Universitat Politècnica de Catalunya

Contribución a la Transmisión de Vídeo en Redes IP con Calidad de Servicio

Tesis Doctoral

Juan José Alins Delgado

Departament de Enginyeria Telemàtica

Universitat Politècnica de Catalunya

Director

Dr. Jorge Mata Díaz

Septiembre 2004

Resumen

Hoy en día, entre los servicios más solicitados por los usuarios en las redes de comunicaciones, se encuentran aquellos que incorporan información multimedia. En Internet, es habitual que un gran número de páginas web incorporen imágenes, información de audio y en algunos casos, información de vídeo. Sin embargo, debido a que Internet sólo ofrece un servicio de entrega best-effort, la información audiovisual recibe el mismo tratamiento que el dispensado a cualquier otro tipo de datos poco sensibles al retardo temporal. Con el fin de desarrollar nuevos marcos de operación que permitan la diferenciación en el tratamiento de los servicios sobre redes basadas en IP, el IETF ha propuesto dos arquitecturas de red con calidad de servicio: IntServ y DifServ. De esta forma se pretende soportar adecuadamente la transmisión de flujos inelásticos de tasa variable mediante la ubicación de recursos especialmente dedicados al transporte de servicios con restricciones temporales acotadas. Actualmente, el tráfico de vídeo se transporta principalmente a través de canales de tasa binaria constante. Sin embargo, este tráfico es por naturaleza variable, con lo que la utilización de los recursos de red se vería sustancialmente mejorada utilizando canales de tasa variable.

En este trabajo se plantea el problema de la transmisión eficiente de tráfico de vídeo comprimido MPEG de tasa variable sobre redes IP con calidad de servicio. Para ello, en primer lugar se realiza un estudio de los algoritmos de compresión más comúnmente utilizados en la actualidad. A continuación, se realiza un análisis detallado de las distintas alternativas para la transmisión de vídeo en redes IP. En primera instancia se ha descrito la problemática del transporte de vídeo en redes de conmutación de paquetes sin garantías de servicio, discutiendo la necesidad de la aplicación de técnicas de control de tasa y suavizado de tráfico. Posteriormente se han introducido las nuevas arquitecturas IntServ y DiffServ especificadas para dar soporte a servicios con un nivel de calidad garantizado. Sobre estas arquitecturas se han discutido las posibles alternativas de transmisión de vídeo sobre redes IP con QoS y la adecuación de los esquemas de codificación de vídeo estandarizados que pueden ser aplicados sobre ellas.

A partir de este estudio se ha puesto de relieve la necesidad de la caracterización del tráfico de vídeo ya que requiere de elevados anchos de banda que además fluctúan ampliamente a lo largo de la conexión. La caracterización del tráfico se ha llevado a cabo a través del modelado MMFP bidimensional de los flujos de vídeo VBR a nivel de GoP. El estudio se ha realizado con un amplio conjunto de secuencias codificadas VBR con diferentes niveles de calidad y resolución espacial. Los resultados han demostrado la aplicabilidad de este modelo de forma general sobre cualquier secuencia de vídeo con independencia de su resolución o nivel de degradación introducido en el proceso de codificación. Una de las principales contribuciones al modelado de estos servicios ha sido el desarrollo de una metodología de ajuste de los parámetros del modelo MMFP bidimensional que permite una automatización del proceso de caracterización del tráfico de vídeo.

Una vez se ha alcanzado un conocimiento de comportamiento del tráfico de vídeo de calidad de imagen constante, se ha abordado su transmisión sobre redes IP con QoS. El marco de trabajo contempla la aplicación de mecanismos de renegociación y de asignación dinámica de recursos. El trafico de vídeo de tasa variable es un ejemplo claro en el que un sistema de renegociaciones maximiza la eficiencia de los recursos de red. La gran variabilidad de tasa binaria exhibida por este tipo de tráfico y su grado de persistencia hacen apropiada la aplicación de técnicas de segmentación sobre las secuencias de vídeo. En este trabajo se han analizado diferentes esquemas de renegociación y se ha estudiado su aplicabilidad en el contexto de redes IP con QoS y facilidades de renegociación de recursos por conexión. Se han analizado las técnicas *work-ahead buffering* donde se explota intensivamente el *buffer* del cliente para almacenar datos enviados "por adelantado". El estudio ha permitido derivar una nueva propuesta denominada 2-RCBR. Complementariamente, se ha derivado una nueva técnica de segmentación basada en el modelo MMFP bidimensional analizado, obteniendose un esquema de segmentación genérico e independiente de la resolución y del factor de cuantificación utilizado el la codificación de las secuencias. Este método, que se ha denominado método de los umbrales genéricos, permite desarrollar una segmentación de una secuencia de vídeo de forma extraordinariamente simple con un elevado nivel de eficiencia. Por otro lado, teniendo en cuenta que se desea fijar una cota máxima del retardo extremo a extremo durante el servicio de *video streaming*, se ha propuesto una nueva técnica de segmentación basada en el retardo máximo aceptado, garantizando un retardo máximo dentro de la red IP para cada segmento transmitido.

La última parte del trabajo ha desarrollado la arquitectura de transmisión escalada sobre redes IP. A raíz de los excelentes resultados obtenidos hasta el momento, se ha procedido a aplicar la misma metodología para la codificación escalable. Se ha caracterizado el flujo escalable de tasa variable mediante el modelo MMFP bidimensional y se han aplicado las técnicas de segmentación propuestas anteriormente. Los resultados obtenidos han seguido el comportamiento esperado, pudiendose considerar satisfactorios.

Índice general

Índice de figuras

Acrónimos

3D Third generation mobile systems

AF Assured Forwarding

AR AutoRegressive

ARIMA AutoRegressive Integrative Moving Average

ATM Asynchronous Transfer Mode

B-ISDN Broadband Integrated Services Digital Network

BAP Batch Arrival Processes

CBQ Class Based Queuing

CBR Constant Bit Rate

CIF Common Image Format

DCT Discrete Cosinus Transform

DNS Domain Name System

DVD Digital Versatile Disc

DIFFSERV Differentiated Services

EF Expedited Forwarding

ES Elementary Stream (MPEG)

ETSI European Telecommunications Standards Institute

FLC Fixed Length Code

FTP File Transfer Protocol (IETF)

GOP Group of Pictures

HDTV High Definition Television

HTTP Hypertext Transfer Protocol (IETF)

IETF Internet Engineering Task Force

IP Internet Protocol

IPP Interrupted Poisson Processes

IPv4 Internet Protocol version 4

IPv6 Internet Protocol version 6

ISP Internet Service Provider

ITU International Telecommunications Union

INTSERV Integrated Services

LRD Long Range Dependence

MMCR Markov Modulated Constant Rate

MMFP Markov Modulated Fluid Processes

MMPP Markov Modulated Poisson Processes

MPEG Motion Picture Experts Group (ISO)

MPEG-1 Motion Picture Experts Group (ISO)

MPEG-2 Motion Picture Experts Group (ISO)

MPEG-4 Motion Picture Experts Group (ISO)

MPEG4 Motion Picture Experts Group (ISO)

MPTS Multiple Program Transport Stream (MPEG)

NTSC National Television System Committee

PDF Probability Distribution Function

PES Paketized Elementary Stream

PS Program Stream (MPEG)

QCIF Quarter of Common Image Format

QoS Quality of Service

RCBR Renegotiated Constant Bit Rate

RSVP Resource Reservation Protocol

RTCP Real Time Control Protocol (IETF)

RTP Real Time Protocol (IETF)

RVBR Renegotiated Variable Bit Rate

RVLC Reversible Variable Length Code

SIF Sequence Intermediate Format

SLA Service Level Agreement

SPTS Single Program Transport Stream (MPEG)

SRD Short Range Dependence

TCP Transmission Control Protocol

TCP/IP Transmission Control Protocol/Internet Protocol

TES Transform-Expand-Sample

TS Transport Stream (MPEG)

TSP Transport Stream Packets (MPEG)

UDP User Datagram Protocol

UMTS Universal Mobile Telecommunications System

VBR Variable Bit Rate

VLC Variable Length Code

VoD Video on Demand

WFQ Weighted Fair Queuing

xDSL x-Type Digital Subscriber Line

CAPÍTULO 1

Introducción

En los últimos años se ha producido un crecimiento sin precedentes de Internet. El número de ordenadores conectados ha pasado de los 5,8 millones a principios de 1995, a los 171,6 millones a principios de 2003 y a los 233,1 millones en enero de 2004[1]. Uno de los principales motivos de este crecimiento ha sido, sin lugar a dudas, la proliferación de los diferentes tipos de contenidos multimedia, tales como imágenes, audio y vídeo, que han llegado a ser parte importante en muchas de las aplicaciones en Internet. Hoy en día, una gran mayoría de páginas Web contienen imágenes, audio y algunas veces servicios de *video streaming.*

El uso de la información multimedia ha sido posible, entre otros factores, gracias a la creciente potencia de los modernos ordenadores y el advenimiento de tecnologías de banda-ancha de conexión a Internet, tales como el cable-modem y xDSL. Sin embargo, las comunicaciones efectivas de vídeo sobre Internet, o sobre redes de paquetes con servicio de entrega best-effort, sigue siendo un desafío, debido a los requerimientos de calidad de servicio (QoS) en el transporte del vídeo comprimido.

Internet, originalmente diseñada para el tráfico de datos, proporciona un canal impredecible y variante en el tiempo en términos de retardo, ancho de banda y pérdida de paquetes. Mientras que las aplicaciones tolerantes al retardo pueden utilizar protocolos como TCP, que esencialmente transforma el ancho de banda variante en el tiempo y la pérdida de paquetes en retardos añadidos, en las aplicaciones de *streaming audio* y *streaming video*, estos retardos son intolerables.

Por otro lado, las técnicas de compresión de vídeo se diseñan con el objetivo principal de la compresión y aunque los diferentes estándares han ido incorporando paulatinamente aspectos de la transmisión, es difícil que estas técnicas se puedan acomodar a la naturaleza dinámica de las redes de paquetes best-effort en términos de pérdidas, anchos de banda variantes y latencia.

Básicamente hay dos enfoques diferentes para paliar el problema entre lo que ofrece la red y lo que las aplicaciones multimedia requieren. El primer enfoque se basa en el desarrollo de aplicaciones *adaptativas* que sean capaces de reaccionar ante las condiciones variables de la red. La habilidad de una aplicación de adaptarse a las condiciones de la red está limitada, por un lado por el soporte mínimo que una red best-effort puede ofrecer, y por el otro lado, por las limitaciones propias de las tecnologías existentes de compresión.

[1]Estos datos se han extraído de `www.isc.org` y hacen referencia al número de ordenadores listados en el DNS (Domain Name Service).

El segundo enfoque se basa en proveer a la red de los mecanismos necesarios para que pueda garantizar un nivel de QoS para el transporte de los datos multimedia. Estos mecanismos son, típicamente, el control de admisión y la reserva de recursos, de forma que se pueda establecer ciertas garantías sobre el retardo, las variaciones del retardo y la tasa de pérdida de paquetes. Las redes ATM y el modelo de servicios integrados (IntServ) desarrollado para Internet, son ejemplos de redes o arquitecturas de red con soporte de QoS.

Los algoritmos de control de admisión y la reserva de recursos se realizan en base a las características estadísticas que presentan los distintos tipos de tráfico, por esta razón, el modelado y caracterización de tráfico tiene una importancia fundamental, tanto en el diseño de los algoritmos de control de admisión como en las estrategias que llevan a una reserva óptima de los recursos de la red. El tráfico de vídeo presenta unas características concretas, como son la gran variabilidad de la tasa de transmisión y el efecto de rafagueo, que hacen que este tipo de tráfico no sea fácilmente gestionable por la red. Durante los últimos años, han aparecido numerosos trabajos de investigación sobre el modelado y caracterización del tráfico de vídeo, propuestas de nuevas arquitecturas de red con QoS, nuevos protocolos de comunicaciones adaptados al transporte de tráfico multimedia – los protocolos denominados *TCP-friendly* –, demostrando, por un lado el interés suscitado en la comunidad científica por la transmisión de datos multimedia, y por otro lado la complejidad y diversidad de las soluciones propuestas.

El trabajo realizado en esta tesis esta centrado en la transmisión de vídeo sobre redes IP con arquitectura IntServ, donde se ha utilizado el estándar de codificación MPEG-2 como algoritmo de compresión y se han explotado los mecanismos de transmisión escalable que ofrece. La presentación de este trabajo se ha organizado en cinco capítulos, además del presente. En el capítulo 2 se realiza una descripción de los mecanismos de compresión utilizados por los estándares MPEG y H.26x que fundamentan las técnicas de codificación empleadas. Se hace un especial hincapié en MPEG-2 y MPEG-4 puesto que son los más ampliamente utilizados en los sistemas de transmisión e incorporan diferentes mecanismos de escalabilidad. La escalabilidad facilita la adaptación de la comunicación cuando la distribución incluye canales variantes en el tiempo o diversidad de terminales. El capítulo 3 centra el marco de transmisión sobre redes IP, describiendo las distintas alternativas propuestas. Se parte de una descripción de la problemática del transporte de vídeo en redes de conmutación de paquetes sin garantías de servicio y posteriormente se introducen las nuevas arquitecturas IntServ y DiffServ especificadas para dar soporte a servicios con un nivel de calidad garantizado. Sobre estas arquitecturas se discuten las posibles alternativas de transmisión de vídeo sobre redes IP con QoS y la adecuación de los esquemas de codificación de vídeo estandarizados que pueden ser aplicados sobre ellas. Una vez se ha especificado el marco donde se desarrolla el trabajo realizado, en el capítulo 4 se caracteriza el tráfico de vídeo a través de una nueva metodología que es capaz de capturar los estadísticos de las secuencias reales con gran precisión. Esta metodología se destaca por uniformizar el proceso de modelado de vídeo con calidad constante. El modelado aplicado se fundamenta en el ajuste cadenas markovianas bidimensionales capaces de capturar la variación temporal de la tasa binaria generada por el codificador de vídeo, fijando un nivel de calidad de imagen. La validación de estos modelos se lleva a cabo contrastando el ajuste de sus estadísticos con las de las secuencias de vídeo codificadas. Los resultados obtenidos facilitan la identificación de invariantes que hasta la fecha no habían sido observados para este tráfico. Estos invariantes serán empleados en el capítulo 5 donde se evaluarán diferentes estrategias de transmisión de vídeo sobre redes IP. En particular, se diseñarán diferentes estrategias de transmisión basadas en la identificación de instantes de renegociación. Estos instantes definirán cambios, o bien en la tasa binaria entregada por el emisor, o bien en los

recursos asignados en la red. Estas variaciones a lo largo de la transmisión de la secuencia de vídeo dan lugar a una asignación dinámica de recursos que maximiza la explotación de los recursos de red. La obtención de garantías en el nivel de calidad de un servicio requiere que la red incorpore mecanismos de control de admisión y ubicación de recursos por clases de servicio. En el capítulo 6 se desarrolla la caracterización del tráfico de vídeo escalado, el cual se adecúa especialmente sobre redes IP que ofrecen diferenciación en los niveles de calidad que se ofrecen a los servicios. En este capítulo se vuelven a aplicar las técnicas de modelado y segmentación desarrolladas en los dos capítulos anteriores. Se observa nuevamente la validez de las técnicas descritas y se identifican los mismos invariantes en los modelos de vídeo del tráfico escalado. El capitulo 7 describe los resultados obtenidos en este trabajo, resaltando la principales conclusiones a las que se ha llegado, y discute algunas de sus posibles aplicaciones.

CAPÍTULO 2

Codificación y compresión de vídeo

El constante y rápido desarrollo de las técnicas de codificación de vídeo y la popularidad de Internet, está provocando la aparición de nuevos servicios cada vez más atractivos para los usuarios. De esta forma, se abre un amplio abanico de posibilidades de negocio, tanto para los fabricantes como para las empresas portadoras y suministradoras de servicios.

La gran cantidad de información generada por las fuentes de vídeo lleva a la necesidad de la aplicación de técnicas de compresión. Esta compresión es posible dada la redundancia, tanto espacial como temporal, presente en dicha información. De entre los algoritmos de codificación de vídeo, el estándar MPEG está siendo el más utilizado tanto para almacenamiento como para transmisión sobre redes de comunicación.

2.1. Introducción

Las empresas proveedoras de servicios de telecomunicaciones han comenzado ya a proporcionar servicios de banda ancha a sus usuarios además de los clásicos servicios de banda estrecha como la telefonía o la transmisión de datos a baja velocidad. Dentro de estos servicios se encuentran, entre otros, la difusión de vídeo o los accesos a alta velocidad a Internet. Por otra parte, las tecnologías en modems para las líneas de abonado digitales proporcionan la posibilidad de llegar con tasas de bit elevadas hasta los usuarios finales. Las arquitecturas de red de banda ancha, junto con las líneas digitales comentadas para la última milla, facilitan la migración entre las redes de banda estrecha y las redes de banda ancha.

Los estándares de audio y vídeo MPEG-2 son los más recientemente adoptados y aceptados internacionalmente para la compresión y transmisión de vídeo y audio digital. Actualmente se utiliza en la mayoría de los sistemas de difusión digital por cable y vía satélite.

A la hora de transmitir vídeo comprimido MPEG-2 por una red de conmutación de paquetes, aparecen una serie de aspectos y problemas a tener en cuenta. Por una parte está la calidad de servicio que se debe mantener en una transmisión de vídeo, la cual en general será muy sensible a las pérdidas y retardos introducidos por la red. Las nuevas conexiones aceptadas por la red no deben reducir notablemente la calidad de las ya existentes. Además, cuando el vídeo se transmite comprimido, es muy importante garantizar una probabilidad de pérdidas muy baja, ya que los errores que se produzcan en una imagen pueden verse reproducidos en imágenes posteriores, a pesar de que se introduzcan técnicas de recuperación de errores. Por otro lado, uno de los principales objetivos en el diseño actual de redes de comunicación es la

utilización lo más eficiente posible de los recursos disponibles. Por lo tanto, se debe maximizar esta eficiencia para la calidad de servicio requerida por los usuarios.

2.2. Codificación de vídeo digital

Las recientes aplicaciones y servicios ofrecidos de vídeo han promovido el desarrollo de nuevos algoritmos de comprensión de vídeo digital que reducen sustancialmente la capacidad de almacenamiento y la tasa binaria de transmisión. De entre los posibles servicios ofrecidos cabe destacar los de telefonía [I.F92b], videoconferencia [I.F92a], distribución de televisión [I.J93], televisión por cable, distribución de televisión de alta resolución [CD94] y vídeo bajo petición [CAE+94]. El vídeo digital presenta diferentes resoluciones dependientes del servicio o aplicación. Los formatos empleados para los servicios de vídeo parten del formato CCIR-601 [I.R72] especificado para televisión. Así, para videoconferencia y para señal de televisión, con calidad de vídeo doméstico, se emplea el formato CIF (*Common Image Format*) y en servicios de telefonía el QCIF (*Quarter of CIF*). Para compatibilizar la señal de vídeo digital proveniente de vídeo NTSC y PAL también se ha especificado el formato SIF (*Sequence Intermediate Format*) como formato estándar de entrada para los algoritmos de codificación MPEG [MPE91].

Las técnicas de compresión que emplean los algoritmos para vídeo digital se basan en la explotación de la redundancia espacial y temporal de la señal. El proceso de compresión puede provocar una distorsión o pérdida respecto a la información original, por lo que aparece un compromiso entre el rango de comprensión y la distorsión obtenida. Otras técnicas de compresión no introducen pérdidas pero el rango de compresión resultante suele ser muy inferior.

Las técnicas de compresión se pueden clasificar en función del tipo de explotación de redundancia que realicen. Las técnicas de explotación de la redundancia espacial procesan cada imagen individualmente aprovechando la semejanza entre los pixels de una misma zona, mientras que las técnicas de explotación de la redundancia temporal se basan en el parecido de los pixels situados en una misma posición de un conjunto de campos consecutivos de una secuencia de imágenes.

Las técnicas de compresión basadas en la reducción de la redundancia espacial se pueden clasificar según el tipo de transformación aplicada sobre la imagen en [Pri94]:

- *Codificación predictiva*: Se basa en la codificación del valor diferencial de un pixel respecto al valor estimado a partir de los pixels previamente codificados de su entorno.

- *Codificación transformacional*: Los métodos transformados buscan la extracción de la redundancia de los pixels de una misma zona de la imagen a través de una transformación lineal, de forma que la codificación de los valores obtenidos en el dominio transformado sea inferior a la de los pixels de la imagen. Se ha demostrado que la transformación lineal óptima es la denominada transformada Karhunen-Loeve [Kou95]. Esta transformación se basa en que los pixels de una zona próxima están muy correlados y en que la distribución de probabilidad de los pixels de una zona es gaussiana. La transformación óptima se puede aproximar por la transformada discreta coseno (DCT), cuando los coeficientes de correlación están próximos a la unidad. En general, las zonas consideradas de la imagen suelen ser bloques rectangulares de pixels.

- *Codificación en subbandas*: Es una descomposición de la señal original utilizando un banco de filtros de distintas bandas frecuenciales y diezmando las señales obtenidas

adecuadamente para que no aparezca aliasing. El resultado de este esquema crítico de descomposición en subbandas es un conjunto de señales con un número total de muestras igual a la original. Cada una de las señales se codifica independientemente y se pueden recomponer para obtener diferentes resoluciones de la imagen original.

- *Codificación jerárquica*: La imagen original se descompone en una serie de señales de resolución menor hasta llegar a un nivel básico. A diferencia de la codificación en subbandas, la codificación de cada nivel de resolución necesita de los resultados de la codificación de resolución inferior. En el proceso de decodificación, la imagen original se reconstruye paulatinamente con la agregación de los distintos niveles de resolución.

- *Codificación por segmentación*: Esta codificación se basa en la detección de los contornos de los objetos que componen la imagen y una descripción de estos objetos según su textura, luminosidad, etc. Esta técnica, si bien proporciona elevados niveles de compresión, requiere de un alto coste computacional.

- *Codificación por modelo*: Cuando las imágenes que se pretenden comprimir mantienen invariantes los contornos, como un rostro en videotelefonía, basta con detectar en la imagen aquellos parámetros que describen el objeto invariante y, posteriormente, los correspondientes a su textura. De esta forma, se pueden alcanzar elevados niveles de compresión.

La explotación de la redundancia temporal se realiza a través de las siguientes técnicas:

- *Codificación transformacional*: De la misma forma que se realizaba sobre una zona de una imagen, se puede aplicar la DCT simultáneamente sobre un grupo de pixels situados en diferentes campos consecutivos, pero en la misma zona espacial de cada campo. De esta forma se obtiene la transformación tridimensional denominada 3D DCT.

- *Codificación predictiva*: En este caso, un bloque de pixels se codifica diferencialmente respecto a otro situado en un campo de referencia temporalmente próximo. En general, esta técnica se aplica buscando el bloque de pixels más similar al que se debe codificar, sobre el campo de referencia. Este mecanismo recibe el nombre de compensación de movimiento (CM), de forma que cada bloque codificado predictivamente va unido a un vector de movimiento o desplazamiento relativo del bloque empleado en el cuadro de referencia.

- *Codificación por relleno condicional*: En este caso, en un campo sólo se codifican aquellos pixels cuyo valor es significativamente diferente de los codificados en el campo previo en la misma localización.

Junto con las técnicas de compresión presentadas también se suelen emplear mecanismos de cuantificación. La cuantificación se puede aplicar a cada muestra del dominio de partida o del domino transformado (cuantificación escalar) o sobre un grupo de muestras (cuantificación vectorial) a fin de aprovechar la similitud de muestras próximas.

Los algoritmos de codificación suelen conjugar diversas técnicas de las expuestas anteriormente para maximizar el rango de la compresión para un nivel de distorsión dado o para una tasa binaria constante. En la tabla 2.1 aparecen varios de estos algoritmos junto con los mecanismos de compresión utilizados.

Algoritmo	Codificación
J.80	Diferencial
H.120	Diferencial y relleno condicional
MJPEG	DCT
H.261	DCT y CM
J.81	DCT y CM
MPEG-1	DCT y CM
MPEG-2	DCT y CM
MPEG-4	Modelo

Tabla 2.1: Algoritmos de compresión de vídeo

2.3. Codificación de vídeo MPEG-2

Formalmente, el Grupo de Expertos en Imágenes en Movimiento (*Motion Pictures Expert Group*, MPEG) se constituyó en 1988 como *Joint ISO/IEC Technical Committee on Information Technology, Subcommittee 29, Working Group 11 (ISO/IEC JTC1 SC29 WG11)*. Se encargó del desarrollo de estándares para la representación codificada de imágenes en movimiento, la información de audio asociada y su combinación para el almacenamiento en medios digitales [Swe97]. Esta primera fase fue completada en 1991, y de este trabajo surgió la especificación MPEG-1 [MPE91][Gal91]. Básicamente, el objetivo estaba en conseguir una calidad similar a la de un vídeocasette VHS con una tasa de bit alrededor de los 1.2 Mbps.

La aparición de nuevos servicios en los cuales el vídeo jugaba un papel importante llevó al desarrollo de un nuevo esquema de codificación, basado en el anterior, que abarcase un abanico más amplio de aplicaciones. De este modo se desarrolló un nuevo estándar, MPEG-2 (ISO/IEC 13818) [MPE96], como un superconjunto de MPEG-1, que adaptaba el anterior a los nuevos servicios emergentes. Originalmente existió también un proyecto MPEG-3 para aplicaciones de televisión de alta definición, pero fue cancelado cuando estas aplicaciones se incluyeron dentro de MPEG-2. Si bien MPEG-2 se asocia generalmente sólo a la compresión de vídeo, en realidad es una familia de estándares que incluye varios aspectos. En total, son ocho las diferentes partes en que se divide MPEG-2, las cuales se muestran en la tabla 2.2, en la cual se ha respetado la nomenclatura inglesa [OS98].

Actualmente, MPEG trabaja en nuevos estándares de codificación para tasa de bit muy bajas. El proyecto es conocido como MPEG-4 [Chi] y está enfocado en aplicaciones de vídeoconferencia con bajos retardos y requisitos estrictos de ancho de banda.

2.3.1. Conceptos básicos

A la hora de definir MPEG-2, uno de los aspectos más importantes era conseguir un alto grado de flexibilidad. Como resultado de ello se permitiría trabajar con distintas resoluciones de vídeo, prestaciones de equipos, requisitos de ancho de banda y calidades de imagen. Así, entre otras características, MPEG-2 no define el método de compresión, sino sólo la cadena de bits resultante. Además, define cómo se debe decodificar dicha cadena de bits.

En este apartado se introducen los conceptos principales de MPEG, comenzando con los objetos básicos definidos. Dichos objetos se muestran en la figura 2.1. La secuencia SIF se estructura en cuatro niveles de codificación: cuadro, tira o *slice*, macrobloque y bloque. El

MPEG-2	Descripción
ISO/IEC 13818-1	Systems
ISO/IEC 13818-2	Video
ISO/IEC 13818-3	Audio
ISO/IEC 13818-4	Compliance
ISO/IEC 13818-5	Software Simulation
ISO/IEC 13818-6	Digital Storage Media-Command and Control (DSM-CC)
ISO/IEC 13818-9	Real-time Interface for Systems Decoders
ISO/IEC 13818-10	DSM Reference Script Format

Tabla 2.2: Partes del estándar MPEG-2

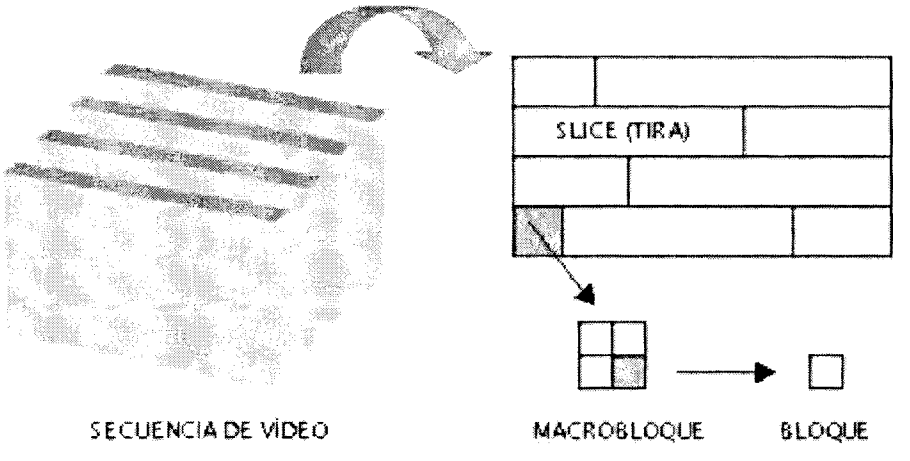


Figura 2.1: Objetos básicos en MPEG-2

cuadro es la unidad básica de presentación cuyo número de *pels* (pixels de 8 bits) depende de la resolución. La imagen se estructura en zonas o bloques de 8 x 8 pels donde se aplica la DCT. La agrupación de 4 bloques de luminancia y uno por cada componente de croma se denomina macrobloque. El macrobloque es la unidad básica donde se aplica la técnica de compensación de movimiento. El conjunto de macrobloques consecutivos horizontalmente se denomina tira o *slice*. La tira es el elemento mínimo donde se puede resincronizar la decodificación en el caso de pérdidas de información. El presente estudio se ha realizado empleando la resolución estandarizada en la recomendación MPEG-1 de 352 por 288 pels, con submuestreo de las componentes de croma tanto vertical como horizontalmente. Las secuencias analizadas se han obtenido a través de la digitalización de la señal de vídeo PAL o NTSC (25 y 30 imágenes por segundo respectivamente).

Los cuadros de la secuencia de vídeo pueden codificarse en tres modos diferentes:

- *Intra* (I): son los cuadros codificados empleando únicamente predicción espacial.
- *Predictivo* (P): son los cuadros codificados con predicción temporal hacia atrás, usando como referencia el anterior cuadro I o P, y con predicción espacial.

- *Predictivo bidireccional* (B): son los cuadros codificados con compensación de movimiento, empleando como referencias la pasada o futura I o P. La compensación de movimiento se puede realizar sobre los macrobloques de una de las referencias o sobre una semisuma de un macrobloque de cada una ellas.

El almacenamiento o transmisión de las imágenes de una secuencia se hace de forma que el decodificador pueda procesar la información lo antes posible. Para ello, en el almacenamiento o transmisión, las imágenes de referencia preceden a aquellas que las necesitan para ser decodificadas. Este efecto produce en aplicaciones en tiempo real un retardo de reordenación, dado que el orden de decodificación de los cuadros es distinto al de su presentación. A su vez, el codificador también introduce un retardo de proceso dado que necesita imágenes que temporalmente son posteriores para codificar otras que las preceden. Por ello, no es aconsejable en este tipo de aplicaciones que el número de imágenes B consecutivas sea superior a 3 [KCJS93].

La secuencia de imágenes transmitida se estructura en los dos niveles siguientes:

- *Grupo de imágenes* (*Group of Pictures*, GoP, tamaño N), compuesto por una imagen I y las imágenes B y P que la han utilizado como referencia.
- *Subgrupo de imágenes* (*Subgroup of Pictures*, SGoP, tamaño M) compuesto por una imagen de referencia I o P y las imágenes B que emplearon la imagen I o P como segunda referencia en su proceso de codificación.

