizan cuando se detecta la pérdida de paquetes, es decir, cuando los números de secuencia de dos paquetes adyacentes no son consecutivos. De la traza capturada en el *host* de destino se determina el tamaño de los paquetes utilizados para la prueba y el tiempo transcurrido entre la pérdida de paquetes anterior y la actual, además, los números de secuencia de los paquetes capturados permiten determinar la cantidad de paquetes que se pierden en cada instante. Con esto se pueden determinar las tasas de salida y entrada al SUT por medio de las ecuaciones 6.1 y 6.2 respectivamente.

Por otro lado, en casos anteriores se ha comentado que la tasa que alcanza el tráfico de prueba puede variar dentro de una misma prueba incluso de una prueba a otra, donde los entornos WiFi son más susceptibles a este fenómeno. Además, la tasa de pérdida de paquetes se encuentra ligada a la velocidad de llenado de los *buffer* y por consiguiente a las tasas de entrada y salida. Esto produce que en ciertas ocasiones la cantidad de paquetes contenidos en cada ráfaga de paquetes perdidos pueda variar durante una misma prueba si la tasa de transmisión cambia.

En la Figura 7.9 se muestran dos ampliaciones diferentes del mapa de pérdidas correspondiente a la prueba de 8 $Mbps$, en la primera ampliación (a) se presentan tres ráfagas con cantidades de paquetes perdidos similares (29, 33 y 27 paquetes), por lo que la tasa es relativamente similar durante todo ese período. Sin embargo, en la ampliación (b) solamente dos valores son parecidos (26 y 32 paquetes) y el tercero prácticamente duplica la cantidad de paquetes perdidos (51 paquetes), por lo que dicho valor no es representativo de la tasa media de la prueba y es conveniente no tenerlo en cuenta para las estimaciones.

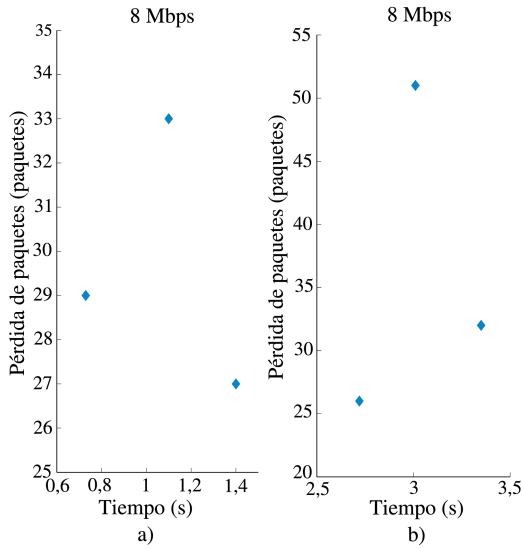

Figura 7.9: Mapa de pérdida de paquetes para el ejemplo con una tasa de prueba de $8\ Mbps$.

La Figura 7.10 muestra dos ampliaciones diferentes del mapa de pérdidas correspondiente a la prueba de $12\ Mbps$ (se debe recordar que la tasa de esta prueba es suficientemente alta para congestionar los dos *buffer* existentes) en la cual existe una cantidad de agrupaciones de ráfagas mayor que en el caso anterior. En la primera ampliación (a) se muestran algunas de las ráfagas de paquetes que han sido descartados por uno de los *buffer*, ahí se nota que dos de las ráfagas se encuentran por encima de 50 paquetes y van a producir una estimación de las tasas que no es fiable. En la otra ampliación (b), se presentan ráfagas de paquetes que se han descartado por el otro *buffer* (cerca de los 125 paquetes) y al caso en el cual los dos *buffer* descartan paquetes al mismo tiempo (al rededor de los 160 paquetes), en ambos casos los dos patrones de pérdidas son bastante estables y por lo tanto las estimaciones derivadas de estos datos serán más fiables.

En general, los resultados obtenidos por medio de las capturas para la prueba de $8\ Mbps$ permiten afirmar que la tasa final de salida (R_3) se encuentra aproximadamente a $7\ Mbps$, mientras que la tasa de entrada ($R_1 = R_2$) alcanzó el valor esperado de $8\ Mbps$. Para el caso de la prueba de $20\ Mbps$, se pudo determinar

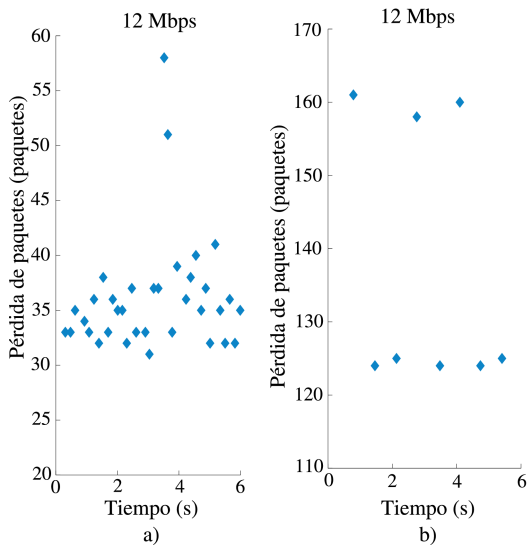

Figura 7.10: Mapa de pérdida de paquetes para el ejemplo con una tasa de prueba de $12\ Mbps$.

la misma tasa de salida ($R_3 = 7\ Mbps$) y la tasa de entrada es de $R_1 = 20\ Mbps$. Sin embargo, para la tasa intermedia se tienen dos posibles valores (debido a los dos patrones observados) según la ecuación 6.3. A continuación se muestra, a modo de ejemplo, dos períodos diferentes (de la traza correspondiente) en los cuales se pueden estimar dichas tasas.

$$R_A = \frac{P_L}{T_{rx}} \times (N + N_A) = \frac{1500 \times 8}{0{,}494} \times (210 + 205) \approx 10\ Mbps$$

$$R_B = \frac{P_L}{T_{rx}} \times (N + N_B) = \frac{1500 \times 8}{0{,}128} \times (150 + 45) \approx 18\ Mbps$$

Para determinar cual de las tasas intermedias (R_A o R_B) es la correcta se realiza una nueva prueba con una tasa de generación de paquetes de $16\ Mbps$ (la prueba con tasa de $12\ Mbps$ también es válida para este caso), la cual se encuentra entre los dos posibles valores. El mapa de pérdidas correspondiente a dicha prueba se muestra en la Figura 7.6, la cual muestra dos patrones de pérdidas de paquetes, por lo tanto, el valor correcto de la tasa intermedia es $R_A = 10\ Mbps$.

7.3.4 Análisis de mapas de pérdidas para inferir ubicaciones

Los mapas de pérdidas también son útiles para determinar la ubicación de los *buffer*. Mediante el análisis realizado para la figura 7.6, cuando las tasas de prueba son mayores a 10 $Mbps$, se pudo detectar la existencia de dos *buffer* diferentes. Para el caso en el cual la tasa de prueba es de 20 $Mbps$, el mapa correspondiente muestra que uno de los *buffer* descarta ráfagas de aproximadamente 50 paquetes, mientras que en el otro dicha cantidad asciende a un valor que supera los 200 paquetes. El *buffer* que descarta la mayor cantidad de paquetes por ráfaga corresponde al que tiene una tasa de llenado (R_{fill}) más grande, es decir, al *buffer* en el cual la diferencia de las tasas de entrada y salida sea mayor ($R_{in} >> R_{out}$).

En la sección anterior se estimaron las tasas de entrada y salida de cada *buffer* detectado, siendo sus correspondientes valores: $R_1 = 20\ Mbps$, $R_2 = 10\ Mbps$ y $R_3 = 7\ Mbps$, por tanto, existe un dispositivo entre R_1 y R_2 con una tasa de llenado $R_{fill_1} = 10\ Mbps$ y otro entre R_2 y R_3 con una tasa de llenado $R_{fill_2} = 3\ Mbps$. Mediante este análisis se puede deducir que el *buffer* que se encuentra físicamente de primero (*Buffer* 1) en el camino de red tiene una tasa de llenado de 10 $Mbps$ y descarta ráfagas de 200 paquetes y el que se encuentra después (*Buffer* 2) tiene una tasa de llenado de 3 $Mbps$ y descarta ráfagas de 50 paquetes.

7.3.5 La estimación del tamaño de los *buffer*

Los parámetros asociados a cada *buffer*, así como el tamaño de los *buffer* del *switch* y del punto de acceso se pudieron determinar mediante los *script* que se describieron anteriormente, en los cuales se utilizó la metodología propuesta en el capítulo 6, analizando un *buffer* a la vez de acuerdo a cada patrón y eliminando el efecto del otro. A modo de ejemplo, a continuación se presenta el cálculo de dicha estimación para el caso en el que se observa la primera pérdida de paquetes para cada *buffer*.

Utilizando la traza capturada se debe determinar el último paquete recibido antes de la primera pérdida para cada uno de los *buffer*. De la traza se observa que dichos paquetes son el número 224 y el 424 para el *Buffer* 1 y el *Buffer* 2

respectivamente, con esta información se estima el tamaño de cada *buffer* mediante la ecuación 6.4.

$$L_{Buffer_1} = n_1 \times \left(1 - \frac{R_B}{R_{in}}\right) = 224 \times \left(1 - \frac{10}{20}\right) = 112\ paquetes$$

$$L_{Buffer_2} = n_2 \times \left(1 - \frac{R_{out}}{R_B}\right) = 424 \times \left(1 - \frac{7}{8}\right) = 53\ paquetes$$

Por otro lado, en ambos casos se observó el comportamiento del *buffer* con límites superior e inferior con un tamaño máximo de aproximadamente 115 paquetes para el *switch* y valores muy cercanos a 55 paquetes para el punto de acceso como se muestra en la Tabla 7.11. Dichos resultados concuerdan con los obtenidos al analizar cada dispositivo de manera aislada con los métodos propuestos en el capítulo 5 y demostrados al inicio de este capítulo. Al igual que en casos anteriores los paquetes de tamaños más grande presentan una mayor precisión en la estimación del tamaño de los *buffer*, esto se debe a que dichos paquetes han demostrado ser mejores para determinar las tasas necesarias en la estimación del tamaño de los *buffer*.

Dispositivo	**Tasa**	**200 *bytes***		**400 *bytes***		**1000 *bytes***		**1500 *bytes***	
	(Mbps)	**LI**	**LS**	**LI**	**LS**	**LI**	**LS**	**LI**	**LS**
Switch	8	–	–	–	–	–	–	–	–
	20	47	65	71	111	89	116	82	112
AP 1	8	–	–	23	48	33	56	30	55
	20	–	–	16	35	22	45	29	53

Tabla 7.11: Estimación del tamaño de cada *buffer* para dos dispositivos concatenados (LI: Límite inferior y LS: Límite superior) con diferentes tasas de prueba y tamaños de paquetes.

Parte III

El impacto del tamaño del *buffer* en la QoS

Algunos de los contenidos de la parte III se han publicado en:

***"The Influence of the Buffer in Packet Loss for Competing Multimedia and Bursty Traffic"**, Proc. International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS, Julio 2013, ISBN 1-56555-352-7, cuyos autores son: Luis Sequeira, Julián Fernández-Navajas, Luis Casadesus, Jose Saldana, Idelkys Quintana, José Ruiz-Mas.*

***"The Effect of the Buffer Size in QoS for Multimedia and bursty Traffic: When an Upgrade Becomes a Downgrade"**, KSII Transactions on Internet and Information Systems, Vol. 2014, cuyos autores son: Luis Sequeira, Julián Fernández-Navajas, Jose Saldana.*

La irracionalidad de una cosa no es un argumento en contra de su existencia, sino más bien una condición de la misma.

Friedrich Wilhelm Nietzsche

CAPÍTULO

8

Análisis de QoS en servicios de tráfico a ráfagas

En este capítulo se presenta un análisis de las características de los *buffer* (especialmente su tamaño y la pérdida de paquetes) en los dispositivos de acceso. En particular se estudia cómo estas características pueden afectar a la calidad de las aplicaciones multimedia cuando éstas generan tráfico a ráfagas en la red local.

En primer lugar, se muestra un escenario con flujos de tráfico a ráfagas en el cual se ha escogido un servicio de videovigilancia con cámaras IP para una red de PYMES, en dicho escenario los flujos comparten el mismo enlace de acceso. Además, se presenta cómo el aumento de capacidad de la red interna podría causar el desbordamiento de los *buffer* y producir una cantidad significativa de pérdida de paquetes que podrían deteriorar la QoS.

