

---

# El impacto del *buffer* en la calidad de servicio

Luis Sequeira

---

004

S773 -i Sequeira Villarreal, Luis Enrique  
El impacto del *buffer* en la calidad de servicio  
Luis Enrique Sequeira Villarreal– 1a. ed.  
– San José: Luis Sequeira, 2021.  
1 recurso en línea ; 1.6Mb.

ISBN 978-9968-49-653-7

1. Redes – 2. Comunicaciones informáticas  
– I. Título.

---

© Luis Sequeira 2021  
Primera edición

ISBN 978-9968-49-653-7



9 789968 496537

---

# Prefacio

En este texto se analiza la respuesta de la transmisión de flujos de datos 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.

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

El contenido es una versión revisada y editada de partes de la tesis doctoral “*Técnicas de estimación de buffer, centradas en las redes de acceso, para la transmisión de flujos IP en tiempo real*” [1] publicada en Zaragoza, España, 2015.

Se agradece a Idelkys Quintana por las revisiones detenidas del libro, señalando errores en el texto, sugiriendo mejoras y por todos sus valiosos comentarios.

Londres, Febrero 2021.

Luis Sequeira

---

# Índice general

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Cuellos de botella	3
1.2	Tráfico a ráfagas	6
1.3	Alcances	6

<b>2</b>	<b>La Calidad de Servicio</b>	<b>9</b>
2.1	Parámetros objetivos de QoS	11
	Retardo	11
	<i>Jitter</i>	12
	Pérdida de paquetes	12
2.2	Medidas objetivas y subjetivas de QoS	13
2.3	Disponibilidad	16
	Métricas relacionadas con el ancho de banda	17
	Técnicas de estimación del ancho de banda	20
	Medidas de disponibilidad	21

---

## **3** El *Buffer* **25**

- 3.1 Dimensionado 26
- 3.2 Disciplinas de gestión de colas 28
- 3.3 El rol del *buffer* en la QoS 32
  - El desbordamiento del *buffer* 33
  - Influencia del *buffer* en diferentes servicios 36

## **4** Los Servicios Multimedia **39**

- 4.1 VoIP 40
- 4.2 Videovigilancia 42
- 4.3 Videoconferencia 43
- 4.4 Video *streaming* 46
- 4.5 P2P-TV 48

## **5** El *Buffer* y las ráfagas **49**

- 5.1 Escenario de red propuesto 50
- 5.2 Tráfico utilizado 51
- 5.3 Análisis de la pérdida de paquetes 53
- 5.4 Distribución de la pérdida de paquetes 55

## **6** El *Buffer* y las aplicaciones **59**

- 6.1 Escenario de red propuesto 60
- 6.2 Tráfico utilizado 61
- 6.3 Análisis de pérdida de paquetes 62
  - Pérdida de paquetes del tráfico combinado 63
  - Pérdida de paquetes por flujo 65

---

6.4	Distribución de la pérdida de paquetes	67
6.5	MOS para llamadas de VoIP	70

## **7** **Conclusión** **75**

---

7.1	El comportamiento del tráfico	75
7.2	La QoS y los <i>buffer</i>	76

---



---

## 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.
DSL	Digital Subscriber Line.
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.
MOS	Mean Opinion Score.
MPEG-TS	Moving Picture Experts Group - Transport Stream.
MTU	Maximum Transfer Unit.
NAT	Network Address Translation.
NS	Network Simulator.
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.
SVC	Scalable Video Coding.
TC	Traffic Class.
TCP	Transport Control Protocol.
ToIP	Telephony over IP.
TOPP	<i>Trains Of Packet Pairs.</i>
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.

---

# CAPÍTULO 1

---

## Introducción

---

El desarrollo de aplicaciones o el despliegue de nuevos servicios que envían datos a red, requiere de una planificación desde el punto de vista de red, ya que esto permitirá garantizar un determinado nivel de calidad, lo cual repercutirá en el grado de satisfacción de los usuarios finales. En este ámbito, resulta curioso cómo profesionales de disciplinas diferentes, o incluso de ámbitos similares, pueden abordar un problema de maneras muy diferentes o identificar potenciales riesgos en soluciones viables. Por ejemplo, en el caso de una aplicación para videoconferencia, el desarrollador de software podría enfocarse en proveer la más alta calidad de video posible, para así obtener un mayor impacto en el mercado e incrementar el número de usuarios satisfechos con una interfaz gráfica de vanguardia y una experiencia de alta calidad en el video. Al mismo tiempo, un profesional de redes podría señalar un potencial riesgo en el deterioro de la calidad debido a la gran cantidad de datos que se debe transmitir en tiempo real desde un usuario a otro utilizando una red como Internet. En una red descentralizada como esta, no se tiene control de la configuración óptima de los dispositivos de red intermedios entre los usuarios finales,

## CAPÍTULO 1. INTRODUCCIÓN

---

y por lo tanto, no se pueden emplear ciertas técnicas de ingeniería de tráfico en los equipos que conforman el camino de red entre dos nodos.

Una posible solución para satisfacer los diferentes puntos de vista descritos anteriormente, podría ser desarrollar la aplicación de manera que se pueda adaptar a diferentes condiciones de la red, las cuales podrían cambiar incluso durante una misma sesión de usuario. La calidad variaría en función de la capacidad o de las limitaciones de la red. La máxima calidad en términos de video se presentaría bajo condiciones óptimas de la red. Por el contrario, la calidad de video bajaría cuando la red presente peores condiciones. Claramente no es el caso ideal, pero el usuario no perdería la sesión y podría seguir utilizando el servicio.

El ejemplo anterior no es un caso aislado si se tiene en cuenta el amplio crecimiento del número de usuarios y de los nuevos servicios multimedia e interactivos en Internet (por ejemplo: video bajo demanda, videoconferencia, juegos en línea, Voice over Internet Protocol (VoIP), *streaming*, videovigilancia, etc.). Estos servicios generan una cantidad significativa de tráfico en la red [2, 3]. Además, la expectativa de crecimiento para estas aplicaciones, indica que la tendencia de uso se incrementará. Además, los usuarios de dichos servicios 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 Quality of Service (QoS) que estos servicios ofrecen para sus aplicaciones; especialmente cuando las tecnologías de acceso deben soportar aplicaciones y servicios multimedia en tiempo real.

En este ámbito, el presente libro ofrece una serie de análisis con ejemplos sencillos y prácticos, enfocados para que profesionales de

diversas áreas puedan comprender el comportamiento del tráfico de diversos servicios y su impacto en la QoS, ya que tiene un efecto 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 el tamaño de la información.

La calidad se describe desde el punto de vista de los *buffer*, debido a que son utilizados como mecanismos de regulación de tráfico en los dispositivos de red. Además, cuando se discute la planificación de una red, el tamaño del *buffer* de los nodos 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 y la calidad de aplicaciones y servicios. Existen muchos estudios en relación al dimensionado de *buffer* [4], pero están especialmente enfocados a los *router* del núcleo de la red y para flujos Transport Control Protocol (TCP), trabajan sobre diferentes estructuras de colas y no contemplan en detalle el comportamiento diferente de los *buffer*.

### 1.1. Cuellos de botella

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, incluso en la nube. En estos puntos, generalmente en el *router* de acceso, la principal causa de pérdida de

## CAPÍTULO 1. INTRODUCCIÓN

---

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, donde el trabajar desde casa es ahora muy común (principalmente durante el inusual año 2020), o para las pequeñas 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 que estos puedan tener. Así, las características de diseño del *buffer* del nodo de la red y las políticas de gestión que este 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 tener en cuenta los parámetros y el comportamiento del *buffer* junto a 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 pueden verse afectados por el comportamiento de los *buffer* que se encuentran entre los equipos terminales [5, 6]. 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 Active Queue Management (AQM), 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 Random Early Detection (RED) y algunos derivados Adaptive Random Early Detection (ARED) o Weighted Random Early Detection (WRED), pero estos algoritmos requieren



de un ajuste cuidadoso de sus parámetros con el fin de proveer un buen rendimiento [7]. También, existen algoritmos de planificación de la QoS como Weighted Fair Queuing (WFQ), 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, los cuales no suelen implementar técnicas avanzadas de gestión de tráfico, e incluso, en la mayoría de los equipos sólo hay un *buffer* tipo First In First Out (FIFO) [8].

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 User Datagram Protocol (UDP), 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 Available Bandwidth Estimations Techniques and Tools (ABETT) [9, 10], 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.

### 1.2. Tráfico a ráfagas

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, podrían 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 conformado de 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.

### 1.3. Alcances

Como se ha mencionado anteriormente, este libro pretende introducir al lector, con una serie de conceptos básicos y ejemplos, en el rol que juega el *buffer* de los dispositivos de red en la QoS. Lo cual es un aspecto a tener en cuenta a la hora de realizar el desarrollo de aplicaciones, despliegue de servicios, la planificación de una red o cuando se quiere proveer ciertos niveles de QoS.

El capítulo 2 presenta brevemente las definiciones más comunes de los parámetros QoS, como el retardo, *jitter* y la pérdida de pa-

quetes; además describe algunas medidas objetivas y subjetivas de la calidad. Finalmente se comentan algunas métricas y técnicas de estimación del ancho de banda disponible. El capítulo 3 describe los principales conceptos de los *buffer*, donde se definen varias técnicas de dimensionado y disciplinas de colas. Además, se habla del efecto del desbordamiento y la influencia en diferentes servicios. También, 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. Por otro lado, se han seleccionado los servicios VoIP, videovigilancia, videoconferencia, video *streaming* y Peer-to-Peer Television (P2P-TV), a modo de ejemplo, para analizar los aspectos más importantes que definen la manera en que dichos servicios envían tráfico a la red; esto se comenta en el capítulo 4.

Los capítulos 5 y 6 tienen un enfoque diferente, un poco más práctico a modo de ejemplos. Se han definido dos casos de uso de flujos Internet Protocol (IP) en tiempo real que pueden considerarse comúnmente utilizados a nivel residencial, pequeñas empresas, ciertos modelos de negocios y grupos de usuarios. El objetivo es analizar la respuesta en la transmisión de los flujos IP en tiempo real cuando estos comparten un enlace con servicios que generan tráfico a ráfagas y sus repercusiones en la QoS, debido al efecto del *buffer*. Además, valorar el aumento de la capacidad de la red interna como posible solución bajo estas condiciones. Finalmente, el capítulo 7 resume los aspectos más relevantes presentados en este libro.

## CAPÍTULO 1. INTRODUCCIÓN

---

## CAPÍTULO 2

---

### La Calidad de Servicio

---

Las 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 [11], como por ejemplo el tráfico de sitios web de Internet y servicios como File Transfer Protocol (FTP). 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 con requerimientos diferentes, así como a un aumento importante en la cantidad de usuarios a nivel global. Muchos de los servicios que se utilizan en la actualidad, son en tiempo real o tienen requerimientos de cierta interactividad, como sucede en los juegos en línea, por mencionar un ejemplo. 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 muchos servicios interactivos.

La congestión es un 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 en un momento dado, y por

lo tanto, generar pérdida de paquetes, degradando la calidad de los servicios. Para hacer frente a esta situación no basta con incrementar la capacidad de una red o en algunos casos no es posible. En dichos casos, se hace necesario implementar mecanismos que gestionen el tráfico y controlen la congestión. Para satisfacer estas necesidades, la Internet Engineering Task Force (IETF) ha desarrollado un conjunto de estándares bajo el marco general de Integrated Service Architecture (ISA) [12]. 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 [13] 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 Type of Service (ToS) de la cabecera IPv4 [14] o Traffic Class (TC) en IPv6 [15]. Esto significa que la negociación será realizada para todo el tráfico de una red, ya sea un Internet Service Provider (ISP) o una empresa. A dichas negociaciones se les llama Service Level Agreement (SLA). 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 estos 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*.