En [J.M96] se llevó a cabo un estudio para la elección de los parámetros óptimos N y M. El estudio se basó tanto en la calidad subjetiva obtenida como en un análisis cuantitativo de la misma. La calidad subjetiva se entiende como un nivel de percepción humano en la calidad de la imagen, mientras que la calidad objetiva es una cuantificación que intenta ponderar el error, o distorsión, de la imagen decodificada respecto a la original. En general, la medida empleada en este caso es el PSNR (*Power Signal to Noise Ratio*) [Wan94], definida como:

$$PSNR = 10 \log \left(\frac{255^2 R}{\sum_{i=1}^{R} (p'(i) - p(i))^2} \right) [dB]$$

donde R es el número de pels en la porción de imagen a analizar, $p(i)$ es el valor del pel original y $p'(i)$ es el valor del pel decodificado. Ambos enfoques llegaron a conclusiones similares, proponiendo los valores N=4 y M=2 o N=6 y M=2. Estos valores han sido también los adoptados en la mayoría de secuencias de prueba utilizadas en este estudio. En la figura 2.2 se ilustra la secuencia de imágenes para los valores N=6 y M=2, indicando su orden de transmisión y de visualización.

El algoritmo de codificación MPEG tiene dos modos de operación, configurables según el tipo de aplicación para la cual se emplea la compresión. En transmisiones sobre circuitos de capacidad fija, el algoritmo se configura para generar una tasa binaria constante, modo CBR. En el caso de que el sistema de comunicaciones soporte servicios de tasa variable, el algoritmo se puede configurar en modo VBR. El modo de funcionamiento VBR presenta la ventaja, respecto al CBR, de poder mantener una calidad, subjetiva u objetiva, constante en toda la secuencia codificada de imágenes [SRVV93]. Además, la utilización de recursos de red puede ser más eficiente [GKL+98].

Las variaciones de la tasa binaria generada en la codificación se deben a razones tanto intrínsecas, debidas al algoritmo de codificación, como extrínsecas debidas a la complejidad

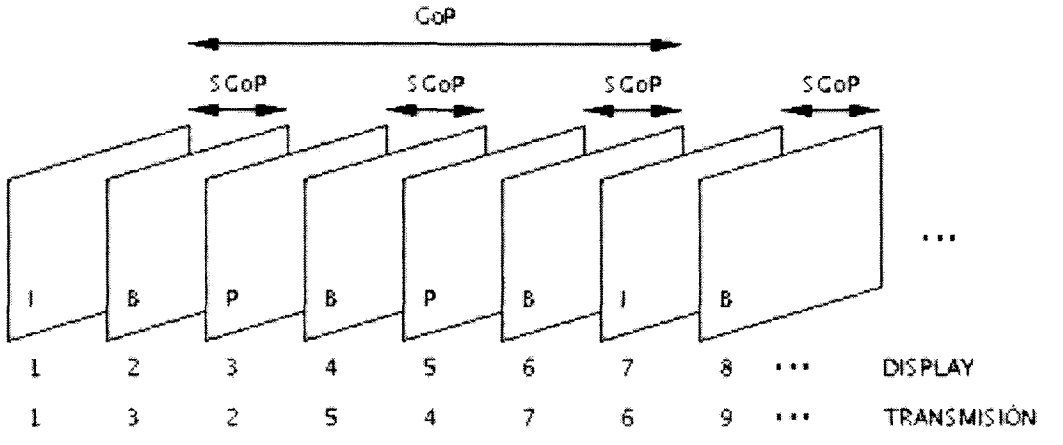


Figura 2.2: Estructura de la secuencia de imágenes (N=6, M=2)

y actividad de la secuencia a codificar. Las razones intrínsecas están relacionadas, fundamentalmente, con los modos de codificación aplicados sobre las imágenes. Así, las imágenes I necesitan un número superior de bits a las imágenes P o B al emplear únicamente la técnica transformada DCT. Asimismo, las imágenes P suelen generar mayor número de bits que las B, dado que sólo emplean compensación de movimiento respecto a las imágenes de referencia anteriores. Dentro de la codificación de las imágenes, otro factor que provoca variaciones de la tasa binaria generada es la explotación de la entropía a través de tablas de códigos de longitud variable, según el tamaño y posición de las ráfagas (run-length) de los coeficientes de la DCT.

Las razones extrínsecas que provocan variaciones en la tasa binaria dependen del contenido de las imágenes a codificar. Las imágenes con mayor grado de detalle o mayor relieve tienen un nivel de complejidad superior y reducen la efectividad de la explotación de la redundancia espacial. Las secuencias de gran actividad, con movimientos rápidos de cámara, zooms y cambios de plano, impiden el empleo de la técnica de compresión predictiva, por lo que, también provocan aumentos en la tasa binaria respecto a secuencias de menor actividad.

Para conseguir una tasa binaria constante en el modo de funcionamiento CBR es preciso intercalar entre la salida del codificador y el canal, una memoria tampón, o buffer, que absorba las variaciones de la tasa binaria generada en la codificación. Los bits almacenados en el buffer son extraídos a velocidad constante, mientras que el codificador llena el contenido del buffer de forma irregular. Para controlar el retardo introducido por la inserción del buffer, se dimensiona éste con una capacidad limitada y se regula la tasa de generación en el proceso de codificación dependiendo del nivel de ocupación del buffer.

La tasa de generación y el nivel de distorsión de la imagen se pueden controlar a través de varios parámetros que intervienen en el proceso de codificación. Los parámetros controlables que afectan a la generación de la tasa binaria son la resolución espacial y temporal de la secuencia, el número N o M de imágenes que componen un GoP o un SGoP, la cuantificación de las tiras de imagen o la cuantificación de los macrobloques individualmente. Los parámetros no controlables son los extrínsecos como el contenido estadístico de la secuencia y la actividad de la escena. En general, los parámetros de resolución y frecuencia de las imágenes se acuerdan al principio de la sesión y no se modifican en su transcurso.

El parámetro más adecuado para controlar la tasa de generación o el nivel de calidad de la imagen es el paso de cuantificación para un macrobloque o para el conjunto de macrobloques de una tira de imagen. Es el más apropiado dado que no introduce una sustancial sobrecarga de señalización y tiene una rápida respuesta temporal sin añadir un elevado coste computacional. También se puede utilizar, como parámetro de control, la variación del número de elementos que componen un GoP o SGoP. Este mecanismo no se puede emplear de forma sostenida cuando aumenta la complejidad de la secuencia, ya que el exceso de imágenes B provoca automáticamente un aumento de la tasa binaria debido al incremento de macrobloques codificados en modo intra.

La compresión en MPEG-2 se consigue en base a tres técnicas. En primer lugar se extrae la redundancia espacial mediante la cuantificación de los coeficientes obtenidos tras aplicar la DCT. Esta cuantificación se lleva a cabo dividiendo por unos factores proporcionados por una matriz de ponderación. Además, está controlada por un factor de escala que permite al usuario ajustar el nivel de compresión. Tras este proceso, se lleva a cabo un barrido en zig-zag del plano de coeficientes DCT, con lo que las altas frecuencias quedan agrupadas al final. Esto provocará un gran número de ceros, lo cual será aprovechado en el siguiente paso.

En segundo lugar, se realiza una codificación de Huffman de los coeficientes. Este tipo de codificación asigna palabras código de mayor número de bits a los coeficientes menos comunes, reservando las de menor número de bits para los coeficientes más usuales (*Variable Length Coding*, VLC).

Finalmente, se utiliza la técnica de compensación de movimiento para los cuadros B y P. Se buscan macrobloques parecidos en el cuadro actual y en el de referencia y se obtiene la diferencia. Dicho valor se transforma mediante la DCT y posteriormente se codifica VLC junto con el vector de movimiento del macrobloque. En el mejor de los casos, un macrobloque se repetirá exactamente de la misma forma en el cuadro actual y en el de referencia, y además permanecerá en la misma posición. De esta forma se tendrá una diferencia y un vector de movimiento nulos.

La estructura sintáctica presentada por MPEG-2 Video es muy variable debido a los diferentes perfiles y aplicaciones. Así, la aparición de algunos campos depende del valor que adopten otros. En otras ocasiones, la aparición o no de algún elemento depende del valor tomado por un bit (*flag*) dentro de otro elemento de control. De esta forma se consigue además reducir la cantidad de bits a transmitir, evitándose la transmisión de ceros o de valores nulos que no aportan información.

2.3.2. Escalabilidad

El estándar de codificación MPEG-2 amplía las aplicaciones a las que estaba dirigido el MPEG-1. Las principales mejoras introducidas en el MPEG-2 son: la posibilidad de operar con imágenes entrelazadas al emplear compensación de movimiento sobre macrobloques de 16 x 8 pels, aumenta la precisión de los coeficientes de continua de la DCT a 10 bits frente a los 8 de MPEG-1, permite la cuantificación no lineal, mejora el control frente a errores en su sintaxis e introduce el concepto de escalabilidad.

Esta última es una de las características más importantes de MPEG-2 Video, proporcionando soporte para un amplio rango de aplicaciones de vídeo. MPEG-2 se puede utilizar para distribución estándar de TV, para TV de alta definición (*High Definition TV*, HDTV), o para la transmisión de imágenes de vídeo a través de redes de telecomunicación.

Para conseguir la escalabilidad la información de vídeo se separa en diferentes flujos o niveles de información, los cuales son complementarios entre sí. Una aplicación básica sería tener un flujo o nivel base para una transmisión de TV estándar (PAL o NTSC), al que se podría añadir un nivel de mejora conteniendo información adicional para proporcionar una transmisión del mismo programa en HDTV. Dependiendo de las características del receptor, se quedaría sólo con el nivel básico o bien decodificaría los dos niveles.

El estándar MPEG-2 define varios modos de escalabilidad:

- *Escalabilidad espacial*: Capacidad para trabajar con diferentes resoluciones de pantalla. En este caso, los niveles básico y de mejora se combinan tras realizar la DCT inversa.
- *Escalabilidad temporal*: Se define como la posibilidad de manejar diferentes tasas de cuadro en un mismo flujo de vídeo. El nivel base, proporcionando una tasa estándar, se puede combinar con el nivel de mejora para alcanzar mayores tasas de cuadro. Los niveles básico y de mejora se combinan, al igual que en el caso anterior, tras realizar la DCT inversa.
- *Escalabilidad en SNR*: Permite manejar al menos dos calidades de vídeo diferentes. La información proporcionada por el nivel base puede ser realzada por dos o más niveles de mejora. Sin embargo, todos los niveles tienen la misma resolución espacial. La principal aplicación es en el encubrimiento de errores. Así, el nivel base podría transportar la información más crítica utilizando un canal más robusto, mientras el nivel de mejora es transmitido por un canal menos fiable. Los errores en este canal de mejora no se harán tan patentes durante la decodificación, ya que al menos será posible presentar las imágenes proporcionadas por el nivel base.
- *Partición de datos*: Este proceso se utiliza para dividir el flujo de información en partes más y menos importantes. De nuevo, las parte más importante, en este caso de la sintaxis del flujo de vídeo MPEG-2, se envía por un canal más fiable mientras que la menos importante se puede transmitir por un medio menos robusto. Una posibilidad sería enviar los elementos sintácticos de alto nivel (como las cabeceras de la secuencia, de los GoPs o de los cuadros), junto con el primer coeficiente de la DCT por el canal básico, mientras que el resto de coeficientes de la DCT se transmitirían por el canal de mejora. Existe un elemento especial, *priority breakpoint*, que define qué partes del flujo de vídeo se ponen en cada partición.

2.3.3. Perfiles y niveles

El gran rango de aplicaciones al que va dirigido MPEG-2 Video tiene como consecuencia un estándar complejo. En muchas ocasiones, los servicios ofrecidos a los usuarios no necesitarán hacer uso de gran parte de las posibilidades ofrecidas. Por otro lado, no tiene sentido en estos casos que los decodificadores sean capaces de entender todas las posibilidades de MPEG-2, lo cual los hace más complejos, si el usuario final no va a poder sacar partido de ellos.

Con objeto de flexibilizar la utilización de MPEG-2 Video se definen por tanto una serie de perfiles (*profiles*) y niveles (*levels*) formados por subconjuntos de las posibilidades ofrecidas por el estándar completo. Estos perfiles y niveles aparecen en la tabla 2.3.

Un perfil se describe como un subconjunto bien definido de la sintaxis de vídeo. Algunos elementos del estándar no serán válidos ni podrán ser decodificados si el decodificador sólo proporciona un perfil bajo. Por ejemplo, el perfil simple (SP) no admite cuadros B. Los

Profiles	Levels
Simple profile (SP)	Low Level (LL)
Main Profile (MP)	Main Level (ML)
SNR Scalable Profile	High 1440 Level (H14)
Spatial Scalable Profile	High Level (HL)
High Profile	

Tabla 2.3: Perfiles y niveles de MPEG-2 Video

perfiles simple y principal (SP y MP) no soportan ningún tipo de escalabilidad. Los perfiles bajos son siempre subconjuntos de los más altos. En la tabla 2.4 se presentan algunas de las características de los perfiles comentados.

Facilidad MPEG-2	Simple P.	Main P.	SNR P.	Spatial P.	High P.
Formato Croma	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0 ó 4:2:2
Cuadros B	No	Si	Si	Si	Si
Mode esc.	Ninguno	Ninguno	SNR	SNR o espacial	SNR o espacial

Tabla 2.4: Requisitos para perfiles MPEG-2

Por otro lado, los niveles definen valores para ciertos parámetros, como por ejemplo el número de líneas por cuadro o el número de cuadros por segundo. Perfiles y niveles son combinados para definir exactamente qué subconjunto de MPEG-2 Video se está utilizando. Una combinación muy importante es la conocida como *"Main Level at Main Profile"* (ML@MP). Esta combinación es adecuada para la difusión de TV, con calidad PAL o NTSC. Algunos de los parámetros aparecen en la tabla 2.5.

Parámetro	Valor en ML@MP
Muestras por línea	720
Lineas por cuadro	576
Cuadros por segundo	30
Muestras de luminancia por segundo	10368000
Tasa máxima de vídeo	15 Mbps
Tamaño máximo del buffer del decodificador	1835008 bits

Tabla 2.5: Valores MPEG-2 ML@MP

2.3.4. MPEG-2 Audio

La especificación de audio de MPEG-2 [MPE98] es una extensión de la que existía en MPEG-1, presentando un alto grado de compatibilidad con ésta, hasta el punto de que un decodificador audio MPEG-1 es capaz de decodificar parte de la información codificada en MPEG-2. Ambos estándares describen tres niveles de compresión, aumentando tanto la compresión como la calidad al pasar del nivel 1 al 2 y del 2 al 3. Los tres niveles son compatibles, en el sentido de que un decodificador de nivel N es capaz de decodificar la información del nivel *N-1*.

La aparición de tres niveles es en gran parte una consecuencia histórica, ya que la especificación de nivel 3 es posterior a las anteriores, con lo que estas dos tenían ganada una amplia cuota de mercado, que se mantiene en la actualidad. Sin embargo, la excelente capacidad de compresión y la gran calidad suministrada, hacen de MPEG-2 Audio nivel 3 la mejor elección en la mayoría de las aplicaciones actuales. Su extensión actual es muy grande, con infinidad de servidores web dedicados exclusivamente a la comercialización de temas musicales codificados en este formato. La ventaja es clara: en un CD-ROM clásico es posible almacenar del orden de 170 canciones comprimidas, las cuales son decodificadas en tiempo real sin problemas en un PC doméstico, ofreciendo una calidad de sonido similar a la de un disco compacto. En la tabla 2.6 se resumen las principales características de los tres niveles comentados.

Nivel	Compresión aprox.	Margen de tasa de bit	Retardo Máximo teórico
1	1:4	32-448 Kbps	19 mseg.
2	1:6	32-384 Kbps	35 mseg.
3	1:10	32-320 Kbps	58 mseg.

Tabla 2.6: Niveles de codificación MPEG-2 Audio

Como se ha comentado MPEG-2 Audio toma como base MPEG-1 Audio. Algunas de las diferencias y mejoras se enumeran a continuación:

- *Frecuencia de muestreo reducida*: En MPEG-2 es posible utilizar una tasa de muestreo reducida a la mitad, y continuar obteniendo una buena calidad de sonido.
- *Extensión multicanal*: Con objeto de obtener una representación estereofónica más realista, se habilitan cinco canales de audio que proporcionan una audición estereofónica "envolvente" (*surround*). Los cinco canales se conocen como izquierdo (*Left*, L), derecho (*Right*, R), central (*Center*, C), envolvente trasero izquierdo (*Left rear Surround*, LS) y envolvente trasero derecho (*Right rear Surround*, RS). Además, se puede incluir un canal especial de baja frecuencia (*Low Frequency Enhancement*, LFE), entre 15 y 120 Hz, principalmente dedicado a la reproducción de efectos especiales.

La compatibilidad comentada entre MPEG-1 y MPEG-2 obliga a ciertas acciones que no permiten alcanzar la máxima calidad posible con la misma tasa de compresión. Como consecuencia, se ha formado un grupo que no respeta la compatibilidad con esquemas anteriores (*Non-Backward Compatible*, NBC), con el objetivo de obtener calidades de sonido superiores a tasas de bit equivalentes.

2.3.5. MPEG-2 Sistemas

La misión principal de MPEG-2 Systems [MPE97] es la de proporcionar una especificación genérica para la multiplexación conjunta de la información codificada de audio y vídeo, independiente de la red física por la que se transmita. Así, esta parte de la especificación MPEG puede considerarse como el interfaz entre los codificadores de audio y vídeo por un lado, y la red de comunicaciones por otro. En la figura 2.3 se esquematiza esta relación.

MPEG-2 Systems distingue entre flujo de programa (*Program Stream*, PS) o flujo de transporte (*Transport Stream*, TS), en función de si la información entregada va a ser almacenada para una posterior visualización, o si por el contrario va a ser transmitida por una red de

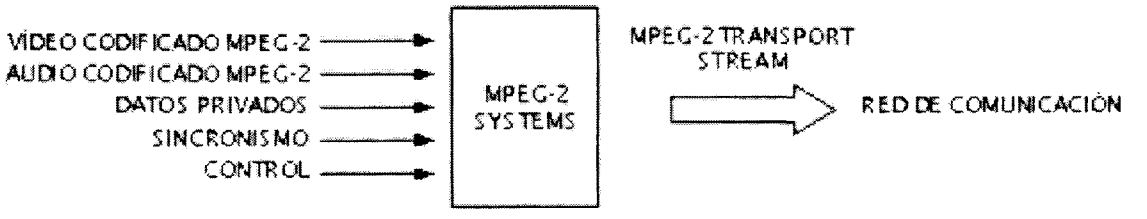


Figura 2.3: MPEG-2 Systems básico

comunicaciones. Por tanto, para el propósito que nos ocupa, nos centraremos en el TS. En este caso, las estructuras de datos utilizadas son cortas y de longitud fija para facilitar su envío por la red.

Una de las misiones fundamentales será la de multiplexar y demultiplexar distintos programas transportando diversos flujos de audio y vídeo. La sincronización necesaria se lleva a cabo añadiendo marcas temporales (*timestamps*). Además, también es posible añadir datos a la transmisión, así como cierta información de control y de gestión. Toda esta información es multiplexada en el TS.

Los diferentes elementos de información manejados se muestran en la figura 2.4 y se explican a continuación. En MPEG-2, el flujo de salida de un codificador de audio o vídeo se conoce como flujo elemental (*Elementary Stream*, ES). Como se ha comentado anteriormente, existe la posibilidad de añadir datos privados a la comunicación, que también formarían un flujo elemental. El flujo elemental se divide en unidades de acceso, que para el caso del vídeo estarían formadas por las distintas imágenes (I, P, B) a transmitir, como se observa en la parte superior de la figura. Este flujo elemental se paquetiza (*Packetized Elementary Stream*, PES), siendo el tamaño del paquete variable y conteniendo exactamente una unidad de acceso. Posteriormente, los paquetes PES se mapean dentro de los paquetes de flujo de transporte MPEG-2 (*Transport Stream Packets*, TSP). Dichos paquetes, de tamaño fijo, forman el MPEG-2 TS. El motivo de esta doble paquetización es el de crear dos niveles con distintos objetivos. Mientras las cabeceras PES contienen información directamente relacionada con el ES, como por ejemplo si se trata de audio, de vídeo o de datos, las cabeceras TS transportan información útil para la transferencia y entrega del flujo de información.

Por otra parte, se distinguen dos tipos de flujo de transporte: *Single Program Transport Stream* (SPTS) y *Multiple Program Transport Stream* (MPTS). El SPTS contiene diferentes flujos PES, los cuales comparten una base de tiempos común. Por su parte, el MPTS multiplexa varios SPTSs, dando lugar a una jerarquía como la mostrada en la figura 2.5.

2.4. MPEG-4

Una pregunta que nos podemos hacer es por qué otro estándar de codificación de vídeo. La respuesta se encuentra en el contexto de los sistemas audiovisuales de hoy. Cuando se desarrolló MPEG-2, la televisión era el paradigma de los sistemas audiovisuales y MPEG-2 tenía como objetivo, entre otros, los sistemas de televisión:

- Servicios de difusión por satélite
- TV por cable

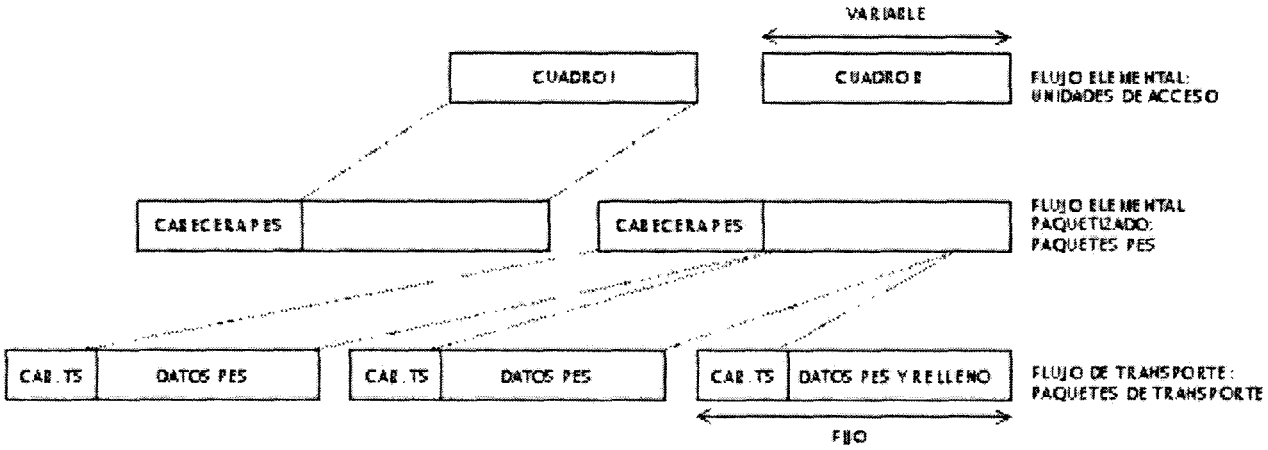


Figura 2.4: Relación entre unidades de acceso, paquetes PES y paquetes de transporte

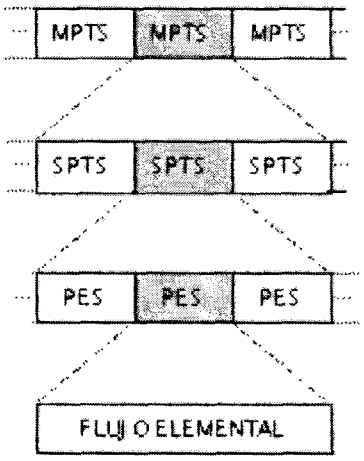


Figura 2.5: Jerarquía en MPEG-2 Systems

- Difusión de TV digital terrestre
- *"Home TV Cinema"*
- Vigilancia remota por vídeo

Sin embargo, la producción de contenidos audiovisuales es hoy más fácil. Las cámaras digitales de fotografía (*still cameras*) almacenan la información directamente en formato JPEG. Las primeras cámaras digitales de vídeo graban en formato MPEG-1. Ambos hechos hacen que haya una mayor aceptación, en los mercados de consumo, de los sistemas audiovisuales de adquisición digital. Cada propietario de uno de estos dispositivos audiovisuales se convierte potencialmente en un productor de contenidos, capaz de crear contenidos que pueden ser fácilmente distribuidos y publicados a través de Internet. Además, cada vez existe mayor contenido audiovisual producido de forma sintética – generados por ordenador – e integrados con material natural en contenidos audiovisuales híbridos.

Por otro lado, la creciente movilidad en las telecomunicaciones es una tendencia cada vez mayor. Por ejemplo, los teléfonos móviles van cambiando cada dos o tres años, sustituyéndose por nuevos terminales móviles con mayor capacidad multimedia incluyendo vídeo, juegos, etc.

La explosión de Internet, y en especial de la WEB y la aceptación del modo de operación interactivo muestran que los usuarios quieren acceder al audio y al vídeo de igual forma que acceden al texto y a los gráficos. Para ello se requiere imágenes y audio de una calidad aceptable con anchos de banda pequeños y acceso interactivo a los contenidos audiovisuales.

Dado que muchas de las aplicaciones audiovisuales emergentes demandan interconexión a través de red, la necesidad de desarrollar un estándar internacional abierto parece evidente. En 1993, MPEG lanzó MPEG-4, ahora conocido oficialmente con el nombre de "Codificación de objetos audiovisuales", con los objetivos, entre otros, de dar respuesta a los requerimientos mencionados. En general, esos requerimientos se pueden englobar en tres grandes categorías: la importancia de los sistemas audiovisuales sobre las redes, la creciente movilidad y la creciente interactividad. Es necesario que el estándar reuna los siguientes aspectos:

- Representación eficiente de diversos tipos de datos:
  - Vídeo, música y voz desde anchos de banda muy pequeños a anchos de banda grandes
  - Objetos genéricos en 3-D dinámicos así como objetos específicos como rostros y cuerpos humanos
  - Voz y música para ser sintetizados en el decodificador, incluyendo soporte para espacio de audio 3-D
  - Texto y gráficos
- Proporcionar, en la capa de codificación, resistencia a los errores residuales para varios tipos de datos, especialmente para condiciones difíciles del canal, tales como los móviles.
- Representación de varios tipos de objetos en las escenas, permitiendo el acceso independiente a cada objeto para poder ser reutilizado.
- Composición de una escena audiovisual a partir de objetos visuales y de audio ya sean sintéticos y/o naturales.
- Descripción de los objetos y eventos en una escena.
- Proporcionar capacidades de hiper-enlace e interacción.
- Manejar y proteger propiedades intelectuales sobre contenidos audiovisuales y algoritmos de manera que solo los usuarios autorizados puedan tener acceso.

La mayor diferencia con los estándares audiovisuales previos, además de las bases de las nuevas funcionalidades, es el modelo de representación basado en objetos audiovisuales. Una escena basada en objetos está construida utilizando objetos individuales que están relacionados entre ellos en espacio y tiempo. Este sistema presenta varias de ventajas. En primer lugar, los diferentes tipos de objetos pueden estar codificados de diferentes formas, por ejemplo, un busto parlante sintético se representa mejor a través de parámetros de animación mientras que el vídeo se representa mejor a través de los valores de los pixels. En segundo lugar, permite

una integración armoniosa de los diferentes tipos de datos que componen una escena: por ejemplo, un dibujo animado en un mundo real, o una persona real en un estudio virtual. En tercer lugar, es posible la interacción con los objetos e incluso los hiper-enlaces.

Las aplicaciones que se pueden beneficiar de MPEG-4 provienen de muchos y muy diferentes entornos. Por esta razón, MPEG-4 se ha diseñado como una caja de herramientas (*toolbox*) en lugar de un estándar monolítico, utilizando perfiles (*profiles*) que proporcionan soluciones a diferentes niveles.

El estándar se compone de diferentes partes mostradas en la figura 2.7, en la que se ha respetado la nomenclatura inglesa.

MPEG-4	Descripción
ISO/IEC 14496-1	Systems
ISO/IEC 14496-2	Video
ISO/IEC 14496-3	Audio
ISO/IEC 14496-4	Conformant Testing
ISO/IEC 14496-5	Reference Software
ISO/IEC 14496-6	Delivery Multimedia Integration Framework (DMIF)

Tabla 2.7: Partes del estándar MPEG-4

2.4.1. Codificación de vídeo MPEG-4

El concepto central definido en el estándar MPEG-4 es el objeto audiovisual como parte fundamental de la representación basada en objetos. Este tipo de representación proporciona acceso directo a los contenidos de las escenas. En MPEG-4, una escena visual puede consistir en uno o más objetos visuales. Cada objeto de vídeo está caracterizado por información temporal y espacial en forma de movimiento, textura y forma (*shape*). A estos objetos visuales se les conoce con el nombre de VOP (*Video Object Plane*). En ciertas aplicaciones, es posible que no se desee la identificación de los objetos de vídeo ya sea por la dificultad en generar esos objetos o bien por la sobrecarga asociada a la propia generación. Para ese tipo de aplicaciones, MPEG-4 vídeo permite la codificación de imágenes rectangulares que representan el caso degenerado de un objeto visual de forma arbitraria.

En general, las técnicas que se utilizan en MPEG-4 para la compresión de vídeo son las mismas que en los estándares anteriores, basándose en la DCT y en la compensación de movimiento. Sin embargo, como las formas de los objetos visuales pueden ser arbitrarias, es necesario cambiar la sintaxis que define tales objetos introduciendo la noción de forma (*shape*). En cuanto a la compensación de movimiento, un VOP puede ser codificado de tres formas diferentes:

- Independientemente de cualquier otro VOP. En este caso se llama Intra-VOP (I-VOP)
- Puede ser codificado utilizando información de un VOP previo y se denomina Predicted-VOP (P-VOP)
- Puede ser codificado utilizando información de VOPs posteriores y anteriores y se denominan Bidirectional Interpolated VOP (B-VOP)

2.4.1.1. Herramientas para la codificación de la forma

En primer lugar, dentro de una escena visual se debe identificar el objeto visual. Para ello se utiliza una plantilla rectangular en la que se encuentra inscrita la parte de la imagen deseada. Esta plantilla determina qué partes (pixels) pertenecen al objeto visual (*Video Object Plane, VOP*) y qué partes no pertenecen al VOP en un cierto instante de tiempo. Esta plantilla se conoce con el nombre de *binary shape information*. En general, esta plantilla es una matriz de elementos de el mismo tamaño que la caja rectangular que acota el VOP y se le conoce con el nombre de máscara binaria. En esta máscara, cada pixel que pertenece al VOP se establece a un valor de 255 y a un valor de 0 si ese pixel no forma parte del VOP. La plantilla se escoge de manera que tenga el número mínimo de bloques de pixels de tamaño 16x16 ya que el proceso de codificación y decodificación se lleva a cabo sobre bloques de este tamaño denominados *Binary Alpha Blocks* (BAB). El estándar proporciona diferentes algoritmos para la compresión de los BAB pero en la última versión se recomienda el uso de la codificación aritmética basada en contenido (*Content-based Arithmetic Encoding*, CAE). Este algoritmo permite la codificación del BAB utilizando vectores para la compensación del movimiento (Inter-CAE) o bien sin utilizar compensación de movimiento (Intra-CAE). Los vectores de movimiento se codifican de forma diferencial. Cada BAB puede ser codificado utilizando uno de los siguientes modos:

1. El bloque se marca como transparente. En este caso no se lleva a cabo la codificación del bloque. La información de la textura tampoco se codifica.
2. El bloque se marca como opaco. La codificación de la forma no se lleva a cabo, pero sí la codificación de la textura.
3. El bloque se codifica utilizando IntraCAE
4. La diferencia de los vectores de movimiento (MVD) es nula y el bloque no se codifica
5. MVD es nula y el bloque se codifica. En este caso se utiliza InterCAE
6. MDV es no nula y el bloque no se codifica
7. MDV es no nula y el bloque se codifica (InterCAE)

El algoritmo CAE se utiliza para codificar cada pixel en un BAB. El codificador aritmético se inicializa al principio de cada BAB y cada pixel se codifica siguiendo estos 3 pasos:

1. Calcular el número de contexto de ese pixel, basándose en plantillas proporcionadas por el estándar
2. Con el número de contexto indexar una tabla de probabilidades proporcionada por el estándar
3. Utilizar la probabilidad obtenida para derivar la codificación aritmética

2.4.1.2. Herramientas para la codificación de la textura

La información sobre la textura de un VOP está presente en la luminancia, Y, y en las dos componentes de la crominancia, C_b y C_r, de la señal de vídeo. En el caso de un I-VOP, la información de la textura reside directamente en las componentes de luminancia y crominancia. En el caso de los VOPs con compensación de movimiento, la información de la textura

representa el error residual remanente después producirse la compensación de movimiento. La codificación de la información de la textura se lleva a cabo a través de la DCT aplicada a bloques de 8x8 pixels. Para llevar a cabo la codificación de un VOP de forma arbitraria, se seleccionan todos aquellos bloques de 8x8 pixels del VOP que caen enteramente dentro del VOP y se les aplica la DCT, se cuantifican y se codifican utilizando un código de longitud variable. Los bloques de 8x8 que forman parte del contorno del VOP tienen un tratamiento diferente. Para ello, se escogen los macrobloques que forman parte del contorno de VOP y se realiza un proceso de relleno de los pixels del macrobloque que caen fuera del VOP. Este proceso de rellenado se hace en base a bloques de 16x16 en la componente de la luminancia y en base a bloques de 8x8 en las componentes de la crominancia. El objeto de este proceso de rellenado es evitar transiciones abruptas dentro de este tipo de bloques. Una vez realizado este proceso de rellenado, se aplica el proceso de codificación basado en DCT.