En segundo lugar, este capítulo muestra que la naturaleza a ráfagas de aplicaciones como la videovigilancia puede perjudicar su QoS, especialmente cuando cierto número de ráfagas se solapan, ya que las ráfagas de pérdidas de paquetes en un *buffer* no se producen únicamente por aplicaciones que generan tráfico a ráfagas, sino, que también pueden ser causadas por el solapamiento de diferentes flujos en un enlace sensible. Para abordar este tema, se han planteado una serie de pruebas con el objetivo de caracterizar el problema que se puede presentar debido al incremento de la demanda de capacidad en este tipo de escenarios. En algunos casos, especialmente cuando se presentan aplicaciones que generan tráfico a ráfagas, el incremento de la capacidad de la red podría conducir a un deterioro de la

calidad. Como conclusión se muestra que en estos casos la principal causa de la degradación de la calidad, en caso de no sobrepasar la capacidad del enlace, se debe al desbordamiento del *buffer*, y que depende de la relación entre el ancho de banda de la red interna y la de acceso.

8.1 Escenario de red propuesto

Como se ha mencionado con anterioridad, se ha seleccionado un servicio de videovigilancia que utiliza cámaras IP, las cuales se pueden acceder mediante un navegador de Internet convencional. El escenario utilizado para las pruebas se muestra en la Figura 8.1, en la cual un número de comunicaciones de videovigilancia comparte el mismo enlace de acceso a Internet hacia el centro de vigilancia. El principal objetivo de las pruebas es determinar la tasa de pérdida de paquetes en el tráfico a ráfagas combinado para diferentes tamaños de *buffer*, y observar los diferentes resultados cuando la capacidad de los enlaces entre las cámaras y el dispositivo de acceso a Internet cambia de 10 $Mbps$ a 100 $Mbps$.

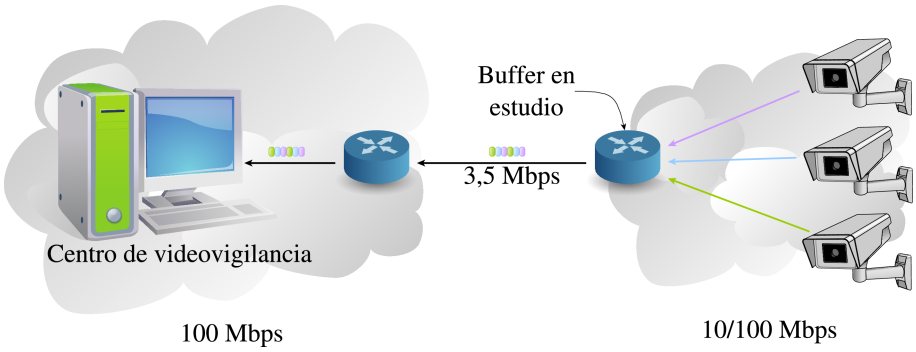

Figura 8.1: Escenario para las pruebas de uno, dos y tres flujos de datos de cámaras IP.

La prueba se repite utilizando tráfico de 1, 2 y 3 cámaras, donde cada una transmite a una tasa media de 1 $Mbps$. La capacidad del enlace de acceso se limita a 3,5 $Mbps$, que es un valor razonable si se compara con un enlace de subida de las actuales redes de acceso DSL. Dicho valor se ha seleccionado con el fin de

establecer el ancho de banda ofrecido al $85\,\%$ de la capacidad del enlace cuando las tres cámaras transmiten al mismo tiempo. Además, estableciendo la utilización del enlace en dicho valor, se puede asegurar que la pérdida de paquetes es causada por la relación entre las características del tráfico y el comportamiento del *buffer* del *router* y no por la escasez del ancho de banda.

8.2 Tráfico utilizado

Con la finalidad de desarrollar las pruebas descritas anteriormente, se han utilizado trazas reales de aplicaciones de videovigilancia, las cuales fueron capturadas en escenarios reales, para luego ser generadas en NS-2, utilizando los mismos tamaños de paquetes y tiempos entre paquetes. La metodología utilizada para la captura de tráfico se ilustra en la Figura 8.2, en la cual se incluye un *sniffer* en la mejor ubicación para que no degrade el rendimiento de las aplicaciones [ZFP12].

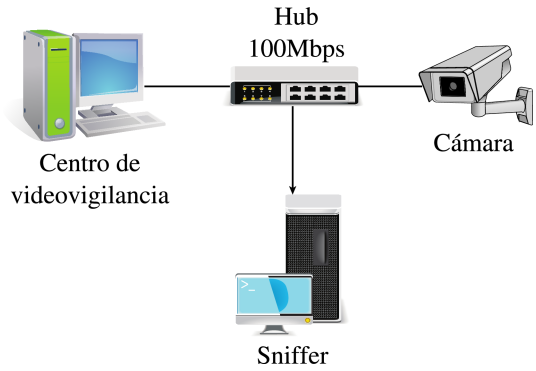

Figura 8.2: Escenario para la captura del tráfico para un sistema de videovigilancia.

Las trazas del tráfico de videovigilancia se obtuvieron utilizando una cámara IP bien conocida en el mercado (AXIS 2120). Este tipo de tráfico es particularmente a ráfagas, en la Tabla 8.1 se muestra la relación entre el nivel de compresión de video y la cantidad de paquetes por ráfaga para dos diferentes resoluciones cuando el ancho de banda de la cámara se establece en $1\ Mbps$. Para todas las pruebas realizadas se han seleccionado trazas con una resolución de $704 \times 576\ px$ y una

compresión de 32 $Kbytes$. El tiempo medio entre las ráfagas es de 0,278 s $\pm$ 0,06 s, la cantidad de paquetes por ráfaga es de 26 y el tamaño de los paquetes es de 1500 $bytes$.

Resolución	Nivel de compresión	Cantidad de paquetes
704×576 $pixeles$	50 $Kbytes$	41
	32 $Kbytes$	26
	16 $Kbytes$	10
352×288 $pixeles$	13 $Kbytes$	9
	4 $Kbytes$	3

Tabla 8.1: Cantidad de paquetes observados por ráfaga, dependiendo del nivel de compresión de la cámara.

8.3 Análisis de pérdida de paquetes

Como ocurre en un escenario real los flujos no se inician al mismo tiempo; por ello, para las simulaciones se ha incluido un período de inicio en el cual todos los flujos comienzan de manera aleatoria. Inicialmente se realiza una serie de pruebas preliminares con 100, 50 y 40 realizaciones, cuyo objetivo es determinar la cantidad de repeticiones necesarias para las pruebas con un adecuado consumo de recursos. Los valores medios de la pérdida de paquetes se calcularon para cada caso considerando su respectiva cantidad de repeticiones (100, 50 y 40), para dichas pruebas los valores fueron muy similares en los tres casos. Por este motivo, cada prueba se repite 40 veces y los resultados que se muestran son los correspondientes a los valores medios de dichos resultados. Además, cada valor incluye un intervalo de confianza del 95 % que se muestra en los gráficos. La duración de cada prueba es de 60 s, tiempo que asegura un patrón de tráfico estable para los flujos.

Los resultados de la pérdida de paquetes cuando se ha utilizado el tráfico de una sola cámara (lo que equivale aproximadamente a un 29 % de la utilización del enlace) es cero para todas las pruebas con diferentes tamaños de *buffer* (desde 30

hasta 65 paquetes) y para diferentes capacidades de la red interna (10 $Mbps$ y 100 $Mbps$). Esto se debe a que el tamaño de la ráfaga (26 paquetes) es menor que el tamaño del *buffer* en todos los casos, y por lo tanto, el *buffer* puede absorber todos los paquetes entrantes sin producir pérdidas que deterioren la calidad de la comunicación.

Para los casos en los que se utilizan dos y tres flujos de datos de cámara el ancho de banda disponible es de 57 % y 85 % respectivamente. En estos casos la pérdida de paquetes puede ser inaceptable como se muestra en las Figuras 8.3 y 8.4 (notar que los ejes "*y*", tienen diferentes escalas para cada figura). La causa es el solapamiento de las ráfagas que provienen de diferentes cámaras, ya que producen una ráfaga de tráfico que puede exceder la capacidad del *buffer*. Como ejemplo se destacan los resultados deficientes que se obtienen cuando se utiliza un tamaño de *buffer* de 30 paquetes y dos flujos de cámaras, en dicho caso, la pérdida de paquetes casi alcanza un 4 % y un 9 % para capacidades de la red interna de 10 $Mbps$ y 100 $Mbps$ respectivamente. Si el número de paquetes generados por cada cámara en una ráfaga es de 26 paquetes, es fácil que el *buffer* se llene cuando diversas ráfagas llegan.

Al mismo tiempo, otro fenómeno interesante se puede observar cuando se comparan los resultados para 10 $Mbps$ y 100 $Mbps$ en ambas figuras. Se puede observar que la pérdida de paquetes es más alta cuando la capacidad de la red es de 100 $Mbps$. Además, hay algunos casos en los que la pérdida de paquetes solamente aparece para la red más rápida, como sucede en la Figura 8.3 para un *buffer* de 45 paquetes. Esto sucede porque la velocidad de la conexión a Internet sigue siendo la misma, y entonces, cuando una ráfaga es generada por la cámara en una red de 100 $Mbps$ el *buffer* se llenará más rápidamente que en una red de 10 $Mbps$. En estos casos, el incrementar la capacidad de la red de 10 $Mbps$ a 100 $Mbps$ producirá ráfagas de paquetes perdidos, degradando el rendimiento de la red.

Como se ha mencionado anteriormente, estas pruebas tienen como objetivo demostrar la manera en que el tamaño de los *buffer* y su comportamiento afecta la pérdida de paquetes. Sin embargo, los resultados obtenidos por cada conjunto

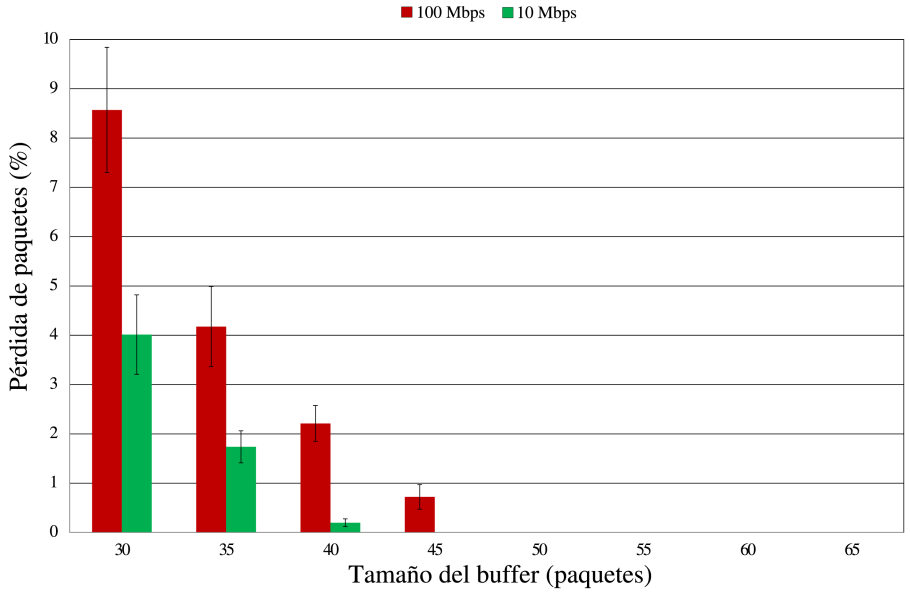

Figura 8.3: Relación entre el tamaño del *buffer* y la pérdida de paquetes para dos flujos de cámara IP.