## 2.1. 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.

### 2.1.1. Retardo

El retardo es un parámetro crítico para ciertas aplicaciones y servicios en tiempo real (por ejemplo en aplicaciones de robótica en la nube, la telecirugía [16] o incluso los vehículos conectados [17, 18]); es también un factor de importancia en el diseño de las redes [19, 20] 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 [21]:

- 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 One-Way Delay (OWD) para referirse al retardo en un sentido de la comunicación (de fuente a destino), además se utiliza Round-Trip Time (RTT) 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. Otro caso más crítico es el de la telecirugía donde los dispositivos hápticos necesitan una respuesta en términos de latencia de menos de 1 *ms* [16], lo cual es muy bajo.

### 2.1.2. *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 de 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 [11]. 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* [22], 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 estas 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.

### 2.1.3. 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 [23], pero esto 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.

## 2.2. 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 (por ejemplo redes móviles o banda ancha). Por otro lado, muchas empresas implementan herramientas de monitorización (usualmente basadas en la nube) de la actividad de los usuarios en los servicios que proveen, la idea es mejorar la experiencia del usuario, ayudar a detectar y detener posibles amenazas de seguridad, entre otras funciones.

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 ellos [24]. 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 medidas objetivas utilizan modelos matemáticos para determinar los niveles de calidad que 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 servicios que transporta una red, se puede decir, que es necesario un modelo diferente por cada tipo de servicio o aplicación.

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, como se muestra en el Ejemplo 2.1 y el Ejemplo 2.2.

### **Ejemplo 2.1 Medida subjetiva de calidad para voz.**

El Mean Opinion Score (MOS) 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 International Telecommunication Union (ITU). El MOS está definido en la recomendación P. 800 [25], en la cual cada percepción de los usuarios se clasifica en una escala subjetiva de 1 a 5, ver Figura 2.1.

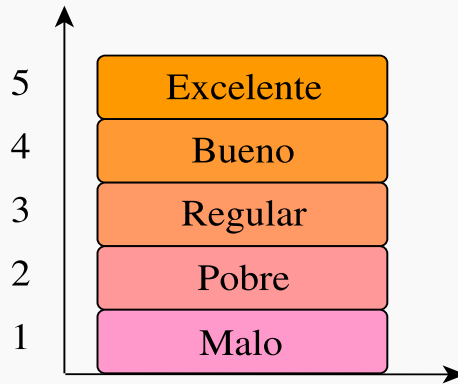


Figura 2.1: Escala de calidad para el MOS según ITU-T.

### Ejemplo 2.2 Medida de calidad utilizando parámetros objetivos.

El *E-model* ofrece una alternativa al MOS y consiste en otra recomendación de la ITU-T [26], la cual se usa como instrumento para estimar un posible nivel de calidad en función de distintos parámetros de QoS medidos de forma objetiva (retardo, pérdidas, etc.), proporcionando un medio para estimar el MOS [27]. 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.

Los juegos en línea forman parte de otra área donde la calidad es muy importante, ya que los usuarios pueden tener una gran interactividad con el juego o con otros usuarios. En estos casos es necesario crear métricas de calidad personalizadas para cada juego en particular con la finalidad de obtener una idea de la calidad que experimentan los jugadores, ver Ejemplo 2.3.

### Ejemplo 2.3 Medida de calidad para *Quake IV*

En [28], 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 First Person Shooter (FPS) 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.

## 2.3. Disponibilidad

En términos generales, la disponibilidad se refiere a cuánto tiempo un dispositivo o sistema está operativo respecto al 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 Peer-to-Peer (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 esta sección se presentan algunas de estas técnicas, así como, herramientas de medición de los parámetros mencionados.

### 2.3.1. Métricas relacionadas con el ancho de banda

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 [29], como sucede en los sistemas inalámbricos IEEE 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:

$$t_{L_3} = \frac{L_{L_3} + H_{L_3} + H_{L_2}}{C_{L_2}} \quad (2.1)$$

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

$$\begin{aligned} C_{L_3} &= \frac{L_{L_3} + H_{L_3}}{t_{L_3}} \\ &= \frac{L_{L_3} + H_{L_3}}{\frac{L_{L_3} + H_{L_3} + H_{L_2}}{C_{L_2}}} \\ &= C_{L_2} \frac{L_{L_3} + H_{L_3}}{L_{L_3} + H_{L_3} + H_{L_2}} \\ &= C_{L_2} \frac{1}{1 + \frac{H_{L_2}}{L_{L_3} + H_{L_3}}} \end{aligned} \quad (2.2)$$

Nótese que la capacidad IP descrita en la ecuación 2.2 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 libro, 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 2.4 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 encabeza-

do,  $H_{L_2}$ , de 38 *bytes*, la capacidad IP,  $C_{L_3}$ , es

$$C_{L_3} = 10 \text{ Mbps} \times \frac{1}{1 + \frac{38 \text{ bytes}}{150 \text{ bytes}}} = 7,97 \text{ Mbps}$$

mientras que si el paquete tiene un tamaño de 1500 *bytes*, la capacidad sería

$$C_{L_3} = 10 \text{ Mbps} \times \frac{1}{1 + \frac{38 \text{ bytes}}{1500 \text{ bytes}}} = 9,75 \text{ Mbps}$$

Por otro lado, en una comunicación serial (como Ethernet, WiFi, etc.) solamente es posible enviar un *bit* a la vez, el cual se envía por el enlace, a la máxima tasa que se puede alcanzar en un determinado instante, desde este punto de vista la utilización del enlace solamente tiene dos estados: ocupado o libre. Por este motivo, el ancho de banda disponible requiere ser definido en términos de una media de tiempo de la utilización instantánea, en una determinada transmisión o período de prueba.

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 anteriormente, 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 [30, 31] 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 estas comparten un mismo enlace.

- La implementación de los mecanismos de control de congestión que se incluyen en la recomendación del RFC 3782 [32], incluida cada variante de TCP (como, Tahoe, Reno, New Reno [33], SACK [34], etc.) que permite alcanzar un nivel diferente de *throughput*.
- 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.
- 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 determinados 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 en una misma sesión, incluso a lo largo del día en función de la carga de la red.

### 2.3.2. Técnicas de estimación del ancho de banda

El desarrollo de las técnicas de estimación de ancho de banda es un tema muy estudiado. En [35] y [36] se proponen las primeras herramientas para la estimación Variable Packet Size (VPS). Además, se encuentra una gran cantidad de técnicas para la estimación del ancho de banda como Packet Pair/Train Dispersion (PPTD) [37, 38], Self-Loading Periodic Streams (SLoPS) [39] y *Trains Of Packet Pairs* (TOPP) [40, 41]. Estas técnicas son conocidas como ABETT y la mayoría de ellas se clasifican en dos grandes tendencias: Probe Gap Model (PGM) y Probe Rate Model (PRM).

Los métodos PGM 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 [42], además, asumen que sólo existe un cuello de botella de extremo a extremo. En [43], 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.

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 [44]. 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 [45], se presenta una herramienta que 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.

### 2.3.3. Medidas de disponibilidad

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 (ver Ejemplo 2.5) que es una de las 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.

### **Ejemplo 2.5 Telefonía IP**

En [46], los autores estiman diferentes parámetros de calidad en la implementación de un sistema Call Admission Control (CAC) 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 [47], 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 procesa-

miento en ambos extremos de la comunicación lo que disminuye la calidad.

En general, un análisis similar puede realizarse para otros servicios como la videoconferencia o los sistemas de videovigilancia, por mencionar algunos que se analizarán más adelante. Esto debido a que la probabilidad de bloqueo (y por lo tanto, la disponibilidad del servicio) es un parámetro que puede utilizarse indistintamente en diferentes servicios.

## CAPÍTULO 2. LA CALIDAD DE SERVICIO

---

## CAPÍTULO 3

---

### El *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* (incluso más, pero no son comunes en redes de acceso), una red WiFi, puede tener una velocidad desde 2 hasta 54 *Mbps* para 802.11g o hasta 600 *Mbps* para IEEE 802.11n, incluso pudiendo superar 1 *Gbps* para IEEE 802.11ac. Por otro lado, las tecnologías asimétricas de acceso como cable módem y Asymmetric Digital Subscriber Line (ADSL) 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 Internet eXchange Point (IXP) 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 velocidades 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 estos 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.

### 3.1. Dimensionado

Desde 1994, en [48] se propuso la denominada *rule of thumb* o también llamada Bandwidth Delay Product (BDP), 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 3.1, 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 3.1 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 (ver Ejemplo 3.1), además, no se tuvo en cuenta el caso de flujos con RTT diferentes.

$$B = C \times RTT \tag{3.1}$$

En 2004, esta regla fue puesta en duda, por el llamado *Stanford model* [49] o también denominado *small buffer* [4], que reduce el tamaño del *buffer*, dividiéndolo por la raíz cuadrada del número de flujos TCP,  $N$ , como se muestra en la ecuación 3.2. 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.

$$B = \frac{C \times RTT}{\sqrt{N}} \quad (3.2)$$

### Ejemplo 3.1 La *rule of thumb* y el *Stanford model*

La cantidad de memoria necesaria para un *buffer*, según la BDP, para una capacidad de 40 *Gbps*, y un RTT de 250 *ms*, es de:

$$B = 40 \text{ Gbps} \times 250 \text{ ms} = 10 \text{ Gbits} = 1,25 \text{ Gbytes}$$

De acuerdo con el *Stanford model* y suponiendo 100 flujos, la cantidad de memoria necesaria sería:

$$B = \frac{40 \text{ Gbps} \times 250 \text{ ms}}{\sqrt{100}} = 1 \text{ Gbits} = 125 \text{ Mbytes}$$

Debido al modelo propuesto por [49] se generó una serie de investigaciones en este ámbito. En [50] 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%. Esto, 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, esto deja un elemento de incertidumbre en cuanto a los modelos de dimensionado de *buffer* para el tráfico de las aplicaciones que utilizamos hoy en día.

Por otro lado, son pocos 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 [51]. En [52] y [53] 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 [54] se presentó una simulación mediante Network Simulator (NS), 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, se dice que un *buffer* es un espacio de memoria en el cual se puede almacenar un determinado número de paquetes. Entonces, la cantidad de paquetes que se puede almacenar en un *buffer* depende del tamaño, tanto de la memoria del *buffer* como del paquete. Por este motivo, algunos investigadores y fabricantes realizan el dimensionado de los *buffer* en términos de *bytes*, mientras que otros, lo hacen en paquetes como se puede observar en [55].

## 3.2. 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 [56]. 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 Class Based Queueing (CBQ) o colas de prioridad Priority Queueing (PQ).

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 (ver Figura 3.1), el cual es un tipo de cola que puede ser configurada en cualquier interfaz de red en los sistemas operativos Linux [57]. 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 de la cabecera IP.

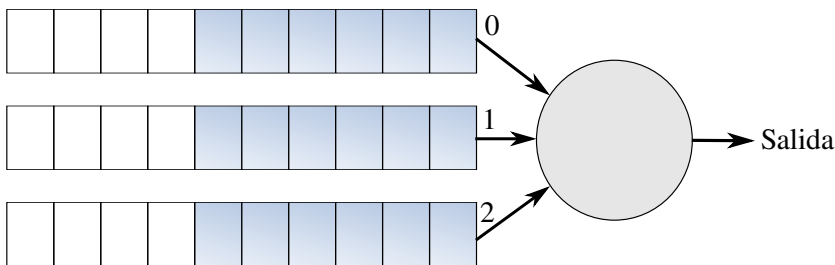


Figura 3.1: Descripción de un *buffer* FIFO-fast.