2.4.2. Estructura y sintaxis de MPEG-4

Un objeto de vídeo puede consistir de una o más capas que soporten una codificación escalable. La sintaxis escalable permite la reconstrucción del vídeo en forma de capas empezando a partir de una capa base a la cual se añaden un número de capas mejoradas. Esta técnica permite a las aplicaciones la generación de un único flujo de bits de vídeo MPEG-4 que tenga unos requerimientos de ancho de banda o una complejidad computacional variables. Como ya se ha visto, una escena visual en MPEG-4 puede consistir de uno o más objetos visuales. En MPEG-4, la jerarquía que describe una escena visual está mostrada en la figura 2.6 y cada uno de los niveles son:

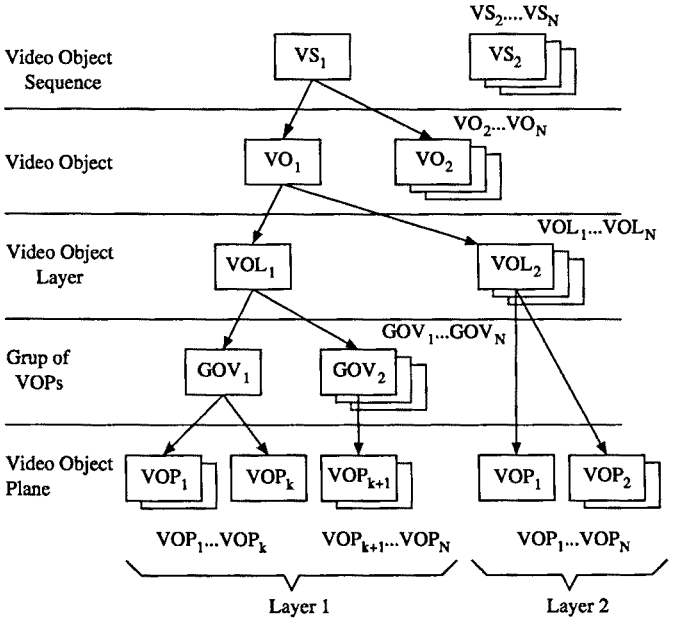


Figura 2.6: Estructura lógica del flujo de bits de vídeo MPEG-4

1. **Visual Object Sequence** (VS): La escena completa MPEG-4 la cual puede contener objetos 2-D o 3-D, sintéticos o naturales y sus capas mejoradas

2. **Video Object** (VO): Un VO corresponde a un objeto particular en la escena. En su forma más simple puede ser una imagen rectangular, o puede ser un objeto con una forma arbitraria.

3. **Video Object Layer** (VOL): Cada uno de los objetos visuales puede ser codificado de forma escalable o no escalable (una sola capa), dependiendo de la aplicación, y se representa por el VOL. El VOL proporciona el soporte necesario para la codificación con escalabilidad. Existen dos tipos de VOL, el primero es el que proporciona la funcionalidad completa de MPEG-4 y el segundo con una funcionalidad reducida pero compatible con H.263.

4. **Group of Video Object Planes** (GOV): Este nivel de la jerarquía es opcional. En este nivel se agrupan los VOPs (ver punto siguiente) y proporciona puntos de entrada en el flujo de bits que permiten el acceso aleatorio a la codificación de cada uno de los VOPs.

5. **Video Object Plane** (VOP): Un VOP es una muestra temporal de un VO.

2.4.3. Escalabilidad

La escalabilidad en MPEG-4 utiliza los mismos sistemas básicos de escalabilidad de MPEG-2 con la salvedad de que las técnicas de codificación escalable de vídeo pueden ser aplicadas selectivamente a los diferentes objetos visuales que forman una escena. Por ejemplo, la escalabilidad espacial soporta el cambio de la calidad de textura (tanto SNR como de resolución espacial) de un objeto. Se obtiene un sistema de escalabilidad de vídeo basado en el contenido que hace posible la mejora de la resolución espacial, la SNR o la exactitud de los contornos de los objetos visuales de interés o de una región de la imagen y que puede ser cambiada de forma dinámica en tiempo de presentación.

En la última versión del estándar MPEG-4 se ha aceptado un nuevo sistema de escalabilidad denominado FGS (*Fine Granularity Scalability*). La escalabilidad FGS utiliza la codificación de planos de bits (*bitplanes coding*) de la DCT para representar los flujos mejorados. La codificación de planos de bits se basa en desglosar el valor binario de un parámetro en tantos niveles como bits tenga la codificación binaria del valor. Así, por ejemplo, si un coeficiente de la DCT está codificado con 7 bits, existen hasta 7 planos de bits posibles. El fundamento de la escalabilidad FGS es enviar en el flujo base un número dado de planos de bits para los coeficientes de la DCT y en los flujos mejorados enviar más planos de bits añadiendo mayor resolución del valor de los coeficientes de la DCT.

2.4.4. Perfiles y niveles

En MPEG-4, los perfiles se definen en términos de los tipos de objetos de vídeo. Un tipo de objeto de vídeo determina un subconjunto de herramientas del estándar MPEG-4 visual que proporcionan una funcionalidad o un grupo de funcionalidades. En la tabla 2.8 se muestran los seis tipos de objetos naturales de vídeo.

Los diferentes perfiles de MPEG-4 se muestran en la tabla 2.9.

Un nivel dentro de un perfil determina las restricciones sobre los parámetros en el flujo de bits relacionados con las herramientas de ese perfil. Actualmente hay once definiciones de perfiles y niveles de vídeo natural y en el que cada uno determina las restricciones de

	MPEG-4 tipos de objetos de vídeo					
MPEG-4 video tools	Simple	Core	Main	Simple Scalable	N-Bit	Still Scalable Texture
Basic	•	•	•	•	•	
Error resilience	•	•	•	•	•	
Short Header	•	•	•		•	
B-VOP		•	•	•	•	
P-VOP with OBMC						
Quantization Method 1 / Method 2		•	•		•	
P-VOP based temporal scalability		•	•		•	
Binary Shape		•	•		•	
Grey Shape			•			
Interlace			•			
Sprite			•			
Temporal Scalability (Rectangular)				•		
Spatial Scalability (Rectangular)				•		
N-Bit					•	
Scalable Still Texture						•

Tabla 2.8: Tipos de objetos de vídeo de MPEG-4

aproximadamente 15 parámetros definidos en cada nivel. La tabla 2.10 muestra los tres perfiles más importantes – Simple, Core y Main – y un subconjunto de restricciones de los niveles.

2.4.5. Modelos de verificación de vídeo

Para poder realizar productos competitivos con implementaciones económicas, MPEG-4 proporciona modelos que permiten determinar los requerimientos máximos de memoria y cálculo computacional. Para conseguir este propósito se define un sistema de verificación de vídeo que consiste en tres modelos normativos. Estos modelos se utilizan en la definición de los perfiles y niveles descritos.

1. Video rate buffer verifier (VBV): Proporciona cotas sobre los requerimientos de memoria del buffer necesario en un decodificador de vídeo.

2. Video complexity verifier (VCV): Proporciona cotas sobre requerimientos de la velocidad de procesado del decodificador de vídeo en términos de macrobloques por segundo.

3. Video reference memory verifier (VMV): Proporciona cotas sobre los requerimientos del tamaño de la memoria del decodificador de vídeo medido en macrobloques.

	MPEG-4 video profiles					
MPEG-4 video object types	Simple	Core	Main	Simple Scalable	N-bit	Still Scalable Texture
Simple	•	•	•	•	•	•
Core		•	•		•	
Main			•			
Simple Scalable				•		
N-bit					•	
Scalable Still Texture			•			•

Tabla 2.9: Perfiles de vídeo de MPEG-4

Profile	Simple Profile			Core Profile		Main Profile		
Level	L1	L2	L3	L1	L2	L1	L2	L3
Typical Scene Size	QCIF	CIF	CIF	QCIF	CIF	CIF	ITU-R601	1920x1088
Bitrate (bit/s)	64 K	128 K	384 K	384 K	2 M	2 M	15 M	38.4 M
Maximum number of objects	4	4	4	4	16	16	32	32
Total mblk memory	198	792	792	594	2376	2376	9720	48960

Tabla 2.10: Subconjunto de perfiles y niveles de vídeo de MPEG-4

CAPÍTULO 3

Transmisión de vídeo sobre redes IP

Durante varios años, el proceso de visualización de una secuencia de vídeo que reside en un servidor, se ha basado en la descarga del archivo de vídeo del servidor al ordenador local y posteriormente, el visualizado de la secuencia de vídeo. Este sistema es el más difundido actualmente y en realidad se basa en los mecanismos genéricos de descarga de ficheros a través de FTP o más modernamente a través de HTTP. Este sistema presenta sus ventajas ya que permite utilizar protocolos de transmisión como TCP que van modificando su tasa de transmisión en función de las condiciones de la red. En este caso, si la red está puntualmente congestionada, la transmisión de la secuencia de vídeo se verá afectada y, posiblemente, retardada pero no el visualizado ya que se produce a posteriori de la descarga. La principal desventaja de este sistema es que los ficheros de vídeo tienden a ser muy voluminosos (del orden de 1 a 8 GB) y por lo tanto el tiempo de descarga es muy alto.

Otra técnica utilizada para la transmisión de vídeo es la llamada *"streaming video"*. Esta técnica se basa en la simultaneidad de la transmisión y la reproducción del vídeo. A medida que la transmisión de vídeo va llegando al receptor, éste va reproduciendo la visualización. Las ventajas frente a la técnica anterior son obvias; por un lado la reproducción de la secuencia de vídeo empieza en poco tiempo después de la petición de descarga. Por otro lado, no es necesario el almacenamiento local del fichero de la secuencia de vídeo. Estas ventajas permiten que servicios del estilo de "pay-per-view" sean viables, como han demostrado las redes de distribución de contenidos (CDN, *Content Delivery Networks*) Akamay, Digital Islands, Spread_It o Imagenio.

3.1. Transmisión de vídeo. Generalidades

En general, un sistema de transmisión digital de vídeo sería como el presentado en la figura 3.1. La señal capturada por la cámara es digitalizada. La información resultante pasa a un codificador donde se aplicará algún proceso de compresión. La secuencia de vídeo es, por tanto, comprimida y enviada a la red de comunicaciones. En general, el codificador llevará asociado un controlador de tasa que adecuará la transmisión sobre la red. Por su parte, el receptor aplicará el proceso inverso, convirtiendo los datos descomprimidos a una señal analógica que será enviada al monitor para ser visualizada.

El sistema de comunicación debe limitar las pérdidas y el retardo de acuerdo a los requisitos concretos de cada aplicación de vídeo. La mayor parte de la distorsión introducida se produce durante el proceso de codificación. Por otra parte, los errores de transmisión pueden provocar

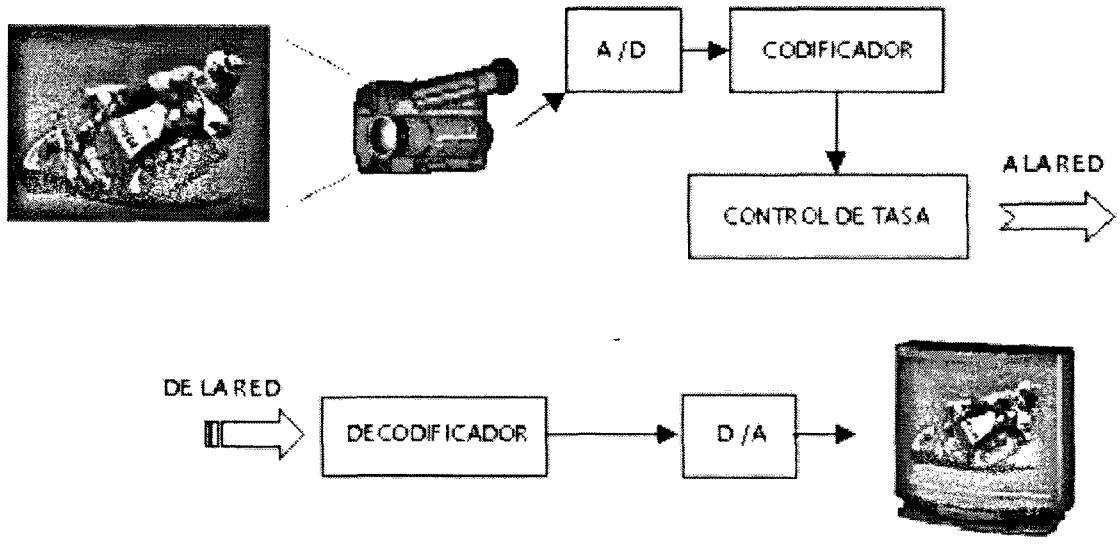


Figura 3.1: Sistema de transmisión de vídeo

distorsión que se extendería a varias imágenes debido a la dependencia existente entre ellas por el algoritmo de compresión.

Las recomendaciones para el máximo retardo de transferencia siguen las especificadas para las conversaciones de voz. Así, la degradación es mínima por debajo de los 150 ms y muy fuerte por encima de los 400 ms [Kar96]. Por su parte, la información de vídeo puede anteceder a la audio hasta en 100 ms o seguirla hasta en 20 ms. Los valores anteriores son válidos para comunicaciones en un sentido, mientras que para servicios interactivos los requisitos serían aún mayores.

Los usuarios de servicios de comunicaciones incluyendo transmisión de vídeo, esperan la mayor calidad con el coste más reducido posible. Esto lleva a una serie de compromisos a la hora de efectuar la comunicación, esquematizados en la tabla 3.1 [Kar96]. Por ejemplo, la

	Maximizar	Minimizar
Sesión	Calidad	Coste
Codificación	Calidad	Tasa de bit
Control de Tasa	Calidad homogénea	Variabilidad de tasa
Transferencia	Utilización de recursos	Espera en colas
Control de errores	Recuperación de errores	Redundancia añadida

Tabla 3.1: Compromisos en la transmisión asíncrona de vídeo

señal de salida de un codificador de vídeo presentará una tasa de bit mayor cuanto más alta deseemos la calidad suministrada. Por lo que respecta a la transmisión, al utilizar multiplexaciones estadísticas a lo largo de la red, aparecerán unos retardos de espera en las colas de los distintos nodos de conmutación, que serán tanto mayores cuanto más queramos utilizar y compartir los recursos disponibles. Finalmente, mejorar la fiabilidad del sistema frente a

errores de transmisión implica utilizar mecanismos de detección y corrección de errores, que disminuyen el rendimiento del sistema debido a la redundancia (*overhead*) añadida.

Uno de los principales tópicos a la hora de transmitir vídeo comprimido es el control de tasa. Como se verá más adelante, la tasa de bit generada por un codificador de vídeo es variable, debido a dos motivos principales. En primer lugar, la variabilidad de las escenas a codificar, que en unas ocasiones serán más complejas que en otras. Por otro lado, los propios algoritmos de compresión más comúnmente utilizados generan variaciones periódicas de la tasa. Así, para mantener una calidad semiconstante en la transferencia de vídeo, será necesario transmitir una secuencia de tasa variable. Por otra parte, en ocasiones es interesante transmitir a tasa semiconstante, lo cual obligará al codificador a variar la relación de compresión media, provocando asimismo variaciones de calidad. Además, el control de tasa se hará también necesario para evitar que el codificador genere un número mayor de bits por unidad de tiempo que el que le es permitido transmitir por la red. En la figura 3.2 se esquematizan las dos posibles formas de transmisión, representando con R la tasa de transmisión y con Q la calidad en función del tiempo.

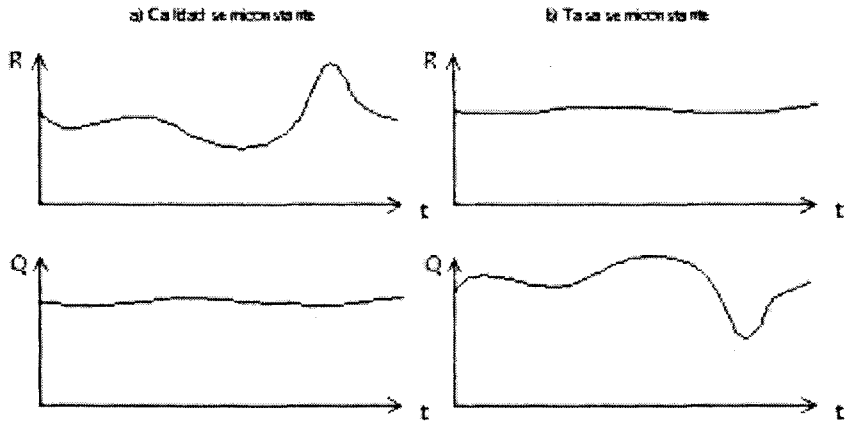


Figura 3.2: Transmisión con tasa variable y con tasa constante.

La codificación a tasa variable (*Variable Bit Rate*, VBR) es común en aplicaciones de almacenamiento, como el disco versátil digital (*Digital Versatile Disc*, DVD). El ahorro de capacidad utilizando esta técnica ha sido analizado en [GKL+98]. Por ejemplo, para almacenar un video-clip de 15 minutos utilizando codificación a tasa constante (*Constant Bit Rate*, CBR) a 6 Mbps se requieren 675 Mbytes, mientras que utilizando VBR con media 3 Mbps y tasa de pico 6 Mbps se necesitan sólo 355 Mbytes. Estos valores se traducen en una ganancia de aproximadamente un 50 %, si bien los resultados varían dependiendo de las secuencias a codificar.

En este momento es conveniente distinguir entre codificación VBR y CBR y transmisión VBR y CBR. Por una parte, y como ya se ha comentado, la codificación VBR presenta la gran ventaja de mantener la calidad de la codificación de vídeo constante a lo largo de toda la secuencia. Sin embargo, a la hora de transmitir esta secuencia, puede ser más cómodo para la red que se realice en modo CBR, debido a su mayor simplicidad. El problema de adaptar el flujo VBR proveniente del codificador al canal CBR de la red ha sido propuesto en estándares de transmisión como H.261 y MPEG-1. En varios trabajos de investigación se

han propuesto técnicas de conformación tanto a nivel espacial como a nivel temporal. Las primeras tratan de obtener un flujo CBR mediante la multiplexación estadística de varios flujos VBR. Como se comentará posteriormente, en muchos escenarios esta solución no va a ser adecuada. En [LT96] y [LC97] se propone una técnica denominada *agregación de tráfico*, en la cual se comprimen y multiplexan los distintos flujos de vídeo antes de ser paquetizados y enviados al transporte CBR. Cuando la suma de todos los flujos de vídeo supere la capacidad de transporte CBR disponible se deberá descartar tráfico. La ventaja del sistema reside en que este descarte se lleva a cabo antes de formar los paquetes, conociendo el tipo de tráfico que se está descartando y consiguiendo que el impacto final en el detrimento de la calidad sea menor. Cuando el descarte se realiza directamente sobre los paquetes, sin importar su contenido, se puede eliminar información muy importante sin la cual enviar otras informaciones no tiene sentido. El principal problema de esta técnica es que sólo es válida en escenarios en los cuales el grupo completo de canales de vídeo deba atravesar conjuntamente toda la red.

Las técnicas de conformación temporal se basan en la inclusión de un buffer entre el codificador y la red, de forma que el tráfico que entra al buffer a tasa variable se va extrayendo de él a tasa constante [LT98]. La principal implicación de esta técnica es el aumento del retardo.

Dentro de las transmisiones a tasa variable, es posible hacer una nueva distinción, dependiendo del grado de control que se aplique a dicha tasa. Así, sería posible distinguir entre tasa variable no regulada, suavizada y regulada, como se esquematiza en la figura 3.3 . En el primer caso, la información se envía a la red a la misma tasa que es generada por el codificador. En el segundo, un buffer intermedio permite un primer intento de suavizado, pero sin llegar a producir una salida a tasa constante. Finalmente, el tercer caso incluye un control de la tasa binaria dependiendo de la cantidad de información que se vaya acumulando en el buffer, con lo que es posible aumentar o reducir la tasa generada por el codificador. Así, la tasa de salida está más controlada, pero se producirán variaciones de calidad con lo que nos podemos acercar al modelo de tasa constante y calidad variable.

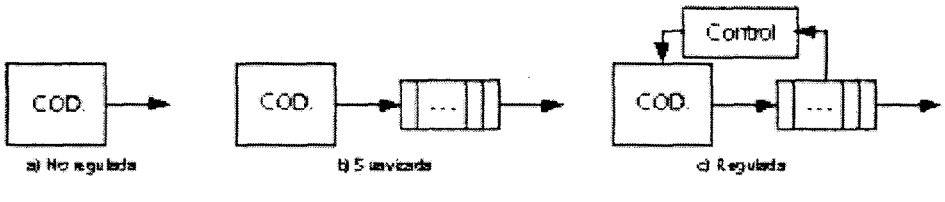


Figura 3.3: Regulación y suavización de tasa variable

La técnica VBR no parece adecuada en primera instancia para transmisión al no limitar la relación de rafagueo de la señal. Posteriormente se hablará con más detalle de este parámetro. Para obtener un ahorro de ancho de banda, como ya se ha comentado, se podrían multiplexar estadísticamente un gran número de flujos de vídeo de forma que el agregado tuviese tasa constante [YJZ93]. De todas formas, esta técnica presenta dos inconvenientes. En primer lugar, el número de fuentes multiplexadas debe ser muy grande, lo cual no va a ser lo más común cuando se habla de fuentes de vídeo. Por otra parte, al transmitir sobre redes conmutadas, los grupos de canales de vídeo no van a ser uniformes a lo largo de ellas. De ambas afirmaciones podemos concluir que, para aplicaciones de transmisión de vídeo sobre redes conmutadas, la tasa debe ser controlada para cada fuente individual.

Sin embargo, aún controlando de forma individual la tasa generada por las fuentes, los recursos de red serían utilizados de forma más eficiente mediante una transmisión VBR [GKL+98]. En [DT97] se lleva a cabo un estudio en base a simulaciones mediante el cual se revela como el número de fuentes VBR multiplexadas puede ser bastante superior al de fuentes CBR, manteniendo los mismos recursos de red y los mismos requisitos de calidad y retardo, especialmente cuando el contenido de la secuencia de vídeo es muy variable. Resultados que corroboran estas afirmaciones se encuentran también en [HOR97][Rea][PZ94]. El problema está por una parte en cómo controlar la tasa del codificador, y por otra en cómo asignar los recursos de red de forma más óptima, manteniendo siempre el grado de servicio por encima del nivel esperado en cada aplicación.

3.2. Transmisión de vídeo en Internet

La transmisión de vídeo en Internet está sujeta a las variaciones de los retardos inherentes a una red donde el servicio de entrega es clásicamente un best-effort, a la probabilidad de pérdida de paquetes y a las variaciones del ancho de banda. Si se utiliza la técnica de *streaming video*, los datos que durante la transmisión se pierden o llegan al receptor después del instante temporal en que debían ser visualizados, no son de ninguna utilidad. El objetivo es diseñar un sistema que permita la entrega de vídeo de alta calidad. Los problemas que presentan la variación del ancho de banda, las variaciones del retardo y las pérdidas de paquetes son sus características dinámicas e impredecibles.

3.2.1. Las variaciones del ancho de banda

En la Internet actual no es posible realizar una reserva de ancho de banda y el ancho de banda proporcionado es dinámico. Los efectos que se derivan de este problema en la transmisión de vídeo afectan directamente a la calidad de la imagen recibida, ya que si se transmite a una tasa superior que el ancho de banda disponible, se producirá congestión en la red y muy posiblemente, los dispositivos de red congestionados descarten paquetes de nuestra transmisión de vídeo con lo que la calidad percibida en recepción descenderá notablemente. Por otro lado si la tasa de transmisión es menor al ancho de banda disponible tendremos una calidad de vídeo en recepción sub-óptima. Una solución es diseñar un sistema que adapte la tasa de transmisión de vídeo al ancho de banda disponible. El control de la tasa se puede realizar desde el transmisor o bien desde el receptor:

1. Control de la tasa de transmisión basado en el transmisor
   En este caso, la fuente ajusta la tasa de generación del codificador de vídeo para que se adapte al ancho de banda disponible. Para ello, utiliza información enviada desde el receptor que le permite estimar el ancho de banda y la tasa de pérdidas de paquetes. Clásicamente, el protocolo utilizado para la transmisión de este tipo de información es RTP y RTCP.
   Los métodos que puede utilizar el receptor para estimar el ancho de banda basándose en la tasa de pérdidas de paquetes son básicamente dos:

   a) Métodos basados en pruebas. La idea básica es realizar experimentos que permitan estimar el ancho de banda disponible. Por ejemplo si se desea mantener la tasa de pérdida de paquetes por debajo de un cierto umbral P_{th}, se puede ir incrementando la tasa de transmisión hasta que la tasa de pérdidas de paquetes medida (ρ)

supere el umbral establecido (P_{th}) y a partir de ese instante decrementar la tasa de transmisión.

b) Métodos basados en modelo matemáticos. El objetivo de estos métodos es asegurar una compartición justa del ancho de banda con los flujos TCP concurrentes en la red. La idea básica es modelar el caudal medio de un flujo TCP y transmitir el vídeo con el mismo caudal como si fuese un flujo TCP. Se está buscando que el caudal de vídeo tenga un comportamiento a nivel "macroscópico" similar a un flujo TCP. En este caso se dice que el caudal de vídeo es *"TCP-friendly"*.

Una pregunta habitual sobre el protocolo de transporte utilizado en *"streaming video"* es porque no se utiliza TCP. La respuesta es que TCP garantiza la entrega a través de mecanismos de retransmisión, los cuales llevan a variaciones temporales del caudal y del retardo. Por otro lado, TCP utiliza un mecanismo de control de tasa del tipo AIMD (*additive-increase multiplicative-decrease*) que provocan oscilaciones de la tasa sobre un valor medio que van en detrimento del sistema de *streaming*. Por estas razones es habitual

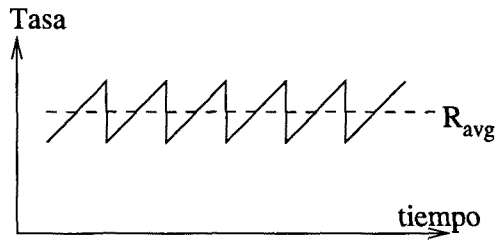


Figura 3.4: Oscilaciones de la tasa de TCP por el efecto AIMD

utilizar UDP como protocolo de transporte sobre redes IP para el *"streaming video"*.

En este modelo de control de tasa basado en el transmisor, un punto importante es cómo conseguir que el transmisor adecúe la tasa de salida del codificador de vídeo al ancho de banda disponible. Durante los últimos años han aparecido varios artículos que describen modificaciones de los codificadores MPEG de vídeo para obtener una tasa controlada a la salida ([LMR99],[CCLS02], [LS03])

2. Control de la tasa de transmisión basado en el receptor
   En este caso, el receptor selecciona la tasa de vídeo de un conjunto de tasas posibles. Un ejemplo claro de este caso es cuando se realiza la transmisión multicast de vídeo escalable. El transmisor ha codificado la secuencia original de vídeo en varios niveles o capas, con un flujo base de baja calidad y un conjunto de flujos que combinados con el flujo base proporcionan una calidad mejorada y asocia cada uno de los flujos a un grupo multicast diferente. El receptor, en función de su ancho de banda estimado, se apunta a uno o más grupos multicast para recibir el flujo base únicamente o el flujo base y los flujos mejorados que pueda aceptar.

3.2.2. Variaciones del retardo

Las variaciones del retardo en la transmisión de paquetes es un problema para los sistemas de *streaming video* ya que una vez que el receptor inicia la presentación de la secuencia de vídeo,

necesita mostrar una imagen cada cierto tiempo y que el vídeo presentado fluya temporalmente de forma correcta; para ello, el receptor necesita que los paquetes que transportan el vídeo estén presentes en recepción en los instantes precisos para poder decodificar y mostrar la imagen. Si un paquete llega más tarde que el instante preciso necesario para que la información de vídeo pueda ser mostrada, esa información ya no es de ninguna utilidad.

El retardo extremo a extremo en Internet depende de los retardos de propagación, los retardos de procesado y encolamiento de los paquetes en los routers y el tiempo de procesado de los extremos. Algunos de estos retardos, son fijos pero otros son variables y dependen de la carga de los sistemas.

La solución ampliamente implementada en los sistemas de *streaming video* para paliar los efectos indeseables de la variación del retardo en la red, es la de utilizar un buffer en el sistema de recepción que absorba estas variaciones. Esta solución presenta el inconveniente de que el tiempo necesario para visualizar la secuencia de vídeo desde el momento de la petición aumenta debido a que el buffer debe ser llenado. El tamaño del buffer suele ser de hasta 8Mb, que corresponde, aproximadamente, a medio minuto de vídeo. La figura 3.5 muestra la relación entre el retardo de transmisión y el retardo de presentación y el efecto del buffer.

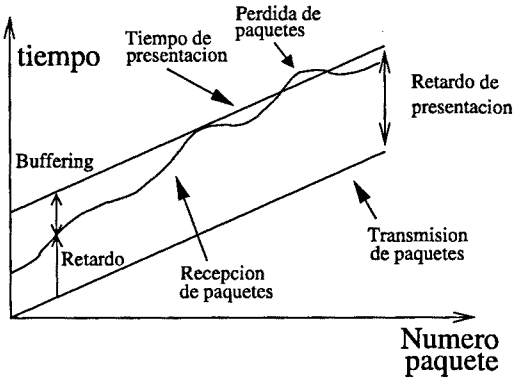


Figura 3.5: Variaciones del retardo

El diseño del tamaño del buffer es crítico ya que cuanto mayor sea el buffer de recepción (y mayor sea el retardo de presentación) existirá una probabilidad menor de que los paquetes lleguen fuera de su tiempo de presentación.