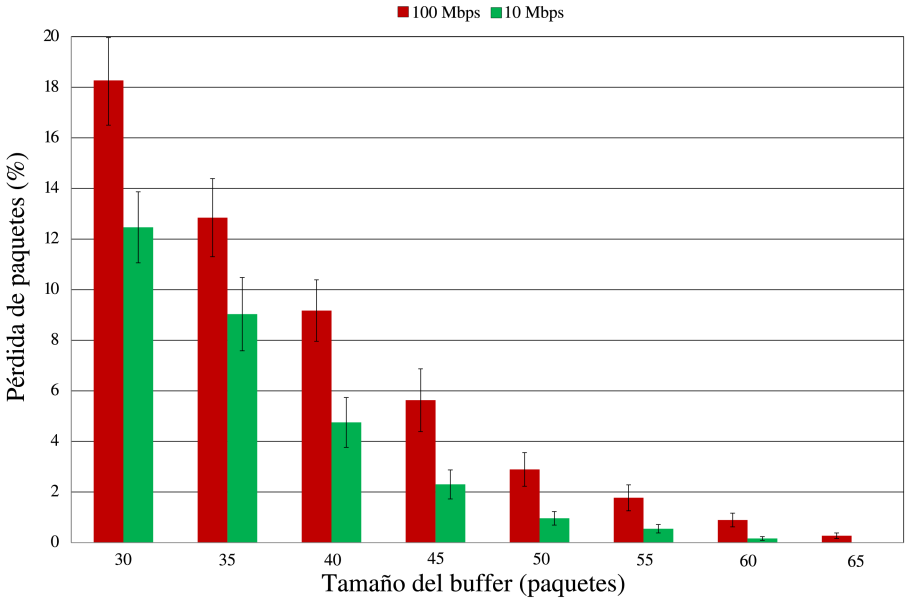

Figura 8.4: Relación entre el tamaño del *buffer* y la pérdida de paquetes para tres flujos de cámara IP.

de pruebas convergen hacia una nueva línea de investigación, ya que el comportamiento de la pérdida de paquetes presenta grandes diferencia entre las pruebas realizadas. Por este motivo, se sugiere un análisis más detallado relacionado a la variabilidad de los datos. En la siguiente sección se analizan algunos aspectos muy generales de los resultados obtenidos en relación a su variabilidad. No obstante, es necesario aclarar que dicho análisis no forma parte de los objetivos de la presente tesis por lo que se debe considerar como una línea futura de investigación.

8.4 Histograma de la pérdida de paquetes

El análisis anterior ha permitido mostrar la relación que existe entre el comportamiento del tráfico a ráfagas y la pérdida de paquetes y por consiguiente la QoS. También, resulta interesante analizar algunos detalles del comportamiento de dicho tráfico, ya que parece haber una distribución no uniforme de los datos, para ello se han seleccionado los resultados correspondientes a un *buffer* con tamaño de 40 paquetes en una red de $100\ Mbps$. Las Figuras 8.5 y 8.6 muestran un histograma de la pérdida de paquetes para las pruebas con dos y tres flujos de cámara respectivamente. En dichos histogramas se puede observar el porcentaje de las iteraciones que han alcanzado un determinado valor de pérdidas.

En ambos casos los datos muestran una distribución no uniforme, presentándose algunas variaciones entre los valores obtenidos de la pérdida de paquetes para las diferentes repeticiones de una misma prueba. Por ejemplo, en la Figura 8.5 se muestra que algunas de las pruebas tienen un $0\,\%$ de pérdidas mientras que en otros casos dicho valor supera el $5\,\%$. En la Figura 8.6 un alto porcentaje de las pruebas ha obtenido una pérdida de paquetes superior al $10\,\%$. Estos resultados sugieren que bajo las mismas condiciones de la red, la QoS de algunas comunicaciones de este tipo de servicio podrían verse drásticamente degradadas. Estos resultados deben ser considerados como datos preliminares y como una iniciativa para futuras líneas de investigación.

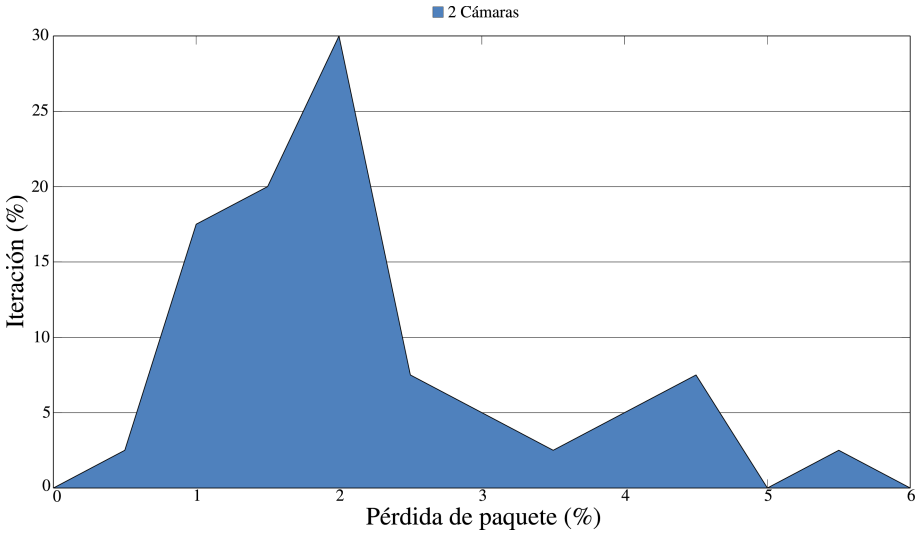

Figura 8.5: Histograma de la pérdida de paquetes para dos flujos de cámara IP que atraviesan un *buffer* de 40 paquetes en una red de 100 $Mbps$.

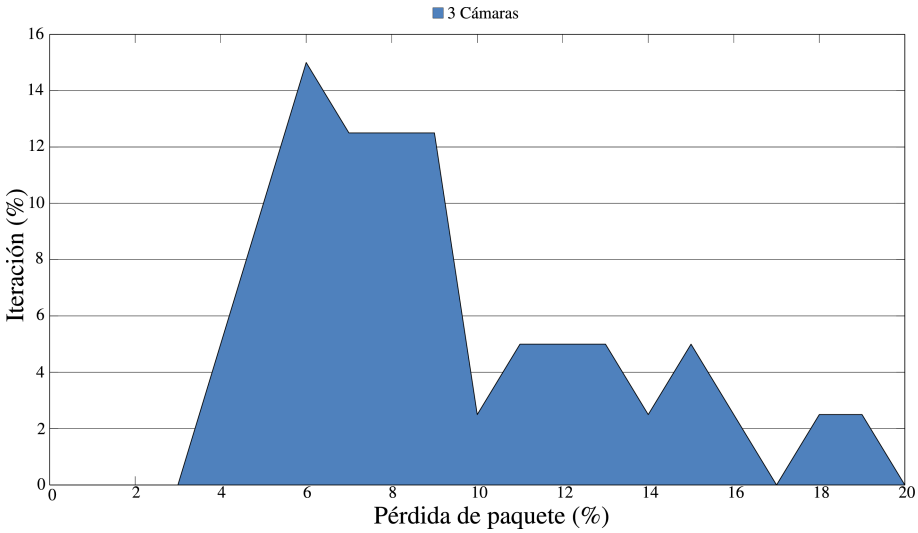

Figura 8.6: Histograma de la pérdida de paquetes para tres flujos de cámara IP que atraviesan un *buffer* de 40 paquetes en una red de 100 $Mbps$.

Hemos aprendido a volar como los pájaros, a nadar como los peces; pero no hemos aprendido el sencillo arte de vivir como hermanos.

Martin Luther King

CAPÍTULO

9

Coexistencia de diversos servicios multimedia

En este capítulo se realiza un análisis del efecto del tamaño del *buffer* en la presencia de tráfico a ráfagas y sus posibles implicaciones en el tráfico de otras aplicaciones que comparten un enlace en común. Para este estudio se ha seleccionado como ejemplo de análisis un entorno de PYMES con un solo enlace de acceso hacia Internet, en el cual convergen servicios de VoIP, videoconferencia y videovigilancia. En este escenario se realizan dos pruebas principales: en la primera se valora el efecto de la variación del tamaño del *buffer* cuando la utilización del enlace se mantiene fija, y en la otra, se observan los efectos del cambio de la utilización del enlace para un determinado tamaño de *buffer*. Además, se analiza el efecto del aumento de la capacidad de la red interna (como se realizó en al capítulo 8) cuando convergen los servicios mencionados.

Dados los resultados obtenidos en el capítulo 8, en este capítulo se describen las distribuciones de pérdida de paquetes por medio de histogramas, ya que la mayoría de los resultados presentan un buen nivel de QoS, y sin embargo, unos pocos presentan peores niveles. El análisis de calidad está basado principalmente en dos parámetros: pérdida de paquetes por flujo y retardo. Además, para el caso de VoIP, también se presentan resultados utilizando estimadores subjetivos de la calidad basados en estos parámetros objetivos, utilizando diferentes valores de retardo de red. Por otro lado, se ha considerado el desbordamiento del *buffer* como la única causa de pérdida de paquetes.

9.1 Escenario de red propuesto

En la Figura 9.1 se muestra el escenario utilizado para las pruebas, el cual consiste en un enlace de acceso a Internet donde convergen dos flujos generados por cámaras IP con un ancho de banda de 1 $Mbps$ cada uno, una sesión de videoconferencia con un ancho de banda medio de 1,5 $Mbps$ y dos llamadas de VoIP con un ancho de banda de 24 $Kbps$ cada una, lo que supone un total de ancho de banda generado de 3,5 $Mbps$.

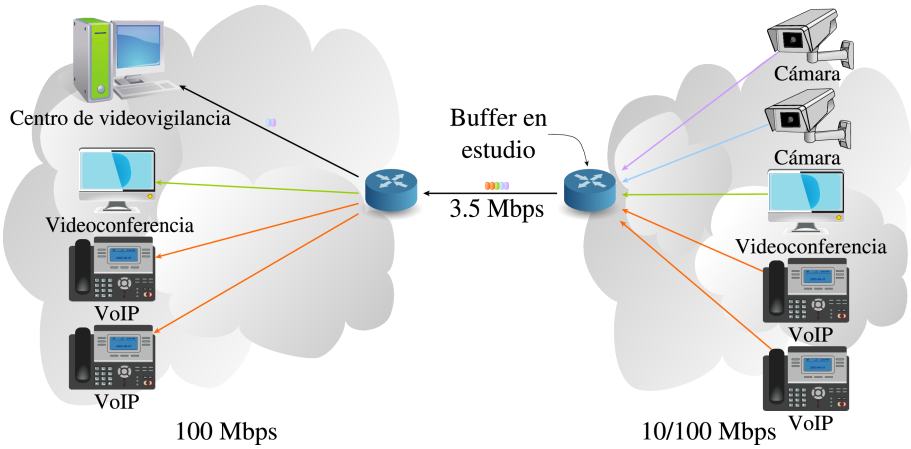

Figura 9.1: Escenario para las pruebas con dos conexiones de cámaras, una videoconferencia y dos llamadas de VoIP.

Para este escenario se plantean dos pruebas diferentes: en la primera, la capacidad del enlace de acceso a Internet se establece en 5 $Mbps$, de esta manera la utilización media del enlace se fija al 70 % y se realizan las pruebas para diferentes tamaños de *buffer*. En la segunda prueba, el tamaño del *buffer* del *router* de acceso a Internet se fija en 40 paquetes y las simulaciones se repiten para diferentes valores de la capacidad de acceso, y por consiguiente, para diferentes niveles de la utilización del enlace, dentro de un rango que va desde el 50 % al 90 %.

9.2 Tráfico utilizado

Con la finalidad de desarrollar las pruebas descritas anteriormente, se han utilizado tres fuentes de tráfico multimedia diferentes: videovigilangia, videoconferencia y VoIP. Para el tráfico de videovigilancia y videoconferencia no se utilizan modelos de tráfico, sino, trazas de tráfico real que fueron capturadas previamente en escenarios reales para luego ser generadas en NS-2, usando sus tamaños de paquetes y tiempo entre paquetes. El tráfico de VoIP es generado mediante un agente CBR (*Constant Bit Rate*) de NS-2.

La metodología utilizada para las capturas del tráfico de videoconferencia se ilustra en la Figura 9.2. Para dichas trazas se ha utilizado la arquitectura de Vidyo^{TM}, la cual incorpora la tecnología AVL que permite la optimización dinámica del video para cada terminal, aprovechando la tecnología de compresión $H.264$-SVC. La aplicación de videoconferencia se configuró con $2\ Mbps$ de ancho de banda (sin embargo, la captura real de dicho tráfico solo alcanza un ancho de banda de $1{,}5\ Mbps$) y una resolución de $800 \times 450\ px$ mientras la cámara capturaba un video con mucho movimiento (un partido de fútbol).