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 [58], 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 [7], el cual fue desarrollado para TCP mediante el reemplazo de las colas *drop-tail*. Los principales objetivos de RED fueron detectar la congestión cuando esta se está iniciando, lograr una equidad entre los flujos a ráfagas que tienen comportamientos diferentes, controlar la latencia, la sincronización global [59], reducir al mínimo la pérdida de paquetes y proporcionar altos niveles de utilización del enlace.

Existe una gran cantidad de implementaciones diferentes de RED [60], 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, ver Figura 3.2.

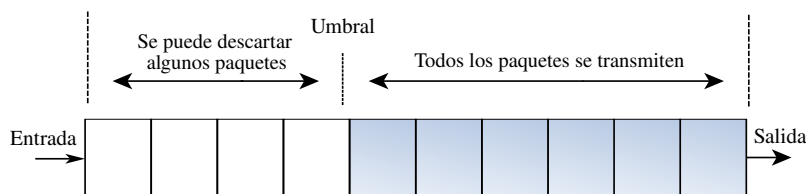


Figura 3.2: Descripción de un *buffer* tipo RED.

El principal problema de esta técnica es la dificultad del ajuste óptimo de sus parámetros para un adecuado funcionamiento [61, 62] 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 [63], 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 de *jitter*, deteriorando el rendimiento global de la red. A este fenómeno se le conoce como *bufferbloat* (como se ha mencionado con anterioridad) 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, 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, los cuales se usan en el núcleo de la red. Como se mencionó en la sección 1, la idea fundamental del libro es presentar 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 y baja) en la QoS para casos de uso a nivel residencial, pequeñas empresas o

grupos de usuarios.

### 3.3. El rol del *buffer* en la QoS

En la actualidad existe un gran número de usuarios y dispositivos utilizando servicios multimedia que generan una cantidad de tráfico significativa en Internet [2, 3] 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 [64]. 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 web, *email* incluso FTP (o servicios en la nube con protocolos similares para la transferencia y sincronización de archivos).

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 describirá en el capítulo 4, 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 [65, 66], 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 [67].

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.

### 3.3.1. El desbordamiento del *buffer*

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 3.3, 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 [68, 69].

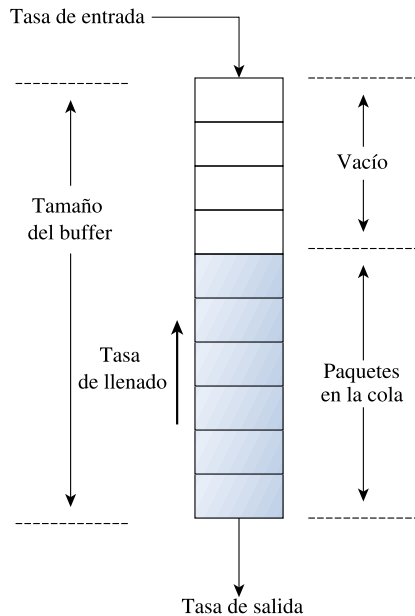


Figura 3.3: Principales características de los *buffer*.

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 este puede entrar en congestión 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, ver Ejemplo 3.2. Es cierto que muchas aplicaciones implementan mecanismos para generar un tráfico suavizado, pero el tráfico global de Internet tiene un comportamiento a ráfagas en todas las escalas [70].

### **Ejemplo 3.2 Desbordamiento del *buffer* debido a ráfagas**

Si una ráfaga con 20 paquetes de video llega a un *buffer* que tiene un tamaño de 15 paquetes, hay 5 paquetes que se descartarán ya que no hay suficiente espacio en el *buffer* para almacenarlos a todos.

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, como se muestra en el Ejemplo 3.3. 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.

### Ejemplo 3.3 Desbordamiento del *buffer* debido a la tasa de entrada

Una red envía datos a una tasa constante de 5 *Mbps* por un enlace de acceso a Internet de 5 *Mbps*. En este caso, la tasa de llenado es de:

$$R_{fill} = 5 \text{ Mbps} - 5 \text{ Mbps} = 0 \text{ Mbps}$$

Esto significa que el *buffer* no se está llenando y puede gestionar la entrega del tráfico sin ningún problema. Si la red incrementa la tasa de envío a 10 *Mbps* y el enlace de acceso se mantiene en el mismo valor, la nueva tasa de llenado sería:

$$R_{fill} = 10 \text{ Mbps} - 5 \text{ Mbps} = 5 \text{ Mbps}$$

Inicialmente el *buffer* encolará los paquetes que no se pueden transmitir en ese momento a una tasa de 5 *Mbps* hasta que el espacio de memoria lo permita. Si la tasa de entrada se mantiene, el *buffer* se llenará completamente y descartará el 50 % del tráfico.

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.

### 3.3.2. 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 [71, 72].

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, el retardo, el *jitter* y la pérdida de paquetes. También se usan evaluaciones de la calidad subjetiva para determinar la percepción de los usuarios para ciertos servicios. Como se mencionó en el capítulo 2, el *E-Model* de la ITU [26, 27], presenta un procedimiento con el objetivo de calcular el MOS, el cual es útil en el planeamiento de transmisión de red. Otros autores [28], 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. A continuación se mencionan algunos de dichos estudios.

- La influencia del *buffer* en VoIP se ha estudiado en [73], 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* [27]. 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 [74], 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 [75], como los servicios de correo, web o FTP. 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 en dichos dispositivos, si estos no pueden gestionar suficientes paquetes por segundo [76]. En este escenario, 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 [77], 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.

## CAPÍTULO 4

---

### Los Servicios Multimedia

---

En la actualidad los usuarios de Internet demandan una gran variedad de servicios multimedia, muchos de ellos con estrictos requerimientos de tiempo real. Además, la pandemia del virus COVID-19 (que inició en el año 2020) ha forzado a millones de personas en todo el mundo a trabajar de manera remota, usualmente desde casa, haciendo uso extensivo de aplicaciones de videoconferencia sobre redes de acceso con mediana o baja capacidad, en muchos casos. También, destacan otras clases de aplicaciones, por ejemplo, la telefonía por Internet, que debido a la reducción de costes ha alcanzado un gran auge. Por otro lado, se pueden resaltar 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 estas, en cuanto al tráfico generado en la red. En este sentido, se desea conocer el tamaño de los paquetes, si el tráfico es a ráfagas y el tamaño de dichas ráfagas, el tiempo entre paquetes, protocolos de transporte o incluso protocolos

a nivel de aplicación que den una idea de cómo una determinada aplicación genera tráfico a la red. Esta información es útil para estimar el impacto que un determinado tipo de tráfico puede tener en la red.

### 4.1. 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 Pequeñas y Medianas Empresas (PYMES) [78, 46]. 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 Session Initiation Protocol (SIP) [79], también, se encuentran implementaciones normalizadas con H.323 o propietarias que se hacen públicas como Inter-Asterisk eXchange protocol (IAX), y por último otras que no se hacen públicas como las de Skype.

Centrándose en SIP, este es uno de los protocolos con mayor impacto en la implementación de Telephony over IP (ToIP) [80, 46]. 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 [47]. Por otro lado, para la transmisión de datos en tiempo real, generalmente se hace uso del protocolo Real-time Transport Protocol (RTP); dicho protocolo se encarga del control de la transmisión en las sesiones de aplicaciones multimedia y utiliza como protocolo de transporte UDP. La Figura 4.1 muestra la estructura de paquetización de VoIP entre dos equipos terminales y el Ejemplo 4.1 describe cómo calcular el ancho de banda a nivel IP de una transmisión de voz. 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.

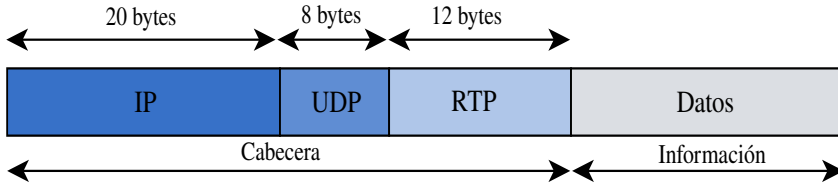


Figura 4.1: Descripción de un paquete VoIP transmitido entre dos estaciones de trabajo.

#### Ejemplo 4.1 Ancho de banda para un flujo VoIP

Al capturar una traza de tráfico VoIP, 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  $20\text{ ms}$  con una desviación estándar de  $\pm 0,62\text{ ms}$ . Cada conexión de este flujo tienen un consumo de ancho de banda a nivel IP de:

$$BW = \frac{IP + UDP + RTP + Datos}{muestras \times 10ms} = \frac{60 \times 8}{20 \times 10^{-3}} = 24\text{ Kbps}$$

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

Por otro lado, hay algunos estudios que relacionan el comportamiento de los usuarios (por ejemplo, a la hora de iniciar, cerrar o reiniciar una sesión en determinadas aplicaciones) y la estabilidad de la red [81], 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. 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, en la distribución de los recursos de la red y el control de congestión.

### 4.2. Videovigilancia

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 las funciones más usuales, se encuentran, por ejemplo, la activación mediante movimiento o sensores, control remoto y gestión a través de Hypertext Transfer Protocol (HTTP). Cada una de dichas funciones produce un efecto diferente en el tráfico generado a la red. El formato de imagen más usual es Joint Photographic Experts Group (JPEG) 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 *frames* por segundo para Phase Alternating Line (PAL)/National Television System Committee (NTSC). Puede trabajar sin problemas sobre una red con ancho de banda de 10 *Mbps* o 100 *Mbps* [82].

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 este 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.

El modelado de tráfico es muy útil para estimar y comprender un determinado flujo de datos. Una manera 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 4.1. 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 × 576 <i>pixeles</i>	50 <i>Kbytes</i>	25
	16 <i>Kbytes</i>	10
352 × 288 <i>pixeles</i>	13 <i>Kbytes</i>	9
	4 <i>Kbytes</i>	3

Tabla 4.1: Cantidad de paquetes por ráfaga en función de la compresión para una cámara IP AXIS 2120.

Otra función que en la actualidad la mayoría de estas cámaras tienen, son 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 en la imagen percibida por la cámara, la cantidad de paquetes enviados lógicamente aumenta.

### 4.3. 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 las redes de acceso. 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. 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 basa en que existen diferentes *codec* que consiguen mayor calidad pero enviando más información. Algunos sistemas de videoconferencia utilizan tecnologías de compresión Scalable Video Coding (SVC), 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 Adaptive Video Layering (AVL), 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, se podría recibir una cantidad de capas menor, manteniendo el servicio a pesar de tener una calidad relativamente inferior [83]. 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 [84]. La ventaja de este 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 envían a la red depende del tamaño del *frame*, y este a su vez, depende del modelo de codificación utilizado y el movimiento del video. En [85], 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.

A continuación se mencionan algunos ejemplos de estudios en relación al análisis del tráfico de aplicaciones de videoconferencia.

- Uno de los principales ejemplos de la videoconferencia (que por cierto, sigue una arquitectura P2P) es Skype. En [86], se analizan las principales funciones de Skype como *login*, Network Address Translation (NAT), 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 [87], los autores comentan que esta aplicación reacciona diferente ante la pérdida de trayectoria y la congestión de la red, además, que 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 [88, 89], analizan la calidad de las llamadas de voz y la Quality of Experience (QoE), proponiendo un modelo para cuantificar el nivel de satisfacción de los usuarios. Algunos [90] han propuesto mecanismos para el control de la congestión en el tráfico de VoIP de Skype.

- Con respecto a la capacidad de respuesta de las llamadas de video de Skype, los autores de [91] midieron las variaciones del ancho de banda y llegaron a la conclusión de que el tiempo de respuesta 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 este es alcanzado. Además, los autores de [92] caracterizaron los sistemas de control de velocidad y la calidad de las video llamadas, mostrando que dicha aplicación es robusta 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 [31], 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.