3.2.3. Pérdidas de paquetes

El objetivo, en este caso, es evitar el impacto de los errores debidos a las pérdidas de paquetes en redes de conmutación de paquetes o a las ráfagas de errores de bit en los enlaces inalámbricos. En general se utilizan cuatro sistemas de control de errores:

- Control de errores mediante codificación de canal
  - Técnicas de FEC (*Forward Error Correction*)
  - Retransmisiones (ARQ, *Automatic Repeat Request*)
- Control de errores mediante codificación de fuente

- Ocultamiento del error (*Error Concealment*)
- Codificación de vídeo resistente a errores (*Error-resilient video coding*)

En general, las técnicas de control de errores basadas en la codificación de canal aplicadas a la transmisión de vídeo tienen las mismas ventajas e inconvenientes que aplicadas a la transmisión en general, con la excepción de que cuando se utilizan técnicas de retransmisiones, deben retransmitirse únicamente aquellos paquetes que todavía puedan alcanzar al receptor antes de su tiempo de presentación (esquema con restricciones de retardo), o bien retransmitir los paquetes más importantes primero (esquema basado en prioridades). Cuando se combinan las técnicas de control de errores basadas en la codificación de canal con el hecho de que en la información de vídeo algunos bits son más importantes que otros y que no es estrictamente necesario realizar una entrega fiable de todos los bits, surgen esquemas que unen la codificación de canal y la codificación de fuente con el objeto de obtener diseños de codificadores de fuente y de canal que exploten esas diferencias. Por ejemplo, en la codificación de vídeo MPEG existen diferentes tipos de imágenes (I,P,B) cuya importancia entre ellas es relativa:

1. Las imágenes I son muy importantes
2. Las imágenes P son menos importantes
3. Las imágenes B son poco importantes (pueden ser descartadas)

Otro ejemplo de importancia relativa ocurre cuando se realiza una transmisión de un vídeo con codificación escalable:

1. El flujo base tiene máxima importancia
2. El primer flujo mejorado tiene una importancia menor
3. El segundo flujo mejorado tiene una importancia mínima

Aplicando estas características a los esquemas de *FEC* y de retransmisiones se obtendrían los siguientes esquemas representados en las tablas 3.2 y 3.3. En ambos casos, el *FEC* utiliza un

	Imagen I	Imagen P	Imagen B
FEC	Máxima	Media	Mínima
ARQ	Máxima	Media	Descartable

Tabla 3.2: Codificación de canal aplicada a la imágenes I,P,B de vídeo

sistema de protección de errores diferente en función de la importancia de los datos a proteger y el sistema de retransmisiones utiliza un esquema de retransmisión con prioridades.

Otras técnicas de control de errores son aquellas que se basan en la codificación de fuente. En las técnica denominada de ocultación de errores (*error concealment*) el objetivo principal es estimar la información perdida con objeto de ocultar el hecho de que ha ocurrido un error que ha llevado a la pérdida de cierta información. La clave está en que el vídeo tiene una correlación fuerte tanto temporal como espacial. Esta correlación, que fue utilizada para obtener la compresión del vídeo, es la que se utiliza para estimar la pérdida de la información. El

	Flujo base	F. Mejorado 1	F. Mejorado 2
FEC	Máximo	Medio	Mínimo
ARQ	Máximo	Medio	Descartable

Tabla 3.3: Codificación de canal aplicada a la codificación de vídeo escalable

objetivo básico en esta técnica es explotar la correlación realizando algún tipo de interpolación (o extrapolación) espacial y/o temporal para estimar la información perdida a partir de los datos correctamente recibidos.

La codificación de vídeo resistente a errores (*error resilient video coding*) trata del conjunto de técnicas aplicadas a la codificación de vídeo para que esa codificación sea resistente a tipos específicos de errores que se pueden producir de forma más o menos habitual. La mayoría de sistemas de compresión de vídeo utilizan arquitecturas similares basadas en la predicción del movimiento entre imágenes, la DCT (u otro tipo de transformada espacial) del error de predicción, seguido de una codificación de entropía (por ejemplo una codificación de Huffman o una codificación *runlength*) de los parámetros. Los dos tipos de problemas, inducidos por los errores, que afectan a sistemas basados en este tipo de arquitecturas son:

1. Pérdida de la sincronización del flujo de bits

2. Estados incorrectos y propagación del error

La primera clase de problemas, pérdida de la sincronización del flujo de bits, hace referencia al caso en el que el error puede producir una confusión en el decodificador de vídeo y éste pierda la sincronización del flujo de bits, es decir, que el decodificador no sepa qué bits corresponden a qué parámetros. La segunda clase de problemas, estados incorrectos y propagación del error, hace referencia a qué ocurre cuando un error afecta a un sistema que utiliza codificación predictiva.

Los problemas derivados de la utilización de códigos de longitud variable (VLC) es que el error en un solo bit de una de las palabras código puede llevar, al decodificador, a la interpretación errónea de la longitud de la palabra código donde se ha producido el error y por esta razón, a la interpretación errónea del resto de palabras código subsiguientes del flujo de bits, al menos hasta que se produzca una cierta resincronización. Es interesante destacar que aunque una codificación de longitud fija (FLC) permite que los errores queden limitados a una única palabra código, no proporcionan un nivel de compresión tan bueno como los VLC. Las soluciones adoptadas para evitar los problemas de pérdida de sincronización son las siguientes:

1. Marcadores de resincronización
   La forma más simple que permite resincronizar el flujo de bits es a través del uso de marcas de resincronización. La idea básica es introducir puntos de entrada a lo largo del flujo de bits que sean fácilmente identificables de manera que si el decodificador pierde el sincronismo, busque el siguiente punto de entrada y empiece de nuevo a decodificar. Esas marcas se definen de forma que sean transparentes a las palabras código y se pone suficiente información después de ellas para permitir que el decodificador pueda reiniciar el proceso de decodificación.

2. Códigos de longitud variable reversibles
   Los códigos de longitud variable convencionales, tales como el de Huffamn, solo son decodificables en una dirección. Los códigos de longitud variable reversibles (RVLCs) tienen la propiedad de ser decodificables en ambas direcciones. Esta propiedad puede ser utilizada para recuperar datos que de otra forma se hubiesen perdido. Por ejemplo, cuando se produce un error, el decodificador salta al siguiente marcador de resincronización y en lugar de descartar todos los datos entre el punto de error y el marcador de resincronización, el decodificador puede iniciar la decodificación desde el marcador y en dirección reversa hasta que se identifica otro error de decodificación, permitiendo la recuperación parcial de la información. El problema es que los RVLC son, en general, menos eficientes que los VLC.

3. *"Data partitioning"*
   Una observación importante es que los datos que siguen a los marcadores de resincronismo tiene más probabilidades de ser decodificados correctamente que aquellos datos que están a más distancia. Esta observación ha generado que en los nuevos estándares como MPEG4 sitúen los datos de mayor importancia (los vectores de movimiento, la componente continua de la DCT, etc) en lugares cercanos a los marcadores de resincronismo y situar los datos de menor importancia, como los coeficientes AC de la DCT en lugares más distantes de estos marcadores.

El otro tipo de problema inducido por los errores es el de los estados incorrectos y propagación de los errores. Cuando se produce una pérdida y aún cuando el flujo de bits haya sido resincronizado, puede ocurrir que el estado de representación del decodificador sea diferente que el del codificador. En particular, cuando se utiliza predicción para la compensación de movimiento, un error provoca que la imagen reconstruida (estado) en el decodificador sea diferente de la del codificador. Por esa razón, las siguientes imágenes basadas en la imagen reconstruida llevan a estados diferentes entre el codificador y el decodificador, provocando que el error se propague a las imágenes posteriores cuando se está utilizando codificación predictiva. En general, en MPEG, se utiliza codificación Intra de forma periódica (GoP) y por esta razón la propagación del error está limitada como máximo al número de imágenes que forman un GoP.

3.3. Calidad de Servicio (*QoS*)

Actualmente, *Internet* ofrece únicamente un servicio de *best-effort* que no es capaz de satisfacer las necesidades de una comunidad creciente de usuarios cuyos servicios demandan un mayor ancho de banda. A medida que los avances tecnológicos permiten incrementar el ancho de banda de las redes, aparecen nuevas aplicaciones en el mercado que también demandan un mayor ancho de banda.

El desarrollo de técnicas avanzadas de compresión han permitido la aparición de nuevas aplicaciones en tiempo real tales como la videoconferencia y el *streaming video*. Estas nuevas aplicaciones multimedia requieren una serie de recursos garantizados (ancho de banda, espacios en *buffers*, potencia de procesado) con el objetivo de ofrecer una alta calidad de servicio a sus usuarios. Algunas aplicaciones necesitan una velocidad de transmisión de datos garantizada que si no puede ser proporcionada provocará que algunos paquetes de datos se pierdan durante la transmisión produciéndose un descenso notable de la calidad ofrecida. Por otro lado, algunas aplicaciones pueden adaptar la calidad ofrecida (velocidad de transmisión de los datos o nivel de compresión de vídeo) dependiendo de las características de la red.

Debido a estas razones, se han desarrollado varios mecanismos, protocolos y sistemas de señalización para proporcionar una calidad de servicio (*QoS*) a las aplicaciones que la necesitan; especialmente las aplicaciones multimedia.

Para poder hablar de *calidad de servicio*, primero se debe definir que se entiende por *calidad de servicio*. No es fácil definir qué es la *QoS*. En [Hus00] el autor utiliza casi todo el primer capítulo del libro para definir la *QoS*. El problema es que los conceptos de *calidad* y de *servicio* son demasiado amplios y por ello *calidad de servicio* es también un concepto amplio y vago. A todo esto, se suman las diferentes interpretaciones que proporcionan para la *calidad de servicio* la ITU/ETSI y el IETF.

En [Har01] se distinguen tres nociones de *QoS* — intrínseca, percibida y valorada — que constituyen el modelo general mostrado en la tabla 3.4.

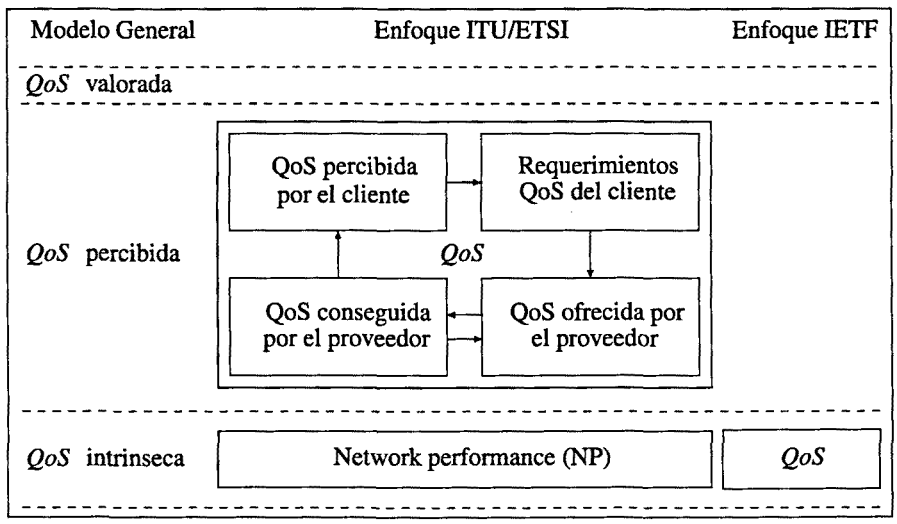


Tabla 3.4: Modelo general de *QoS*

La *QoS* intrínseca pertenece a las características del servicio que tienen que ver con aspectos técnicos y está determinada por el diseño de la red de transporte y cómo se hayan estipulado los accesos, las terminaciones y conexiones a la red. La calidad requerida se consigue, entre otros factores, a través de una selección apropiada del protocolo de transporte, los mecanismos que aseguran la *QoS* y los valores de los parámetros relacionados. La forma de evaluar la *QoS* intrínseca es a través de la comparación de las características medidas y esperadas.

La *QoS* percibida refleja las experiencias del cliente cuando utiliza un servicio particular y está influenciada por las expectativas que tiene el cliente comparadas con las características observadas del servicio. Es importante comentar que las expectativas personales de un cliente pueden verse afectadas por las experiencias previas de un servicio similar que haya utilizado ese cliente así como por las opiniones de otros clientes. De esta forma, una *QoS* con las mismas característica intrínsecas puede ser percibida de forma distinta por diferentes clientes, de lo que se deduce que asegurando ciertos parámetros de red para un servicio dado puede no ser suficiente para satisfacer a los diferente clientes quienes a su vez no tienen porque saber cómo se está proporcionando ese servicio.

La *QoS* valorada comienza a ser tenida en cuenta cuando el cliente decide si desea continuar utilizando el servicio o no. Está decisión depende de la calidad percibida, el precio del servicio

y el comportamiento del proveedor de servicios ante las demandas y consultas del cliente frente a posibles problemas. Ni la ITU, ni la ETSI, ni el IETF hacen alusión a este tipo de *QoS*.

El enfoque que tienen tanto la ITU como la ETSI sobre calidad de servicio es casi la misma y utilizan la siguiente definición básica [ITU93] :

El efecto colectivo de las características del servicio que determinan el grado de satisfacción de un usuario del servicio

Esta definición tiene que ver con la calidad de servicio percibida más que con la calidad de servicio intrínseca. La ITU introduce la noción de *network performance* (NP) para cubrir los aspectos técnicos haciendo una clara distinción entre *QoS*, entendida como algo que afecta a los efectos percibidos por el usuario y NP que engloba a todos aquellos parámetros de red esenciales para proporcionar un servicio. Los parámetros de la *QoS* están definidos dentro del ámbito del usuario y no tienen una traducción directa a los parámetros de red. Por otro lado, las características de los parámetros de red determinan la calidad observada por los clientes y por lo tanto existe un mapeo consistente entre los parámetros de la *QoS* y los parámetros de red.

El enfoque que tiene el IETF sobre la calidad de servicio es de una *QoS* intrínseca y no dice nada sobre la percepción o la valoración del servicio por parte del cliente. La IETF define la *QoS* de la siguiente manera [CNRS98]:

El conjunto de requerimientos de un servicio que debe proporcionar la red mientras transporta un flujo

Esta definición es equivalente a la noción de NP (*Network Performance*) de la ITU/ETSI y está definida en términos de parámetros.

Durante los últimos años, el IETF ha puesto mucha atención en la calidad de servicio en redes IP. Ha propuesto dos arquitecturas de red que soportan *QoS*: *IntServ* [BCS94] y *DiffServ* [BBC+98]. Ha estandarizado un protocolo de señalización, *Resource Reservation Protocol* (RSVP), originalmente relacionado con el modelo de implementación de *IntServ* y extendido posteriormente para otros propósitos.

El IETF define cierto número de parámetros de *QoS* independientes de la arquitectura de red así como otro conjunto específico de parámetros relacionados directamente con los dispositivos de red como los medidores de tráfico, marcadores de paquetes, clasificadores de paquetes, etc., que constituyen una arquitectura de red particular. En general, en la *QoS* en redes IP existe una relación intrínseca entre los dispositivos de red y la "calidad" experimentada por los paquetes.

La *QoS* intrínseca en redes de conmutación de paquetes queda definida mediante el siguiente conjunto de parámetros:

- La tasa de transmisión que se puede conseguir para un cierto tipo de servicio
- El retardo experimentado por los paquetes mientras atraviesan la red. El retardo puede ser considerado extremo a extremo o simplemente el retardo producido por un cierto dispositivo de red.
- El *jitter*, las variaciones en el retardo de transmisión.
- El índice de las pérdidas de paquetes expresado como el número de paquetes perdido frente al total de paquetes enviados.

Este conjunto de parámetros expresan la calidad con la que van a ser tratados los paquetes al atravesar una red. Estos parámetros se pueden traducir a otro conjunto de parámetros relacionados con una arquitectura de red en particular para asegurar la *QoS*.

3.4. Arquitecturas de red con *QoS* en IP

Tal y como se ha comentado en la sección 3.3 de este capítulo, el concepto de *QoS* en redes IP está estrechamente relacionado con las capacidades de los dispositivos de red de dar un tratamiento diferente a los distintos paquetes que son procesados por este dispositivo. En las redes IP el dispositivo de red que tiene una funcionalidad fundamental es el router, cuya misión principal es la de enrutar los paquetes hasta el destino final en función de la dirección destino del paquete procesado. Para que un router pueda formar parte de una red con *QoS* debe tener una serie de funcionalidades adicionales tales como:

- Clasificar los paquetes recibidos. La acción básica de un router IP es tratar todos los paquetes IP exactamente de la misma forma. Todos los paquetes se procesan en el mismo orden en que llegan y tienen la misma respuesta de servicio. Si se desea que cierto tipo de paquetes tengan un tratamiento diferente, el router debe estar equipado con alguna función de clasificación de paquetes. La clasificación se hace, básicamente, de dos formas:
  - Los paquetes transportan ellos mismos una marca que permite al router clasificarlos.
  - El router tiene una lista de acceso (*access list*) que le indica cómo debe clasificar los paquetes, ya sea en función del protocolo (UDP, TCP, etc.), la dirección IP origen, la dirección IP destino, e incluso filtrando y clasificando mediante los puertos origen y destino del protocolo de transporte del paquete. En este caso, el router, de nivel 3, debe acceder al nivel de transporte para llevar a cabo la clasificación del paquete.
- Disciplinas de colas. Una vez clasificados los paquetes, son encolados para ser posteriormente procesados y retransmitidos por el router. El grado de servicio experimentado por un paquete, dependerá de la política de colas que esté configurada en el router. Por ejemplo, si el router se configura con colas con prioridad, los paquetes de la cola de mayor prioridad serán procesados. Sólo se pasará a procesar los paquetes de otra cola menos prioritaria si las colas de mayor prioridad están vacías. Existen otros mecanismos de encolamiento llamados *class-based queuing* (CBQ) cuya finalidad es que el nivel de servicio aplicado a cada una de las colas sea relativo a su prioridad, sin que ninguna cola llegue a monopolizar todos los recursos del router tal y como pasa con las colas con prioridad.

Estas dos funciones adicionales descritas atienden a un concepto de bajo nivel (clasificación, encolamiento) pero cuyas repercusiones son importantes ya que con los progresivos refinamientos de las disciplinas de encolamiento se incrementa la necesidad de definir la manera en cómo se van a distribuir los recursos compartidos entre los paquetes que compiten por ellos. El objetivo de tales esfuerzos es la definición de políticas de contención y resolución en los routers que determinen la prioridad relativa de los paquetes encolados que están esperando ser servidos y determinar la prioridad relativa de los paquetes que serán descartados cuando las colas se hayan saturado.

En redes IP, la forma de especificar un cierto perfil de tráfico es a través del modelo denominado *token bucket*. En este modelo, existe una cesta (*bucket*) con una cierta capacidad b

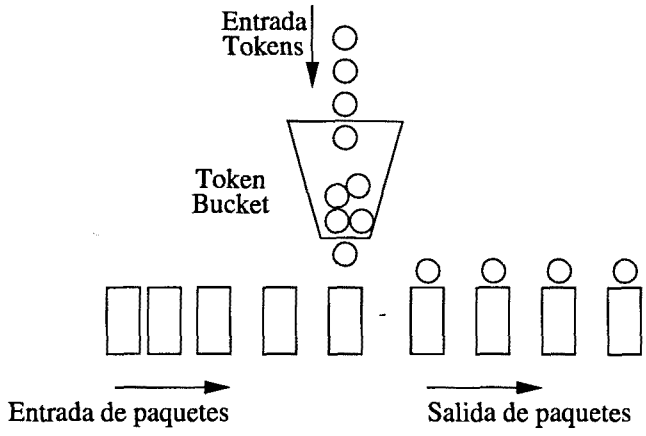


Figura 3.6: Token Bucket

de tokens que se llena a una tasa de r tokens/seg. Si la cesta llega hasta su máxima capacidad, los siguientes tokens de entrada se descartan. Los paquetes entran al mecanismo desde la izquierda y mientras existan tokens en la cesta se podrá extraer paquetes hacia la salida de forma inmediata. Si no hay tokens en la cesta y siguen llegando paquetes pueden ocurrir dos casos:

1. Si existe una cola de entrada, el paquete deberá esperar a que haya suficiente crédito para poder ser transmitido. En este caso el modelo de la *token bucket* se está utilizando como conformador de tráfico

2. Si no existe una cola de entrada, el paquete se descarta y se ha producido una violación, por parte del tráfico entrante, del perfil establecido.

El valor máximo de paquetes a la salida de la token bucket en un cierto instante de tiempo T es de $r \cdot T + b$. El tráfico de salida tendrá una tasa media de r paq/seg y podrá tener un valor máximo de ráfaga de b paquetes.

El IETF contempla dos enfoques básicos que permiten proporcionar calidad de servicio. Uno de los enfoques es crear un estado de reserva para un flujo de datos a través de la red que corresponde a la petición por parte de un servicio y mantener este estado de reserva durante todo el tiempo de aplicación del servicio. Este enfoque es el que se utiliza en los Servicio Integrados. El otro enfoque es la definición estática, en la red, de diferentes clases de servicios con diferentes garantías de *QoS* y el marcado y la clasificación de los paquetes que pertenecen a las diferentes clases de servicios. Este es el enfoque utilizado en los Servicios Diferenciados.

3.5. Servicios Integrados – *IntServ*

La arquitectura denominada Servicios Integrados (*IntServ*) fue diseñada para proporcionar un conjunto de extensiones al mecanismo tradicional de entrega de paquetes en redes IP denominado "best-effort". El objetivo de esta arquitectura es dotar a la red de mecanismos especiales que permitan manejar de forma diferente ciertos tipos de tráfico y proporcionar mecanismos a las aplicaciones para que puedan escoger entre los múltiples servicios de entrega

de paquetes para el tráfico generado. El motivo real que impulsó la definición arquitectura fue el deseo de tener soporte para las transmisiones multimedia en redes IP, tal y como se menciona en [BCS94]:

*... todavía falta un elemento técnico importante: las aplicaciones en tiempo real, a menudo, no funcionan bien en Internet debido a las variaciones de los retardos de las colas y a las pérdidas por congestión. Internet, tal y como se concibió originalmente, únicamente ofrece una calidad de servicio (*QoS*) muy simple, un servicio de entrega extremo a extremo de tipo "best-effort". Antes de que aplicaciones en tiempo real tales como vídeo remoto, videoconferencia, y realidad virtual puedan ser utilizadas ampliamente, la infraestructura de Internet debe ser modificada para soportar calidad de servicio de tiempo real, lo que proporciona cierto control sobre los retardos extremo a extremo de los paquetes.*

El hecho de que la transmisión multimedia sea uno de los factores que han determinado esta arquitectura implica la necesidad de distinguir, dentro de este ámbito de trabajo, entre tráfico elástico y tráfico inelástico, ya que serán las características de estos tipos de tráfico las que van a determinar los parámetros de *QoS* utilizados en esta arquitectura.

El tráfico elástico puede adaptarse, dentro de un amplio margen, a variaciones del retardo y del caudal dentro de la red manteniendo las necesidades mínimas de las aplicaciones que los generan. Este es el tráfico tradicional soportado por redes basadas en TCP/IP. Las aplicaciones que pueden clasificarse como generadoras de tráfico elástico son las clásicas de Internet tales como transferencia de ficheros, correo electrónico, acceso remoto, gestión de red y acceso a servicios de WEB. No todas estas aplicaciones tienen los mismos requerimientos, por ejemplo:

- El correo electrónico es el menos sensible a cambios en el retardo ya que no es una aplicación interactiva.
- La transferencia de ficheros se realiza *on-line*, como es habitual, el usuario espera que el retardo sea proporcional al tamaño del fichero y por lo tanto es sensible a las variaciones del caudal.
- Las aplicaciones interactivas como el acceso remoto o el acceso a servicios WEB son bastante sensibles al retardo

El tráfico inelástico no se adapta fácilmente o en absoluto a las variaciones en el retardo y en el caudal. Los ejemplos principales de tráfico inelástico son la transmisión de voz y de vídeo. Los requerimientos para el tráfico inelástico son :

- El caudal. Es posible que una aplicación requiera de un valor mínimo de caudal.
- El retardo.
- Las variaciones del retardo. La magnitud de la variación del retardo es un factor crítico en aplicaciones en tiempo real. Algunas aplicaciones requieren una cota superior del retardo extremo a extremo experimentado por los paquetes.
- Las pérdidas de paquetes.

Estos requerimientos son difíciles de conseguir en un entorno en el que se producen variaciones del retardo y pérdidas por congestión. El tráfico inelástico introduce nuevos requerimientos

en la arquitectura de Internet. Por una parte, hace falta algún mecanismo que de un trato preferente al tráfico generado por ciertas aplicaciones con una demanda mayor de requerimientos. Las aplicaciones deben ser capaces de determinar qué necesidades tienen para satisfacer un cierto servicio e indicar a la red cuáles son sus requerimientos. Para ello es necesario un protocolo de señalización que permita establecer la reserva de recursos. Por otro lado, en situaciones de congestión y dado que el tráfico inelástico no reduce sus necesidades ante tales eventualidades, es necesario un control de admisión que pueda denegar las nuevas peticiones de servicios.

El modelo de componentes de los servicios integrados hace referencia a las funciones que debe tener implementadas un router capaz de funcionar dentro de esta arquitectura y se

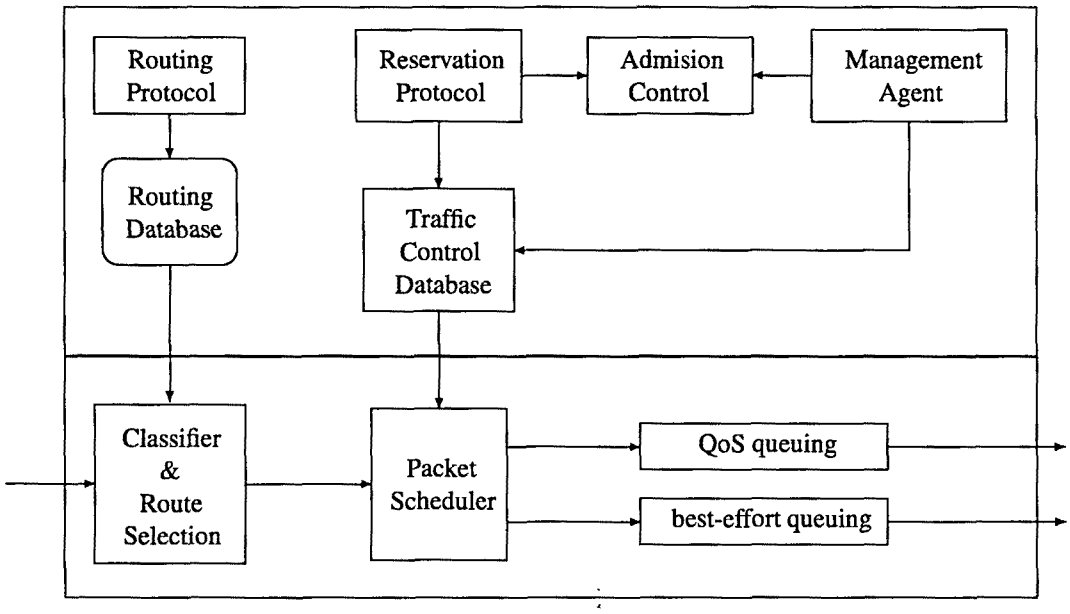


Figura 3.7: Modelo de referencia de IntServ

muestra en la figura 3.7. Por debajo de la línea horizontal gruesa se muestran las funciones de *forwarding* del router; estas funciones se ejecutan por cada paquete procesado y deben estar optimizadas al máximo. Por encima de la línea gruesa se muestran las funciones que manejan el sistema de *forwarding*, denominadas funciones de segundo plano. Las principales funciones de segundo plano son:

- Protocolo de Reserva. Este protocolo es el que se utiliza entre los routers y entre routers y sistemas finales para reservar recursos para un nuevo flujo con un cierto nivel de *QoS*. El protocolo de reserva es responsable de mantener información de estado específica del flujo en los sistemas finales y en los routers que se encuentran a lo largo del camino por el que circularán los paquetes de este flujo. El protocolo de reserva refresca la base de datos de control de tráfico (*traffic control database*) utilizada por el selector de paquetes (*packet scheduler*) para determinar el nivel de servicio proporcionado a los paquetes de cada flujo.

- Control de Admisión. Cuando se hace una petición de reserva para un nuevo flujo, el protocolo de reserva invoca la función de control de admisión. Esta función determina

si existen suficientes recursos para el nivel de servicio demandado por este nuevo flujo. Esta determinación está basada en los recursos actualmente reservados por otros flujos así como por el nivel de carga actual de la red.

- Agente de Gestión. Un agente de gestión de red es capaz de modificar la base de datos de control de tráfico y manejar el módulo de control de admisión para establecer nuevas políticas de control.
- Protocolo de rutado. Es el responsable de mantener la base de datos de ruta que proporciona el siguiente salto en función del destino de un flujo.

Las funciones que realizan la parte de reenvío (*forwarding*) dentro del router son:

- El Clasificador y Selector de Ruta. Para poder llevar a cabo las tareas de reenvío y control de tráfico, es necesario que los paquetes que entran al router sean mapeados en clases. Una clase puede corresponder a un único flujo o a un conjunto de flujos con los mismos requerimientos de *QoS*. Por ejemplo, los paquetes de todos los flujos de vídeo o los paquetes de todos los flujos que pertenecen a una misma organización pueden ser tratados idénticamente. La selección de la clase se basa en campos específicos de los paquetes. Basándose en la clase del paquete y su dirección destino, se determina la dirección del siguiente salto para este paquete.
- El Selector de Paquetes. Esta función gestiona una o más colas por cada interfaz de salida. Determina el orden en el que los paquetes encolados deben ser transmitidos y cuales de ellos deben ser descartados si ello llega a ser necesario. Las decisiones se realizan en base a la clase del paquete, el contenido de la base de datos de control de tráfico y de la actividad pasada y actual del interfaz de salida. Parte de las tareas del selector de paquetes es realizar un control de policía determinando si un cierto flujo esta excediendo los recursos reservados y decidir cómo debe ser tratado el paquete en ese caso.

3.5.1. Clases de servicio de IntServ

El servicio *IntServ* para un flujo de paquetes se define en dos niveles. Primeramente, se definen un número general de categorías (clases) de servicio, las cuales proporcionan ciertas garantías de servicio generales. En segundo lugar y dentro de cada clase de servicio se define el grado de *QoS* para un cierto flujo a través de ciertos parámetros que se denominan TSpec (*Traffic Specification*). Actualmente hay tres categorías definidas:

- Servicio Garantizado (*Guaranteed Service*)
- Carga Controlada (*Controlled Load*)
- Best Effort

Una aplicación puede realizar una petición de reserva para un flujo de un servicio garantizado o de carga controlada y con un TSpec que define la cantidad exacta de servicio que desea recibir. Si la petición es aceptada, entonces el TSpec se convierte en parte del contrato entre el flujo de datos y el servicio. El servicio se compromete a proporcionar la *QoS* requerida por el flujo siempre y cuando el tráfico del flujo de datos no viole el TSpec. Los paquetes que no forman parte de un flujo con reserva son servidos por la clase de Best-Effort por defecto.

La especificación del tráfico (TSpec) se compone de los siguientes parámetros:

- r : Tasa media en unidades de bytes de datagrama IP por segundo
- b : La profundidad del *bucket* en unidades de bytes
- p : La tasa de pico del tráfico en unidades de bytes de datagrama IP por segundo
- m : El tamaño mínimo del paquete que será aceptado, medido en bytes
- M : El tamaño máximo de paquete que será aceptado, medido en bytes

La definición de m (*Minimum Policed Unit*) se realiza para que se pueda obtener una estimación razonable de los recursos por paquete necesarios para procesar un flujo. La tasa máxima de transmisión de un paquete puede calcularse en base a m y b, dividiendo b por m. El tamaño de los paquetes incluyen los datos de las aplicaciones así como las cabeceras de los protocolos asociados desde el nivel IP hacia arriba. No se incluye el nivel de enlace que puede variar a medida que viajan los paquetes a través de la red. En la figura 3.8 se muestra el modelo que

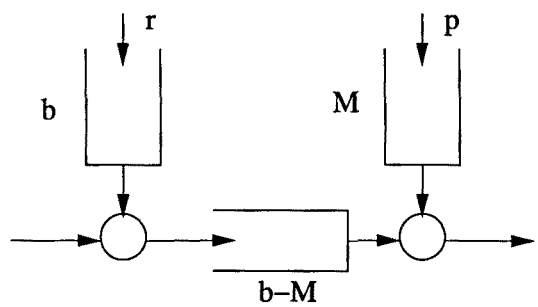


Figura 3.8: Especificación *Token-Bucket* del IETF

utiliza el IETF para definir los parámetros descritos.