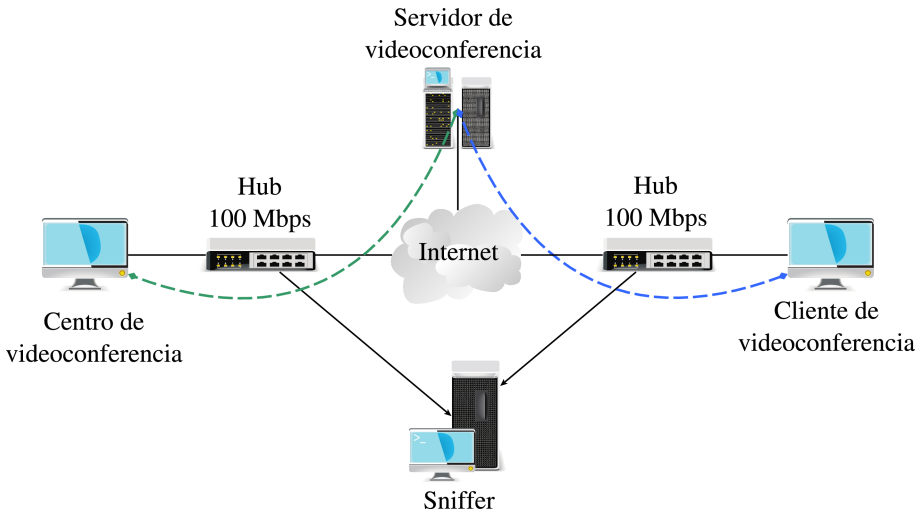

Figura 9.2: Escenario para la captura del tráfico de una videoconferencia.

El tráfico de voz se genera de acuerdo a la recomendación $G.729$ con un tiempo entre paquetes de $20\ ms$ y 2 muestras por paquete, resultando en un tamaño de paquete de $60\ bytes$. Para las trazas de videovigilancia, se han utilizado las mismas que se usaron para obtener los resultados del capítulo 8.

9.3 Análisis de pérdida de paquetes

Este análisis se enfoca en la calidad que se obtiene para el tráfico combinado y para cada uno de los servicios que comparten la red, lo cual corresponde a un estudio más detallado del comportamiento del tráfico en el escenario en cuestión. Además, el análisis del tráfico combinado se centra en la relación de pérdida de paquetes para los casos en los que la red interna es de $10\ Mbps$ y $100\ Mbps$, mientras que el análisis de los flujos lo hace en el caso de una red interna a $100\ Mbps$, ya que esta capacidad de red es la que presenta el peor de los casos en términos de pérdida de paquetes.

9.3.1 Pérdida de paquetes del tráfico combinado

Los resultados para el tráfico combinado de los tres tipos de flujo que comparten la red, correspondientes a la primera prueba, se muestran en la Figura 9.3. En la cual se observa la pérdida de paquetes para diferentes tamaños de *buffer* cuando la utilización del enlace es del $70\,\%$ con sus respectivos intervalos de confianza del $95\,\%$. En dicho gráfico se puede observar el mismo fenómeno mencionado en el capítulo 8: la pérdida de paquetes es mayor cuando la capacidad de la red local es de $100\ Mbps$. Además, dicho efecto aumenta cuando el tamaño del *buffer* disminuye.

A pesar de que el tráfico no se muestra de manera separada para cada flujo (este tema se analiza en la siguiente sección), la pérdida de paquetes afecta a todas las aplicaciones, de esta manera se observa que la presencia de aplicaciones que generan tráfico a ráfagas (videovigilancia) causa la pérdida de paquetes para todas las aplicaciones que coexisten, incluso para aquellas que generan tráfico a una tasa de bit constante (VoIP).

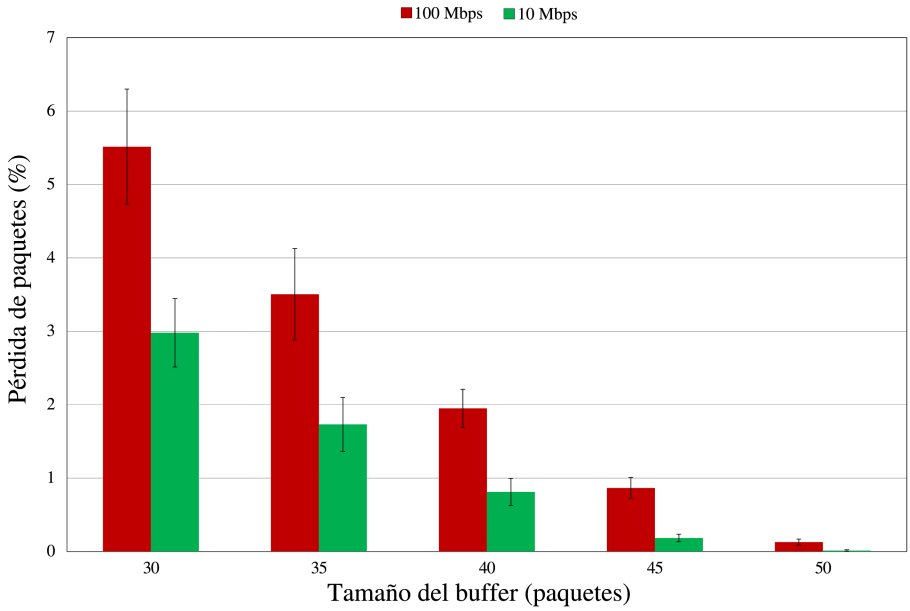

Figura 9.3: Relación entre el tamaño del *buffer* y la pérdida de paquetes para una utilización del enlace del 70 %.

Para la segunda prueba, los resultados se muestran en la Figura 9.4, la cual describe la relación de la pérdida de paquetes y la utilización del enlace para este escenario en particular. Como era de esperar, la pérdida de paquetes se incrementa cuando la utilización del enlace crece para el caso de un tamaño de *buffer* de 40 paquetes. De nuevo, la pérdida de paquetes es mayor cuando la capacidad de la red es de 100 $Mbps$.

9.3.2 Pérdida de paquetes por flujo

Para este caso se han desarrollado dos tipos diferentes de pruebas: en la primera se considera un escenario con la utilización de enlace fijada y se varía el tamaño del *buffer* y en la segunda se fija el tamaño del *buffer* variando la utilización del enlace. Las Figuras 9.5 y 9.6 muestran la pérdida de paquetes por flujo usando una utilización del enlace fija (70 %) con sus correspondientes intervalos de confianza del 95 %.

La principal causa de pérdida de paquetes es la presencia de una aplicación

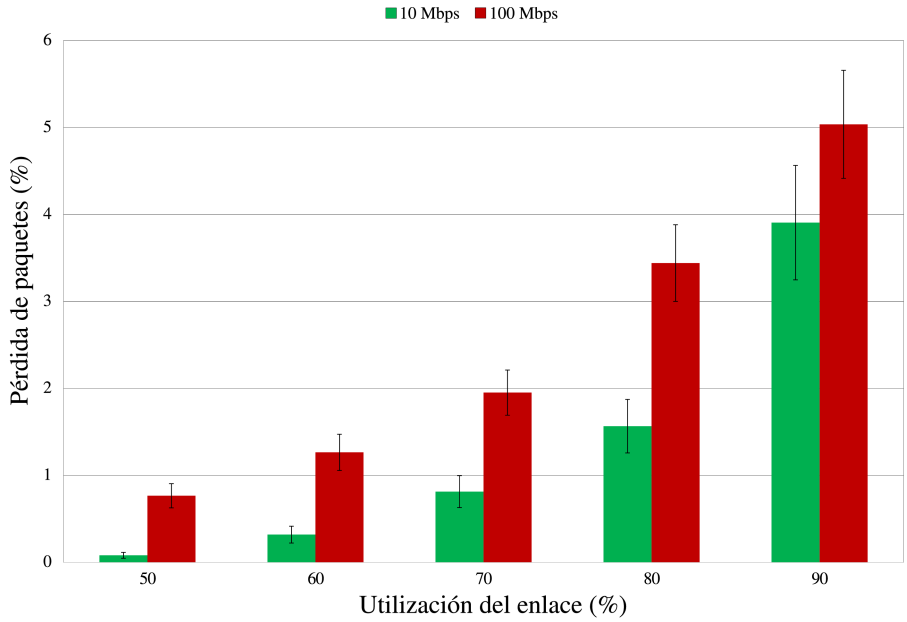

Figura 9.4: Relación entre la utilización del enlace y la pérdida de paquetes para un tamaño de *buffer* de 40 paquetes.

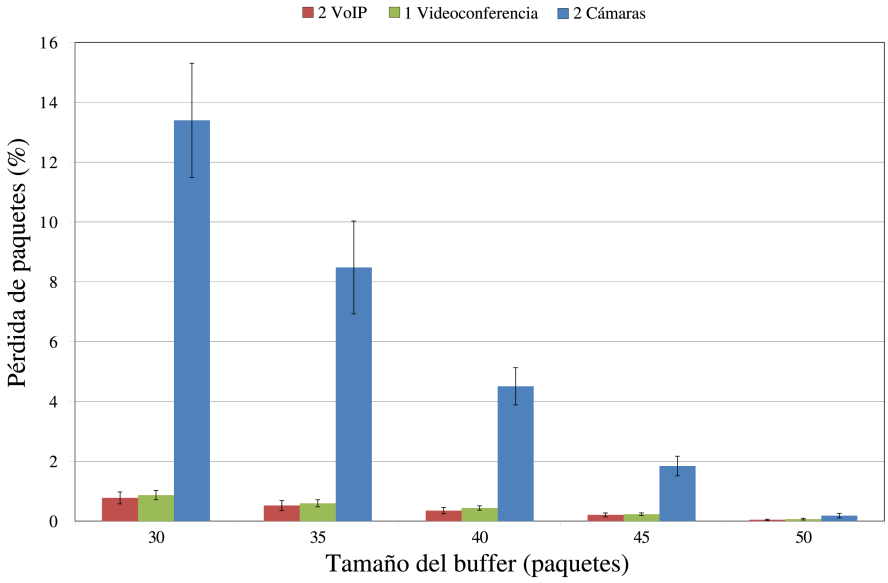

Figura 9.5: Pérdida de paquetes por flujo cuando la utilización del enlace es del 70 % para diferentes tamaños de *buffer*.

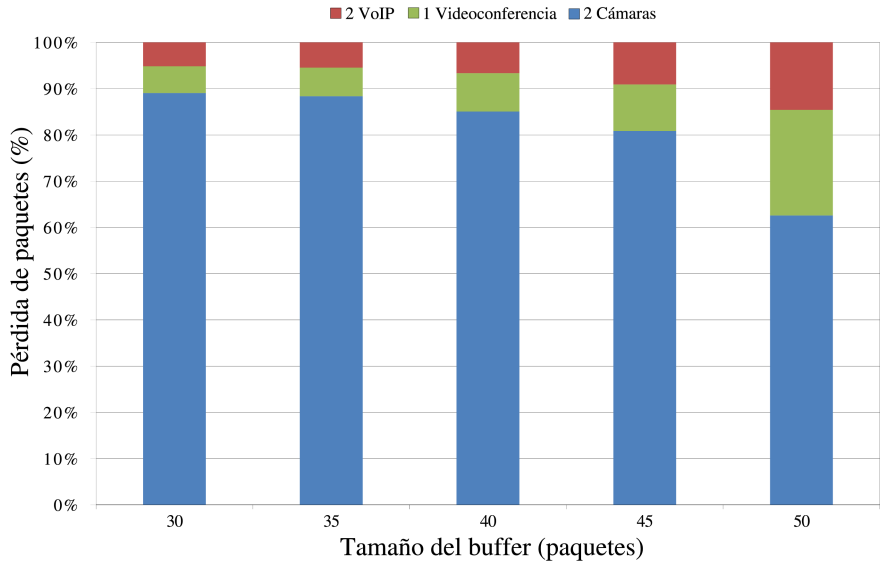

Figura 9.6: Distribución por flujo de la pérdida de paquetes cuando la utilización del enlace es del 70 % para diferentes tamaños de *buffer*.

que genera tráfico a ráfagas (videovigilancia), la cual causa un desbordamiento del *buffer* y degrada la calidad de todas las aplicaciones coexistentes. Se puede observar que la pérdida de paquetes decrece cuando el tamaño del *buffer* se incrementa, porque los *buffer* más grandes pueden absorber mejor las ráfagas producidas por el tráfico mezclado. No obstante, la videoconferencia y VoIP obtienen mejores resultados debido a su perfil de tráfico menos a ráfagas.

Por otro lado, en la Figura 9.6 se puede ver que la distribución de la pérdida de paquetes no es la misma para todas las pruebas con diferentes tamaños de *buffer*. Los *buffer* pequeños aumentan el problema causado por el tráfico de videovigilancia (el más a ráfagas), incrementando la tasa de paquetes perdidos correspondientemente a este servicio.