### 4.4. Video *streaming*

El crecimiento a nivel mundial en el acceso a Internet por parte de los usuarios móviles ha generado el desarrollo de diversas aplicaciones y nuevos modelos de negocios utilizando servicios de *streaming*, entre ellos se encuentran muchas plataformas de redes sociales, la radio y la televisión por Internet (por mencionar algunos) [51] y [93]. 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 Moving Picture Experts Group - Transport Stream (MPEG-TS). También, existen estudios que han caracterizado y medido el impacto en el tráfico de la red para aplicaciones de video *streaming* [94].

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 Transport Stream (TS) 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* [95] y los paquetes IP deben contener múltiplos de estos, por lo tanto el mayor tamaño posible si consideramos un Maximum Transfer Unit (MTU) 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 *streaming* de televisión es un servicio más sensible a los retardos debido a sus características, como se ve por ejemplo en los programas en vivo.

### 4.5. P2P-TV

Internet Protocol Television (IPTV) 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 [94]. Los sistemas P2P-TV son una técnica de difusión de contenidos usando una arquitectura P2P. 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 [96].

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 [97], 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 este 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 [98]. El comportamiento del tráfico de un *peer* se caracterizó en [99], 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.

## CAPÍTULO 5

---

### El *Buffer* y las 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 estas 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* (cuando se mantiene la misma capacidad del enlace de salida) 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.

### 5.1. 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 5.1, en la cual varias comunicaciones de videovigilancia comparten 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*.

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*, dicho valor se ha seleccionado con el fin de establecer la utilización 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*

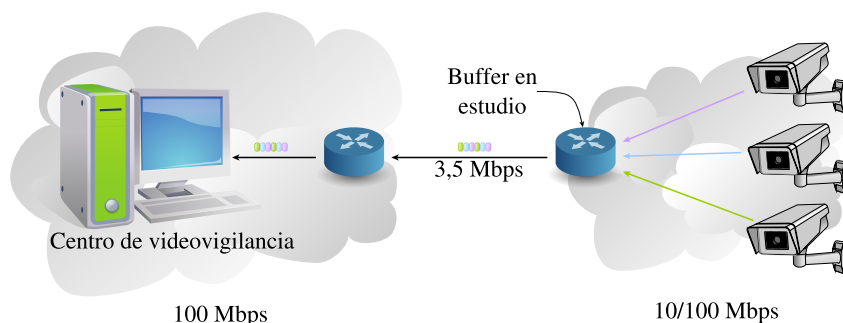


Figura 5.1: Escenario para las pruebas de uno, dos y tres flujos de datos de cámaras IP.

y no por la escasez del ancho de banda.

## 5.2. 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, utilizando los mismos tamaños de paquetes y tiempos entre paquetes. La metodología utilizada para la captura de tráfico se ilustra en la Figura 5.2, en la cual se incluye un *sniffer* en la mejor ubicación para que no degrade el rendimiento de las aplicaciones [100].

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 5.1 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

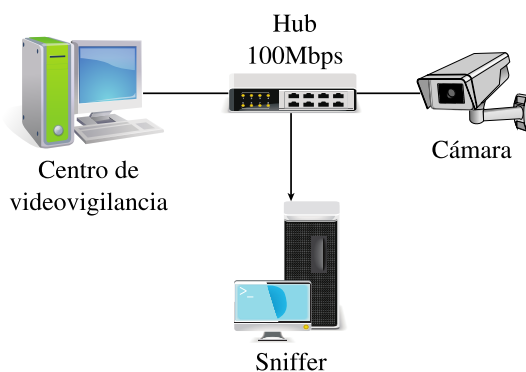


Figura 5.2: Escenario para la captura del tráfico para un sistema de videovigilancia.

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  $0,278 \text{ s} \pm 0,06 \text{ s}$ , 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
$704 \times 576$ <i>pixeles</i>	50 <i>Kbytes</i>	41
	32 <i>Kbytes</i>	26
	16 <i>Kbytes</i>	10
$352 \times 288$ <i>pixeles</i>	13 <i>Kbytes</i>	9
	4 <i>Kbytes</i>	3

Tabla 5.1: Cantidad de paquetes observados por ráfaga, dependiendo del nivel de compresión de la cámara.



### 5.3. Análisis de la 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 5.3 y 5.4 (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

## CAPÍTULO 5. EL *BUFFER* Y LAS RÁFAGAS

---

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.

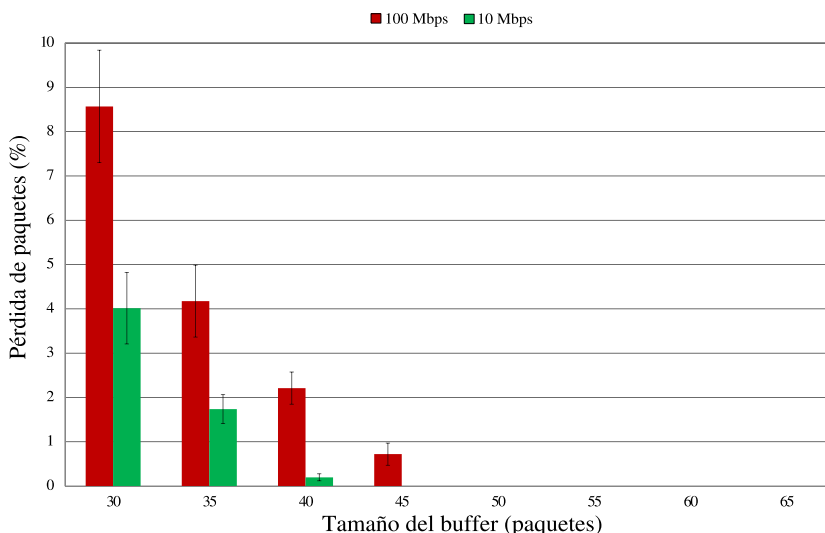


Figura 5.3: Relación entre el tamaño del *buffer* y la pérdida de paquetes para dos flujos de cámara IP.

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 5.3 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

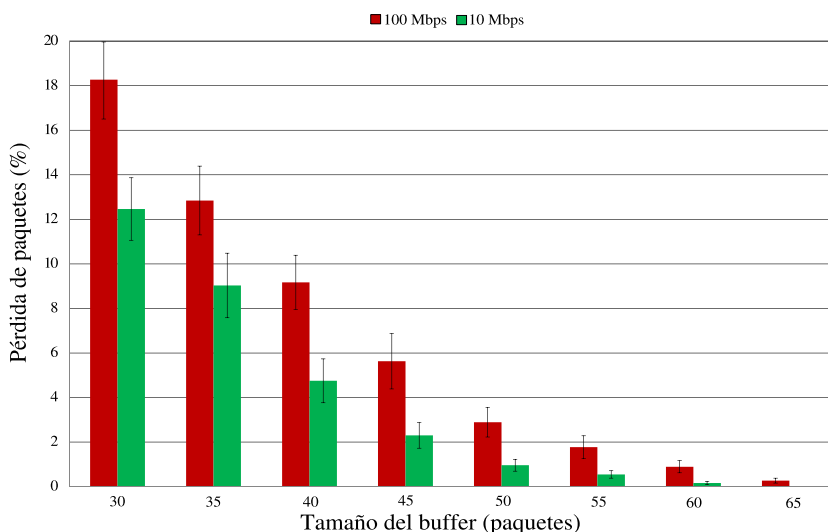


Figura 5.4: Relación entre el tamaño del *buffer* y la pérdida de paquetes para tres flujos de cámara IP.

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.

## 5.4. Distribución 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 se-

leccionado los resultados correspondientes a un *buffer* con tamaño de 40 paquetes en una red de 100 *Mbps*. Las Figuras 5.5 y 5.6 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.

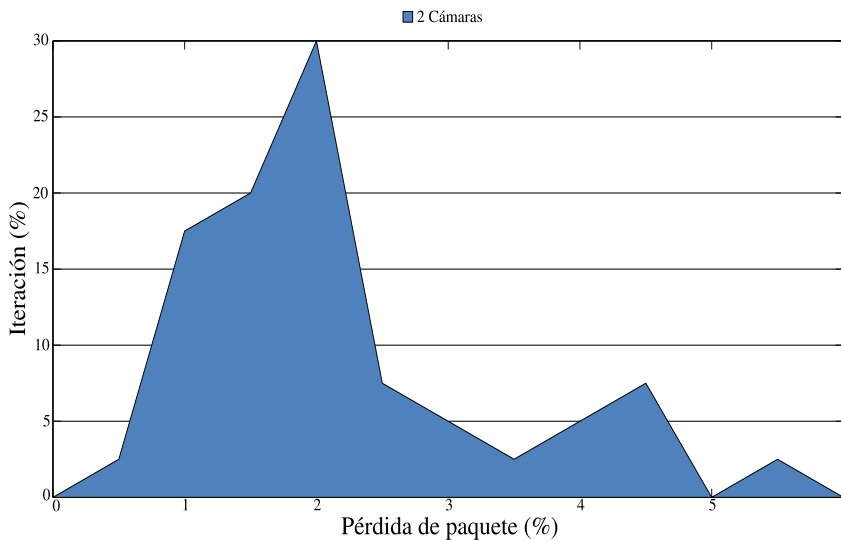


Figura 5.5: Pérdida de paquetes para dos flujos de cámara IP que atraviesan un *buffer* de 40 paquetes en una red de 100 *Mbps*.

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 5.5 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 5.6 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.

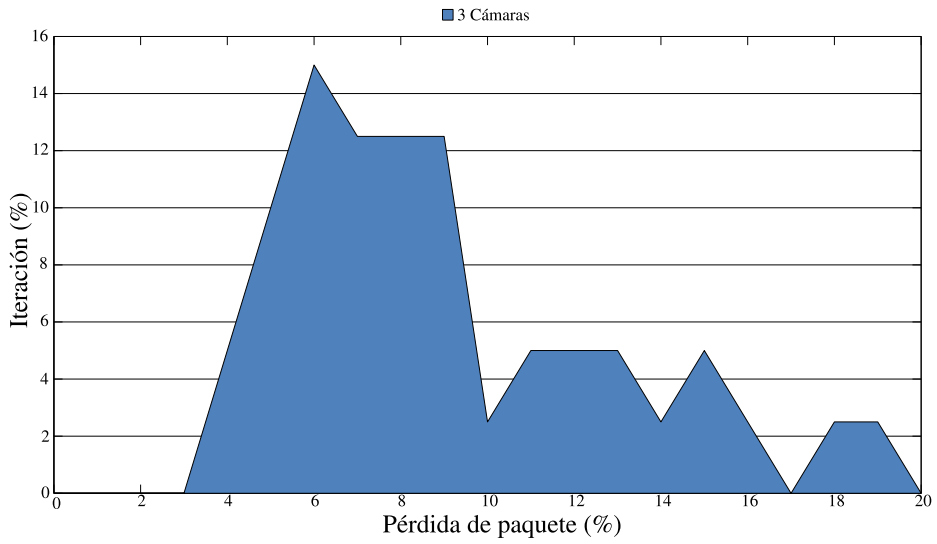


Figura 5.6: Pérdida de paquetes para tres flujos de cámara IP que atraviesan un *buffer* de 40 paquetes en una red de 100 *Mbps*.



## CAPÍTULO 6

---

### El *Buffer* y las aplicaciones

---

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. Se ha seleccionado como ejemplo de análisis un entorno de PYMES (pero igualmente aplicable al caso de una persona trabajando remotamente, por ejemplo desde la casa) 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.
- En la segunda, 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 el capítulo 5) cuando convergen los servicios mencionados. Dados los resultados obtenidos en el capítulo

5, 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.

### 6.1. Escenario de red propuesto

En la Figura 6.1 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*. 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%.