3.5.1.1. Servicio Garantizado

Los elementos que definen un servicio garantizado son los siguientes:

1. El servicio proporciona una tasa asegurada.
2. Existe una cota superior del retardo a través de la red. Esta cota superior no contempla retardos por propagación.
3. No hay pérdidas en las colas. Es decir, ningún paquete se puede perder debido a la saturación de buffers. Las pérdidas de paquetes pueden ser debidas a fallos en la red o cambios en los caminos de ruta.

El servicio garantizado, descrito en [SPG97], está pensado para aplicaciones en tiempo real sensibles al retardo tales como audio interactivo de alta calidad y transmisión de vídeo. Este servicio no intenta acotar el *jitter* sino que acota el retardo máximo de las colas, extremo a extremo, a lo largo de la ruta que seguirán los paquetes. Matemáticamente se establece que el retardo por encolamiento es debido a dos factores: la profundidad del *bucket* b y la tasa de

transmisión r, del TSpec. El retardo extremo a extremo máximo puede calcularse utilizando el modelo de fluidos aplicado a la figura 3.8 y son (ver apéndice A):

$$\begin{aligned} Q_d &= \frac{(b-M)\cdot(p-R)}{(p-r)\cdot R} + \frac{M}{R} + \frac{C_{tot}}{R} + D_{tot} \qquad p > R \geq r \\ Q_d &= \frac{M}{R} + \frac{C_{tot}}{R} + D_{tot} \qquad R \geq p \geq r \end{aligned}$$

siendo R la tasa de salida del modelo. Los parámetros C_{tot} y D_{tot} son factores de corrección para ajustar la desviación del modelo ideal de fluidos al sistema real. C_{tot} representa el retardo que podría experimentar un paquete a causa de parámetros relacionados con la tasa de transmisión, como por ejemplo mecanismos de serialización; las unidades de este parámetro son de bytes. D_{tot} representa el retardo, independiente de la tasa de transmisión, impuesto por el tiempo que el paquete necesita esperar antes de ser transmitido como por ejemplo si se debe refrescar la memoria de ruta para poder enviar el paquete. Cada uno de los routers por los que circulará un flujo, se apartará del modelo ideal de fluidos añadiendo un valor C_i y D_i que modifican el retardo experimentado por un paquete en cada nodo. C_{tot} y D_{tot} se obtienen como suma de todos los C_i y D_i de los routers por los que circularán los paquetes entre los dos extremos. En la especificación del servicio garantizado no se establece la manera en que los extremos pueden llegar a tener conocimiento de la acumulación total de estos parámetros sino que se supone que el protocolo de reserva será utilizado para este menester.

El servicio garantizado es invocado por el transmisor enviando, a través del protocolo de reserva, un descriptor de tráfico (TSpec) con las características del flujo a transmitir. El receptor, una vez ha recibido el TSpec y desea obtener ese servicio, envía un mensaje con el nivel de servicio requerido, el RSpec. El RSpec incluye una tasa de transmisión R, y un término de ajuste S denominado *Slack Term.* La tasa requerida por el receptor, R, debe ser mayor o igual que la establecida en el TSpec, es decir $R \geq r$. De esta forma se consiguen menores retardos. El parámetro de ajuste S representa la diferencia entre el retardo deseado y el retardo obtenido por el uso de una tasa de transmisión R y no puede ser negativo. Este parámetro puede ser utilizado por los dispositivos de la red para relajar los niveles de reserva de un flujo.

En [SPG97] se establece el comportamiento de un servicio garantizado con estas palabras:

La definición del servicio garantizado recae sobre el hecho de que el retardo de un fluido obedeciendo un modelo token bucket *(r,b) y siendo servido por una línea con ancho de banda R está acotado por b/R siempre y cuando R no sea menor que r. El servicio garantizado con una tasa de servicio R, donde ahora R es una parte compartida del ancho de banda en lugar del ancho de banda de una línea dedicada, se aproxima a este comportamiento.*

Según este comportamiento, el receptor de un descriptor de tráfico TSpec puede calcular el parámetro de ajuste S, teniendo en cuenta que, establecida la tasa del servicio R, el retardo máximo que se va a obtener es

$$\frac{b}{R} + \frac{C_{tot}}{R} + D_{tot}$$

Si el retardo deseado es D_{req}, el parámetro de ajuste S se obtiene de:

$$S = D_{req} - \left(\frac{b}{R} + \frac{C_{tot}}{R} + D_{tot}\right)$$

El control de admisión en servicios garantizados es estricto debido a las requerimientos que debe proporcionar este servicio. Así, si un router tiene asignada una tasa de salida C_{max} para

el conjunto de servicios garantizados, debe ocurrir que la suma de las tasas individuales de todos los servicios garantizados que pueda estar sirviendo ese router debe ser menor o igual a C_{max}. Matemáticamente, si se están sirviendo N servicios garantizados a una tasa C_i, entonces

$$C = \sum_{i=1}^{N} C_i \leq C_{max}$$

Si llega una nueva petición de servicio garantizado con una tasa C_k, el servicio no será aceptado si

$$C + C_k > C_{max}$$

Obviamente, no únicamente se contrasta la tasa total para realizar un control de admisión, también deben existir suficiente memoria para alojar el tamaño del bucket, etc.

3.5.1.2. Carga Controlada

Los elementos principales que definen un servicio de carga controlada se describen en [Wro97] y son los siguientes:

1. El servicio de carga controlada se asemeja al comportamiento visible por las aplicaciones recibiendo un servicio best-effort en condiciones de carga baja.

2. No existe una cota superior del retardo en colas a través de la red. Sin embargo, el servicio asegura que un porcentaje muy alto de paquetes no van a experimentar retardos que excedan de forma sensible el tiempo mínimo de tránsito (es decir, el retardo debido a la propagación más el tiempo de procesado por los routers sin que exista retardos en las colas).

3. Un porcentaje muy alto de paquetes transmitidos serán entregados satisfactoriamente, es decir que casi no habrán pérdidas en las colas.

Para poder asegurar estas características, las aplicaciones que utilicen este tipo de servicio deben proporcionar a la red una estimación del tráfico a transmitir mediante los descriptores de tráfico TSpec. A su vez, cada nodo que gestiona las peticiones de un servicio de carga controlada asegura que existirán suficientes recursos para acomodar las necesidades de reserva de la petición. El grado de exactitud con el que el descriptor de tráfico TSpec se ajusta a los recursos reales de la red no tiene porque ser demasiado estricto. Si los recursos requeridos caen fuera de los límites de los recursos asequibles, el tráfico de ese flujo puede experimentar cierto incremento del retardo o bien un nivel ligeramente mayor de paquetes descartados. Sin embargo, las variaciones del retardo o del nivel de descarte de paquetes deberían ser lo suficientemente pequeñas para que las aplicaciones adaptativas en tiempo real puedan seguir funcionando sin una degradación notable.

El control de admisión para servicios de carga controlada puede prever la posibilidad de multiplexación estadística debido a esas variaciones aceptadas en el servicio.

3.5.2. *Resource Reservation Protocol* (RSVP)

Es indudable que la especificación de servicios integrados se basa en la existencia de un protocolo de reserva y señalización y aunque no especifica cuál debe ser este protocolo, en redes IP el que más éxito ha tenido no es otro que RSVP (*Resource Reservation Protocol*) [BZB+97].

Mediante este protocolo, las aplicaciones pueden notificar a los distintos dispositivos de red sus necesidades de servicio y sus descriptores de tráfico, y los dispositivos de red lo utilizan para intercambiar información relativa a la *QoS* entre ellos.

Existe una separación lógica entre el control de la *QoS* en *IntServ* y RSVP. RSVP, como protocolo de señalización de mantenimiento de ruta, puede ser utilizado para soportar una amplia variedad de servicios, y a su vez, el control de la *QoS* en *IntServ* ha sido diseñado para ser gestionado, de forma transparente, por otros protocolos de control y señalización.

RSVP es meramente un protocolo de señalización, es decir, RSVP no define el formato interno de los objetos del protocolo relacionados con la caracterización de *QoS*, simplemente los trata como objetos opacos que debe transportar entre distintos nodos. Si RSVP es un protocolo de señalización, la información de la caracterización de *QoS* es el contenido de la señal. RSVP tampoco es un protocolo de ruta, sino que necesita de protocolos de rutado ya establecidos y es capaz de interoperar con ellos, tanto con protocolos de rutado *unicast* como *multicast*.

En términos generales, RSVP se utiliza para transportar, a todos los nodos intermedios entre dos extremos por los que fluirán los datos, los requerimientos de *QoS* y mantener el estado de reserva en cada uno de los nodos durante todo el tiempo del servicio. Para ello, RSVP establece y mantiene estados "blandos" (*soft states*) en cada uno de los nodos a lo largo de la ruta de la comunicación. Un *soft state* es un estado que se mantiene a través de envíos periódicos de mensajes de refresco. En ausencia de estos mensajes periódicos y al cabo de un cierto tiempo de temporización, el estado de reserva expira. Los *soft states* son apropiados en redes IP donde las rutas entre extremos pueden cambiar durante una comunicación. De esta forma los estados de reserva no están asociados a una ruta específica.

RSVP también ofrece una *QoS* dinámica ya que es capaz de señalizar cambios en las reservas establecidas durante la sesión.

Método de operación de RSVP

RSVP fue originalmente diseñado con las aplicaciones *multicast* en mente. Por ello, RSVP tiene un enfoque denominado "orientado a receptor". Es decir, es el receptor de los datos de una aplicación servidora el responsable de pedir unos requerimientos específicos de *QoS*.

Una aplicación servidora envía mensajes RSVP indicando las características de tráfico que desea servir (TSpec) hacia el receptor. Estos mensajes se denominan mensajes de PATH. Cada uno de los nodos intermedios a lo largo de la ruta entre el servidor y el receptor utiliza la información de los mensajes de PATH para mantener un estado interno de PATH, incluyendo la información de los descriptores de tráfico. Cuando el RSVP receptor recibe el mensaje de PATH, envía un mensaje de reserva (RESV) hacia el servidor a través de la misma ruta, en sentido inverso, que ha seguido el mensaje de PATH. En el mensaje de reserva, el receptor indica cual es el nivel de *QoS* requerido. A medida que este mensaje de reserva pasa por los nodos intermedios, éstos pasan a un estado de reserva ubicando los recursos necesarios para la transmisión que va a empezar. Si algún nodo intermedio no tuviera suficientes recursos, se encargaría a través de RSVP de indicar este hecho al servidor y receptor. Cuando el mensaje de reserva llega al servidor, este inicia la transmisión. Los nodos intermedios (routers) utilizan las direcciones IP de origen y destino, el protocolo de transporte y los puertos origen y destino, obtenidos a través de los mensajes de reserva, para establecer los filtros que determinarán si un paquete pertenece a un cierto flujo con reserva o no.

De este modo de operación, se deduce que RSVP es un protocolo simplex. La reserva de recursos se realiza solamente en una dirección, del servidor al receptor.

3.6. Servicios Diferenciados – *DiffServ*

La arquitectura de servicios integrados presenta una serie de problemas:

- La cantidad de información de los estados se incrementa proporcionalmente con el número de flujos. Esto obliga a los routers a tener una gran capacidad de almacenamiento y una sobrecarga de procesado que impide que esta arquitectura escale bien en el centro de Internet (*Internet Core*)

- Los requerimientos de los routers son altos ya que deben soportar RSVP, control de admisión, clasificación de paquetes en base a direcciones IP y puertos de protocolos de transporte – a este tipo de clasificadores se le conoce con el nombre de clasificadores MF (*Multifield*) y selectores de paquetes con sistemas avanzados de selección (CBQ, WFQ, ...)

- La posibilidad de ofrecer un servicio garantizado extremo a extremo en Internet obliga a que todos los routers de Internet deban soportar necesariamente este tipo de servicio por lo que no existe la posibilidad de establecer de forma progresiva este servicio en Internet si no se cambian todos los routers que no sean capaces de soportar este servicio
  El servicio de carga controlada se puede establecer de forma incremental en Internet introduciendo routers que soporten este tipo de servicio en aquellos puntos que experimenten un elevado grado de congestión.

A mediados de 1997 y acabada la especificación de RSVP, varios proveedores de servicios de Internet (*Internet Service Provider*, ISP) expresaron al IETF sus dificultades en implementar en sus redes, de forma efectiva, servicios con *QoS* basándose en el modelo de Servicios Integrados debido a las razones expuestas. Como respuesta a estos problemas de escalabilidad el IETF empezó a trabajar en una arquitectura diferente cuyo objetivo principal fue la de proporcionar servicios con *QoS*, que fuese fácilmente implementable y que escalase bien en Internet. Para ello se basaron en dos publicaciones recientes ([CW97], [NJZ97]) en las que se establece el entorno sobre el cual ofrecer una calidad de servicio diferenciada basándose en un proceso de marcaje individual de los paquetes IP para determinar el nivel de prioridad relativa ofrecido a cada uno de los paquetes.

El marco de trabajo fundamental de la arquitectura de servicios integrados se describe en [NBBB98, BBC+98]. Las propiedades de escalabilidad de la arquitectura DiffServ se consigue a través del marcaje de la cabecera de cada paquete con un código estandarizado. Todos los paquetes que contienen un mismo código reciben un tratamiento idéntico al atravesar los distintos routers a lo largo de una ruta. De esta forma se evita la necesidad de estados o de decisiones complejas de reenvío de paquetes en los routers centrales como era el caso del modelo IntServ. La posibilidad de tener diferentes códigos lleva a la posibilidad de crear clases de servicios diferentes y por ello, el modelo DiffServ es esencialmente un esquema de prioridad relativa.

El objetivo de la arquitectura DiffServ es sencillo: Definir métodos simples y efectivos de proporcionar clases de servicio diferenciadas para el tráfico de Internet. No se hace ningún

intento de que las redes DiffServ respondan a sesiones individuales, en lugar de ello, la arquitectura DiffServ se diseña para ofrecer los servicios a clases agregadas, donde un número de flujos de sesiones individuales se agrupan y son tratados consistentemente por la red.

Para que un cliente pueda recibir un servicio de los definidos en DiffServ por parte de un ISP, primero debe obtener un acuerdo de nivel de servicio (*Service Level Agreement*, SLA) con su ISP. Un SLA especifica, básicamente, las clases de servicio soportadas y la cantidad permitida de tráfico en cada clase. Un SLA puede ser estático o dinámico. Los SLA estáticos se negocian en base a unos periodos regulares (p.e mensuales o anuales). Los clientes que desean obtener un SLA dinámico deben utilizar un protocolo de señalización (p.e RSVP) para requerir los servicios demandados.

Los clientes marcan cada paquete individualmente para indicar el servicio deseado. Cuando los paquetes entran en la red del ISP, se clasifican y se realiza un control de policía y, posiblemente, se realiza una conformación de tráfico para que reuna las condiciones de policía asociadas a esa clasificación. La clasificación y el control de policía que se utilizan en los puntos de acceso a la red se derivan de estos acuerdos de nivel de servicio (SLA), así como la cantidad de espacio de almacenamiento necesario para estas operaciones.

Un concepto importante dentro de la arquitectura DiffServ es el de dominio. Un dominio DiffServ corresponde a un conjunto de routers que ofrecen un mismo servicio a cada uno de los códigos de marcaje de paquetes. Es decir, el comportamiento por router (*Per Hop Behaviour*) es el mismo para el mismo código. El modelo de DiffServ no especifica una relación explícita entre código y PHB y deja el establecimiento de esta relación dentro de un dominio al administrador y gestor de la red. Esto implica que el campo de marcado de los paquetes no es transparente extremo a extremo ya que cuando un paquete cambia de dominio, la nueva red puede modificar el valor de marcado para poder seguir ofreciendo los servicios requeridos por el cliente o bien en función de sus políticas de gestión de recursos.

3.6.1. Elementos de red de DiffServ

Los elementos de red de una arquitectura DiffServ son los siguientes:

- El campo de marcado DS
- El clasificador de tráfico
- El medidor de perfil de tráfico
- El marcador de códigos
- El Router DiffServ

El campo de marcado DS (DiffServ)

El campo de la cabecera de los paquetes IP utilizado en la arquitectura DiffServ es el campo *Type of Service* (TOS) de la cabecera de los paquetes IPv4 o el campo *Traffic Class* de la cabecera de los paquetes IPv6. El valor de este campo es interpretado por los routers DiffServ dentro de la red para indicar un tratamiento particular de la función de reenvío en cada uno de los nodos de la red a lo largo de la ruta que sigue el paquete. Dentro de la arquitectura DiffServ, a este campo se le conoce con el nombre de campo DS, donde DS hace referencia al papel de indicador que juega dentro de una red DiffServ. De los 8 bits que tiene este campo, los primeros 6 bits contienen el código de Servicios Diferenciados (*Differentiated Services*

Codepoint, DSCP), el cual se traducirá al PHB que recibirá el paquete en ese nodo. El mapeo entre el DSCP y el PHB no es necesariamente uno a uno. Los dos últimos bits, denominados CU (*Currently Unused*) no son de aplicación dentro de esta arquitectura y deben ser ignorados por los dispositivos de red DiffServ. La estructura del campo DS es incompatible con la definición existente en IPv4 del campo TOS; aunque la función es similar, la semántica y la asignación de los bits dentro del campo son bastante diferentes.

Es importante destacar dos cambios críticos en la definición del campo DS: en primer lugar, mientras que el resto de campos de la cabecera de un paquete IP tienen una semántica universal, el campo DS tiene valores que son locales a cada dominio DiffServ. En segundo lugar y como se ha comentado anteriormente, este campo no es transparente ni simétrico extremo a extremo.

El clasificador de tráfico

Todo el tráfico que entra a una red DiffServ pasa a través de un clasificador de tráfico. El clasificador selecciona los paquetes basándose en la política de control de admisión y en los campos de la cabecera de los paquetes. La arquitectura DiffServ define dos formas de clasificación:

- Una basada exclusivamente en el campo DS y conocida con el nombre de clasificador de comportamiento agregado (*Aggregated Behaviour Classifier*).
- Otra basada en unas condiciones de clasificación mas generales. Estas condiciones pueden incluir cualquier campo de las cabeceras de IP, TCP o UDP, tales como direcciones origen o destino o puertos origen o destino, el valor DSCP, el tipo de protocolo, etc. A este tipo de clasificador se le conoce con el nombre de MF (*Multifield*).

Es importante señalar que en esta arquitectura existe el mismo problema que en el modelo IntServ sobre la fragmentación IP de los paquetes. Dentro de los fragmentos de un paquete IP, no existe suficiente información en la cabecera que permita que un paquete sea correctamente clasificado. Esto implica que una aplicación debe utilizar algún método para descubrir el tamaño máximo de transmisión (MTU) entre su extremo y el extremo remoto para evitar la fragmentación de los paquetes.

Los paquetes clasificados se pasan al medidor de perfil de tráfico.

El medidor de perfil de tráfico

Un medidor de perfil de tráfico DiffServ examina cada flujo de tráfico agregado y determina si el paquete está dentro o fuera del perfil asociado por el SLA. Por ejemplo, un perfil de tráfico en un nodo de acceso a una red DiffServ puede estar basado en un modelo *Token Bucket* y clasificar los paquetes con un valor específico de DSCP en función de una tasa de ráfaga específica:

```
Si DSCP=XXX, usar TB(r,b)
```

Este ejemplo sencillo de perfil indica que todos los paquetes con valor DSCP=XXX son evaluados respecto a un modelo de *token bucket* con tasa r y tamaño del bucket b. Los paquetes que excedan estos valores serán considerados que están fuera del perfil y los paquetes que se ajusten serán considerados que están dentro del perfil.

También es necesario definir cuáles son las acciones que se deben llevar a cabo cuando un paquete ha sido considerado que está fuera del perfil. Estos paquetes pueden ser marcados de nuevo con una prioridad menor o bien se puede indicar al conformador de tráfico que retarde la expedición del paquete para que se ajuste al perfil o bien, pueden ser descartados.

El marcador de códigos

El marcador de paquetes obtiene el paquete del clasificador e influido por el medidor de perfil de tráfico, verifica o marca el paquete con el valor DSCP adecuado en el campo DS de la cabecera.

El Router DiffServ

El modelo general de un router DiffServ capaz de realizar las tareas de clasificación y acondicionamiento de tráfico se muestra en la figura 3.9. Cuando un paquete entra en la red, el primer router que procesa el paquete fuerza el SLA y marca los paquetes de una manera consistente con el PHB deseado. Hay tres tipos de routers genéricos en una arquitectura DiffServ. El nodo de acceso (*Ingress Node*) es responsable de las funciones de acondicionamiento de tráfico y marcado de paquetes. Los nodos interiores son responsables de la selección del PHB basándose en el valor DSCP de los paquetes. Los nodos de salida (*Egress Node*) pueden ser responsables de cambiar el marcado de los paquetes y realizar funciones de acondicionamiento de tráfico basándose en la políticas de salida e identidad del siguiente dominio al cual se reenvía el paquete .

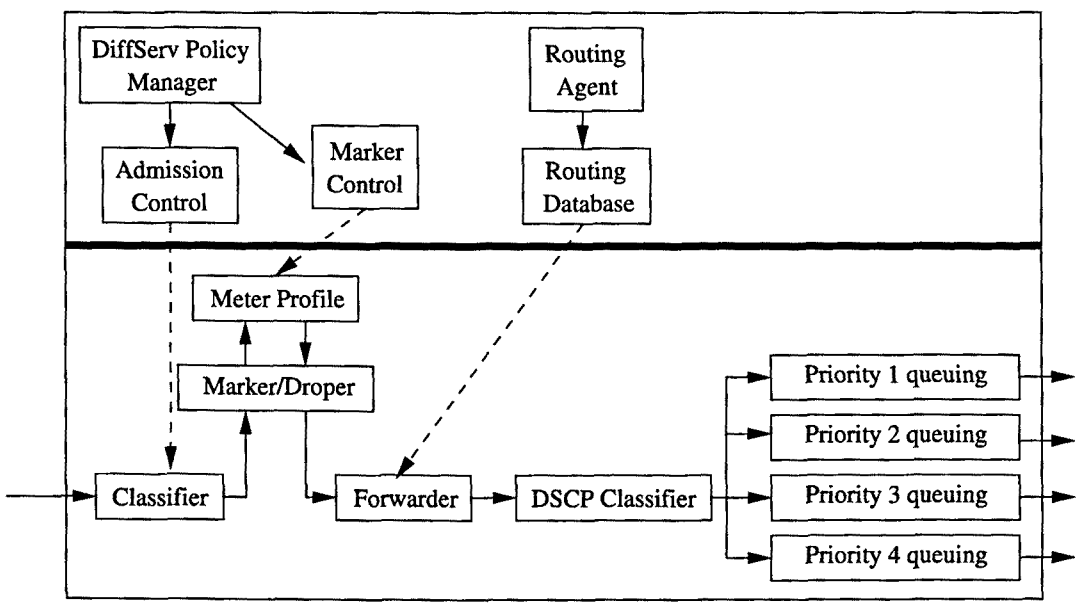


Figura 3.9: Modelo general de un router DiffServ

3.6.2. PHB estandarizados

La IETF define tres grupos de comportamientos por salto (PHB):

Grupo Class Selector PHB

La arquitectura DiffServ intenta mantener cierta compatibilidad con la funcionalidad del campo TOS de la cabecera IP. La especificación DiffServ recomienda que el valor DSCP 000000b sea mapeado a un servicio de tipo best-effort.

Además, la especificación DiffServ intenta preservar la semántica de los bits de precedencia[1] del campo TOS. Es decir, cualquier DSCP que sea de la forma xxx000 (donde x puede ser 0 ó 1) está reservado. A estos DSCP se les conoce normalmente con el nombre de DSCP selectores de clase.

Expedited Forwarding (EF)

En este tipo de PHB intenta proporcionar un servicio extremo a extremo de bajas pérdidas, baja latencia, variaciones de retardo bajas y ancho de banda asegurado. A este tipo de servicio se le conoce con el nombre de *Premium*. Se supone que este servicio será utilizado por aplicaciones de voz y de vídeo interactivas.

Para conseguir estas características, la cantidad de tráfico agregado que entra en la red no puede ser superior a la cantidad de ancho de banda reservado para el servicio EF. Esto implica que el control de policía para este tipo de tráfico debe ser muy estricto y los paquetes que no cumplan el perfil de tráfico de este tipo de servicio serán descartados o marcados con una prioridad mucho menor (p.e. *best-effort*).

Assured Forwarding (AF)

Assured Forwarding es una colección de PHB que ofrecen un alto nivel de seguridad en la entrega de cada paquete siempre y cuando el tráfico se atenga a las especificaciones establecidas en el SLA. Aunque una fuente de tráfico pueda exceder su SLA, y la red acepte este exceso de tráfico, el tráfico excedido puede ser descartado con mayor probabilidad. Sin embargo, si no se descarta y se entrega al extremo receptor, todos los paquetes, incluyendo los paquetes con tasa excedida, llegarán ordenados.

Se definen cuatro clases AF, donde a cada clase AF se le asignan cierta cantidad de recursos (ancho de banda y espacio de almacenamiento) en cada nodo DiffServ. Este tipo de PHB permite a los ISPs proporcionar un tipo de servicio denominado genéricamente *Olympic* en el que se definen diversos niveles de servicio: oro, plata y bronce, donde cada uno recibe distintos niveles de ancho de banda, espacio de almacenamiento y política de descarte.

La idea del PHB AF es la minimización de los eventos de congestión local de larga duración mientras que permite ráfagas de tráfico de corta duración. El tipo de clientes que se espera que utilicen este tipo de servicio son las aplicaciones de *streaming* y de datos interactivos.

3.7. RTP y RTCP

Como se ha ido viendo a lo largo de este capítulo, la transmisión multimedia sobre redes IP utiliza el protocolo de transporte UDP. En general, las diferencias entre TCP y UDP que afectan a la transmisión multimedia mediante *"streaming"* son:

- TCP opera sobre el flujo de bytes mientras que UDP está orientado al paquete

[1]Estos bits son los 3 primeros bits del campo TOS de la cabecera IP

- TCP garantiza la entrega a través de retransmisiones, pero a causa de ello el retardo no está acotado. UDP no garantiza la entrega, pero el retardo de los paquetes entregados es más predecible y menor.

- TCP proporciona control de flujo y control de congestión. UDP no proporciona ningún mecanismo para ello. Esto proporciona mayor flexibilidad para que las aplicaciones determinen los procedimientos apropiados de control de flujo y congestión.

Estas diferencias llevan a que, en general, sea UDP el protocolo utilizado en transmisiones multimedia. Sin embargo UDP únicamente proporciona unos servicios muy básicos, que incluyen un *checksum* para verificar paquetes erróneos y el direccionamiento de puerto para la demultiplexación del tráfico recibido.

El IETF diseñó dos protocolos para la transmisión de información multimedia en *streaming*: *Real Time Protocol* (RTP) y *Real Time Control Protocol* (RTCP). RTP se utiliza para la transferencia de datos, mientras que RTCP se utiliza para los mensajes de control. Ninguno de estos dos protocolos proporcionan servicios en tiempo real, sino que debe ser la red la que disponga de estos servicios, sin embargo, RTP y RTCP proporcionan las funcionalidades para soportar servicios en tiempo real. RTP no garantiza QoS ni un servicio de entrega fiable, pero implementa los mecanismos necesarios que necesitan las aplicaciones que tienen restricciones temporales, proporcionando funcionalidades genéricas para este tipo de tráfico como marcas temporales, numeración de secuencia y especificación del tipo de datos transportados. RTP permite la detección de pérdidas de paquetes. RTCP proporciona un mecanismo de realimentación, con mensajes que permiten determinar la calidad de los datos entregados. Proporciona información al transmisor sobre la QoS en términos del número de paquetes perdidos, la variación del retardo de los paquetes, el retardo, etc. RTCP especifica la periodicidad de los paquetes de realimentación, de forma que no se gaste más de un 5 % del ancho de banda total de la sesión y al menos se envía un paquete cada 5 segundos. El transmisor puede utilizar los mensajes de RTCP recibidos para adaptarse a las condiciones de la red, por ejemplo ajustando la tasa de transmisión. El enfoque convencional para la transmisión de multimedia en *streaming* es utilizar RTP/UDP para la transmisión de los datos multimedia y RTCP/TCP o RTCP/UDP para el control.

3.8. Arquitectura de un sistema de transmisión de *Video Streaming*

La arquitectura básica de un sistema de *video streaming* no varía sustancialmente de la representada en la figura 3.10. En el servidor se almacena la información de vídeo y audio una vez se ha codificado y comprimido. Cuando el cliente realiza una petición para recibir el video y audio comprimido, el módulo de control de QoS adapta los flujos de audio y vídeo según las características de la red y el nivel de QoS ofrecido o bien, negocia con la red los recursos necesarios para la transmisión.

Las características de la red, así como los servicios ofrecidos por ésta, determinan las diferentes estratégias en la transmisión de vídeo con el fin de intentar mantener unos niveles mínimos de calidad en el receptor. Si la red únicamente proporciona un servicio de entrega best-effort, se utilizan las técnicas descritas en la sección 3.2 de este capítulo. En este caso, se intenta minimizar el impacto que tienen las variaciones del ancho de banda, las variaciones del retardo y las pérdidas de paquetes sobre la transmisión del vídeo. Las técnicas utilizadas

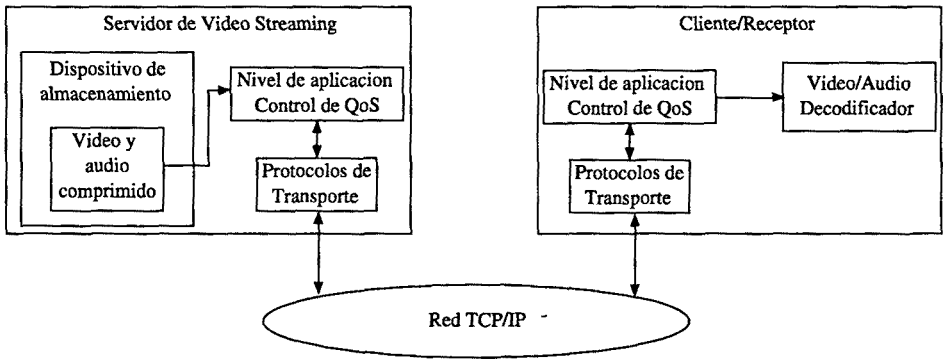


Figura 3.10: Arquitectura básica de un sistema de video streaming

son el uso de buffers en recepción para minimizar el efecto de las variaciones del retardo, el control de la tasa de transmisión para adaptarse al ancho de banda y minimizar las pérdidas de paquetes y el uso de protocolos (RTP/RTCP) que permitan determinar el estado de la transmisión y los instantes de congestión para adaptarse a estas condiciones.

Si la red proporciona servicios de QoS, entonces es posible combinar los diferentes tipos de servicios de QoS proporcionados por la red con las diferentes técnicas de codificación escalable de vídeo para obtener una estrategia de transmisión que permita mantener unos niveles de calidad mínimos en recepción.