Las Figuras 9.7 y 9.8 muestran los resultados de las pruebas con el tamaño del *buffer* fijo (40 paquetes). En los gráficos se representa en el eje "x", la utilización media del enlace de acuerdo al ancho de banda generado por las aplicaciones. Como era de esperar, la pérdida de paquetes se incrementa cuando la utilización

del enlace crece. De nuevo, la distribución de la pérdida de paquetes no es la misma para todas las pruebas (Figura 9.8), aunque las diferencias no son significativas. Así, teniendo en cuenta los resultados de las Figuras 9.7 y 9.8, se puede ver que el tamaño del *buffer* tiene una fuerte influencia en la distribución de la pérdida de paquetes por flujo.

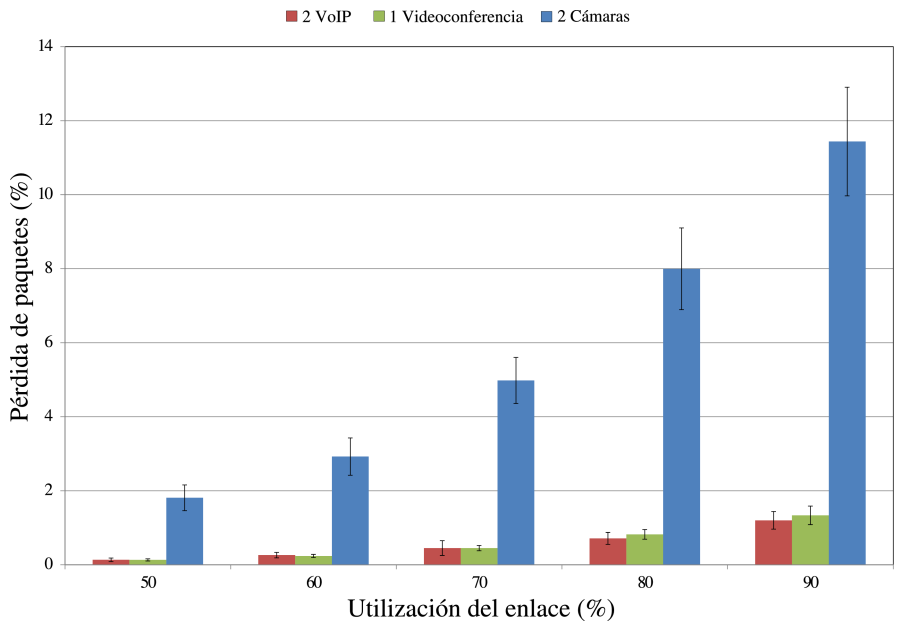

Figura 9.7: Relación entre la pérdida de paquetes por flujo y la utilización del enlace para un *buffer* de 40 paquetes.

9.4 Histograma de la pérdida de paquetes

En la sección anterior se han presentado la media de los resultados para una serie de pruebas y se han obtenido valores pequeños de los intervalos de confianza. Sin embargo es necesario describir la distribución de pérdida de paquetes entre las comunicaciones establecidas en las diferentes pruebas, ya que la pérdida de paquetes podría no ser uniforme entre ellas, como se ha comentado en el capítulo 8. Mientras que en algunas pruebas no se pierden paquetes, en otras, algunos flujos presentan altas tasas de paquetes perdidos, porque en esos casos el solapamiento

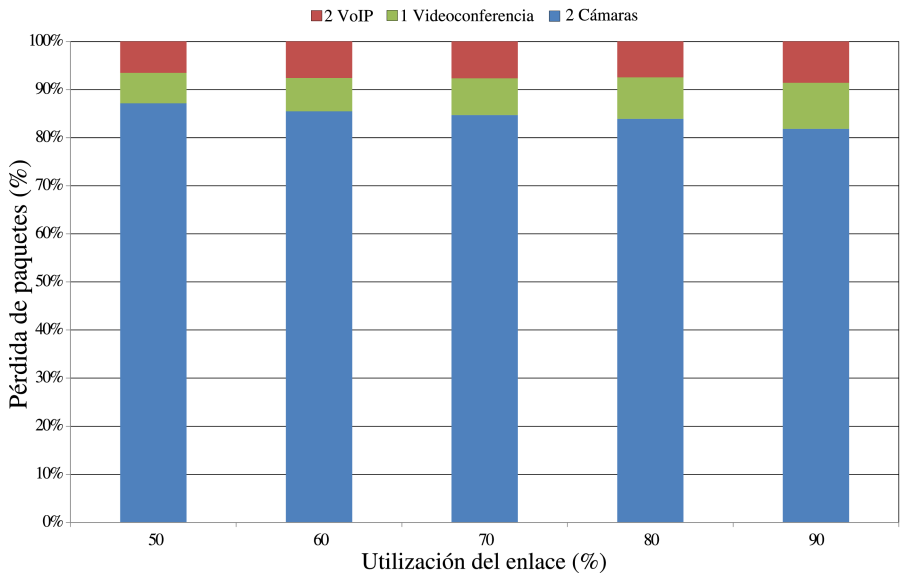

Figura 9.8: Distribución por flujo de la pérdida de paquetes para un *buffer* de 40 paquetes en función de la utilización del enlace.

de los flujos es más grande. Como resultado, en la misma red habrá momentos en que con las mismas condiciones, una comunicación puede obtener una muy buena calidad mientras que en otros presenten valores significativamente peores. La principal causa de este efecto es la distribución aleatoria de las superposiciones entre las ráfagas.

Con la finalidad de incentivar este tipo de análisis y sugerir una línea futura de investigación, a continuación se introduce una forma de medir este fenómeno. Para esto, se ha seleccionado un escenario con una utilización del enlace del 70 % y un tamaño de *buffer* de 40 paquetes, y las mismas aplicaciones descritas anteriormente. En este escenario específico, las pruebas se han repetido 200 veces (a pesar que en el capítulo 8 se mencionó que con 40 repeticiones se obtienen resultados muy similares) para observar mejor el solapamiento de los flujos comentados con anterioridad, y su relación con la pérdida de paquetes. Los resultados se presentan por medio de un histograma correspondiente al tráfico total y para cada servicio donde en el eje "x", se muestra el porcentaje de pérdida de paquetes, y en el eje "y", el porcentaje de iteraciones en la cual se ha obtenido ese valor de pérdida de

paquetes.

En la Figura 9.9 se presenta un histograma de la pérdida de paquetes para el tráfico total en una red a $100\ Mbps$. El valor medio de la pérdida de paquetes correspondiente a los resultados de las 200 repeticiones es de $2{,}11\ \%$. Como se puede observar en dicha figura, existe una gran cantidad de iteraciones que se encuentran por debajo de la media, incluso el $1\ \%$ de los casos sin pérdida de paquetes, a la vez, muchas de las pruebas duplican la media y algunas se encuentran por encima del $5\ \%$.

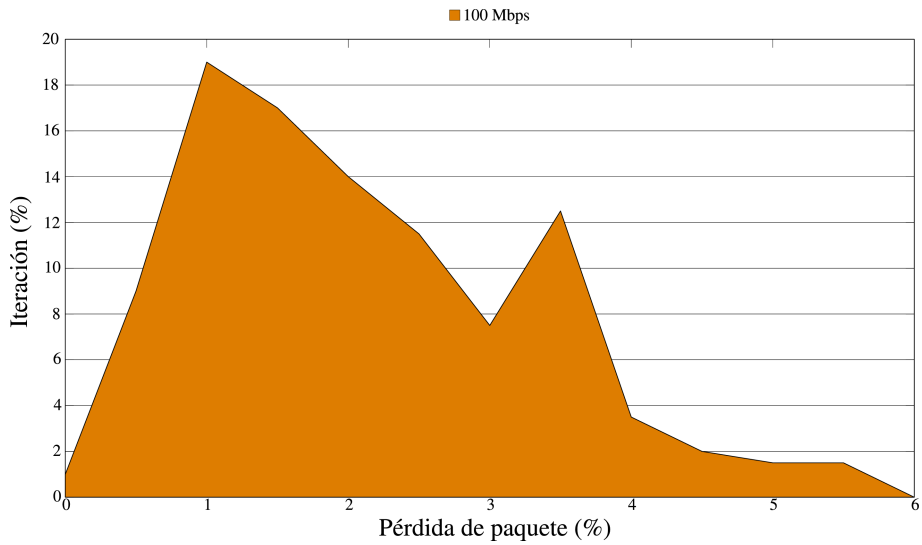

Figura 9.9: Histograma de la pérdida de paquetes para el tráfico combinado con un *buffer* de 40 paquetes y una utilización del enlace del $70\ \%$.

El caso de VoIP se presenta en la Figura 9.10, en la cual casi el $80\ \%$ de las llamadas presentan un valor de pérdida de paquetes menor al $0{,}75\ \%$. La pérdida de paquetes aumenta hasta un $3\ \%$ o más en el $0{,}5\ \%$ de los casos (equivalente a 20 llamadas) en los cuales la QoS sería significativamente degradada. Esto confirma que hay un porcentaje de llamadas en las cuales la calidad obtenida no será lo suficientemente buena para los usuarios.

El servicio de videoconferencia presenta un comportamiento similar (Figura 9.11). La tasa de pérdida de paquetes es baja para un alto porcentaje de las pruebas.

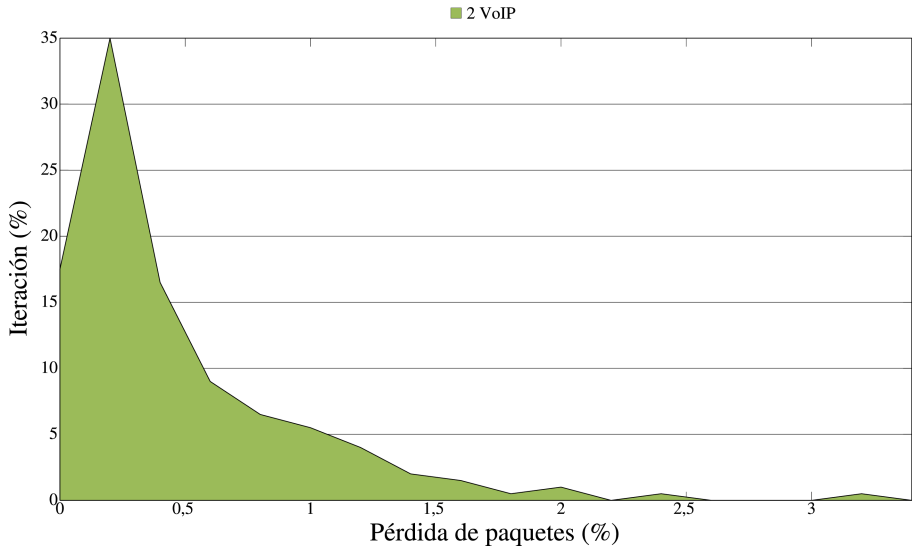

Figura 9.10: Histograma de la pérdida de paquetes para el tráfico de VoIP con un *buffer* de 40 paquetes y una utilización del enlace del $70\,\%$.

Sin embargo, estas pérdidas pueden afectar a la calidad de la videoconferencia. Por otro lado, los resultados de las comunicaciones del servicio de videovigilancia (Figura 9.12) muestran el nivel más alto de pérdida de paquetes (hasta un $14\,\%$ en algunos casos), el cual podría degradar significativamente la QoS de este servicio.