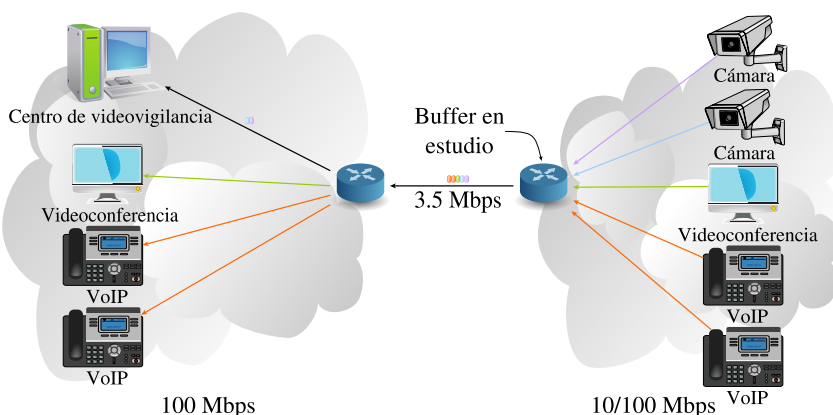


Figura 6.1: Escenario para las pruebas con dos conexiones de cámaras, una videoconferencia y dos llamadas de VoIP.

## 6.2. Tráfico utilizado

Con la finalidad de desarrollar las pruebas descritas anteriormente, se han utilizado tres fuentes de tráfico multimedia diferentes: videovigilancia, 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, usando sus tamaños de paquetes y tiempo entre paquetes. El tráfico de VoIP es generado mediante un agente Constant Bit Rate (CBR) de NS.

La metodología utilizada para las capturas del tráfico de videoconferencia se ilustra en la Figura 6.2. Para dichas trazas se ha utilizado la arquitectura de Vidyo™, 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).

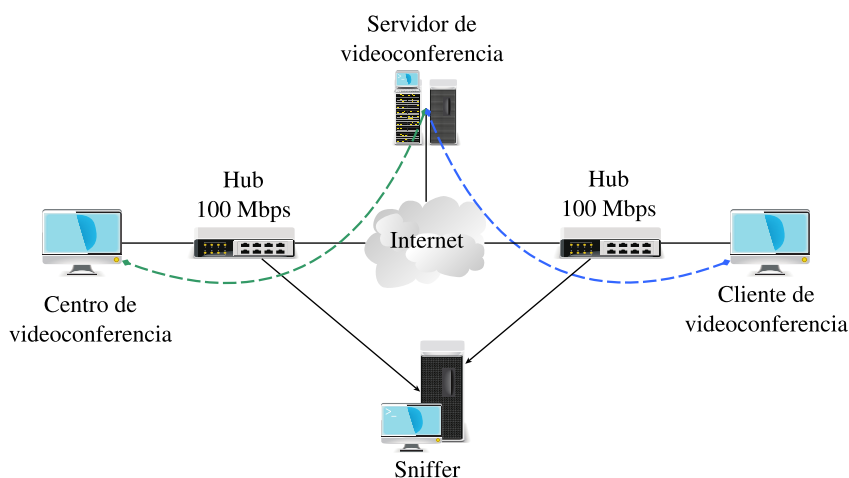


Figura 6.2: Escenario para la captura del tráfico de una videoconferencia.

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 5.

### 6.3. 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.

### 6.3.1. 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 6.3. 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 5: 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.

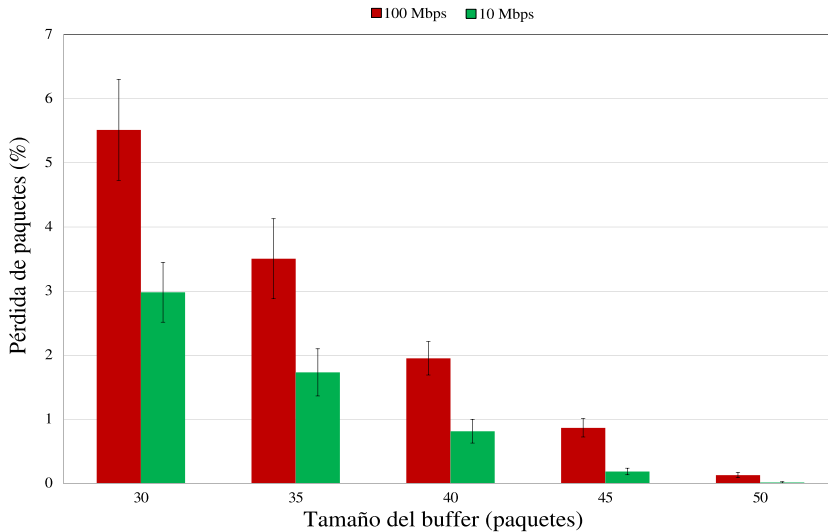


Figura 6.3: Relación entre el tamaño del *buffer* y la pérdida de paquetes para una utilización del enlace del 70 %.

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).

Para la segunda prueba, los resultados se muestran en la Figura 6.4, 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*.

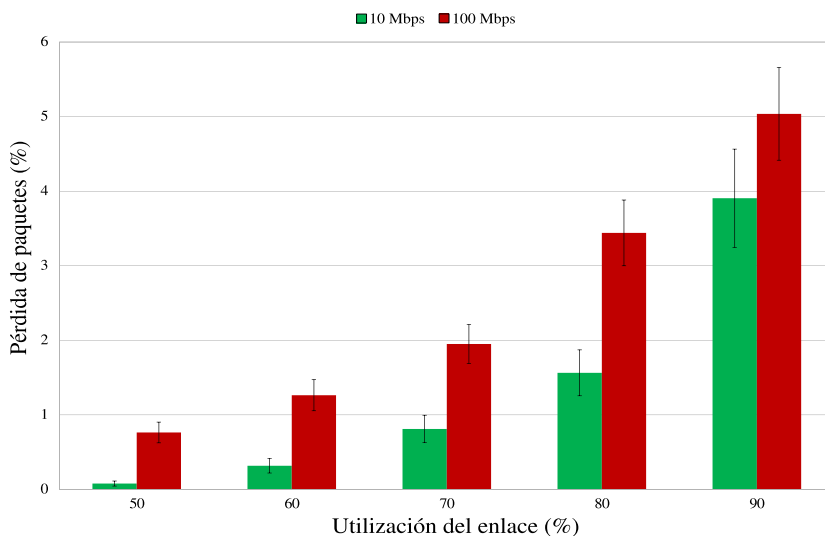


Figura 6.4: Relación entre la utilización del enlace y la pérdida de paquetes para un tamaño de *buffer* de 40 paquetes.

### 6.3.2. 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*, en la segunda se fija el tamaño del *buffer* variando la utilización del enlace. Las Figuras 6.5 y 6.6 muestran la pérdida de paquetes por flujo usando una utilización del enlace fija (70 %) con sus correspondientes intervalos de confianza del 95 %.

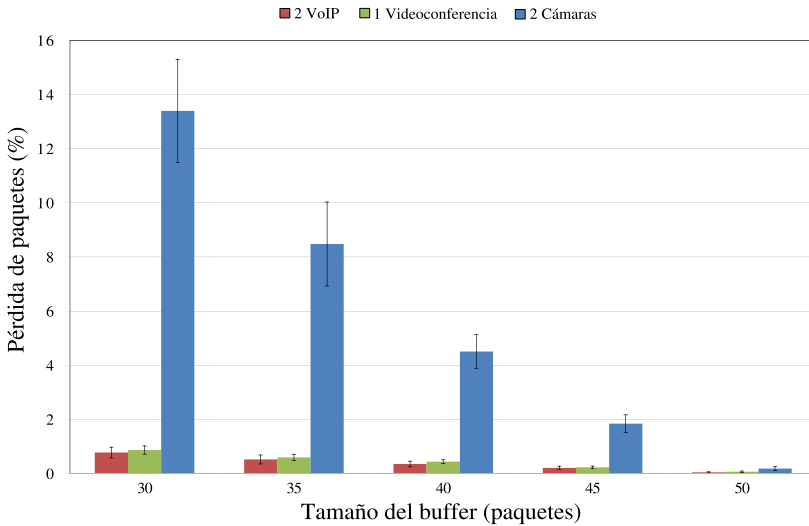


Figura 6.5: Pérdida de paquetes por flujo cuando la utilización del enlace es del 70 % para diferentes tamaños de *buffer*.

La principal causa de pérdida de paquetes es la presencia de una aplicación 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. Sin embargo, esto podría incrementar el tiempo que

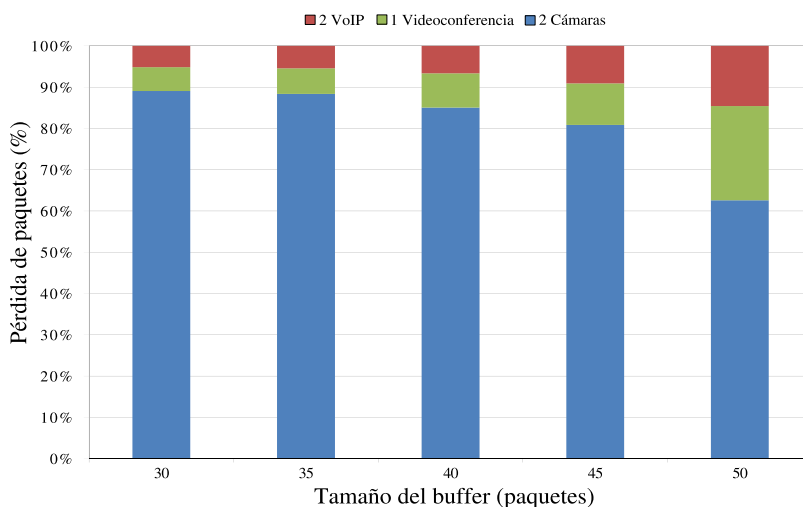


Figura 6.6: Distribución por flujo de la pérdida de paquetes cuando la utilización del enlace es del 70 % para diferentes tamaños de *buffer*.

un paquete está encolado durante períodos de congestión, generando un mayor retardo. No obstante, la videoconferencia y VoIP obtienen mejores resultados debido a que su perfil de tráfico tiene menos ráfagas.

Por otro lado, en la Figura 6.6 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 que presenta más ráfagas), incrementando la tasa de paquetes perdidos correspondientemente a este servicio.

Las Figuras 6.7 y 6.8 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 6.8), aunque las diferencias no son significativas. Así, teniendo en cuenta los resultados de las Figuras 6.7 y 6.8, se puede ver que el tamaño del *buffer* tiene una fuerte influencia en la distribución de la pérdida de paquetes por flujo.

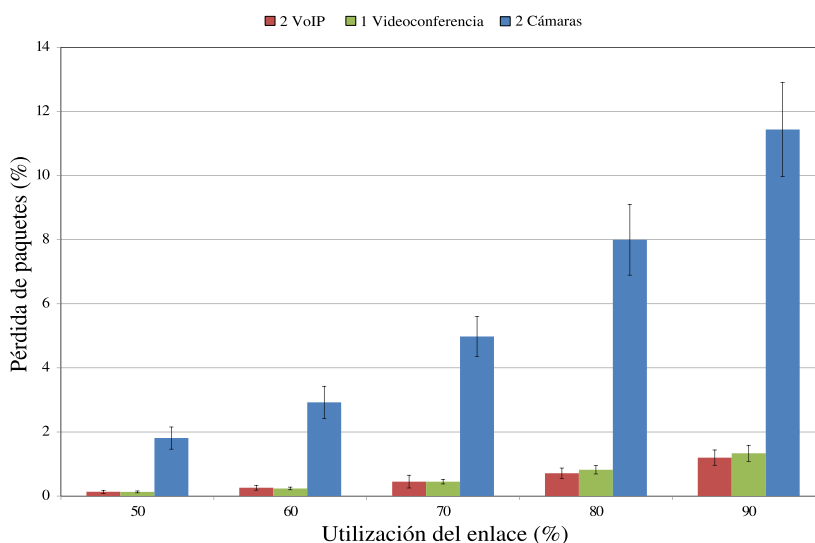


Figura 6.7: Relación entre la pérdida de paquetes por flujo y la utilización del enlace para un *buffer* de 40 paquetes.