3.8.1. Estrategias de transmisión de vídeo en IntServ y DiffServ

En una arquitectura de red de servicios integrados (IntServ) existen dos clases de servicios diferentes con QoS, carga controlada y servicio garantizado. El servicio de carga controlada es equivalente a un best-effort en condiciones de baja carga y aunque no asegura una cota superior en el retardo de los paquetes ni en el nivel de pérdidas de paquetes en las colas, se espera que los paquetes que puedan experimentar este tipo de sucesos sea mínimo. El servicio garantizado proporciona una tasa asegurada, una cota superior en el retardo de los paquetes y asegura un nivel de pérdidas de paquetes nulo en las colas. Este servicio es el más costoso desde el punto de vista de recursos de la red y no permite multiplexación estadística de los flujos transmitidos.

Dados estos dos tipos de servicios con QoS en esta arquitectura de red, nos podemos plantear de cuántas formas se puede transmitir un flujo de vídeo teniendo en cuenta los diferentes modos de codificación de vídeo (VBR o CBR) y la posibilidad de la codificación escalable. A continuación se exponen varias estrategias de transmisión de vídeo sobre una arquitectura IntServ, que sin ser todas las posibles, si son las más representativas.

1. Transmisión de vídeo CBR sobre un servicio de carga controlada. Esta estrategia es la menos costosa en cuanto a recursos de red, sin embargo el servicio de carga controlada no asegura una cota máxima en el retardo de los paquetes y no permite relajar las condiciones del tamaño del buffer de recepción. Por otro lado, la transmisión CBR no proporciona una calidad de imagen constante y, por lo tanto, para obtener un nivel de calidad mínima aceptable es necesario consumir un ancho de banda elevado.

2. Transmisión de vídeo VBR sobre un servicio de carga controlada. En este caso, y debido a la gran variación de la señal de vídeo codificada, el buffer en recepción deberá ser grande ya que como se ha comentado anteriormente, el servicio de carga controlada no asegura una cota máxima en el retardo de los paquetes. Por otro lado, esta estrategia presenta la ventaja del uso de multiplexación estadística que permite maximizar la explotación de los recursos. Sin embargo y debido a que la probabilidad de pérdida de paquetes no es nula, no se puede garantizar un valor mínimo de la calidad de la imagen recibida.

3. Transmisión de vídeo CBR sobre un servicio garantizado. Esta estrategia adolece de que la calidad de vídeo no es constante, aunque en este caso, los receptores no necesitan disponer de un buffer grande ya que por un lado, la codificación CBR proporciona un flujo de datos con un nivel de ráfagas mínimo y por otro lado, el retardo de los paquetes está acotado por el servicio garantizado.

4. Transmisión de vídeo VBR sobre un servicio garantizado. Con esta estrategia no se explotan las ventajas de la multiplexación estadística y requiere de un exceso de asignación de recursos para garantizar el funcionamiento en las condiciones más activas de la transmisión.

5. Transmisión de vídeo escalable con un flujo base CBR sobre un servicio garantizado y un flujo mejorado VBR sobre un servicio de carga controlada. Un esquema de codificación escalable de vídeo codifica y comprime el vídeo en varios flujos. Uno de esos flujos es el flujo base, el cual puede ser decodificado independientemente de los demás y proporciona una calidad visual mínima seleccionable. Los otros flujos se denominan flujos mejorados y deben ser decodificados conjuntamente con el flujo base dando lugar a una mayor calidad visual. Esta estrategia de transmisión permite que el usuario final reciba un nivel mínimo y asegurado de calidad, establecido por el flujo base y por el servicio garantizado y, en función de las condiciones de transmisión o bien en función de la capacidad del receptor, decodificar los flujos mejorados con objeto de obtener un nivel de calidad visual superior. Además, este tipo de transmisión presenta diversas ventajas. Por un lado, permite reducir el buffer utilizado en el servicio de carga controlada respecto a una transmisión VBR sobre un servicio de carga controlada debido a la codificación escalable. Por otro lado garantiza una calidad mínima durante toda la transmisión y finalmente, reduce el riesgo de degradación de la imagen.

Las mismas estrategias de transmisión de vídeo que se han comentado se pueden aplicar en redes con arquitectura DiffServ, utilizando *Expedited Forwarding* y las diferentes clases de *Assured Forwarding*. Por ejemplo, se puede utilizar una estrategia de transmisión de vídeo escalable transmitiendo el flujo base CBR sobre *Expedited Forwarding* y los flujos mejorados VBR sobre distintas clases de *Assured Forwarding*.

CAPÍTULO 4

Modelado del tráfico de vídeo a nivel de GoP

Este capítulo se centra en la caracterización y modelado de tráfico de vídeo de tasa variable. El tráfico de vídeo presenta unas características de efecto de rafagueo debido a la estructura de codificación del algoritmo MPEG que afectan negativamente a la ganancia de multiplexación estadística en redes con QoS. En este trabajo, se ha supuesto que el tráfico entregado a la red ha sido previamente suavizado y el modelado del tráfico se lleva a cabo considerando que la tasa generada se produce en intervalos de duración de un GoP. En el presente capítulo se expone y se analiza el modelo MMFP bidimensional, proponiendo una metodología de ajuste que permite una automatización del proceso de modelado y caracterización del tráfico de vídeo. Esta metodología de ajuste ha sido aplicada sobre un conjunto extenso de secuencias de vídeo verificando que la estructura bidimensional del modelo MMFP se adapta adecuadamente en todos los casos.

4.1. Introducción

Se entiende por modelo de tráfico a una abstracción matemática que trata de representar una o varias características estadísticas de un tipo de tráfico real o de un flujo concreto en particular. En general, los motivos que llevan al desarrollo del modelo de un sistema suelen ser dos:

- La explicación formal de aspectos del propio sistema. En otras palabras, el modelado del sistema con objeto de entender y profundizar en los mecanismos de funcionamiento del sistema.
- La imitación de dichos aspectos, realizada con el propósito de tomar una decisión acerca del sistema objeto del modelo. En otras palabras, obtener salidas sintetizadas mediante el modelo cuyos parámetros estadísticos y temporales se asemejen en el máximo grado posible a las salidas del propio sistema con objeto de realizar simulaciones en diferentes entornos.

El modelado de tráfico de vídeo digital codificado es especialmente importante ya que los servicios de multimedia, vídeo bajo demanda, distribución de vídeo además de estar en continuo crecimiento, son los mayores consumidores de ancho de banda de la red.

Las aplicaciones de los modelos de tráfico son muchas, entre ellas se pueden destacar:

- Dimensionar la red para soportar una carga de tráficos heterogéneos simultáneos.

- Evaluar las prestaciones de los dispositivos o del comportamiento de la red extremo a extremo.
- Establecer criterios de control de admisión de nuevas llamadas con un nivel de calidad de servicio especificado.
- Definir funciones reguladoras de congestión preventiva que monitorice el comportamiento de la conexión de forma que se cumpla con los parámetros especificados.
- Predecir el comportamiento del tráfico, simple o multiplexado, para aumentar el grado de servicio ofrecido y la explotación de recursos.

Un modelo de tráfico debe capturar los comportamientos del tráfico generado por el servicio que son significativos a la hora de desarrollar las funciones especificadas anteriormente. La bondad de un modelo debe ser evaluada en tanto en cuanto capture estos comportamientos. La gran mayoría de modelos intentan caracterizar el comportamiento de uno o varios de los parámetros relacionados con la tasa de generación (λ) [And93]:

1. Tasa media de generación (m):
$$m = E\{\lambda\}$$
2. Varianza de la tasa de generación (σ^2):
$$\sigma^2 = E\{(\lambda - m)^2\}$$
3. Coeficiente de variación de la tasa de generación (c). Se obtiene como cociente de la desviación estándar y la media de la tasa generada:
$$c = \frac{\sigma}{m}$$
4. Momentos de orden superior de la tasa generada:
$$m^{(j)} = E\{\lambda^j\},\ j = 3, 4, \ldots$$
5. Función de distribución de probabilidad de la tasa de generación.
6. Función de autocorrelación de la tasa de generación. Esta función permite ajustar la relación temporal de las tasas generadas por el modelo.
7. Relación de rafagueo (Burstiness, B). Se obtiene como el cociente entre la tasa máxima de generación (R_p) y la tasa media (m):
$$B = \frac{R_p}{m}$$
8. Parámetro de Hurst (H) o relación de autosemejanza. Intenta relacionar la variación a corto y a largo plazo de la tasa de generación.

Debido al algoritmo de codificación MPEG de una secuencia de vídeo, el flujo de bits obtenido se puede caracterizar a diferentes niveles relacionados con la escala temporal escogida. Estos niveles pueden ser:

- Nivel de secuencia: En este nivel se utiliza toda la secuencia de vídeo y la duración temporal va desde varios minutos a horas
- Nivel de escena: Las escenas son intervalos de tiempo donde el contenido de las imágenes es parecido. La duración temporal es del orden de segundos
- Nivel de GoP: Se utiliza el periodo de un GoP. La duración es de cientos de milisegundos
- Nivel de imagen (frame): El periodo básico es de una imagen y la duración es de decenas de milisegundos
- Nivel de tira (slice): Fragmento horizontal de una imagen

La elección del nivel sobre el que se desea obtener un modelo depende del uso al que esté destinado ese modelo. Si se desea obtener un modelo para estimar razonablemente las probabilidades de pérdida en *buffers* de diversos tamaños es necesario utilizar un modelado a nivel de imagen (*frame*) que capture la periodicidad de la función de autocorrelación [Ros95]. Tal

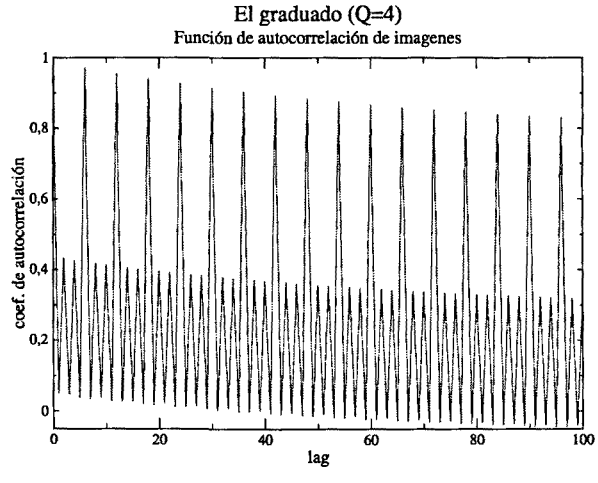


Figura 4.1: autocorrelación a nivel de imagen

periodicidad mostrada en la figura 4.1 es debida al patrón IBBPBB utilizado en la fase de codificación.

Es obvio que una vez elegido el nivel sobre el que se desea trabajar, se pueden obtener modelos para ese mismo nivel y para niveles superiores ya que los datos que forman la base para la caracterización de los parámetros del modelo tienen la información necesaria de la escala temporal mínima elegida y por ende para las escalas superiores.

Durante la década de los 90 se publicaron varios trabajos de investigación sobre técnicas de suavizado a nivel de imagen en la transmisión del tráfico de vídeo VBR ([dlCAM98]). En este caso, el tráfico entregado a la red se puede modelar a nivel de GoP sin pérdida de generalidad. Esta situación se muestra en la figura 4.2 en la que se representa el tamaño en bits de las primeras 3000 imágenes de la película *El graduado* y el valor medio del tamaño de la imagen en un GoP.

Trabajando a nivel de GoP es sencillo observar la dependencia a largo plazo (LRD, *Long Range Dependence*) exhibida por el tráfico de vídeo VBR tal y como se muestra en la figura 4.3.

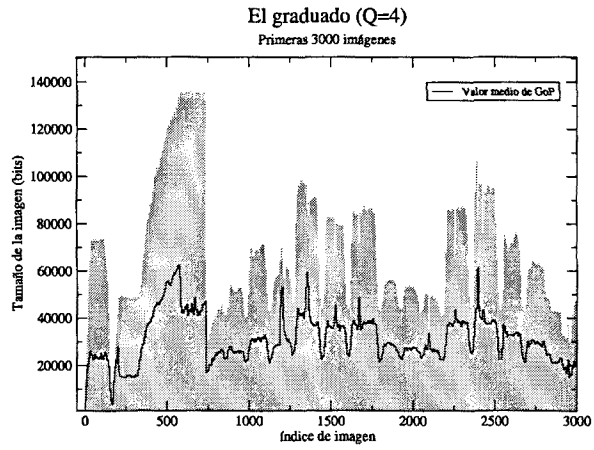


Figura 4.2: Tamaño de Imágenes frente a tamaño medio de GoP

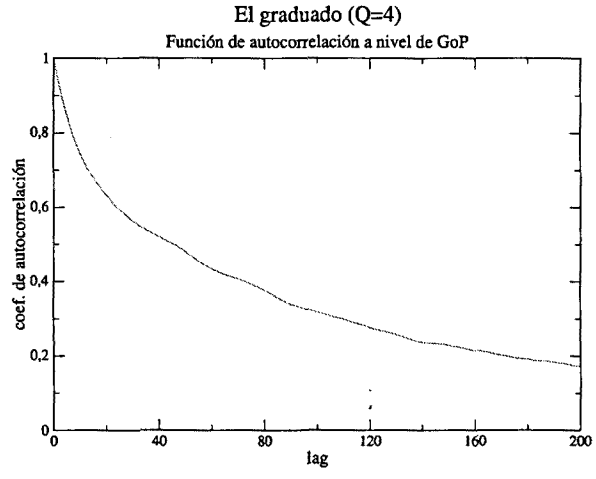


Figura 4.3: autocorrelación a nivel de GoP

La dependencia a largo plazo de la función de autocorrelación de un proceso estocástico indica un comportamiento fractal o autosemejante de dicho proceso. En otras palabras, y aplicado al tráfico de vídeo VBR, se observa que el comportamiento a ráfagas de dicho tráfico se repite a diferentes escalas de tiempo. Diferentes estudios demuestran que la presencia de tráfico autosemejante en las colas de los multiplexores y conmutadores de las redes de banda ancha lleva a la necesidad de tener un mayor número de recursos para mantener la misma calidad de servicio (QoS, *Quality of Service*) que cuando el tráfico no es autosemejante. La LRD implica que pueda darse una persistencia considerable en procesos de tráfico a ráfagas. El crecimiento de los elementos almacenados en las colas de espera de los dispositivos de las redes de conmutación de paquetes se debe a que el tráfico es en ocasiones superior al que puede ser servido provocando que los episodios de congestión puedan extenderse aumentando la probabilidad de pérdida o el retardo introducido por los dispositivos de red congestionados. A estos resultados se llega en [AM95] y en [CGL96] a través de la realización de simulaciones.

Los modelos clásicos para el tráfico en redes de paquetes basados en procesos de llegada de Poisson, utilizados tradicionalmente en el análisis de redes de telefonía, no capturan el comportamiento autosemejante debido al hecho de que un proceso de Poisson asume que las llegadas son independientes. Estos modelos llevan a errores cuando se utilizan para conducir simulaciones con el fin de dimensionar el tamaño de los *buffers* y la capacidad de los enlaces de las redes de paquetes. En general, estos modelos llevan a resultados demasiado optimistas obteniéndose tamaños de *buffers* menores de los necesarios para una *QoS* determinada. Desde mediados hasta finales de los años 90, se han publicado diversos trabajos de investigación en los que se buscan nuevos modelos del tráfico de vídeo VBR que integren tanto la dependencia a corto (SRD) como a largo plazo (LRD).

Los diferentes modelos propuestos en la literatura se pueden clasificar, o bien por los parámetros de tráfico que ajustan, o bien por el nivel temporal donde son aplicados. Las series temporales generadas por los modelos de tráfico son eventos que pretenden definir la tasa instantánea de generación o la tasa media de generación en un intervalo dado. Los procesos de generación de tasa media en intervalos de duración dados proporcionan como eventos el volumen de información a transferir en un intervalo, mientras que los procesos de generación de llegadas hacen hincapié en cómo se producen las transferencias de información indicando el tiempo entre dos llegadas consecutivas de paquetes de información. Dentro de los trabajos presentados en la literatura se han desarrollado también modelos compuestos. Estos modelos conjugan la generación de tasas en intervalos dados y las tasas instantáneas. Se basan en desarrollar un proceso que sintetice el tiempo entre llegadas y otro proceso que determine el número de llegadas en ese instante. A estos modelos se les denomina genéricamente como procesos de llegada en grupo (*Batch Arrival Processes, BAP*).

La clasificación de los modelos de tráfico se puede realizar atendiendo a múltiples criterios; como en los parámetros que sintetizan, modelos de tasa media en un intervalo o modelos de generación instantánea, etc. Se pueden encontrar trabajos de gran calidad sobre clasificaciones pormenorizadas de los modelos tráfico de vídeo, como por ejemplo en [Cas98]. En la tabla 4.1 se muestra una clasificación de los diferentes modelos en función de los parámetros ajustados tal y como se expone en el capítulo 4 en [Cas98]. Sin embargo no es objetivo de esta tesis la realización de tal nivel de pormenorización de la clasificación de los modelos de tráfico de vídeo, por ello y seguidamente se presenta una clasificación en cuatro grandes categorías según las técnicas de génesis de eventos que emplean.

Modelos markovianos de renovación

Los modelos de renovación sintetizan procesos de generación de tasa instantánea con distribución de probabilidad genérica. Los eventos generados son independientes e idénticamente distribuidos. El principal inconveniente presentado es la incorrelación de las generaciones dado que la captura de la relación temporal entre llegadas es imprescindible para hallar los efectos provocados por la generación en avalancha en los dispositivos de red.

Para capturar la correlación temporal de los eventos se controlan las generaciones a través de una cadena de Markov. Estos procesos reciben el nombre de procesos de renovación markovianos, y se caracterizan por un conjunto de estados, que forman una cadena de Markov, y los tiempos de transición entre los estados. La función de distribución de probabilidad de los tiempos de transición es genérica y depende exclusivamente del estado previo a la transición a un nuevo estado de la cadena de Markov. Este tipo de modelos se emplea para determinar la tasa instantánea, o de forma equivalente, el tiempo entre llegadas consecutivas. Así, cada

Propiedad Ajustada	Modelos
PDF	Histográmicos PDF concretas: Normal, Gamma, Weibull, Log-Normal, Pareto, ...
SRD	Modelos AR, ARMA
PDF y SRD	Modelo NARMA Cadenas de Markov Modelos DAR Modelos TES Modelos orientados a escenas con duración exponencial
LRD	Ruidos Gaussianos Fraccionarios (FGN) Procesos de renovación fractal (FRP), mapas caóticos
PDF y LRD	FGN proyectado
SRD y LRD	Modelo FARIMA Planteamiento no estacionario: Modelo ARIMA
PDF, SRD y LRD	Modelo FARIMA proyectado Procesos de renovación espacial o SRP Modelos orientados a escena con duración de escena subexponencial
Otros efectos: Cambios de planos	Cadenas de Markov con estados específicos y evolución determinista

Tabla 4.1: Clasificación de los modelos

vez que se produce una transición entre estados se considera que se ha producido una nueva generación de un paquete. Este modelo puede se ampliado con la consideración de llegadas en grupo, asociando un segundo proceso independiente del tiempo entre llegadas. El proceso de llegadas define el número de paquetes de una llegada. En general, este proceso de generación en grupo se liga a otra cadena de Markov con un número finito de estados, donde cada estado define el número de paquetes de llegada.

Modelos de tasa modulada por Markov

Estos modelos sintetizan la tasa de generación según una función de distribución de probabilidad que depende del estado en que se encuentra una cadena de Markov. Como caso particular, se emplea una distribución exponencial de generación de tasas, dando lugar a los procesos de Poisson modulados por Markov (*Markov-Modulated Poisson Processes, MMPP*). Estos procesos conjugan las características exponenciales del tiempo de permanencia en los estados y el tiempo entre generaciones. Así, en cada estado de la cadena de Markov se generarán tiempo entre llegadas exponenciales según una tasa de generación poissoniana que depende del estado en curso. Estos modelos permiten un tratamiento analítico y han sido ampliamente estudiados en [FMH93]. Los casos más simples de MMPP son los de un estado, es decir, un proceso de Poisson, y los modelos ON/OFF, también denominados procesos de Poisson interrumpidos (*Interrupted Poisson Processes, IPP*). Los IPP son procesos de dos estados donde la tasa de generación poissoniana en un estado es nula y en el otro es constante [Kuc73].

Aproximación de Fluidos

Los modelos de fluidos pueden considerarse como un caso particular de los modelos de tasa modulados por Markov cuando la tasa generada está controlada por una cadena de Markov y la generación es determinista en cada estado. Estos procesos se denominan procesos de fluidos modulados por Markov (*Markov Modulated Fluid Processes, MMFP*). La aproximación realizada en estos modelos es considerar la tasa de generación constante durante el intervalo de tiempo en que se encuentra la cadena de Markov en un estado. Por ello también se denominan *Markov Modulated Constant Rate* (*MMCR*). La cadena de Markov discretiza las posibles tasas de generación en un número de niveles igual al número de estados, como se ilustra en la figura 4.4. Esta aproximación es válida en los casos donde el número de unidades básicas generadas durante un intervalo es muy elevado. En consecuencia, la casuística de una unidad generada carece de importancia y el análisis se debe centrar en el volumen de información transferido.

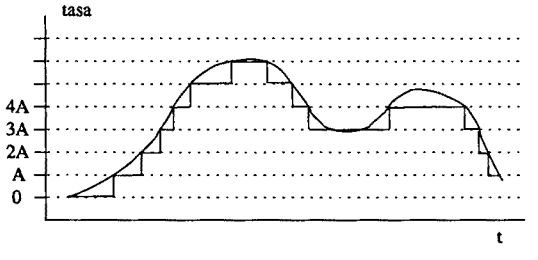


Figura 4.4: Aproximación de la tasa mediante un proceso MMFP

Dentro de los MMFP, los utilizados más frecuentemente son:

1. MMFP binomial ($M/M/\infty//S$). Estos modelos presentan una función de distribución de probabilidad binomial para la generación de tasas y una función de autocorrelación exponencial [MAS+98]. La cadena de Markov es una cadena de nacimiento y muerte que puede describir un sistema con un numero infinito de servidores, con distribución de tiempo exponencial de parámetro β, donde las llegadas se producen desde una población finita de S elementos con tasa individual de generación poissoniana α. Este sistema, según la notación Kendall, sería un $M/M/\infty//S$. Según el estado de la cadena de Markov se genera a una tasa constante. Se puede considerar un valor mínimo de generación en el estado 0 del valor A_{min}. Las tasas de generación entre estados consecutivos se suelen fijar de forma que difieran en un valor constante A. Según esta definición, la aproximación se basa en discretizar la tasa de generación en un conjunto $S+1$ estados, donde, en un estado $i \in \{0,1,\ldots,S\}$ de la cadena de Markov el tiempo de permanencia está distribuido exponencialmente con tasa $(S-i)\alpha+i\beta$ y la tasa binaria generada es de valor $iA+A_{min}$. De esta forma, obtendríamos una aproximación como la presentada en la figura 4.7. El caso más simple de los MMFP con distribución binomial son los denominados MMCR ON/OFF, donde la cadena de Markov sólo dispone de dos estados con generación nula y generación a tasa A. Los procesos MMFP binomiales tiene la propiedad fundamental de poderse descomponer en la agregación de procesos elementales ON/OFF. Así, un proceso genérico como el presentado en la figura 4.7, se puede descomponer en S procesos simples, tal como se muestra en la figura 4.5, donde

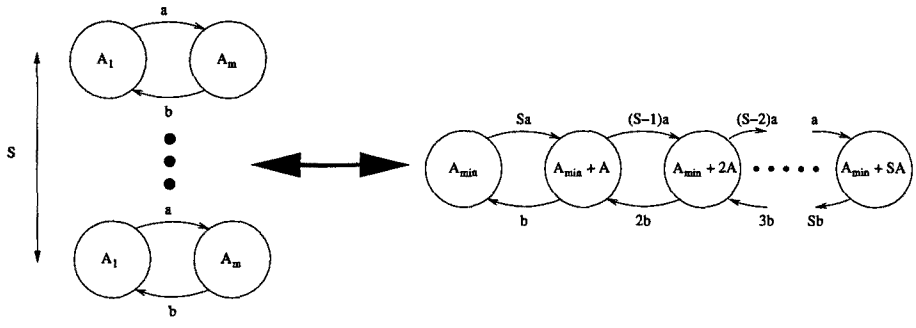


Figura 4.5: Descomposición de una fuente multiestado en minifuentes

$$\begin{aligned} A_{min} &= SA_1 \\ A_m &= A + A_I \end{aligned}$$

2. Procesos autoregresivos de media móvil discretos (*Discrete AutoRegresive Moving Average, DARMA*). Estos procesos también generan una tasa constante según el estado de una cadena de Markov. Su tratamiento analítico es muy complejo, aunque, presentan como cualidad el ajuste de cualquier función de distribución de probabilidad y de la función de autocorrelación [JL83]. Para funciones de distribución simples y considerando una función de autocovarianza con decaimiento exponencial, estos modelos presentan una estructura regular fácilmente sintetizable, denominada DAR(1) [HTL92]. La cadena de Markov, en general, permite las transiciones entre cada par de estados. La probabilidad de encontrar el proceso en un estado de la cadena viene fijada por la función de densidad de probabilidad sintetizada.

Procesos autoregresivos

Estos procesos han sido estudiados ampliamente en la literatura y en su forma más general se denominan procesos autoregresivos, integrativos de media móvil (*autoregressive integrative moving average, ARIMA*) [BJ94]. Los modelos autoregresivos se emplean en el contexto de fuentes de tráfico sintéticas o en predicción de tráfico para la generación de tasas medias en intervalos de duración fija [GCMOO91]. Los modelos ARIMA(p,d,q) se componen de una parte autoregresiva de orden p, una parte integrativa de orden d y una parte de media móvil de orden q. La parte autoregresiva refleja la dependencia entre la generación actual y las pasadas p generaciones. Así, para un proceso AR(p) los valores generados en una serie temporal $Y = \{y_0, y_1, \ldots, y_n\}$ se obtienen de los p valores pasados y un factor independiente de la serie temporal, modelable como un proceso de valores idénticamente distribuidos e independientes entre sí $W = \{w_0, w_1, \ldots, w_n\}$. Habitualmente, los valores de la serie W se sintetizan a partir de la realización de una variable aleatoria gaussiana con una media y una desviación típica relacionadas directamente con los correspondientes momentos del proceso AR a generar. De forma que

$$y_n = a_1 y_n + b_1 y_{n-1} + b_2 y_{n-2} + \cdots + a_p y_{n-p} + w_n$$

donde los términos a_i son coeficientes constantes.

La parte MA(q) del proceso refleja la dependencia en la generación de los valores pasados del proceso independiente que contribuye en el valor obtenido. Así, un proceso MA(q) podría expresarse como:

$$x_n = b_0 w_n + b_1 w_{n-1} + b_2 w_{n-2} + \cdots + b_q w_{n-q}$$

donde los términos b_i son coeficientes constantes.

La componente integrativa pretende modelar la no estacionareidad de los momentos del proceso estocástico. Si bien podría considerarse dentro de la parte AR por su formulación, su síntesis depende de factores distintos de la parte autoregresiva. Así, la parte integrativa también muestra la dependencia con valores pasados de la realización pero depende de los momentos del proceso no estacionario más que de la relación temporal entre generaciones. El orden d de la componente integrativa queda fijado por el orden del momento del proceso estocástico no estacionario. En general, esta dependencia se puede expresar:

$$z_n = c_1 z_{n-1} + c_2 z_{n-2} + \cdots + c_d z_{n-d} + w_n$$

donde los c_i se obtienen a través de:

$$c_i = \binom{d}{i}(-1)^{i+1} \; i \in \{1, 2, \ldots, d\}$$

Como caso de aplicación, un proceso cuya media no es estacionaria, pero sí sus momentos de orden superior, tendría una parte integrativa de orden 1. Los procesos integrativos de orden 1 reciben el nombre de *random walk* y no están acotados.

Interpretando el proceso ARIMA(p,d,q) a través de la transformada **z**, se obtiene la siguiente relación

$$Y(z) = \frac{B(z)}{A(z)C(z)} W(z) \tag{4.1}$$

siendo $Y(z)$, $B(z)$, $A(z)$, $C(z)$ y $W(z)$ las transformadas **z** de la secuencias Y, la secuencia B formada con los valores de los coeficientes b_i, $B = \{b_1, \ldots, b_q\}$, la secuencia formada con los valores de los coeficientes a_i, $A = \{a_1, \ldots, a_p\}$, la secuencia C formada con los valores de los coeficientes c_i, $C = \{c_1, \ldots, c_d\}$ y la secuencia W. Finalmente, interpretando la relación expresada en (4.1) como la relación entrada-salida de un filtro discreto, la función de transferencia de este filtro es

$$H(z) = \frac{Y(z)}{W(z)} = \frac{B(z)}{A(z)C(z)}$$

cuyo modelo esquemático esta representado en la figura 4.6.

A la serie temporal w se le denomina serie residual y se suele considerar como la parte impredictible de la siguiente generación a partir de los valores anteriores. Estos procesos estocásticos son incorrelados y su distribución suele ser gaussiana.

Dentro de los modelos autoregresivos se han introducido mejoras para ajustar, además de las funciones de autocorrelación, las funciones de distribución de probabilidad [GCMOO91], [Ens94]. En particular cabe resaltar los modelo *Transform-Expand-Simple* (TES). Los modelos TES realizan una transformación de una secuencia uniformemente distribuida para obtener la función de distribución deseada. La generación del proceso uniforme, además, captura las principales características de la función de autocorrelación.

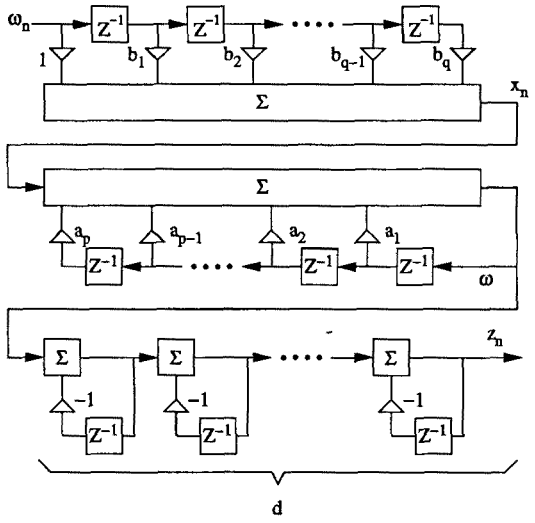


Figura 4.6: Esquema de un filtro ARIMA (p,d,q)

Procesos autosemejantes

Los modelos autosemejantes sintetizan la tasa media generada en un intervalo fijo. A diferencia del resto de modelos presentados anteriormente, intentan capturar el comportamiento que presentan las fuentes de tráfico a largo plazo, en vez de a corto plazo. En general, la mayoría de los procesos estocásticos reducen su relación temporal cuanto más distantes son los eventos, aunque, en particular, las fuentes de vídeo manifiestan una relación temporal que disminuye mucho más lentamente de lo habitual. Por ello los modelos autosemejantes también reciben el nombre de persistentes, o bien modelos de efecto Hurst.

Existen diferentes maneras de observar el efecto Hurst a través del análisis de las series temporales. La persistencia queda patente cuando la densidad espectral de potencia de las series tiene un decaimiento con relación a la frecuencia de la forma $f^{-\alpha}$, con α un valor constante y positivo.

Los procesos persistentes también se caracterizan por disponer de una función de autocovarianza no sumable, lo cual implica que la autocorrelación decae de forma más lenta que una exponencial. Se ha comprobado que los procesos de Hurst presentan un decaimiento hiperbólico.