9.5 MOS para llamadas de VoIP

Ahora, se analizará el efecto de las ráfagas de pérdidas de paquetes en la calidad subjetiva de VoIP, ya que es un servicio en tiempo real con requerimientos muy específicos de retardo y pérdida de paquetes. En este caso se han utilizado los resultados del histograma de la pérdida de paquetes para el tráfico de VoIP analizado anteriormente. Con el objetivo de estimar la calidad subjetiva que se obtendría para cada llamada, se ha calculado el R_{factor} de acuerdo con [CR01] mediante la siguiente ecuación:

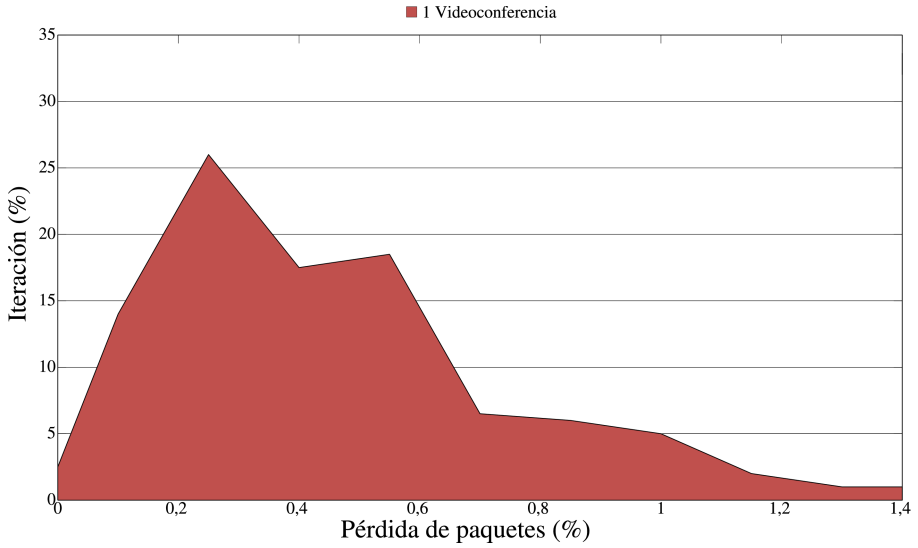

Figura 9.11: Histograma de la pérdida de paquetes para el tráfico de videoconferencia con un *buffer* de 40 paquetes y una utilización del enlace del 70 %.

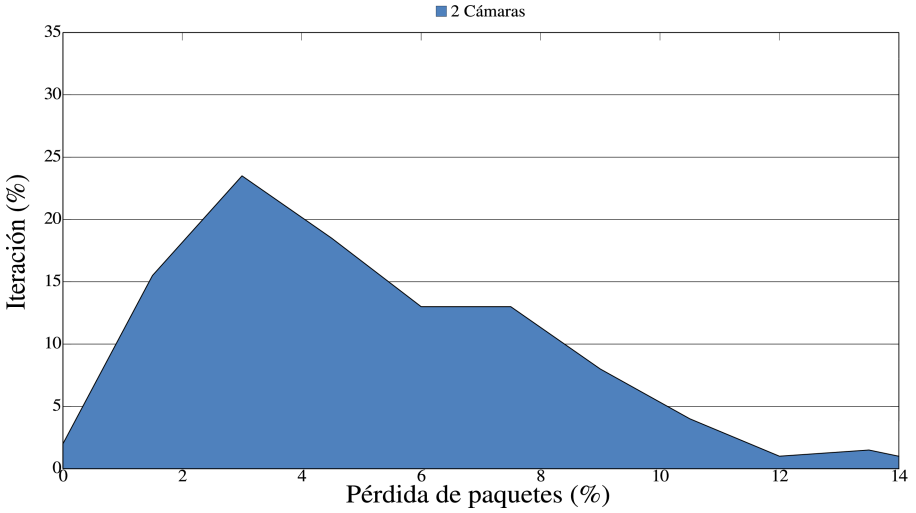

Figura 9.12: Histograma de la pérdida de paquetes para el tráfico de cámara IP con un *buffer* de 40 paquetes y una utilización del enlace del 70 %.

$$\begin{aligned} R_{factor} &= 94{,}2 - 0{,}24 \times delay_{total} - 0{,}11(delay_{total} - 177{,}3) \\ &\quad \times H(delay_{total} - 177{,}3) - 11 \\ &\quad -40ln(1 + (10 \times delay_{total})) \end{aligned} \tag{9.1}$$

Donde $delay_{total}$ es el retardo OWD y $H(x)$ es una función escalón. Así, si el retardo se encuentra por debajo de 177,3 ms, entonces no afecta al R_{factor}. Sin embargo, si excede este valor, entonces el R_{factor} sería significativamente menor. Esto responde al fenómeno citado en [CR01]: *"Para el OWD menor que* 177,3 ms*, las conversaciones ocurren con normalidad, mientras que cuando el retardo se excede de* 177,3 ms *la conversación comienza con deformaciones y rupturas; a menudo degenerando en conversaciones tipo simplex para los valores de retardo más alto"*.

A continuación, se obtiene el MOS a partir del R_{factor}, utilizando la conversión citada en el mismo artículo [CR01]. Para el retardo total se ha considerado incluir el retardo causado por el *buffer* del *router* y la red; además, se ha incluido un *buffer* de *de-jitter* con la finalidad de absorber las variaciones de retardo generadas por el *buffer* del *router*, así los *buffer* del *router* y el de *de-jitter* se compensan mutuamente.

Se han utilizado seis valores diferentes de retardo de red (20, 40, 60, 100, 120 y 140 ms) que producen un retardo total de 116, 136, 156, 196, 216 y 236 ms, respectivamente. Los resultados se presentan por medio de un histograma (Figura 9.13) del MOS obtenido para cada prueba. Para los tres valores más bajos del retardo de red (20, 40 y 60), la figura muestra una cantidad significativa de llamadas con una calidad media según el *E-model* de la ITU-T [Rec14, CR01]. Esto representaría malos resultados para los usuarios de VoIP, ya que de este escenario se esperaría que proporcionara la mejor calidad en todos los casos. Además, las colas a la izquierda de la Figura 9.13 representan a unas cuantas llamadas con niveles inaceptables de calidad. Por otro lado, para los tres valores de retardo de red más altos (100, 120 y 140 ms), en los cuales el retardo total excede el umbral de 177,3 ms, se puede observar un comportamiento peor en términos de MOS. El

incremento del retardo de la red produce una reducción significativa en la calidad subjetiva, dando como resultado una calidad baja en algunos casos.

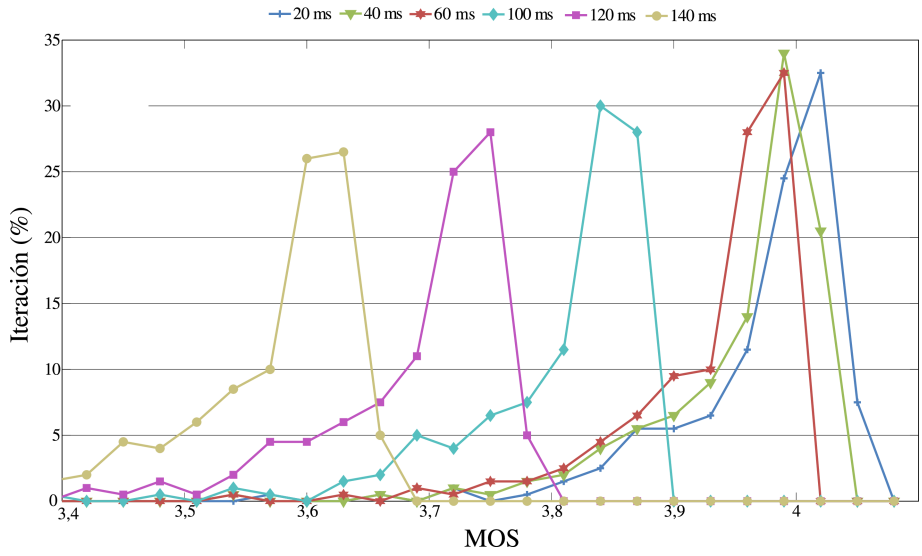

Figura 9.13: Histograma del MOS con diferentes retardos de red (OWD) para un *buffer* de 40 paquetes y una utilización del enlace del 70 %.

Los resultados anteriores muestran el efecto que tiene el tráfico a ráfagas en la calidad subjetiva del servicio de VoIP. No obstante, sería interesante analizar el efecto que dicho tráfico tiene en los otros servicios. Por esto, se sugieren como futuras líneas de investigación, el desarrollo de modelos que permitan obtener un MOS a partir de parámetros objetivos de la calidad para otros servicios como los analizados en la presente tesis. Además, la evaluación de los mismos en entornos de tráfico concurrentes como los que se han analizado anteriormente.

Parte IV

Conclusiones y líneas futuras

Ahora bien, este no es el final. No es ni siquiera el principio del fin. Pero es, quizás, el final del principio.

Winston Churchill

CAPÍTULO
10

Conclusiones y líneas futuras

10.1 Conclusiones

En este capítulo se comentan las principales conclusiones relacionadas con el comportamiento de los flujos de datos y su impacto en la red, además de los métodos de detección de *buffer* y su efecto en el tráfico. También se incluyen aspectos relacionados a la calidad de servicio en la coexistencia de tráficos concurrentes de varios servicios. Por último, se sugieren algunas líneas futuras de investigación que se derivan del presente trabajo.

10.1.1 El comportamiento del tráfico

El comportamiento del tráfico de las aplicaciones está ligado a cómo éstas envían datos a la red, y por lo tanto, a las necesidades y requerimientos del servicio asociado a la aplicación y la forma en que dicha aplicación ha sido desarrollada. En este contexto, algunas aplicaciones generan datos a una tasa constante mientras que en otros casos los patrones de tráfico pueden llegar a ser más complejos, produciendo ráfagas de paquetes que contienen un número de paquetes diferente dependiendo de cada servicio.

Además, el tamaño de la información juega un papel importante en el tráfico de la red, ya que las aplicaciones generan paquetes de tamaños muy diversos en función de aspectos como: la interactividad, requerimientos temporales y la propia naturaleza de la información. El tamaño de los paquetes puede variar desde unas pocas decenas de $bytes$ en los casos de aplicaciones de VoIP o juegos *online*, hasta

el máximo MTU que la red permite como en los casos de servicios que necesitan enviar una gran cantidad de información (IPTV, video *streaming*, entre otros).

El tráfico en la red está formado por el conjunto de cada uno de los flujos de información generados por cada una de las aplicaciones utilizadas por los usuarios finales. Por lo tanto, el comportamiento del tráfico en la red es producido por la combinación de todos los tráficos concurrentes, generando patrones de tráfico aún más complejos. En esta combinación de flujos, es muy probable que se formen ráfagas de paquetes por la agrupación aleatoria de diversos flujos, incluso ráfagas aún más grandes cuando coinciden ráfagas de diversas aplicaciones.

10.1.2 La detección de *buffer*

En la presente tesis se ha propuesto un procedimiento que permite descubrir y describir características de redes de forma alternativa a los métodos tradicionales y obteniendo información que pasa desapercibida. El procedimiento propuesto se basa en la utilización de modelos de *buffer* presentes e influyentes en un camino de red. Dicho modelo consiste en la caracterización del comportamiento del *buffer* y sus principales parámetros como son su tamaño, sus límites y las tasas de entrada y salida. De manera muy general, el procedimiento está basado en el envío de una ráfaga de paquetes UDP desde la máquina fuente hasta la de destino, donde todos los paquetes tienen el mismo tamaño y son identificados con un número de secuencia incluido en el *payload*. El objetivo principal de las pruebas es producir un desbordamiento del *buffer* que actúa como punto crítico en el camino de red, para luego, analizar las capturas de tráfico de ambos extremos de la comunicación y obtener los parámetros del modelo.

Los métodos propuestos se pueden clasificar según su tipo de acceso: físico o remoto. El primero de ellos es el método exacto y se utiliza para comparar la exactitud de los otros métodos, sin embargo, requiere tener acceso físico al dispositivo a medir. El segundo es útil en entornos donde las medidas deben ser realizadas de forma desatendida o en los casos donde no se puede tener acceso físico. Desde otra perspectiva, la clasificación se puede realizar según la cantidad de *buffer* a detectar, existiendo métodos para detectar un *buffer* y otro para detectar diversos *buffer*

concatenados.

Además, se ha propuesto un método que permite detectar más de un *buffer* en congestión con sus respectivos parámetros, lo cual permite ampliar el modelo y obtener más información útil de las mediciones realizadas.

Los métodos descritos en el presente trabajo han sido validados mediante implementaciones reales con dispositivos comerciales bien conocidos y de uso común en diversas redes. Dichos dispositivos se han estudiado en escenarios controlados de laboratorio en redes cableadas e inalámbricas. Los resultados muestran que los métodos planteados permiten descubrir y describir el comportamiento y los parámetros del modelo de *buffer* propuesto con altos niveles de exactitud.

10.1.3 La QoS y la coexistencia del tráfico multimedia

Se ha estudiado la pérdida de paquetes causada por el *buffer* de un nodo de la red en la presencia de aplicaciones que generan tráfico a ráfagas, y su influencia en la calidad subjetiva de VoIP. Además, se han desarrollado diversas pruebas en diferentes escenarios utilizando trazas reales de aplicaciones multimedia.