## 6.4. Distribución 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 distri-

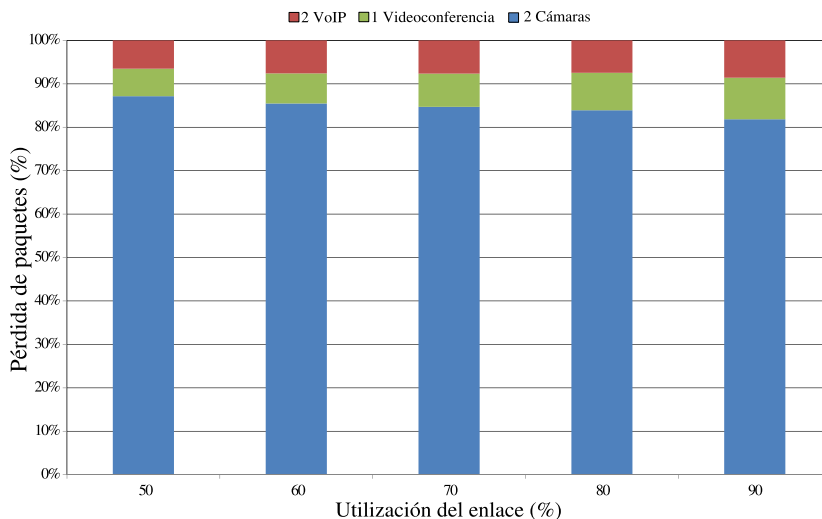


Figura 6.8: Distribución por flujo de la pérdida de paquetes para un *buffer* de 40 paquetes en función de la utilización del enlace.

bución 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 5. 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 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 la calidad presenta valores significativamente peores. La principal causa de este efecto es la distribución aleatoria de las superposiciones entre las ráfagas.

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 5 se mencionó que con 40 repeticiones se obtienen resultados muy similares) para observar mejor el solapamiento de los flujos 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 6.9 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%.

El caso de VoIP se presenta en la Figura 6.10, 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 6.11). La tasa de pérdida de paquetes es baja para un alto porcentaje de las pruebas. 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 6.12) 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.

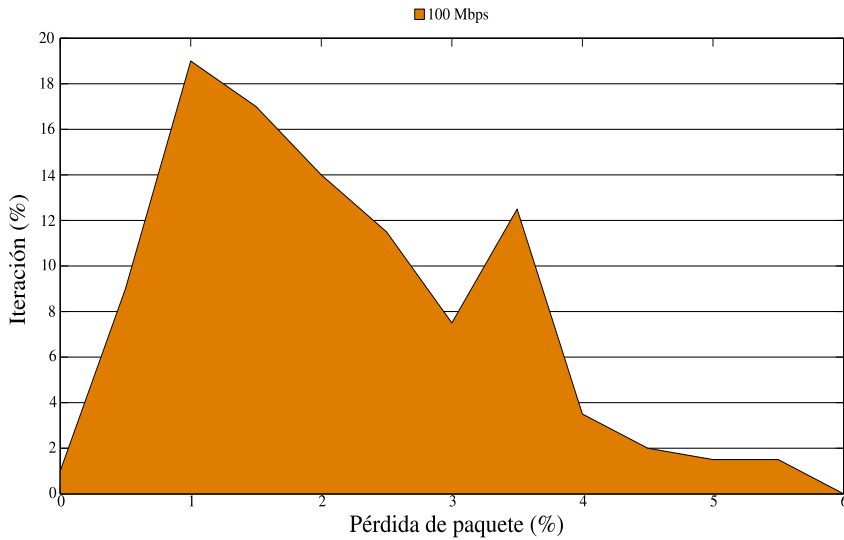


Figura 6.9: Pérdida de paquetes para el tráfico combinado con un *buffer* de 40 paquetes y una utilización del enlace del 70 %.

## 6.5. 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 [27] mediante la siguiente ecuación 6.1.

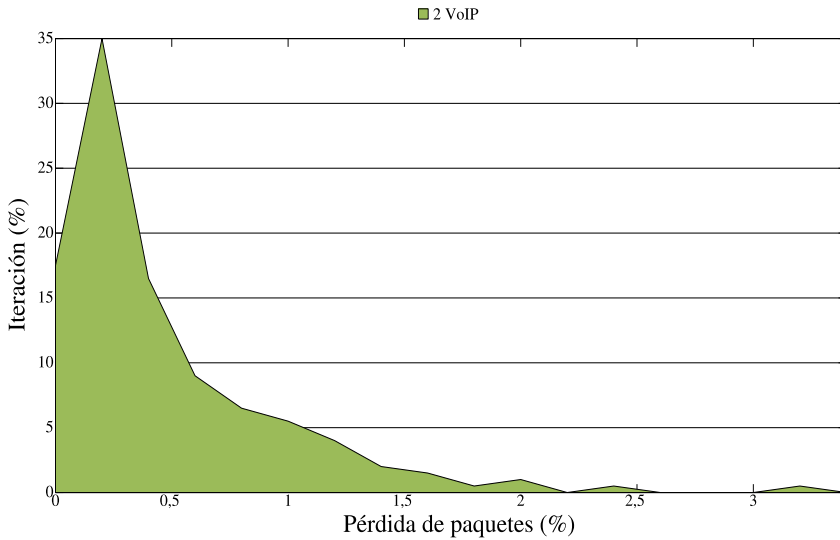


Figura 6.10: Pérdida de paquetes para el tráfico de VoIP con un *buffer* de 40 paquetes y una utilización del enlace del 70 %.

$$\begin{aligned}
 R_{factor} = & 94,2 - 0,24 \times delay_{total} - 0,11(delay_{total} - 177,3) \\
 & \times H(delay_{total} - 177,3) - 11 \\
 & - 40 \ln(1 + (10 \times delay_{total}))
 \end{aligned} \tag{6.1}$$

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 [27]:

*“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*

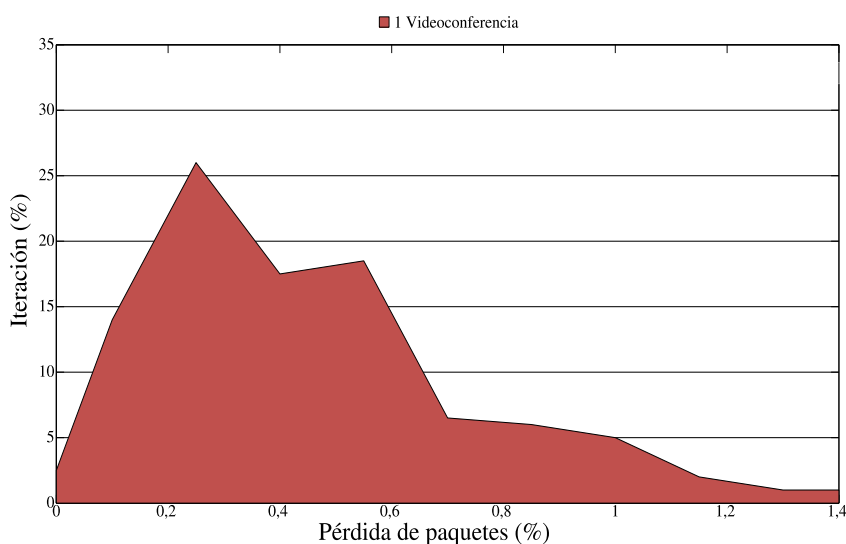


Figura 6.11: Pérdida de paquetes para el tráfico de videoconferencia con un *buffer* de 40 paquetes y una utilización del enlace del 70%.

*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 [27]. 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 6.13) del MOS obtenido para

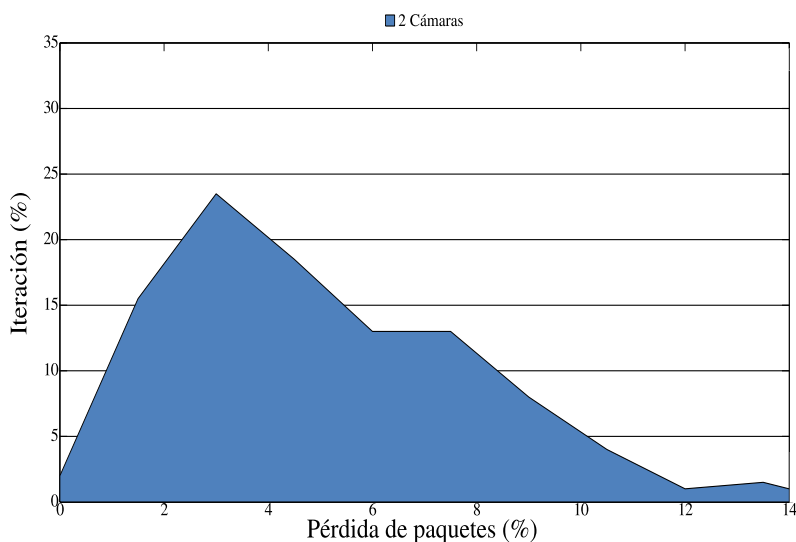


Figura 6.12: Pérdida de paquetes para el tráfico de cámara IP con un *buffer* de 40 paquetes y una utilización del enlace del 70 %.

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 [26, 27]. 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 6.13 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.

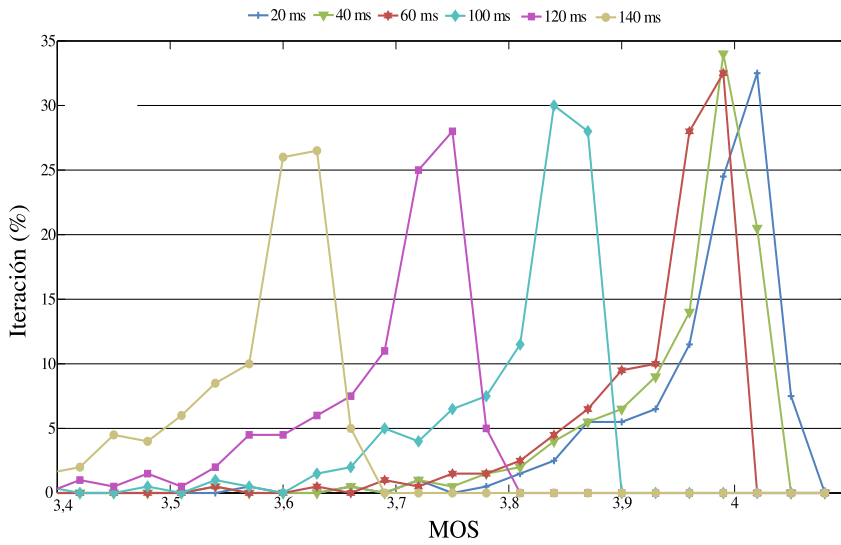


Figura 6.13: MOS con diferentes retardos de red (OWD) para un *buffer* de 40 paquetes y una utilización del enlace del 70 %.

# CAPÍTULO 7

---

## Conclusión

---

En este capítulo se comentan las principales conclusiones relacionadas con el impacto de los *buffer* en la QoS, el comportamiento de los flujos de datos y su impacto en la red. También se incluyen aspectos relacionados a la QoS en la coexistencia de tráficos concurrentes de varios servicios.