Otra manera de identificar los procesos persistentes es a través de las series temporales agregadas. Se considera una serie temporal agregada S^m como la serie obtenida a partir de una serie original S, donde el valor de las muestras es resultado de la adición de las muestras de la serie original que pertenecen a un mismo intervalo de duración a^m, donde a es una base con valor real, positivo y constante. En general, se emplean como intervalos de agregación $\{10^1, 10^2, 10^3, 10^4, \ldots\}$. Se observa que las series temporales agregadas reducen su varianza según la relación:

$$Var(S^m) \approx \frac{Var(S)}{m^\beta} \; 0 < \beta \leq 1$$

Cuando los procesos estocásticos estacionarios no presentan una dependencia a largo plazo el valor de β es 1. Valores de β inferiores nos determinan que, para diferentes escalas temporales, los procesos mantienen un parecido en su comportamiento, razón por la cual se denominan

autosemejantes. Este parecido se diluye mucho más lentamente que en los procesos sin relación a largo plazo, los cuales, reducen su varianza rápidamente cuando se considera la agregación en pocos órdenes de magnitud. Los procesos autosemejantes se identifican a través del parámetro de Hurst (H), el cual se relaciona con el valor de β según

$$H = 1 - \frac{\beta}{2}$$

La estimación más precisa de H se puede llevar a cabo con la técnica *reescaled adjusted range statistics* (R/S) presentada en [GW94]. Los efectos de dependencia a largo término pueden ser sintetizados por los procesos denominados procesos de diferenciación fraccional. Esos procesos se derivan de los procesos integrativos puros descritos en los modelos autoregresivos. Por ello, también se suelen denotar como procesos ARIMA(0,d,0) donde d no necesariamente debe ser un número entero, como en los procesos integrativos. Esta generalización de los procesos integrativos se hace en base a la expresión del polinomio en C(z) que caracterizaba la multiplicidad del polo $z = 1$ en el filtro asociado a los modelos ARIMA con parte integrativa no nula. Así,

$$C(z) = (1 - z^{-1})^d = \sum_{i=0}^{\infty} \binom{d}{i} (-1)^i z^{-i} \quad -0{,}5 < d < 0{,}5$$

donde

$$\binom{d}{i}(-1)^i = \frac{\Gamma(-d+i)}{\Gamma(-d)\Gamma(i+1)}$$

La relación entre la multiplicidad del polo (d) y el parámetro de Hurst es:

$$d = H - \frac{1}{2}$$

Los modelos autosemejantes, además, se conjugan con alguna transformación de la función de distribución gaussiana obtenida en el proceso ARIMA(0,d,0) para ajustar el comportamiento de las series temporales empíricas.

4.2. Modelo de fluidos bidimensional

En la transmisión de vídeo MPEG VBR es muy aconsejable el empleo de técnicas de suavizado que reduzcan el efecto de rafagueo (*burstiness*) de la estructura de codificación del algoritmo MPEG. Mediante estas técnicas de suavizado se consigue deshacer las fluctuaciones periódicas de la tasa de transmisión exhibidas por el tráfico VBR debido a los modos de codificación del algoritmo MPEG permitiendo maximizar la ganancia de multiplexación y reducir los recursos ubicados en los diferentes dispositivos de red. El suavizado del tráfico se puede realizar a través del almacenamiento de la información en un *buffer* entre el codificador y la interfaz de usuario. Dependiendo de los requerimientos temporales del servicio se puede llegar a almacenar toda la información referente a un GoP para entregarla posteriormente a la red a tasa constante durante el tiempo de GoP. Para servicios con requerimientos temporales más restrictivos, se hace imprescindible la reducción del tiempo de almacenamiento, del orden de 80 *ms*, por lo que es necesario el empleo de técnicas predictivas [dlCAM98].

Puesto que las fuentes de tráfico emplearán técnicas de conformación, desde el punto de vista de la red, parece adecuado obtener un modelo que capture las características del tráfico suavizado, ya que dicho tráfico será el que realmente se entregue a la red. El modelo que

se presenta es un modelo de fluidos modulado por Markov (MMFP, *Markov Modulated Fluid Process*) presentado y validado previamente en [J.M96] y en [dlCFAM98]

Los modelos clásicos MMFP binomiales presentan una función de autocovarianza con decaimiento exponencial y una función de distribución binomial. Este tipo de procesos corresponden a una cadena de Markov de nacimiento y muerte mostrado en la figura 4.7. El estudio y ajuste

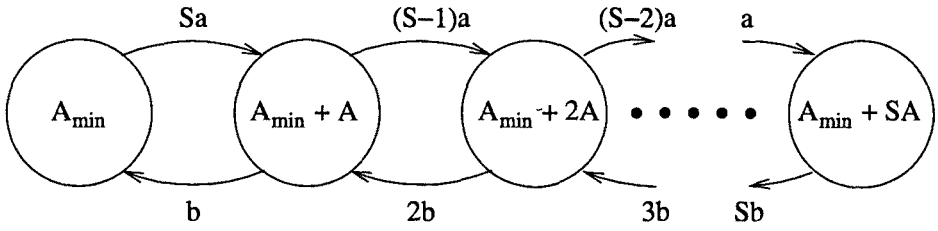


Figura 4.7: Proceso MMFP binomial

de este tipo de procesos puede realizarse a través del análisis del sistema de cola $M/M/\infty//S$ que presenta una distribución binomial de parámetro p. Definiendo la variable aleatoria Z asociada al número de elementos del sistema $M/M/\infty//S$ y X otra variable aleatoria binomial relacionada con Z a través de

$$X = A \cdot Z$$

se pueden interpretar los estados del sistema como el número de minifuentes ON/OFF que generan en estado ON a una tasa constante A y en estado OFF una generación nula. Puesto que las series temporales no alcanzan un valor nulo de generación, se modifica el modelo para que la tasa de generación en estado OFF sea un valor mínimo. Formalmente, se define una variable aleatoria Y relacionada con X a través de

$$Y = X + k$$

donde k es el valor mínimo generado por el modelo. La variable aleatoria Y se puede interpretar como la agregación de un número S de minifuentes ON/OFF con tasa de generación constante en cada estado. El modelo de minifuente que se utiliza es el presentado en la figura 4.8, donde

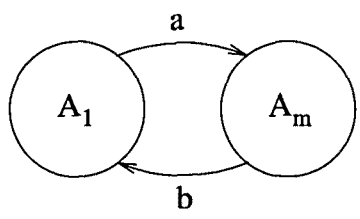


Figura 4.8: Modelo de minifuente de dos estados

la diferencia entre la tasa de generación en estado ON (A_m) y la tasa de generación en estado OFF (A_1) es constante y de valor A :

$$A_m - A_1 = A > 0$$

Para determinar la función de autocorrelación de estas fuentes es necesario recurrir a la evolución temporal del proceso. El régimen transitorio puede calcularse a través del balance de flujos de probabilidad de la cadena de Markov de dos estados según las ecuaciones:

$$\begin{array}{rcl} \frac{dP_{off}(t)}{dt} & = & -aP_{off}(t) + bP_{on}(t) \\ \frac{dP_{on}(t)}{dt} & = & aP_{off}(t) - bP_{on}(t) \end{array} \tag{4.2}$$

La ecuación (4.2) corresponde a un sistema de ecuaciones diferenciales ordinarias de primer orden cuya solución es:

$$\begin{array}{rcl} P_{off}(t) & = & \pi_{ON} + (P_{off}(0) - \pi_{ON})e^{-(a+b)t} \\ P_{on}(t) & = & \pi_{ON} + (P_{on}(0) - \pi_{ON})e^{-(a+b)t} \end{array} \tag{4.3}$$

donde π_{ON} y π_{OFF} son las probabilidades de encontrarse en estado ON u OFF, respectivamente, en régimen permanente y sus valores son:

$$\begin{array}{rcl} \pi_{ON} & = & \frac{a}{a+b} \\ \pi_{OFF} & = & \frac{b}{a+b} \end{array} \tag{4.4}$$

Las probabilidades de transición de estados ($P_{00}(t)$, $P_{01}(t)$, $P_{10}(t)$, $P_{11}(t)$), se derivan a partir de (4.3) imponiendo las condiciones de contorno necesarias:

$$\begin{array}{rcl} P_{00}(t) = P_{off}(t)|_{P_{off}(0)=1} & = & \pi_{OFF} + \pi_{ON}e^{-(a+b)t} \\ P_{10}(t) = P_{off}(t)|_{P_{on}(0)=1} & = & \pi_{OFF} - \pi_{OFF}e^{-(a+b)t} \\ P_{01}(t) = P_{on}(t)|_{P_{off}(0)=1} & = & \pi_{ON} - \pi_{ON}e^{-(a+b)t} \\ P_{11}(t) = P_{on}(t)|_{P_{on}(0)=1} & = & \pi_{ON} + \pi_{OFF}e^{-(a+b)t} \end{array} \tag{4.5}$$

Sea $Y(t)$ un proceso aleatorio modulado por la minifuente de la figura 4.8. En este caso $Y(t)$ tendrá los siguientes valores:

$$Y(t) = \begin{cases} A_m & con\, probabilidad\, \pi_{ON} \\ A_1 & con\, probabilidad\, \pi_{OFF} \end{cases}$$

El valor medio de $Y(t)$ lo obtenemos a partir de la esperanza:

$$\eta = E\{Y(t)\} = \sum_i y_i P[Y(t) = i] = y_o P[Y(t) = 0] + y_1 P[Y(t) = 1] = A_m \pi_{ON} + A_1 \pi_{OFF}$$

La función de autocorrelación la obtenemos también a partir de la esperanza:

$$r_{yy}(\tau) = E\{Y(t)Y(t+\tau)\} = \sum_i \sum_j y_i y_j P[Y(t) = y_i, Y(t+\tau) = y_j] \tag{4.6}$$

Como $Y(t)$ es un proceso de Markov, se cumple la siguiente relación:

$$P[Y(t) = y_i, Y(t+\tau) = y_j] = P[Y(t+\tau) = y_j | Y(t) = y_i] P[Y(t) = y_i]$$

y si además se impone que sea un proceso homogéneo en tiempo, se cumple que

$$P[Y(t+\tau) = y_j | Y(t) = y_i] = P[Y(\tau) = y_j | Y(0) = y_i] = P_{ij}(\tau)$$

Sustituyendo estas igualdades en la ecuación (4.6) se simplifica para obtener:

$$r_{yy}(\tau) = \sum_i \sum_j y_i y_j P_{ij}(\tau) P[Y(0) = i]$$

Sumando esta última ecuación para valores $i, j = 0 \ldots 1$ se obtiene que la función de autocorrelación del proceso $Y(t)$ modulado por la minifuente de la figura 4.8 es:

$$r_{yy}(\tau) = {A_1}^2 P_{00}(\tau)\pi_{OFF} + A_1 A_m P_{01}(\tau)\pi_{OFF} + A_m A_1 P_{10}(\tau)\pi_{ON} + {A_m}^2 P_{11}(\tau)\pi_{ON}$$

y utilizando las probabilidades de transición de estado de la ecuación (4.5) y el valor medio de $Y(t)$ obtenido:

$$r_{yy}(\tau) = \eta^2 + A^2 \pi_{ON}\pi_{OFF} e^{-(a+b)\tau} \tag{4.7}$$

siendo $A = A_m - A_1$. La autocovarianza es:

$$COV_{yy}(\tau) = r_{yy}(\tau) - \eta^2 = A^2 \pi_{ON}\pi_{OFF} e^{-(a+b)\tau} \tag{4.8}$$

y la varianza la obtenemos a través de la autocovarianza

$${\sigma_{yy}}^2 = COV_{yy}(0) = A^2 \pi_{ON}\pi_{OFF} \tag{4.9}$$

Analizada la minifuente, el comportamiento de la fuente $M/M/\infty//S$ se obtiene a partir de la adición de variables aleatorias independientes asociadas a las S minifuentes, de donde se obtiene que la tasa media generada, la varianza y la autocovarianza se pueden expresar matemáticamente:

$$\begin{array}{lcll} m_S & = & S\eta & = S(A_m \pi_{ON} + A_1 \pi_{OFF}) \\ {\sigma_s}^2 & = & S{\sigma_{yy}}^2 & = SA^2 \pi_{ON}\pi_{OFF} \\ COV_{SS}(t) & = & S \cdot COV_{yy}(t) & = SA^2 \pi_{ON}\pi_{OFF} e^{-(a+b)t} \end{array}$$

El problema que presenta este modelo es que no es capaz de capturar la dependencia a largo plazo (LRD) exhibida por el tráfico de vídeo MPEG VBR al tener una función de autocorrelación con decaimiento exponencial. Por esta razón, el modelo que se utilizará es una extensión bidimensional del modelo clásico MMFP cuya estructura se muestra en la figura 4.9. La idea de la extensión bidimensional es que mientras la dimensión horizontal se encarga de capturar la dependencia a corto plazo (SRD), la dimensión vertical capture la dependencia a largo plazo (LRD). Para ello es necesario analizar el modelo propuesto. Sin embargo el modelo bidimensional se puede analizar a través de una cola $M/M/\infty//S_1 + S_2$ mediante la agregación de S_1 fuentes en la dimensión horizontal y S_2 fuentes en la dimensión vertical. Para ello definimos:

$$\begin{array}{lcl} X & = & \textit{variable aleatoria asociada al proceso } M/M/\infty//S_1 + S_2 \\ Y & = & \textit{variable aleatoria asociada al proceso } M/M/\infty//S_1 \\ Z & = & \textit{variable aleatoria asociada al proceso } M/M/\infty//S_2 \end{array}$$

entonces

$$\begin{array}{lcl} X & = & Y + Z \\ m_X & = & m_Y + m_Z \\ {\sigma_X}^2 & = & {\sigma_Y}^2 + {\sigma_Z}^2 \\ COV_{XX}(\tau) & = & COV_{YY}(\tau) + COV_{ZZ}(\tau) \end{array}$$

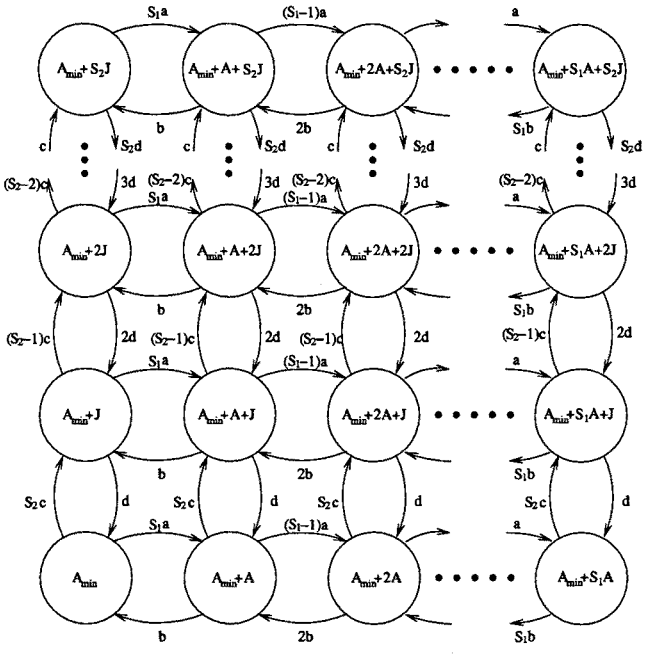


Figura 4.9: Modelo de fluidos bidimensional

A partir de los resultados obtenidos para el modelo $M/M/\infty//S$ podemos expresar:

$$m_x = S_1pA + S_2qJ \tag{4.10}$$

$$\sigma_X{}^2 = S_1p(1-p)A^2 + S_2q(1-q)J^2 \tag{4.11}$$

$$COV_{XX}(\tau) = S_1p(1-p)A^2e^{-\gamma_1\tau} + S_2q(1-q)J^2e^{-\gamma_2\tau} \tag{4.12}$$

Respecto a las ecuaciones (4.10), (4.11) y (4.12), se debe resaltar lo siguiente:

- La media m_X se ha establecido suponiendo que la tasa de generación en estado OFF de las minifuentes horizontales y verticales es nulo. Este hecho no influye en la estructura del modelo y debe ser tenido en cuenta en el momento del ajuste del modelo al conjunto de datos disponible, añadiendo a m_X el valor mínimo de generación A_{min} que se obtiene de los datos.

- Los parámetros p y q se relacionan con los parámetros del modelo de la figura 4.9 de la siguiente manera:

$$p = \frac{a}{a+b}$$
$$q = \frac{c}{c+d}$$

- Las constates de tiempo de las exponenciales de la función de autocovarianza se relacionan con los parámetros del modelo de la siguiente manera:

$$\gamma_1 = (a+b)$$
$$\gamma_2 = (c+d)$$

4.2.1. Proceso de ajuste del modelo MMFP bidimensional

Para ajustar el modelo MMFP bidimensional a un conjunto de datos que representan la tasa de transmisión por GoP, se empieza por ajustar la función de autocovarianza de los datos a una curva matemática de la forma

$$\sigma_Y{}^2 e^{-\gamma_Y \tau} + \sigma_Z{}^2 e^{-\gamma_Z \tau} \tag{4.13}$$

Para poder descomponer la función de autocovarianza de los datos en la suma de dos exponenciales se realiza la hipótesis de que

$$\gamma_Y \gg \gamma_Z > 0 \tag{4.14}$$

Bajo esta hipótesis, el primer sumando de (4.13) debe capturar la dependencia a corto plazo (SRD) y el segundo sumando debe capturar la dependencia a largo plazo (LRD). Además, con esta hipótesis es matemáticamente sencillo obtener la descomposición de la suma ya que observando la figura 4.10, a partir de $\tau > 100$, la influencia de la primera exponencial es despreciable y se puede considerar que:

$$COV_{XX}(\tau) \cong \sigma_Z{}^2 e^{-\gamma_Z \tau}, \ \forall \tau \geq 100$$

Para hallar γ_Z y $\sigma_Z{}^2$ se utiliza una regresión lineal del logaritmo natural de la función de autocovarianza de los datos a partir un valor $\tau = 100$ hasta la longitud máxima de la autocorrelación.

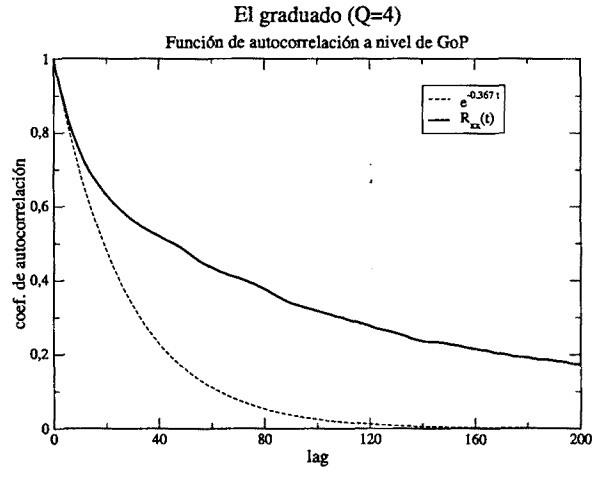


Figura 4.10: Autocorrelación de los datos y aproximación exponencial

En este momento ya se puede obtener la exponencial que representa el SRD ya que:

$$\sigma_Y{}^2 e^{-\gamma_Y \tau} = COV_{XX}(\tau) - \sigma_Z{}^2 e^{-\gamma_Z \tau}$$

Aplicando la misma técnica de estimación que para la relación a largo plazo se obtiene γ_Y y $\sigma_Y{}^2$.

El proceso de ajuste de la función de autocorrelación se ha automatizado mediante un algoritmo. Dicho algoritmo realiza un barrido en τ, obteniendo la estimación de la función

de autocorrelación y calculando un índice de error a base de integrar la resta de la función de autocorrelación y la estimación para el τ escogido. Finalmente, devuelve la estimación que proporciona un índice de error menor. La figura 4.11 muestra la aproximación obtenida por este método para la película *el graduado* codificada VBR con MPEG-2 con un paso de cuantificación $Q = 4$. Tal como se observa, los valores obtenidos para las constantes de tiempo son

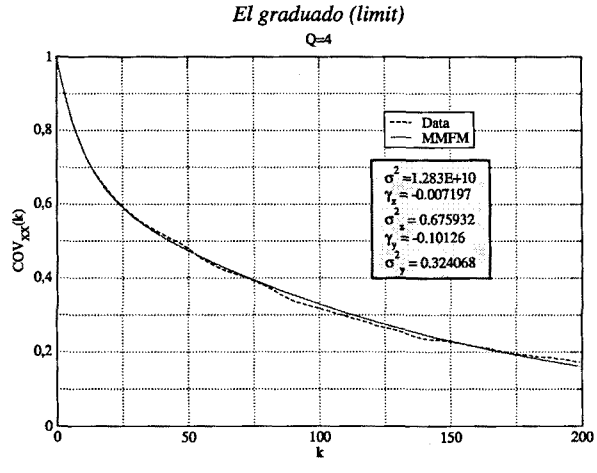


Figura 4.11: Autocorrelación y estimación obtenida

$$\begin{aligned} \gamma_Y &= 0{,}10126 \\ \gamma_Z &= 0{,}007197 \end{aligned}$$

y estos valores cumplen la desigualdad (4.14) y por lo tanto cumplen la hipótesis que se ha asumido inicialmente, suponiendo que una relación mínima de 10 entre las constantes de tiempo es equivalente al "mucho mayor" de la ecuación (4.14).

El siguiente paso en el proceso de ajuste del modelo es estimar los valores restantes de las ecuaciones (4.10), (4.11) y (4.12) es decir p, q, A, J, S_1 y S_2. El análisis de los datos de la serie temporal proporciona las restricciones necesarias para el planteamiento de un conjunto de ecuaciones que permita obtener los valores de los parámetros pendientes. A partir de los datos de la serie temporal se puede obtener:

$$\begin{aligned} A_{max} &= max(X) \\ A_{min} &= min(X) \\ m_X &= E\{X\} \\ {\sigma_X}^2 &= E\{(X - m_x)^2\} \end{aligned}$$

que junto con las varianzas ${\sigma_Y}^2$ y ${\sigma_Z}^2$ obtenidas en el paso de ajuste anterior nos permiten establecer las siguientes relaciones respecto al modelo de la figura 4.9:

- La tasa máxima de generación del modelo se produce en el estado superior derecho y es

$$A_{min} + S_1 A + S_2 J$$

y en este caso debe ajustarse al valor máximo de los datos de la serie $A_{max} = max(X)$

- El valor medio del modelo esta descrito por la ecuación (4.10) y modificado según se comenta a continuación de la ecuación debe ajustarse al valor medio obtenido directamente de la serie temporal ($m_x = E\{X\}$):

$$S_1 pA + S_2 qJ = m_X - A_{min}$$

- Dado que ${\sigma_X}^2 = {\sigma_Y}^2 + {\sigma_Z}^2$ y a través de la descomposición realizada y los valores hallados de ${\sigma_Y}^2$ y ${\sigma_Z}^2$, la siguiente igualdad se deduce de la ecuación (4.11) y (4.12)

$$\begin{aligned} S_1 p(1-p)A^2 &= {\sigma_Y}^2 \\ S_2 q(1-q)J^2 &= {\sigma_Z}^2 \end{aligned}$$

El conjunto de ecuaciones obtenido es:

$$\begin{aligned} &J = \tfrac{A_{max} - S_1 A - A_{min}}{S_2} \\ &S_1 pA + S_2 qJ = m - A_{min} \\ &S_1 p(1-p)A^2 = {\sigma_Y}^2 \\ &S_2 q(1-q)J^2 = {\sigma_Z}^2 \end{aligned} \tag{4.15}$$

El sistema de ecuaciones (4.15) es un sistema indeterminado cuyas incógnitas son S_1, S_2, p, q, A y J. El proceso para obtener el valor de las incógnitas es:

1. Escoger los valores de S_1 y S_2
2. Resolver numéricamente la ecuación no lineal (4.15). Si el sistema no tiene solución, volver al paso 1.
3. Validar los resultados obtenidos del modelo respecto a los datos. Si el ajuste no es lo suficientemente exacto para nuestros propósitos, volver al paso 1.

La elección de los parámetros S_1 y S_2 determinarán el número total de estados del modelo bidimensional. Analizando los valores obtenidos de p y q (con $p > q$), presentados en la tabla 4.2, se observa que estos no varían sustancialmente cuando se incrementa el valor de S_2, lo cual revela que el aumento de S_2 sólo lleva a realizar una ampliación de los niveles alcanzables. Estos niveles se visitan con probabilidades extraordinariamente pequeñas ($\approx 10^{-10}$) cuando $S_2 \geq 3$.

La síntesis de un modelo realista, no sobredimensionado, nos queda fijada a la consideración de los casos $S_2 = 1$ y $S_2 = 2$. Las soluciones obtenidas para el caso $S_2 = 1$ tampoco son válidas, dado que, para el conjunto de tasas sintetizadas existe una fuerte transición entre la máxima tasa generada, cuando la minifuente de tipo 2 está inactiva, y la mínima generada, cuando la minifuente está activa. Este modelo provocaría la no consideración de un conjunto de tasas de generación intermedia, por lo que, el modelo presentaría una deficiencia en la síntesis de estas tasas. Este efecto se podría interpretar, en algunos casos, como una probabilidad de generación nula de tasas intermedias, lo cual se aleja del comportamiento real. En la figura 4.12 se pone de manifiesto el comportamiento descrito ya que entre los valores 384000 y 672000 del eje de abscisas de la función de probabilidad del modelo se mantiene aproximadamente constante indicando que la síntesis de estas tasas es nulo.

En general, se ha hallado que el valor que captura bien las características de generación del tráfico real es $S_2 = 2$.

El proceso de ajuste aplicado a la película *"el graduado"* codificada con $Q = 4$ presenta los siguientes parámetros:

S_1	S_2	p	q	A	J
4	1	0.7688	0.0220	76470	634193
5	1	0.7261	0.0233	64660	616772
6	1	0.6876	0.0247	56795	599302
7	1	0.6526	0.0263	51185	581779
12	1	0.495	0.0401	35768	475084
4	2	0.7459	0.0437	74058	321920
5	2	0.6997	0.0465	62905	312773
6	2	0.6580	0.0495	55491	303562
7	2	0.6203	0.0529	50216	294281
12	2	0.4715	0.0774	37287	246311

Tabla 4.2: Parámetros característicos de las minifuentes de los modelos MMFP bidimensionales

A_{max}	A_{min}	$mean$	σ^2
960000	19928	269070.25	$1{,}283 \cdot 10^{10}$

El resultado del proceso de ajuste es el siguiente:

S_1	S_2	p	q	A	J
14	2	0.4235	0.0937	34876	225904

La función de probabilidad acumulada obtenida del modelo ajustado está representada en la figura 4.13. La estructura bidimensional con las tasas de generación para cada estado de este modelo se muestra en la tabla 4.3.

Obsérvese que, por la propia estructura bidimensional del modelo, los estados de generación del conjunto de minifuentes de tipo 1 (horizontales) que corresponden a un mismo nivel de actividad de las minifuentes de tipo 2 (verticales), definen, de forma natural, tres niveles de actividad de la fuente de vídeo (figura 4.14). El tiempo de permanencia en cada uno de los niveles de actividad captura la dependencia a largo plazo de la tasa de generación y viene condicionado por la tasa de transición de las minifuentes de tipo 2. A su vez, la transición entre niveles de actividad simula los posibles cambios significativos en la tasa de generación que muestran típicamente las secuencias de vídeo. La función de densidad de probabilidad del modelo se ajusta de forma más precisa al tráfico real ya que por construcción, la función de densidad binomial de las minifuentes de tipo 1 se adiciona ponderada y desplazadamente para cada uno de los niveles de actividad tal como se muestra en la figura 4.15.

Analizando las tasas de cada uno de los estados del modelo bidimensional mostrados en la tabla 4.3, se obtienen los valores a partir de los cuales se produce un cambio de nivel de actividad, observándose que se producen unos ciclos de histéresis que dificultan o facilitan el cambio de nivel. El diagrama de histéresis se muestra en la figura 4.16.

471737	506613	541489	576365	611241	646117	680993	715869	750744	785620	820496	855372	890248	925124	960000
245833	280708	315584	350460	385336	420212	455088	489964	524840	559716	594592	629468	664344	699220	734095
19928	54804	89680	124556	159432	194308	229184	264059	298935	333811	368687	403563	438439	473315	508191

Tabla 4.3: Tasas de cada estado del modelo de la película *el graduado*

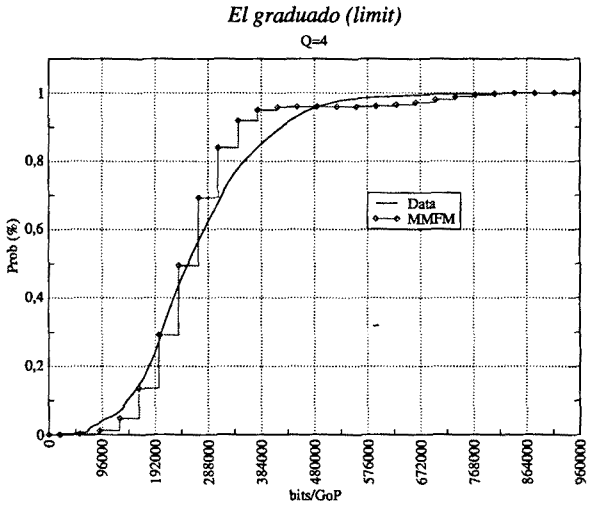


Figura 4.12: Función de probabilidad y ajuste con $S_1 = 14$ y $S_2 = 1$ de la secuencia de *el graduado*

4.2.2. Validación del ajuste del modelo MMFP bidimensional

Utilizando los valores de las tasas de transición entre niveles, se puede analizar de forma numérica el conjunto de datos de la secuencia de vídeo de forma que nos indique la bondad del ajuste realizado. Para ello se ha realizado un programa cuyo algoritmo básico es el siguiente:

1. Leer las tasas de transición entre niveles de actividad
2. Leer línea a línea el fichero de datos de la secuencia de vídeo. Cada línea del fichero de datos indica el número de bits por GoP generados en la codificación de ese GoP.
3. Procesar cada entrada del fichero de datos, determinando el nivel de actividad al que pertenece esa entrada teniendo en cuenta las tasas de transición entre niveles y los ciclos de histéresis.
4. Una vez clasificadas todas las entradas del fichero de datos en los diferentes niveles de actividad, obtener las estadísticas, para cada nivel de actividad, de:
   - *a*) Tiempo medio de permanencia en el nivel
   - *b*) Desviación típica del tiempo medio de permanencia en el nivel
   - *c*) Tiempo máximo de permanencia
   - *d*) Tiempo mínimo de permanencia
   - *e*) Número de veces que se ha entrado en el nivel
   - *f*) Número total de bits generados en el nivel
   - *g*) Valor medio del número de bits/GoP generados en el nivel
   - *h*) Probabilidad del nivel

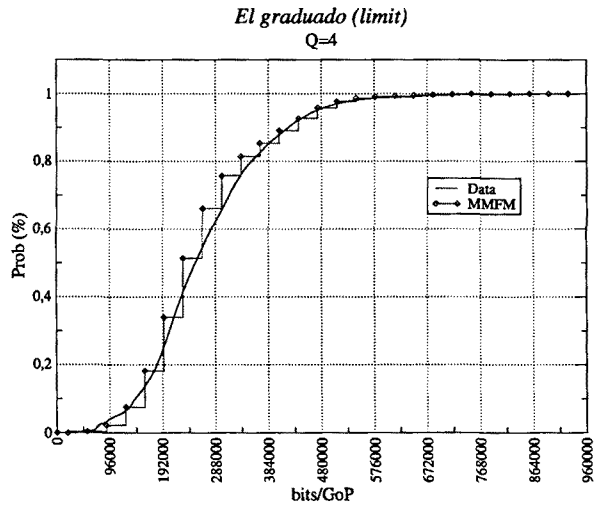


Figura 4.13: Función de probabilidad de *el graduado*

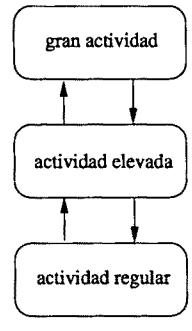


Figura 4.14: Niveles de actividad derivados del modelo MMFP

Teniendo la probabilidad de cada uno de los niveles de actividad, se puede obtener la probabilidad, q, del modelo, teniendo en cuenta que para tres niveles de actividad, la probabilidad del nivel de *"actividad regular"* es $1-q^2$, la probabilidad del nivel de *"Actividad elevada"* es $2q(1-q)$, y la probabilidad del nivel de *"Gran actividad"* es q^2. Se trata, pues, de resolver las ecuaciones:

$$\begin{aligned} p_0 &= (1-q_0)^2 \\ p_1 &= 2q_1(1-q_1) \\ p_2 &= {q_2}^2 \end{aligned}$$

siendo p_0, p_1, p_2 las probabilidades de cada nivel obtenidas a través del análisis de los datos en el punto $4h$ del algoritmo descrito y q_0, q_1 y q_2 las probabilidades de estado del modelo bidimensional. Idealmente debería ocurrir que $q_i = q_j, \forall i,j$, sin embargo, en general, no va a ocurrir que todas las q_i sean iguales porque la serie temporal no tiene porque tener un comportamiento exacto al descrito por el modelo. Así pues, el valor de q escogido es el valor medio de las q_i siempre y cuando el número de eventos de ese nivel de actividad sea suficientemente representativo.