El tamaño del *buffer* se ha identificado como un parámetro crítico a la hora de realizar el planeamiento de una red en este tipo de entornos. La razón de esto es la relación entre el tamaño del *buffer* y el número de paquetes contenidos en una ráfaga de tráfico que generan las aplicaciones, ya que dicho número debe ser consistente con la cantidad de paquetes que un *buffer* puede absorber durante períodos de congestión, con la finalidad de prevenir la pérdida de paquetes debido al desbordamiento del *buffer*. Además, se debe tener en cuenta que la tasa de llenado del *buffer* está determinada por la relación entre la velocidad de la red interna y el acceso a Internet, ya que esto podría producir pérdida de paquetes cuando se envían ráfagas con cantidades de paquetes muy grandes, desde la red interna hacia Internet.

Los resultados de diversas pruebas muestran que la presencia de aplicaciones que generan tráfico a ráfagas en la red interna, podrían producir pérdida de paquetes, la cual podría aumentar si se incrementa la velocidad de la red interna. En este

sentido, se han realizado pruebas con diferentes aplicaciones multimedia, velocidades de acceso y tamaños de *buffer*, y en todos los casos la pérdida de paquetes es mayor para redes de $100\ Mbps$ que para las de $10\ Mbps$.

Además, se ha observado que el tráfico a ráfagas que generan algunas aplicaciones afectan a otros servicios que comparten el mismo enlace. Con la finalidad de mostrar el efecto de la naturaleza a ráfagas del tráfico de estas aplicaciones, se ha medido el MOS en llamadas de VoIP concurrentes. Los resultados muestran que dichas llamadas sólo son capaces de obtener una calidad media, fallando en alcanzar mejores resultados incluso cuando la utilización del enlace es del $70\,\%$. Ya que la causa de este problema es la naturaleza a ráfagas de muchas aplicaciones, en estos casos las técnicas que permiten suavizar el tráfico se pueden considerar como una ventaja.

10.2 Líneas futuras

Con respecto a la caracterización de un camino de red, se podría afirmar que en una conexión a través de una red como Internet, no se puede estar seguro de la cantidad de dispositivos que se encuentran en un determinado camino de red, tampoco de su orden o su comportamiento. La metodología propuesta en esta tesis para descubrir y caracterizar el comportamiento y los parámetros de los *buffer*, se ha limitado a la concatenación de un máximo de dos *buffer* diferentes, sin embargo, esta metodología puede ser aplicada para descubrir un número mayor de *buffer*. Por esto se sugiere como línea de investigación futura, analizar sistemas con una cantidad mayor de dispositivos en el camino de red de los que se han estudiado en esta tesis. Es cierto que en un determinado enlace lo más usual es que el tráfico se vea principalmente afectado por un dispositivo que produce un punto de congestión, pero existen situaciones en las que el comportamiento y las limitaciones de diversos *buffer* pueden generar un número mayor de puntos de congestión.

Por otro lado, las aplicaciones multimeda se están convirtiendo muy populares en entornos móviles donde los recursos son todavía más limitados (por ejemplo, la capacidad de procesamiento, consumo de energía y memoria). En este tipo de escenarios, los problemas resaltados en el presente estudio podría afectar la QoS

de diferentes servicios si éstos tienen determinados requerimientos temporales o relacionados a la pérdida de paquetes, principalmente si se tiene en cuenta que las características del enlace pueden cambiar debido a la movilidad (por ejemplo, durante un *handoff* o un *roaming*) en este tipo de redes. Además, los *buffer* son implementados por los fabricantes en función de ciertos criterios muy particulares y directamente relacionados al tipo de red. Esto se ha comprobado en la presente tesis ya que se han analizados dispositivos *Ethernet* y WiFi y sus parámetros difieren en buena medida en ambos casos. Por esta razón, se plantea como línea futura, el análisis del comportamiento y los parámetros de los *buffer* en entornos móviles cuando concurren tráfico multimedia y a ráfagas.

Por otro lado, se sugiere analizar el efecto del tráfico a ráfagas en otros servicios multimedia desde la perspectiva de la calidad percibida por los usuarios. Para esto, es necesario el desarrollo de modelos que permitan obtener un MOS a partir de parámetros objetivos de la calidad para otros servicios como la videovigilancia, videoconferencia y otros. Por último, se propone realizar la evaluación de dichos modelos de calidad en entornos de tráfico donde concurren diversos tipos de flujos como los que se han analizado en la presente tesis.

Bibliografía

[AB02] Axis Communications AB. *AXIS 2120 User's Manual*, 2.01 edition, February 2002. 13

[Ada13] R. Adams. Active Queue Management: A Survey. *Communications Surveys Tutorials, IEEE*, 15(3):1425–1476, Third 2013. 35

[AKM04] Guido Appenzeller, Isaac Keslassy, and Nick McKeown. Sizing router buffers. In Raj Yavatkar, Ellen W. Zegura, and Jennifer Rexford, editors, *SIGCOMM*, page 281–292. ACM, 2004. 34

[ANFN+11] José I. Aznar, Eduardo Viruete Navarro, Julián Fernández-Navajas, José Ruiz-Mas, José Ma Saldaña, and Jenifer Murillo. QMoEs: A Bandwidth Estimation and Monitoring Tool for QoE-Driven Broadband Networks. In *NTMS*, page 1–4. IEEE, 2011. 4

[BBC+98] Steven Blake, David L. Black, Mark A. Carlson, Elwyn Davies, Zheng Wang, and Walter Weiss. An Architecture for Differentiated Services. *Network Working Group*, RFC 2475, December 1998. 22

[BCS94] Bob Braden, David Clark, and Scott Shenker. Integrated Services in the Internet Architecture: an Overview. *Network Working Group*, RFC 1633, June 1994. 21

[Bel92] Steven Michael Bellovin. A best-case network performance model. Technical report, February 1992. 30

[BLT06] T. Bu, Yong Liu, and D. Towsley. On the TCP-Friendliness of VoIP Traffic. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, page 1–12, April 2006. 13

[BMM+08] D. Bonfiglio, M. Mellia, M. Meo, N. Ritacca, and D. Rossi. Tracking Down Skype Traffic. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, page –, April 2008. 16

[Bol93] Jean-Chrysostome Bolot. Characterizing End-to-End Packet Delay and Loss in the Internet. *J. High Speed Networks*, 2(3):305–323, 1993. 30

[BS06] S.A. Baset and H.G. Schulzrinne. An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, page 1–11, April 2006. 16

[CC96] Robert L. Carter and Mark Crovella. Measuring Bottleneck Link Speed in Packet-Switched Networks. *Perform. Eval.*, 27/28(4):297–318, 1996. 30

[CHHL06] Kuan-Ta Chen, Chun-Ying Huang, Polly Huang, and Chin-Laung Lei. Quantifying Skype User Satisfaction. In *Proceedings of ACM SIGCOMM 2006*, Pisa Italy, Sep 2006. 16

[CMP08] L. De Cicco, S. Mascolo, and V. Palmisano. A mathematical model of the Skype VoIP congestion control algorithm. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, page 1410–1415, Dec 2008. 16

[CR01] R. G. Cole and J. H. Rosenbluth. Voice over IP performance monitoring. *SIGCOMM Comput. Commun. Rev.*, 31(2):9–24, April 2001. 26, 41, 42, 50, 127, 129

[DD06] Amogh Dhamdhere and Constantine Dovrolis. Open issues in router buffer sizing. *Computer Communication Review*, 36(1):87–92, 2006. 35

[DH98] Stephen E. Deering and Robert M. Hinden. Internet protocol, version 6 (IPv6) specification. RFC 2460, December 1998. 22

[DMP11] Luca De Cicco, Saverio Mascolo, and Vittorio Palmisano. Skype Video congestion control: An experimental investigation. *Computer Networks*, 55(3):558–571, 2011. 16

[EGG+05] Mihaela Enachescu, Yashar Ganjali, Ashish Goel, Nick McKeown, and Tim Roughgarden. Part III: routers with very small buffers. *Computer Communication Review*, 35(3):83–90, 2005. 34

[FCFW02] Wu-Chang Feng, Francis Chang, Wu-Chi Feng, and Jonathan Walpole. Provisioning on-line games: a traffic analysis of a busy counter-strike server. In *Internet Measurement Workshop*, page 151–156. ACM, 2002. 42

[FHG04] Sally Floyd, Tom Henderson, and Andrei Gurtov. The NewReno modification to TCP's fast recovery algorithm. RFC 3782, April 2004. 29

[FJ93] Sally Floyd and Van Jacobson. Random early detection gateways for congestion avoidance. *Networking, IEEE/ACM Transactions on*, 1(4):397–413, 1993. 3, 37

[FLK+08] B. Fallica, Yue Lu, F. Kuipers, R. Kooij, and P. Van Mieghem. On the Quality of Experience of SopCast. In *Next Generation Mobile Applications, Services and Technologies, 2008. NGMAST '08.*

The Second International Conference on, page 501–506, sept. 2008. 19

[FS08] S. Fleck and W. Strasser. Smart Camera Based Monitoring System and Its Application to Assisted Living. *Proceedings of the IEEE*, 96(10):1698–1714, oct. 2008. 1, 38

[FSFN+14] Carlos Fernández, Jose Saldana, Julián Fernández-Navajas, Luis Sequeira, and Luis Casadesus. Video Conferences through the Internet: How to Survive in a Hostile Environment. *The Scientific World Journal*, 2014, 2014. 17

[GL10] Cesar D Guerrero and Miguel A Labrador. On the applicability of available bandwidth estimation techniques and tools. *Computer Communications*, 33(1):11–22, 2010. 4

[GRT10] Emanuele Goldoni, Giuseppe Rossi, and Alberto Torelli. ASSOLO: an Efficient Tool for Active End-to-end Available Bandwidth Estimation. *International Journal On Advances in Systems and Measurements*, 2(4):283–292, 2010. 31

[HHCW10] Te-Yuan Huang, Polly Huang, Kuan-Ta Chen, and Po-Jung Wang. Could Skype be more satisfying? a QoE-centric study of the FEC mechanism in an internet-scale VoIP system. *Network, IEEE*, 24(2):42–48, March 2010. 16

[HMTG01a] C. V. Hollot, Vishal Misra, Donald F. Towsley, and Weibo Gong. On Designing Improved Controllers for AQM Routers Supporting TCP Flows. In *INFOCOM*, page 1726–1734, 2001. 37

[HMTG01b] C.V. Hollot, V. Misra, D. Towsley, and W.-B. Gong. A control theoretic analysis of RED. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, page 1510–1519, 2001. 37

[Hub12] Bert Hubert. *Linux Advanced Routing & Traffic Control HOWTO*, May 2012. 36

[HYC04] Gao Huang, Meng Ye, and Long Cheng. Modeling system performance in MMORPG. In *Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004. IEEE*, page 512–518. IEEE, 2004. 1, 38

[iee11] IEEE Standard for Local and metropolitan area networks–Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks–Amendment 17: Priority-based Flow Control. *IEEE Std 802.1Qbb-2011 (Amendment to IEEE Std 802.1Q-2011 as amended by IEEE Std 802.1Qbe-2011 and IEEE Std 802.1Qbc-2011)*, page 1–40, Sept 2011. 54

[iee12] IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007)*, page 1–2793, March 2012. 28

[Jac97] Van Jacobson. Pathchar: A tool to infer characteristics of Internet paths. Technical report, April 1997. 30

[JD03] Manish Jain and Constantinos Dovrolis. End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput. *IEEE/ACM Trans. Netw.*, 11(4):537–549, 2003. 30

[JD05] Hao Jiang and Constantinos Dovrolis. Why is the internet traffic bursty in short time scales? In Derek L. Eager, Carey L. Williamson, Sem C. Borst, and John C. S. Lui, editors, *SIGMETRICS*, page 241–252. ACM, 2005. 40

[KS08] U.R. Krieger and R. Schwessinger. Analysis and quality assessment of peer-to-peer IPTV systems. In *Consumer Electronics, 2008. ISCE 2008. IEEE International Symposium on*, page 1–4, april 2008. 19

[LDS06] Li Lao, Constantine Dovrolis, and M. Y. Sanadidi. The probe gap model can underestimate the available bandwidth of multihop paths. *Computer Communication Review*, 36(5):29–34, 2006. 30

[LGL08] Yong Liu, Yang Guo, and Chao Liang. A survey on peer-to-peer video streaming systems. *Peer-to-Peer Networking and Applications*, 1(1):18–28, March 2008. 18, 19

[MA01] Matt Mathis and Mark Allman. A framework for defining empirical bulk transfer capacity metrics. RFC 3148, July 2001. 29