### 7.1. El comportamiento del tráfico

El comportamiento del tráfico de las aplicaciones está ligado a cómo estas 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 implementada. 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, por ejemplo, imágenes de alta calidad y archivos con tamaños muy grandes. 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). Muchas de estas aplicaciones incluso generan paquetes de tamaños variados en un mismo flujo, donde un porcentaje de los paquetes son pequeños, y usualmente están relacionados al transporte de información para la gestión y control de la aplicación, mientras que otros son de mayor tamaño y generalmente están asociados al envío de datos propios de la función principal (por ejemplo, video en el caso de sistemas de videoconferencia o videovigilancia).

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.

### 7.2. La QoS y los *buffer*

El tamaño del *buffer* se ha identificado como un parámetro crítico a la hora de realizar el planeamiento de una red, principalmente en entornos donde no es posible aumentar la capacidad del enlace de acceso. 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 datos presentados 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 (y se mantiene la tasa de salida al mismo valor). En este sentido, los capítulos 5 y 6 presentan simulaciones 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 datos 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 de conformado de tráfico, que permiten modificar la manera en que el tráfico es generado, se pueden considerar como una ventaja.

## CAPÍTULO 7. CONCLUSIÓN

---

---

## Bibliografía

---

- [1] Luis Sequeira. “Técnicas de estimación de buffer, centradas en las redes de acceso, para la transmisión de flujos ip en tiempo real”. Tesis doct. Universidad de Zaragoza, 2015.
- [2] Gao Huang, Meng Ye y Long Cheng. “Modeling system performance in MMORPG”. En: *Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004. IEEE*. IEEE. 2004, págs. 512-518.
- [3] S. Fleck y W. Strasser. “Smart Camera Based Monitoring System and Its Application to Assisted Living”. En: *Proceedings of the IEEE* 96.10 (2008), págs. 1698-1714.
- [4] Arun Vishwanath, Vijay Sivaraman y Marina Thottan. “Perspectives on router buffer sizing: recent results and open problems.” En: *Computer Communication Review* 39.2 (21 de dic. de 2009), págs. 34-39.
- [5] Luis Sequeira, Julián Fernández-Navajas, Luis Casadesus, Jose Saldana, Idelkys Quintana y José Ruiz-Mas. “The influence of the buffer size in packet loss for competing multimedia and bursty traffic”. En: *Performance Evaluation of Computer*

## BIBLIOGRAFÍA

---

- and Telecommunication Systems (SPECTS), 2013 International Symposium on.* IEEE. 2013, págs. 134-141.
- [6] L. Sequeira, J. Fernández-Navajas y J. Saldana. “The Effect of the Buffer Size in QoS for Multimedia and Bursty Traffic: When an Upgrade Becomes a Downgrade”. En: *KSI Transactions on Internet and Information Systems* 8.9 (2014), págs. 3159-3176.
- [7] Sally Floyd y Van Jacobson. “Random early detection gateways for congestion avoidance”. En: *Networking, IEEE/ACM Transactions on* 1.4 (1993), págs. 397-413.
- [8] Rade Stanojevic y Robert Shorten. “Trading link utilization for queueing delays: An adaptive approach.” En: *Computer Communications* 33.9 (2010), págs. 1108-1121.
- [9] Frederic Thouin, Mark Coates y Michael Rabbat. “Large scale probabilistic available bandwidth estimation.” En: *Computer Networks* 55.9 (2011), págs. 2065-2078.
- [10] Cesar D Guerrero y Miguel A Labrador. “On the applicability of available bandwidth estimation techniques and tools”. En: *Computer Communications* 33.1 (2010), págs. 11-22.
- [11] William Stallings. *Redes e Internet de alta velocidad Rendimiento y Calidad de Servicio*. Segunda. Madrid: Pearson Education, 2004.
- [12] Bob Braden, David Clark y Scott Shenker. “Integrated Services in the Internet Architecture: an Overview”. En: *Network Working Group RFC* 1633 (1994).
- [13] Steven Blake, David L. Black, Mark A. Carlson, Elwyn Davies, Zheng Wang y Walter Weiss. “An Architecture for Differentiated Services”. En: *Network Working Group RFC* 2475 (1998).
- [14] Jon Postel. “Internet Protocol”. En: *RFC* 791 (1981).

- 
- [15] Stephen E. Deering y Robert M. Hinden. “Internet protocol, version 6 (IPv6) specification”. En: RFC 2460 (1998).
- [16] Luis Sequeira, Konstantinos Antonakoglou, Maliheh Mahlouji y Toktam Mahmoodi. “Haptic Networking Supporting Vertical Industries”. En: *Enabling 5G Communication Systems to Support Vertical Industries*. John Wiley & Sons, Ltd, 2019. Cap. 3, págs. 41-73.
- [17] Omar Nassef, Luis Sequeira, Elias Salam y Toktam Mahmoodi. “Building a Lane Merge Coordination for Connected Vehicles Using Deep Reinforcement Learning”. En: *IEEE Internet of Things Journal* (2020).
- [18] Luis Sequeira, Adam Szefer, Jamie Slome y Toktam Mahmoodi. “A lane merge coordination model for a V2X scenario”. En: *2019 European Conference on Networks and Communications (EuCNC)*. IEEE. 2019, págs. 198-203.
- [19] Idelkys Quintana-Ramirez, Anthony Tsiopoulos, Maria A Lema, Fragkiskos Sardis, Luis Sequeira, James Arias, Aravindh Raman, Ali Azam y Mischa Dohler. “The making of 5g: Building an end-to-end 5g-enabled system”. En: *IEEE Communications Standards Magazine* 2.4 (2018), págs. 88-96.
- [20] Jose Saldana, Luis Sequeira, Jose Ruiz Mas, Julian Fernandez Navajas, Alessandro Raschella, Almodovar Jose y Ali Arsal. “Specification of Smart AP solutions”. En: *Deliverable 3.3* (2016).
- [21] Kun I Park. *QoS in packet networks*. Primera. Boston: Springer, 2005.
- [22] E.A.V. Navarro, J.R. Mas, J.F. Navajas y C.P. Alcega. “Performance of a 3g-based mobile telemedicine system”. En: *Consumer Communications and Networking Conference, CCNC IEEE*. Vol. 2. 2006, págs. 1023-1027.
- [23] Jon Postel. “Transmission Control Protocol”. En: RFC 793 (1981).

## BIBLIOGRAFÍA

---

- [24] Abdulhussain E. Mahdi y Dorel Picovici. “Advances in voice quality measurement in modern telecommunications.” En: *Digital Signal Processing* 19.1 (2009), págs. 79-103.
- [25] ITU-T Rec. “P. 800: Métodos de determinación subjetiva de la calidad de transmisión”. En: *UIT-T. CALIDAD DE TRANSMISIÓN TELEFÓNICA* (1996).
- [26] ITU-T Rec. “G. 107: The E-Model a computational model for use in transmission planning”. En: (2014).
- [27] R. G. Cole y J. H. Rosenbluth. “Voice over IP performance monitoring”. En: *SIGCOMM Comput. Commun. Rev.* 31.2 (2001), págs. 9-24.
- [28] A. F. Wattimena, R. E. Kooij, J. M. van Vugt y O. K. Ahmed. “Predicting the perceived quality of a first person shooter: the Quake IV G-model”. En: *Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games. NetGames '06*. Singapore: ACM, 2006.
- [29] “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”. En: *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007)* (2012), págs. 1-2793.
- [30] R. Prasad, C. Dovrolis, M. Murray y K. Claffy. “Bandwidth estimation: metrics, measurement techniques, and tools”. En: *Network, IEEE* 17.6 (2003), págs. 27-35.
- [31] Carlos Fernández, Jose Saldana, Julián Fernández-Navajas, Luis Sequeira y Luis Casadesus. “Video Conferences through the Internet: How to Survive in a Hostile Environment”. En: *The Scientific World Journal* 2014 (2014).
- [32] Matt Mathis y Mark Allman. “A framework for defining empirical bulk transfer capacity metrics”. En: RFC 3148 (2001).

- 
- [33] Sally Floyd, Tom Henderson y Andrei Gurtov. “The NewReno modification to TCPs fast recovery algorithm”. En: RFC 3782 (2004).
- [34] Matt Mathis, Jamshid Mahdavi, Sally Floyd y Allyn Romanow. “TCP selective acknowledgment options”. En: RFC 2018 (1996).
- [35] Steven Michael Bellovin. *A best-case network performance model*. Inf. téc. 1992.
- [36] Van Jacobson. *Pathchar: A tool to infer characteristics of Internet paths*. Inf. téc. 1997.
- [37] Jean-Chrysostome Bolot. “Characterizing End-to-End Packet Delay and Loss in the Internet.” En: *J. High Speed Networks* 2.3 (1993), págs. 305-323.
- [38] Robert L. Carter y Mark Crovella. “Measuring Bottleneck Link Speed in Packet-Switched Networks.” En: *Perform. Eval.* 27/28.4 (1996), págs. 297-318.
- [39] Manish Jain y Constantinos Dovrolis. “End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput.” En: *IEEE/ACM Trans. Netw.* 11.4 (2003), págs. 537-549.
- [40] B. Melander, M. Bjorkman y P. Gunningberg. “A new end-to-end probing and analysis method for estimating bandwidth bottlenecks”. En: *Global Telecommunications Conference, Globecom IEEE*. Vol. 1. 2000, 415-420 vol.1.
- [41] Bob Melander, Mats Bjorkman y Per Gunningberg. “Regression based available bandwidth measurements”. En: *International Symposium on Performance Evaluation of Computer and Telecommunications Systems*.
- [42] Li Lao, Constantine Dovrolis y M. Y. Sanadidi. “The probe gap model can underestimate the available bandwidth of multihop paths.” En: *Computer Communication Review* 36.5 (2006), págs. 29-34.

## BIBLIOGRAFÍA

---

- [43] Jacob Strauss, Dina Katabi y M. Frans Kaashoek. “A measurement study of available bandwidth estimation tools.” En: *Internet Measurement Conference*. ACM, 2003, págs. 39-44.
- [44] Vinay Ribeiro, Rudolf Riedi, Richard Baraniuk, Jiri Navratil y Les Cottrell. “pathchirp: Efficient available bandwidth estimation for network paths”. En: *Passive and active measurement workshop*. Vol. 4. 2003.
- [45] Emanuele Goldoni, Giuseppe Rossi y Alberto Torelli. “ASSOLO: an Efficient Tool for Active End-to-end Available Bandwidth Estimation”. En: *International Journal On Advances in Systems and Measurements 2.4* (2010), págs. 283-292.
- [46] J. Murillo Royo, J. M. Saldaña Medina, J. Fernández Navajas, J. Ruiz Mas, E. A. Viruete Navarro y J. I. Aznar Baranda. “Análisis de QoS para una Plataforma Distribuida de Telefonía IP”. En: 2010, págs. 63-70.
- [47] J. Saldana, J. Murillo, J. Fernández-Navajas, J. Ruiz-Mas, E. Viruete Navarro y J.I. Aznar. “Evaluation of multiplexing and buffer policies influence on VoIP conversation quality”. En: *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*. 2011, págs. 378-382.
- [48] Curtis Villamizar y Cheng Song. “High performance TCP in ANSNET”. En: *SIGCOMM Comput. Commun. Rev.* 24.5 (1994), págs. 45-60.
- [49] Guido Appenzeller, Isaac Keslassy y Nick McKeown. “Sizing router buffers.” En: *SIGCOMM*. Ed. por Raj Yavatkar, Ellen W. Zegura y Jennifer Rexford. ACM, 2004, págs. 281-292.
- [50] Mihaela Enachescu, Yashar Ganjali, Ashish Goel, Nick McKeown y Tim Roughgarden. “Part III: routers with very small buffers.” En: *Computer Communication Review* 35.3 (2005), págs. 83-90.



- 
- [51] J. Ruiz Mas, J. I. Aznar Baranda, J. M. Saldaña Medina, J. Fernández Navajas, B. Hernández Ortega, L. Blasco Arcas y J. Jiménez Martínez. “Evaluación de nuevos canales de distribución en servicios interactivos IP”. En: 2010.
- [52] Arun Vishwanath y Vijay Sivaraman. “Routers With Very Small Buffers: Anomalous Loss Performance for Mixed Real-Time and TCP Traffic.” En: *IWQoS*. Ed. por Hans van den Berg y Gunnar Karlsson. IEEE, 2008, págs. 80-89.
- [53] Arun Vishwanath, Vijay Sivaraman y George Rouskas. “Considerations for Sizing Buffers in Optical Packet Switched Networks.” En: *INFOCOM*. IEEE, 2009, págs. 1323-1331.
- [54] Amogh Dhamdhere y Constantine Dovrolis. “Open issues in router buffer sizing.” En: *Computer Communication Review* 36.1 (2006), págs. 87-92.
- [55] Joel Sommers, Paul Barford, Albert G. Greenberg y Walter Willinger. “An SLA perspective on the router buffer sizing problem.” En: *SIGMETRICS Performance Evaluation Review* 35.4 (2008), págs. 40-51.
- [56] Richelle Adams. “Active Queue Management: A Survey”. En: *Communications Surveys and Tutorials, IEEE* 15.3 (2013), págs. 1425-1476.
- [57] Bert Hubert. *Linux Advanced Routing & Traffic Control HOWTO*. 2012.
- [58] Seungwan Ryu, Christopher Rump y Chunming Qiao. “Advances in Active Queue Management (AQM) Based TCP Congestion Control.” En: *Telecommunication Systems*. 25 (2004), págs. 317-351.
- [59] Vishal Misra, Wei-Bo Gong y Don Towsley. “Fluid Based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED”. En: *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*. SIGCOMM '00. New

## BIBLIOGRAFÍA

---

- York, NY, USA: Association for Computing Machinery, 2000, págs. 151-160.
- [60] Seungwan Ryu, C. Rump y Chunming Qiao. “Advances in internet congestion control”. En: *Communications Surveys Tutorials, IEEE* 5.1 (2003), págs. 28-39.
- [61] C.V. Hollot, V. Misra, D. Towsley y W.-B. Gong. “A control theoretic analysis of RED”. En: *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. Vol. 3. 2001, págs. 1510-1519.
- [62] C. V. Hollot, Vishal Misra, Donald F. Towsley y Weibo Gong. “On Designing Improved Controllers for AQM Routers Supporting TCP Flows.” En: *INFOCOM*. 2001, págs. 1726-1734.
- [63] Seungwan Ryu y Chulhyoe Cho. “PI-PD-Controller for Robust and Adaptive Queue Management for Supporting TCP Congestion Control”. En: *Proceedings of the 37th Annual Symposium on Simulation. ANSS '04*. Washington, DC, USA: IEEE Computer Society, 2004, págs. 132-139.
- [64] P. Svoboda, W. Karner y M. Rupp. “Traffic Analysis and Modeling for World of Warcraft”. En: *Communications, ICC IEEE International Conference on*. 2007, págs. 1612-1617.
- [65] Ahmad Vakili y Jean-Charles Grégoire. “QoE management for video conferencing applications”. En: *Computer Networks* 57.7 (2013), págs. 1726-1738.
- [66] Idelkys Quintana, Luis Sequeira, J Fernandez, Jose Ruiz y Jose Saldana. “Minimizing the Impact of P2P-TV Applications in Access Links.” En: *IEEE Latin America Transactions* 17.02 (2019), págs. 183-192.
- [67] Idelkys Quintana, Luis Sequeira y Jose Ruiz. “An Edge-Cloud Approach for Video Surveillance in Public Transport Vehicles.” En: *IEEE Latin America Transactions* 100.1e (2021).

- 
- [68] L. Sequeira, J. Fernández-Navajas, J. Saldana y L. Casadesus. “Empirically characterizing the buffer behaviour of real devices”. En: *Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2012 International Symposium on*. 2012, págs. 1-6.
- [69] Luis Sequeira, Julián Fernández-Navajas, Jose Saldana, José Ramón Gállego y María Canales. “Describing the Access Network by means of Router Buffer Modelling: A New Methodology.” En: *The Scientific World Journal* 2014 (2014).
- [70] Hao Jiang y Constantinos Dovrolis. “Why is the internet traffic bursty in short time scales?” En: *SIGMETRICS*. Ed. por Derek L. Eager, Carey L. Williamson, Sem C. Borst y John C. S. Lui. ACM, 2005, págs. 241-252.
- [71] Jose Saldana, Luis Sequeira, Julián Fernández-Navajas y José Ruiz-Mas. “Traffic optimization for tcp-based massive multi-player online games”. En: *2012 International Symposium on Performance Evaluation of Computer & Telecommunication Systems (SPECTS)*. IEEE. 2012, págs. 1-8.
- [72] Jose Saldana, Julian Fernandez Navajas, Jose Ruiz Mas, Luis Sequeira y Luis Casadesus. “Comparison of Multiplexing Policies for FPS Games in terms of Subjective Quality”. En: *Proc. II Workshop on Multimedia Data Coding and Transmission 2012, Jornadas Sarteco* (2020).
- [73] José Ma Saldaña, Julián Fernández-Navajas, José Ruiz-Mas, Jenifer Murillo, Eduardo Viruete Navarro y José I. Aznar. “Evaluating the influence of multiplexing schemes and buffer implementation on perceived VoIP conversation quality.” En: *Computer Networks* 56.7 (2012).
- [74] José Ma Saldaña, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro y Luis Casadesus. “Influence of on-line games traffic multiplexing and router buffer on subjective quality.” En: *CCNC*. IEEE, 2012, págs. 462-466.

## BIBLIOGRAFÍA

---

- [75] José Ma Saldaña, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro y Luis Casadesus. “The utility of characterizing packet loss as a function of packet size in commercial routers.” En: *CCNC*. IEEE, 2012, págs. 346-347.
- [76] Wu-Chang Feng, Francis Chang, Wu-Chi Feng y Jonathan Walpole. “Provisioning on-line games: a traffic analysis of a busy counter-strike server.” En: *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement*. ACM, 2002, págs. 151-156.
- [77] Alex Borges Vieira, Pedro Gomes, José Augusto Miranda Nacif, Rodrigo Mantini, Jussara M. Almeida y Sérgio Vale Aguiar Campos. “Characterizing SopCast client behavior.” En: *Computer Communications* 35.8 (2012).
- [78] José Ma Saldaña, Jenifer Murillo, Julián Fernández-Navajas, José Ruiz-Mas, Eduardo Viruete Navarro y José I. Aznar. “QoS and Admission Probability Study for a SIP-Based Central Managed IP Telephony System.” En: *NTMS*. IEEE, 2011, págs. 1-6.
- [79] Jonathan Rosenberg, Henning Schulzrinne, Gonzalo Camarillo, Alan Johnston, Jon Peterson, Robert Sparks, Mark Handley, Eve Schooler y col. “SIP: session initiation protocol”. En: RFC 3261 (2002).
- [80] José Ma Saldaña, José I. Aznar, Eduardo Viruete, Julián Fernández Navajas y José Ruiz. “QoS Measurement-Based CAC for an IP Telephony System.” En: *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*. Springer, 2009, págs. 3-19.
- [81] T. Bu, Yong Liu y D. Towsley. “On the TCP-Friendliness of VoIP Traffic”. En: *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*. 2006, págs. 1-12.

- 
- [82] Axis Communications AB. *AXIS 2120 User's Manual*. 2.01. 2002.
- [83] S. Tanwir y H. Perros. "A Survey of VBR Video Traffic Models". En: *Communications Surveys and Tutorials, IEEE* 15.4 (2013), págs. 1778-1802.
- [84] Patrick Seeling y Martin Reisslein. "Video Transport Evaluation With H.264 Video Traces." En: *IEEE Communications Surveys and Tutorials* 14.4 (2012), págs. 1142-1165.
- [85] Geert Van der Auwera, Prasanth T David, Martin Reisslein y Lina J Karam. "Traffic and quality characterization of the H. 264/AVC scalable video coding extension". En: *Advances in Multimedia* 2008.2 (2008), pág. 1.
- [86] S.A. Baset y H.G. Schulzrinne. "An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol". En: *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*. 2006, págs. 1-11.
- [87] D. Bonfiglio, M. Mellia, M. Meo, N. Ritacca y D. Rossi. "Tracking Down Skype Traffic". En: *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*. 2008.
- [88] Kuan-Ta Chen, Chun-Ying Huang, Polly Huang y Chin-Laung Lei. "Quantifying Skype User Satisfaction". En: *Proceedings of ACM SIGCOMM 2006*. Pisa Italy, 2006.
- [89] Te-Yuan Huang, Polly Huang, Kuan-Ta Chen y Po-Jung Wang. "Could Skype be more satisfying? a QoE-centric study of the FEC mechanism in an internet-scale VoIP system". En: *Network, IEEE* 24.2 (2010), págs. 42-48.
- [90] L. De Cicco, S. Mascolo y V. Palmisano. "A mathematical model of the Skype VoIP congestion control algorithm". En: *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*. 2008, págs. 1410-1415.

## BIBLIOGRAFÍA

---

- [91] Luca De Cicco, Saverio Mascolo y Vittorio Palmisano. “Skype Video congestion control: An experimental investigation”. En: *Computer Networks* 55.3 (2011), págs. 558-571.
- [92] Xinggong Zhang, Yang Xu, Hao Hu, Yong Liu, Zongming Guo y Yao Wang. “Modeling and Analysis of Skype Video Calls: Rate Control and Video Quality”. En: *IEEE Transactions on Multimedia*. 15.6 (2013), págs. 1446-1457.
- [93] Padmavathi Mundur y Poorva Arankalle. “Optimal server allocations for streaming multimedia applications on the Internet.” En: *Computer Networks* 50.18 (2006), págs. 3608-3621.
- [94] Yong Liu, Yang Guo y Chao Liang. “A survey on peer-to-peer video streaming systems”. En: *Peer-to-Peer Networking and Applications* 1.1 (2008), págs. 18-28.
- [95] ITU-T Rec. “G. 729: Coding of speech at 8 kbit/s using conjugate structure algebraic-code-excited linear-prediction (CS-ACELP)”. En: (2012).
- [96] U.R. Krieger y R. Schwessinger. “Analysis and quality assessment of peer-to-peer IPTV systems”. En: *Consumer Electronics, 2008. ISCE 2008. IEEE International Symposium on*. 2008, págs. 1-4.
- [97] Thomas Silverston y Olivier Fourmaux. “Measuring p2p iptv systems”. En: *Proc. of ACM NOSSDAV*. Vol. 7. 2007.
- [98] B. Fallica, Yue Lu, F. Kuipers, R. Kooij y P. Van Mieghem. “On the Quality of Experience of SopCast”. En: *Next Generation Mobile Applications, Services and Technologies, 2008. NGMAST '08. The Second International Conference on*. 2008, págs. 501-506.
- [99] Idelkys Quintana-Ramírez, Jose Saldana, José Ruiz-Mas, Luis Sequeira, Julián Fernández-Navajas y Luis Casadesus. “Optimization of P2P-TV traffic by means of header compression and multiplexing”. En: *2013 21st International Conference*

- on Software, Telecommunications and Computer Networks- (SoftCOM 2013)*. IEEE. Split, 2013, págs. 1-5.
- [100] L. Zabala, A. Ferro y A. Pineda. “Modelling packet capturing in a traffic monitoring system based on Linux”. En: *2012 International Symposium on Performance Evaluation of Computer & Telecommunication Systems (SPECTS)*. 2012, págs. 1-6.

## BIBLIOGRAFÍA

---