[MA06] Padmavathi Mundur and Poorva Arankalle. Optimal server allocations for streaming multimedia applications on the Internet. *Computer Networks*, 50(18):3608–3621, 2006. 17

[MBG] Bob Melander, Mats Bjorkman, and Per Gunningberg. Regression-based available bandwidth measurements. In *International Symposium on Performance Evaluation of Computer and Telecommunications Systems*. 30

[MBG00] B. Melander, M. Bjorkman, and P. Gunningberg. A new end-to-end probing and analysis method for estimating bandwidth bottlenecks. In *Global Telecommunications Conference, 2000. GLOBECOM '00. IEEE*, volume 1, page 415–420 vol.1, 2000. 30

[MBM+10] J. Ruiz Mas, J. I. Aznar Baranda, J. M. Saldaña Medina, J. Fernández Navajas, B. Hernández Ortega, L. Blasco Arcas, and J. Jiménez Martínez. Evaluación de nuevos canales de distribución en servicios interactivos IP. Septiembre 2010. 17, 35

[MGT00] Vishal Misra, Weibo Gong, and Donald F. Towsley. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED. In *SIGCOMM*, page 151–160, 2000. 37

[MMFR96] Matt Mathis, Jamshid Mahdavi, Sally Floyd, and Allyn Romanow. TCP selective acknowledgment options. RFC 2018, October 1996. 29

[MP09] Abdulhussain E. Mahdi and Dorel Picovici. Advances in voice quality measurement in modern telecommunications. *Digital Signal Processing*, 19(1):79–103, 2009. 24

[NMNA06] E.A.V. Navarro, J.R. Mas, J.F. Navajas, and C.P. Alcega. Performance of a 3g-based mobile telemedicine system. In *Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE*, volume 2, page 1023–1027, Jan 2006. 23

[Par05] Kun I Park. *QoS in packet networks*. Springer, Boston, primera edition, 2005. 22

[PDMC03] R. Prasad, C. Dovrolis, M. Murray, and K. Claffy. Bandwidth estimation: metrics, measurement techniques, and tools. *Network, IEEE*, 17(6):27–35, Nov 2003. 29

[PNM+11] L. A. Casadesus Pazos, J. Fernández Navajas, J. Ruiz Mas, J. M. Saldana Medina, J. I. Aznar Baranda, and E. Viruete Navarro. Herramienta para Automatización de Medidas de Tiempo Real Extremo a Extremo. Leganés (España). ISBN 9788493393458. Sept. 2011. 81

[Pos81a] Jon Postel. Internet Protocol. RFC 791, September 1981. 22

[Pos81b] Jon Postel. Transmission Control Protocol. RFC 793, September 1981. 24

[QRSRM+13] Idelkys Quintana-Ramírez, Jose Saldana, José Ruiz-Mas, Luis Sequeira, Julián Fernández-Navajas, and Luis Casadesus. Optimization of P2P-TV traffic by means of header compression and multiplexing. In *Software, Telecommunications and Computer Networks (SoftCOM), 2013 21st International Conference on*, page 1–5, Croatia, September 2013. IEEE. 19

[RC04] Seungwan Ryu and Chulhyoe Cho. PI-PD-Controller for Robust and Adaptive Queue Management for Supporting TCP Congestion Control. In *Proceedings of the 37th Annual Symposium on Simulation*, ANSS '04, page 132–139, Washington, DC, USA, 2004. IEEE Computer Society. 37

[Rec96] ITU-T Rec. P. 800: Métodos de determinación subjetiva de la calidad de transmisión. *UIT-T*, 1996. 25

[Rec12] ITU-T Rec. G. 729: Coding of speech at 8 kbit/s using conjugate structure algebraic-code-excited linear-prediction (CS-ACELP). June 2012. 18

[Rec14] ITU-T Rec. G. 107: The E-Model a computational model for use in transmission planning. February 2014. 25, 41, 129

[RMN+10] J. Murillo Royo, J. M. Saldaña Medina, J. Fernández Navajas, J. Ruiz Mas, E. A. Viruete Navarro, and J. I. Aznar Baranda. Análisis de QoS para una Plataforma Distribuida de Telefonía IP. page 63–70, Valladolid. Septiembre 2010. 11, 12, 31

[RRB+03] Vinay Ribeiro, Rudolf Riedi, Richard Baraniuk, Jiri Navratil, and Les Cottrell. pathchirp: Efficient available bandwidth estimation for network paths. In *Passive and active measurement workshop*, volume 4, April 2003. 30

[RRQ03] Seungwan Ryu, C. Rump, and Chunming Qiao. Advances in internet congestion control. *Communications Surveys Tutorials, IEEE*, 5(1):28–39, Third 2003. 37

[RRQ04] Seungwan Ryu, Christopher Rump, and Chunming Qiao. Advances in Active Queue Management (AQM) Based TCP Congestion Control. *Telecommunication Systems*, 25(3-4):317–351, 2004. 36

[RSC+02] Jonathan Rosenberg, Henning Schulzrinne, Gonzalo Camarillo, Alan Johnston, Jon Peterson, Robert Sparks, Mark Handley, Eve Schooler, et al. SIP: session initiation protocol. RFC 3261, June 2002. 11

[SAV+09] José Ma Saldaña, José I. Aznar, Eduardo Viruete, Julián Fernández-Navajas, and José Ruiz. QoS Measurement-Based CAC for an IP Telephony System. In Novella Bartolini, Sotiris E. Nikoletseas, Prasun Sinha, Valeria Cardellini, and Anirban Mahanti, editors, *QSHINE*, volume 22 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, page 3–19. Springer, 2009. 12

[SBGW08] Joel Sommers, Paul Barford, Albert G. Greenberg, and Walter Willinger. An SLA perspective on the router buffer sizing problem. *SIGMETRICS Performance Evaluation Review*, 35(4):40–51, 2008. 35

[SF07] Thomas Silverston and Olivier Fourmaux. Measuring p2p iptv systems. In *Proc. of ACM NOSSDAV*, volume 7, 2007. 19

[SFNRM+11] J. Saldana, J. Fernández-Navajas, J. Ruiz-Mas, J.I. Aznar, E. Viruete, and L. Casadesus. Influence of the router buffer on online games traffic multiplexing. In *Performance Evaluation of Computer Telecommunication Systems (SPECTS), 2011 International Symposium on*, page 253–258, 2011. 3

[SFNRM+12a] José Ma Saldaña, Julián Fernández-Navajas, José Ruiz-Mas, Jenifer Murillo, Eduardo Viruete Navarro, and José I. Aznar. Evaluating the influence of multiplexing schemes and buffer imple-

mentation on perceived VoIP conversation quality. *Computer Networks*, 56(7):1893–1919, 2012. 41, 42

[SFNRM+12b] José Ma Saldaña, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro, and Luis Casadesus. Influence of online games traffic multiplexing and router buffer on subjective quality. In *CCNC*, page 462–466. IEEE, 2012. 42

[SFNRM+12c] José Ma Saldaña, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro, and Luis Casadesus. The effect of router buffer size on subjective gaming quality estimators based on delay and jitter. In *CCNC*, page 482–486. IEEE, 2012. 3

[SFNRM+12d] José Ma Saldaña, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro, and Luis Casadesus. The utility of characterizing packet loss as a function of packet size in commercial routers. In *CCNC*, page 346–347. IEEE, 2012. 42

[SFNSC12] L. Sequeira, J. Fernández-Navajas, J. Saldana, and L. Casadesus. Empirically characterizing the buffer behaviour of real devices. In *Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2012 International Symposium on*, page 1–6, 2012. 39

[SKK03] Jacob Strauss, Dina Katabi, and M. Frans Kaashoek. A measurement study of available bandwidth estimation tools. In *Internet Measurement Comference*, page 39–44. ACM, 2003. 30

[SKR07] P. Svoboda, W. Karner, and M. Rupp. Traffic Analysis and Modeling for World of Warcraft. In *Communications, 2007. ICC '07. IEEE International Conference on*, page 1612–1617, june 2007. 38

[SMFN+11a] J. Saldana, J. Murillo, J. Fernández-Navajas, J. Ruiz-Mas, E. Viruete Navarro, and J.I. Aznar. Evaluation of multiplexing and

buffer policies influence on VoIP conversation quality. In *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*, page 378–382, 2011. 12, 32

[SMFN+11b] José Ma Saldaña, Jenifer Murillo, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro, and José I. Aznar. QoS and Admission Probability Study for a SIP-Based Central Managed IP Telephony System. In *NTMS*, page 1–6. IEEE, 2011. 11

[SR12] Patrick Seeling and Martin Reisslein. Video Transport Evaluation With H.264 Video Traces. *IEEE Communications Surveys and Tutorials*, 14(4):1142–1165, 2012. 15

[SS10] Rade Stanojevic and Robert Shorten. Trading link utilization for queueing delays: An adaptive approach. *Computer Communications*, 33(9):1108–1121, 2010. 3

[Sta04] William Stallings. *Redes e Internet de alta velocidad Rendimiento y Calidad de Servicio*. Pearson Education, Madrid, segunda edition, 2004. 21, 23

[TCR11] Frederic Thouin, Mark Coates, and Michael Rabbat. Large scale probabilistic available bandwidth estimation. *Computer Networks*, 55(9):2065–2078, 2011. 4

[TP13] S. Tanwir and H. Perros. A Survey of VBR Video Traffic Models. *Communications Surveys Tutorials, IEEE*, 15(4):1778–1802, Fourth 2013. 15

[VDRK08] Geert Van der Auwera, Prasanth T David, Martin Reisslein, and Lina J Karam. Traffic and quality characterization of the H. 264/AVC scalable video coding extension. *Advances in Multimedia*, 2008(2):1, 2008. 16

[VG13] Ahmad Vakili and Jean-Charles Grégoire. QoE management for video conferencing applications. *Computer Networks*, 57(7):1726–1738, 2013. 39

[VGN+12] Alex Borges Vieira, Pedro Gomes, José Augusto Miranda Nacif, Rodrigo Mantini, Jussara M. Almeida, and Sérgio Vale Aguiar Campos. Characterizing SopCast client behavior. *Computer Communications*, 35(8):1004–1016, 2012. 42

[VS94] Curtis Villamizar and Cheng Song. High performance TCP in ANSNET. *SIGCOMM Comput. Commun. Rev.*, 24(5):45–60, October 1994. 34

[VS08] Arun Vishwanath and Vijay Sivaraman. Routers With Very Small Buffers: Anomalous Loss Performance for Mixed Real-Time and TCP Traffic. In Hans van den Berg and Gunnar Karlsson, editors, *IWQoS*, page 80–89. IEEE, 2008. 35

[VSR09] Arun Vishwanath, Vijay Sivaraman, and George Rouskas. Considerations for Sizing Buffers in Optical Packet Switched Networks. In *INFOCOM*, page 1323–1331. IEEE, 2009. 35

[VST09] Arun Vishwanath, Vijay Sivaraman, and Marina Thottan. Perspectives on router buffer sizing: recent results and open problems. *Computer Communication Review*, 39(2):34–39, 2009. 2, 34

[WKvVA06] A. F. Wattimena, R. E. Kooij, J. M. van Vugt, and O. K. Ahmed. Predicting the perceived quality of a first person shooter: the Quake IV G-model. In *Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games*, NetGames '06, New York, NY, USA, 2006. ACM. 26, 41, 50

[ZFP12] L. Zabala, A. Ferro, and A. Pineda. Modelling packet capturing in a traffic monitoring system based on Linux. In *Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2012 International Symposium on*, page 1–6, 2012. 111

[ZXH+13] Xinggong Zhang, Yang Xu, Hao Hu, Yong Liu, Zongming Guo, and Yao Wang. Modeling and Analysis of Skype Video Calls: Rate Control and Video Quality. *Multimedia, IEEE Transactions on*, 15(6):1446–1457, Oct 2013. 17

Declaración

Por la presente declaro que he producido esta obra sin la prohibición de terceros y sin hacer uso de los medios que no sean los especificados. Además, que no existe ningún conflicto de intereses en relación con la publicación de esta tesis. El autor no trabaja con ninguna de las empresas cuyos productos se citan en el presente trabajo, ni tiene ninguna relación comercial o asociación con dichas empresas.

Este trabajo de tesis se llevó a cabo a partir del año 2011 hasta el 2015 bajo la supervisión de Dr. Julián Fernández Navajas en la Universidad de Zaragoza.

Zaragoza, 28 de mayo de 2015

Luis Sequeira

Esta tesis se terminó de escribir en Zaragoza el
28 de mayo de 2015