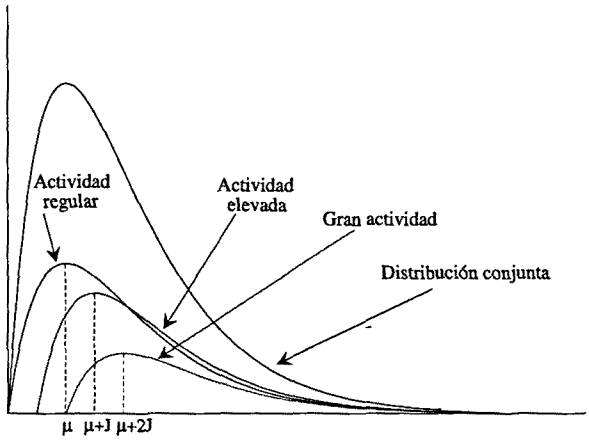


Figura 4.15: Función de densidad de probabilidad del modelo $M/M/\infty/S_1+S_2$ por adición de funciones de densidad de cada nivel de actividad

A través del valor medio del número de bits/GoP generado en cada nivel (punto $4g$ del algoritmo descrito), podemos obtener el valor de J de los datos. También en este caso obtendremos, para el caso de tres niveles de actividad, un valor J_{01} entre los niveles de *"Actividad regular"* y *"Actividad elevada"* , y un valor J_{12} entre los niveles de *"Actividad elevada"* y *"Gran actividad"* . Finalmente nos quedaremos con un valor medio de J.

A continuación se presentan los resultado del análisis de los datos de la película *el graduado* codificada VBR con paso de cuantificación $Q = 4$, utilizando las tasas de transición entre niveles mostradas en la figura 4.16:

```
Eventos Totales 25326

Tiempo medio en estado 0: 656.096774
devstd tiempo en estado 0: 808.277855
Tiempo máximo en estado 0: 3026
Tiempo mínimo en estado 0: 48
Tiempo total en estado 0: 20339
Número de eventos de estado 0: 31
Bits totales generados en estado 0: 4784596345
Bits/GoP en media de estado 0: 235242.457594
Probabilidad del estado 0: 0.803088

Tiempo medio en estado 1: 156.161290
devstd tiempo en estado 1: 154.842521
Tiempo máximo en estado 1: 686
Tiempo mínimo en estado 1: 18
Tiempo total en estado 1: 4841
Número de eventos de estado 1: 31
Bits totales generados en estado 1: 1925018805
Bits/GoP en media de estado 1: 397648.999174
Probabilidad del estado 1: 0.191147
```

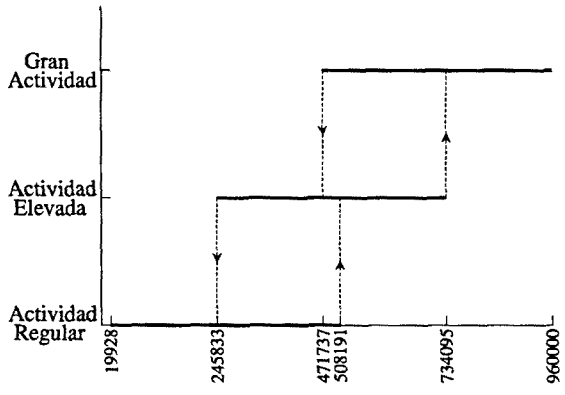


Figura 4.16: Tasas de transición entre niveles y ciclos de histéresis

```
Tiempo medio en estado 2: 36.500000
devstd tiempo en estado 2: 44.455971
Tiempo máximo en estado 2: 103
Tiempo mínimo en estado 2: 10
Tiempo total en estado 2: 146
Número de eventos de estado 2: 4
Bits totales generados en estado 2: 104857926
Bits/GoP en media de estado 2: 718204.972603
Número de renegociaciones por GoP del estado 2: 0.027397
Probabilidad del estado 2: 0.005765

Bits Totales de la película: 6814473076

q de estado 0 (q_0) 0.103848
q de estado 1 (q_1) 0.107029
q de estado 2 (q_2) 0.075926
q_mean (sum(q_i)/N) 0.095601

J01: 162406.541580
J12: 320555.973429
J_m: 241481.257504
```

La obtención de la tasa media por nivel de actividad a partir del modelo bidimensional, puede obtenerse multiplicando la probabilidad del estado por la tasa de generación de ese estado y sumando para todos los estados del mismo nivel de actividad:

$$m_i = \sum_{j=0}^{S_1} \binom{S_1}{j} p^j (1-p)^{S_1-j} R_i(j), \; donde \; i = 0 \ldots S_2$$

La obtención de los tiempos medios de permanencia a partir del modelo se obtienen a través de las tasas de salida de los estados del modelo bidimensional (figura 4.9) y los parámetros γ_Z y q:

$$\begin{aligned} c &= \gamma_Z \cdot q \\ d &= \gamma_Z \cdot (1-q) \end{aligned}$$

ya que $\gamma_Z = (c+d)$ y $q = c \cdot (c+d)^{-1}$. El tiempo de permanencia medio en un estado en una cadena de Markov es igual al inverso de la tasa de salida del estado, por lo tanto:

$$\begin{aligned} T_0 &= \frac{1}{2c} \\ T_1 &= \frac{1}{c+d} \\ T_2 &= \frac{1}{2d} \end{aligned}$$

La siguiente tabla muestra la comparación entre los resultados obtenidos mediante el modelo y el análisis de los datos:

	Modelo	Análisis
Tiempo de permanencia Nivel 0[a]	741,043054	656,096774
Tasa de generación Nivel 0[b]	226712,80	235242,46
Probabilidad Nivel 0	0,8213	0,8031
Tiempo de permanencia Nivel 1	138,946783	156,161290
Tasa de generación Nivel 1	452617,32	397648,99
Probabilidad Nivel 1	0,1699	0,1911
Tiempo de permanencia Nivel 2	76,660363	36,50
Tasa de generación Nivel 2	678521,84	718204,97
Probabilidad Nivel 2	0,0088	0,0058
Valor medio de q	0,093751	0,095601
Valor medio de J	225904,52	241481,26

[a]Las unidades temporales es 1 T_{GoP}

[b]Las unidades de las tasa de generación son de *bits/GoP*

Si el ajuste obtenido no es lo suficientemente exacto, las siguiente regla heurística puede ayudar a la elección de S_1 completando el punto 3 en la página 72 del algoritmo del ajuste de parámetros:

- Dado un valor máximo y mínimo de la tasa de generación, aumentar el valor de S_1 manteniendo el valor de S_2, implica que el valor de q obtenido será mayor y el valor de J obtenido será menor. La explicación de este comportamiento es la siguiente. Al aumentar el valor de S_1, se incrementan el número total de estados entre los que se repartirán las tasas de transmisión ($A_{max} - A_{min}$). Para el nivel de actividad menor ,*"Actividad regular"*, el valor mínimo continúa siendo el mismo (A_{min}), mientras que el valor de transición de nivel tiende a aumentar y también el valor medio de los bits/GoP generados en este nivel. Para el nivel de actividad mayor, *"Gran actividad"*, el valor máximo continúa siendo el mismo (A_{max}) mientras que el valor de transición de nivel tiende a disminuir. Ahora este

nivel de actividad tiene estados con tasas de generación menores que en el caso anterior y que se alcanzan con la misma probabilidad que anteriormente, como consecuencia debe aumentar el valor de q para poder acceder a estos estados. Por otro lado se comprueba que mientras que el valor de J aumenta en el nivel de actividad menor, para los dos siguientes niveles de actividad la J disminuye, obteniéndose diferencias menores y un valor medio de J menor.

Se ha realizado un extenso estudio sobre un conjunto de secuencias codificadas con diferentes factores de cuantificación y diferentes valores de resolución espacial. Los resultados obtenidos del proceso de caracterización mediante el modelo MMFP bidimensional pueden consultarse en el apéndice B.

4.3. Conclusiones

A partir de la consideración de que el tráfico generado por el codificador será suavizado antes de su entrega a la red, ya que las características periódicas de la codificación MPEG VBR reducen las ventajas de la multiplexación estadística, se ha analizado el modelo MMFP bidimensional de tráfico para la tasa binaria generada en intervalos de duración de un GoP. Se ha elegido este tipo de modelo por su tratabilidad matemática en el análisis de las prestaciones de la multiplexación estadística y por los resultados que se habían obtenido en trabajos preliminares donde no se demostraba su aplicabilidad de forma general sobre cualquier secuencia de vídeo. Este modelo de fluidos se caracteriza por estar compuesto de dos tipos de minifuentes diferentes, lo cual da lugar a una estructura bidimensional de estados de la fuente. La estructura bidimensional de este modelo ha permitido ajustar perfectamente las funciones de probabilidad acumulada obtenidas del tráfico real. La síntesis del modelo parte del comportamiento presentado por la función de autocovarianza. Se ha demostrado que esta función se puede descomponer en un término de decaimiento a corto plazo y otro a largo plazo. El decaimiento a largo plazo muestra el efecto de semejanza de los valores generados durante un intervalo de tiempo prolongado. Este comportamiento refleja la composición escénica de las secuencias de vídeo.

La metodología de ajuste de los parámetros del modelo MMFP bidimensional ha sido mejorada respecto a la propuesta inicial, permitiendo una automatización del proceso de modelado y caracterización del tráfico de vídeo. Para ello, se ha desarrollado un algoritmo de ajuste de las funciones de autocovarianza que permite identificar las dependencias a corto y largo plazo. A partir de esta información, se desarrolla un proceso iterativo que facilita el ajuste de los parámetros restantes del modelo. La validación de los valores obtenidos se determina sobre el conjunto de datos reales, reiterando el proceso hasta que dicha validación alcanza el nivel de precisión deseado.

Esta metodología de ajuste ha sido aplicada sobre un conjunto extenso de secuencias de vídeo verificando que la estructura bidimensional del modelo MMFP se adapta adecuadamente en todos los casos. La elección de una clasificación escénica basada en tres niveles de actividad se ha mostrado acertada en todas las secuencias de vídeo estudiadas, independientemente del factor de cuantificación utilizado para la codificación.

La comparación de los parámetros que caracterizan las diferentes secuencias revelan que, en general, el valor de q hallado es muy parecido, y esa similitud se mantiene independientemente del factor de cuantificación utilizado para la codificación de las secuencias de vídeo, mostrando que el nivel de actividad escénica de todas las secuencias sigue un patrón similar. De la misma

forma, una comparación de los valores del parámetro p del conjunto de las secuencias indica que, independientemente del valor S_1 de los modelos, la probabilidad de estado de la dimensión horizontal toma valores muy similares a pesar del factor de cuantificación utilizado.

Todo ello lleva a que considerando únicamente los niveles de actividad, el comportamiento estadístico de las distintas secuencias presenta un fuerte parecido para todos los factores de cuantificación y para las diversas resoluciones espaciales consideradas, permitiendo considerar un modelo escénico genérico que se adapte con suficiente exactitud a cualquier secuencia de vídeo. Considerando los invariantes del modelo Markoviano, se podría desarrollar un control de admisión específico para el tráfico de vídeo aplicado en redes con diferenciación de servicios y en particular para servicios de *streaming,* como es el caso de UMTS o DiffServ.

CAPÍTULO 5

Asignación dinámica de recursos de red

El tráfico de vídeo con calidad de imagen constante presenta una gran variabilidad en su intensidad y un alto grado de correlación. Las fluctuaciones del tráfico son debidas principalmente a dos causas. La primera, tiene como origen los diferentes modos de codificación de las imágenes debido a las técnicas de codificación de vídeo propias de los estándares H.26x y MPEG. La segunda es debida a los diversos grados de complejidad de las secuencias a codificar. La primera de ellas provoca fluctuaciones periódicas con una duración del orden de milisegundos, y la segunda provoca variaciones no periódicas de larga duración cuya duración temporal es del orden de segundos.

Durante los últimos años se han propuesto diversas técnicas de suavizado a nivel de imagen para resolver los problemas derivados de las fluctuaciones periódicas de corta duración [dlCAM97][dlCAM98]. Sin embargo, debido a la variabilidad de las escenas a codificar, el tráfico de vídeo con calidad de imagen constante sigue presentando un comportamiento variable dado que las técnicas de suavizado aplicables no pueden absorber las fluctuaciones que lo caracterizan. Por consiguiente, la ubicación de recursos en la redes de conmutación de paquetes para este tipo de tráfico es un reto que en la actualidad suscita gran interés en redes basadas en IP con calidad de servicio. La primera aproximación realizada por los proveedores de este tipo de servicios es ubicar la cantidad de recursos necesarios para los instantes en los que el tráfico de vídeo presenta una mayor actividad (QoS). Este sobredimensionamiento provoca una baja explotación de los recursos de red ya que en los momentos de actividad media o baja los requerimientos son muy inferiores. Según las consideraciones realizadas hasta el momento sobre las variaciones de actividad de las secuencias de vídeo y la posibilidad de caracterizar su comportamiento a través de modelos bidimensionales MMFP, se propone el empleo de técnicas de renegociación que faciliten la asignación dinámica de recursos en redes IP como solución efectiva para reducir los recursos utilizados en la transmisión de vídeo.

5.1. Introducción

Las técnicas de suavizado como la propuesta en [dlCAM97], son útiles para extraer la variabilidad producida en el tráfico de vídeo por el algoritmo de codificación MPEG, quedando aún presente la producida por la distinta complejidad de los diferentes cuadros y escenas dentro de la secuencia completa. El tráfico resultante continúa teniendo características de gran variabilidad, mostrando periodos de actividad elevada, media y baja. Para mantener la calidad de servicio al nivel deseado, se deberá contratar con la red una conexión que permita la

correcta transmisión durante esos periodos de actividad elevada. Como consecuencia, durante los periodos menos activos, se mantendrán reservados unos recursos de red que no serán necesarios. En general, la variabilidad de la tasa generada por un codificador de vídeo puede ser debida a múltiples factores, entre ellos se pueden citar los siguientes [RRH96]:

- Complejidad variada de las escenas.
- Utilización por parte del usuario de prestaciones añadidas, como el avance rápido o la pausa en un servicio VoD.
- Necesidades instantáneas de mayor resolución.

Así, los mecanismos que incorporen la transmisión de servicios de vídeo deberán ser lo más flexibles posible, adaptándose a los requisitos variables que van a presentar tanto usuarios como servicios y operadores de red. Con objeto de conseguir la citada flexibilidad se puede utilizar un sistema de asignación dinámica de recursos en las transmisiones de vídeo. De este modo, en los periodos de actividad media o baja, se podrán ceder los recursos a la red para que sean utilizados por otras conexiones. Por otro lado, en instantes de actividad elevada, se solicitará una cantidad mayor de recursos con objeto de mantener la calidad de servicio requerida [RRH96], [Ada96], [RRR95].

La determinación de los instantes en los cuales se demanda la variación del nivel de los recursos contratados se conoce con el nombre de instantes de renegociación y la elección de dichos instantes es determinante para maximizar el uso de recursos de la red. Los instantes de renegociación dependen directamente del perfil de tráfico que se desea transmitir. Por ello, es necesario analizar la secuencia a transmitir, determinando los instantes de renegociación óptimos y los parámetros que determinan los recursos necesarios en los intervalos de tiempo entre los instantes de renegociación [WJW+01]. Este proceso se conoce con el nombre de segmentación del tráfico de vídeo. En la literatura se exponen diversos métodos de segmentación ([KRK00], [SRS03]). La idea principal de estos algoritmos es buscar intervalos temporales (segmentos) en los que el tráfico de vídeo presente unas características similares que determinan los recursos de red necesarios para ser transmitidos. Como alternativa, se pueden emplear técnicas cuyo propósito es gestionar el ritmo de transmisión de la secuencia codificada para que el receptor siempre disponga de información para presentar. Estas técnicas se basan en utilizar un buffer en el receptor el cual se carga previamente a la presentación de las imágenes por lo que se denominan técnicas *"work-ahead"*. Estas técnicas definen también unos instantes de renegociación pero su determinación depende de la previsión realizada sobre el contenido de buffer y no del grado de complejidad de la imagen en cuadros consecutivos de la secuencia. Los instantes de renegociación en las técnicas *work-ahead* se basan en incrementos de la tasa de transmisión cuando se considera que el buffer de recepción se vacía o, por el contrario, reducciones en la tasa cuando llega a su nivel máximo de ocupación permitido [APMdlC02].

5.2. Servicios con asignación dinámica de recursos

En este apartado se analizan propuestas de servicio con asignación dinámica, en los cuales es el usuario el que decide llevar a cabo la renegociación de los parámetros descriptores de su conexión. Los tipos de servicio con renegociación dinámica y su nomenclatura vienen del entorno de ATM, pero pueden ser aplicados a otros tipos de redes sin pérdida de generalidad. Del mismo modo que para servicios con asignación estática de recursos existe la posibilidad

entre solicitar servicios con tasa constante (CBR) o tasa variable (VBR), se puede llevar a cabo una distinción entre los servicios con asignación dinámica. Así, la renegociación puede implementarse sobre servicios a tasa constante o a tasa variable, dando lugar a los siguientes servicios:

- Renegociación de tasa binaria constante (*Renegotiated CBR*, RCBR)
- Renegociación de tasa binaria variable (*Renegotiated VBR*, RVBR)

5.2.1. Renegociación de tasa binaria constante

De entre todos los mecanismos de asignación dinámica de recursos, el RCBR ([GKT97], [SZKT98]) es el más sencillo que puede implementarse. Mediante RCBR, una fuente puede renegociar su tasa de servicio mediante el envío de un mensaje de señalización en el que se solicita el incremento o decremento de la tasa actual.

Si la renegociación es admitida por la red, la fuente puede enviar datos a una nueva tasa constante manteniendo el nivel de calidad adecuado al servicio ofrecido. En caso contrario, la fuente deberá adaptarse a la tasa que tenía disponible antes de la solicitud, lo que puede llevar a retardos inaceptables en el servicio. Una posible solución es la de solicitar a la fuente la disminución de la tasa de transmisión. Esta solución es posible en la transmisión de vídeo ya que se puede aumentar el paso de cuantificación en el codificador de vídeo para obtener un flujo con una tasa menor, a pesar de degradar la calidad de la imagen codificada. Este mecanismo puede ser viable dentro de unos ciertos márgenes ya que es preferible degradar la calidad de la imagen transmitida que la recepción de la secuencia de vídeo con retardos elevados lo que puede llevar a una presentación visual no sincronizada de dicha secuencia.

Dentro del ámbito de la transmisión de vídeo se ha publicado diversos trabajos relacionados con RCBR. En [Ada98] propone una estrategia que permite la predicción del ancho de banda necesario para la transmisión del siguiente GoP con codificación VBR. A partir de esta predicción, el autor determina una política de renegociación RCBR utilizando la tasa de pico del próximo GoP a transmitir.

5.2.2. Renegociación de tasa binaria variable

La segunda solución para servicios con renegociación dinámica de recursos solicitada por el usuario consiste en la renegociación a tasa variable. Esta modalidad ha sido utilizada en trabajos como [RRH96] que se basa en la renegociación de los parámetros descriptores de tráfico para una conexión clásica VBR, es decir, la tasa de pico (λ_p), tolerancia de ráfaga (*Burst Tolerance*, BT) y tasa sostenida (λ_s). La aplicación es para un sistema MPEG VBR, calculando los parámetros necesarios cuadro a cuadro (tomando como base el GoP) y solicitando la renegociación en función de la información almacenada en el buffer local.

En los últimos años se han publicado varios trabajos en los que se proponen la modificación del codificador de vídeo MPEG con el objeto de que la tasa de salida del codificador esté controlada por alguna función específica para poder adaptarse a los recursos asignados en la red. En [LS03] se propone una función de realimentación del codificador de vídeo MPEG para obtener una tasa de salida acotada que se ajuste a un descriptor de tráfico TSpec, en redes IP con arquitectura IntServ, sin un perjuicio notable en la calidad de la imagen. Otros trabajos en esta dirección son los presentados por [KRK00] y [YH01] en los que se utiliza el concepto de C-VBR (*Constrained VBR*) el cual determina una conexión ATM con una tasa de pico (λ_p),

una tolerancia de ráfaga (*Burst Tolerance*, BT) y una tasa sostenida (λ_s) a la que se debe adaptar las características de salida del codificador de vídeo MPEG. En [CCLS02], los autores proponen un entorno de trabajo analítico que permita evaluar el comportamiento del sistema de codificación y transmisión para distintas funciones de realimentación.

5.2.3. Implementación del mecanismo de renegociación

Uno de los problemas más importantes a solucionar en la asignación dinámica de recursos es cuándo deben realizarse las renegociaciones. En el apartado 5.2, se ha destacado la importancia del proceso de segmentación del tráfico de vídeo. En este caso se puede proponer un algoritmo que calcule los instantes de renegociación óptimos para toda la secuencia. Como pasa habitualmente en el mundo de la ingeniería, el cálculo de los valores óptimos de renegociación está sujeto al siguiente compromiso: cuanto mayor sea el número de puntos de renegociación, mejor será el aprovechamiento de los recursos de la red por parte del flujo analizado; sin embargo, el exceso de señalización para requerir esos recursos puede sobrecargar la red de forma inaceptable.

Otro punto importante es el tiempo necesario que empleará la red para ubicar la cantidad de recursos demandados. En redes IP con arquitectura IntServ donde se utiliza como protocolo de señalización RSVP, es importante recordar que la demanda de recursos parte del cliente del flujo. En este caso, la fuente debe enviar un mensaje de PATH con los nuevos requisitos de QoS. Dicho mensaje debe pasar y ser procesado por cada uno de los nodos intermedios y finalmente el receptor realizará la nueva reserva enviando un mensaje RESV hacia la fuente. El tiempo necesario para efectuar todo este proceso variará en función de los nodos intermedios entre ambos extremos y determinará el intervalo de tiempo mínimo entre renegociaciones.

Cuando se trata de trabajar con fuentes de vídeo en tiempo real, es decir, sin la posibilidad de trabajar sobre una información almacenada de antemano, el problema es distinto. Las diferentes soluciones propuestas para la determinación de los intervalos de renegociación son en gran medida heurísticas, como el caso de [RRH96] o bien pasan por utilizar predictores con un cierto horizonte que permitan un cierto margen de maniobra como en el caso de [Ada98].

5.3. Segmentación basada en técnicas *work-ahead*

En servicios VBR, la solución más ampliamente adoptada para mantener la calidad de servicio requerida por el cliente es contratar con la red una disponibilidad de recursos que permita la correcta transmisión de los periodos de más elevada tasa. Esta solución conlleva una disminución considerable de la eficiencia de la conexión ya que, durante los periodos de menor actividad, se están reservando unos recursos que en realidad no son necesarios.

Diversos estudios han propuesto los servicios con asignación dinámica de recursos como solución para incrementar la eficiencia total en la transmisión de una secuencia de vídeo. De esta forma, en los momentos de baja actividad será posible llevar a cabo una liberación de recursos que podrán ser empleados por otras conexiones mientras que cuando la actividad sea más alta, se solicitarán mayores recursos con objeto de mantener la calidad de servicio deseada.

Del mismo modo que para servicios con asignación estática de recursos existe la posibilidad entre solicitar conexiones a tasa constante o variable (CBR o VBR), se puede llevar a cabo una distinción entre los servicios con asignación dinámica. Así, la renegociación podría implementarse sobre conexiones a tasa constante o a tasa variable.

De entre todos los mecanismos de asignación dinámica de recursos, el RCBR, o renegociación de conexiones a tasa constante, es el más sencillo que puede implementarse. Mediante RCBR, una fuente puede renegociar su tasa de servicio mediante el envío de un mensaje de señalización en el que se solicita el incremento o el decremento de la tasa actual. Si la renegociación es admitida por la red, la fuente puede enviar datos a una nueva tasa constante. En caso contrario ésta deberá adaptarse a la tasa que tiene disponible o la información excedente se perderá produciéndose una degradación de la calidad de servicio.

Los dos parámetros más importantes a la hora de realizar un servicio RCBR son los instantes de renegociación y los niveles de tasa solicitada. Las políticas seguidas para determinar dichos parámetros dependen en gran medida del tipo de servicio que se quiera prestar. Si lo que se quiere es transmitir una secuencia de vídeo previamente almacenada, se puede calcular un programa de renegociaciones óptimo a priori. Si por contra lo que se quiere es trabajar con codificación en tiempo real se deberá hallar mecanismos heurísticos, que permitan decidir el nivel de contrato con la red únicamente con la información disponible en tiempo real.

En [SZ98] se propone una técnica de suavizado óptima para un tamaño de buffer de cliente determinado. El sistema propuesto aprovecha el conocimiento del vídeo a transmitir para calcular la manera óptima de entregar la secuencia a la red mediante tramos a tasa constante. Se aprovecha el buffer del cliente para almacenar datos enviados "por adelantado", es decir, que aun no se necesitan para decodificar el vídeo, pero que se van a necesitar al cabo de un cierto intervalo de tiempo. Este tipo de técnicas recibe el nombre de *Work-ahead buffering.*

El algoritmo *work-ahead optimal smoothing* aunque es la forma más suave de entregar el tráfico a la red, y configurable para un tamaño de cliente dado, nos lleva a soluciones con excesivas renegociaciones de tasa y un número no determinado de niveles de renegociación.

En el presente trabajo se hace una propuesta que pretende establecer un mecanismo simple y eficiente de entrega de tráfico de vídeo almacenado a la red mediante intervalos a tasa constante. En los servicios de vídeo bajo demanda las secuencias son conocidas a priori. Parece lógico aprovechar este hecho para realizar una entrega inteligente a la red. Por otro lado, se debe controlar que el estudio de la secuencia de vídeo no derive en una complejidad de cálculo excesiva. Parece razonable buscar un sistema que encuentre un compromiso entre eficiencia y simplicidad de cálculo. Este es el objetivo del mecanismo 2-RCBR.

5.3.1. Modelo matemático

Tal como se define en [SC00], un servicio de vídeo bajo demanda debe garantizar que no se produzca ni inanición ni saturación de buffers. Se debe procurar que en todo momento se tenga información suficiente para decodificar y a su vez se disponga de espacio de memoria suficiente para guardar la que no se decodificará de inmediato.

A continuación se presentará el modelo matemático que nos servirá para tratar estas condiciones de una manera formal.

Definamos la secuencia de vídeo como una función discreta en tiempo y amplitud $v(n)$, donde n indica el índice del cuadro codificado pudiendo tomar valores desde 1 hasta la longitud de la secuencia en número de cuadros (N). Esta función tiene como imágenes el número de bits necesarios para codificar el cuadro de índice n.

Sea *V(n)* la función que representa el total de bits necesarios para codificar todos los cuadros desde 1 hasta n. Esta función se puede calcular a partir de *v(n)* tal como se expresa en (5.1).

$$V(n) = \sum_{i=1}^{n} v(i) \qquad n = 1 \ldots N \tag{5.1}$$

Por otra parte, definamos *c(n)* como el contrato en bps con la red para cada tiempo de cuadro y *C(n)* como el total contratado con la red desde el instante inicial hasta el tiempo de cuadro n. De igual forma que en (1) puede definirse *C(n)* como la función acumulativa de *c(n)* como indica la expresión (5.2).

$$C(n) = \sum_{i=1}^{n} c(i) \qquad n = 1 \ldots N \tag{5.2}$$

Finalmente definamos *e(n)* como el tráfico que realmente se entrega a la red en un determinado tiempo de cuadro n y que depende de la técnica de transmisión empleada. Similarmente a lo realizado con *v(n)* y *c(n)*, definiremos *E(n)* como el tráfico total entregado a la red desde el instante inicial hasta el tiempo de cuadro n, véase expresión (5.3).

$$E(n) = \sum_{i=1}^{n} e(i) \qquad n = 1 \ldots N \tag{5.3}$$

Si no se utilizan técnicas de *buffering*, el tráfico entregado a la red es VBR puro. Este hecho se puede expresar tal como se indica en (5.4).

$$e(n) = v(n) \tag{5.4}$$

En este caso, las restricciones de no violación del contrato y no inanición de buffer se pueden formalizar según la expresión (5.5).

$$c(n) \geq e(n) = v(n) \quad \forall n \tag{5.5}$$

Es decir, en todo momento se debe estar contratando como mínimo igual cantidad de recursos a la red que los requeridos instantáneamente por la secuencia de vídeo y en todo momento el tráfico entregado debe ser superior al requerido para decodificar la secuencia, que en este caso particular, al no realizarse técnicas de *buffering*, coinciden. Esta restricción suele siempre cumplirse en exceso y ello conlleva decrementos drásticos de la eficiencia.

Cuando se usan técnicas de *buffering* el tráfico entregado a la red no tiene porqué coincidir con el codificado instantáneamente o con el necesario en decodificación en un determinado momento. La condición de uso de *buffering* queda reflejada en (5.6). En este tipo de mecanismos, el estudio de las funciones acumulativas nos lleva a resultados mucho más fáciles de entender.

$$e(n) \neq v(n) \tag{5.6}$$

Si la técnica de *buffering* se realiza en emisión, como es el caso típico de todas las técnicas de conformación de tráfico, el tráfico total acumulado entregado a la red es menor que el total codificado en un cierto instante de tiempo (5.17).

$$E(n) \leq V(n) \tag{5.7}$$

Esta diferencia se traduce en un retardo de decodificación en recepción (D expresado en tiempos de cuadro). Además se debe controlar el tamaño del buffer en emisión (que denominaremos B) para que no se produzca saturación de buffer. Esta condición se explicita en (5.8).

$$V(n) - E(n) < B \tag{5.8}$$

En este caso la restricción de no inanición de buffer se puede expresar como que el total de datos entregados a la red debe ser siempre igual o superior a los totales decodificados en un cierto instante de cuadro menos el retardo introducido por el buffer de emisión, véase expresión (5.9).

$$E(n) > V(n - D) \tag{5.9}$$

Si la técnica de *buffering* se realiza en recepción, como en el caso de todas las técnicas *work-ahead*, el tráfico total acumulado entregado a la red es mayor que el total decodificado en un cierto instante de tiempo, tal como se expresa en (5.10).

$$E(n) \geq V(n) \tag{5.10}$$

Esta diferencia, aunque también nos lleva a un control del tamaño del buffer, expresión (5.11), produce un retardo nulo. En este caso, como el retardo es nulo, la no inanición de buffer se expresa igual que en (5.10) y se deriva de forma natural del propio mecanismo de *work-ahead buffering.*

$$E(n) - V(n) \leq B \tag{5.11}$$

En cualquiera de los dos casos la expresión (5.12) define la eficiencia, calculada como el cociente entre el total entregado a la red y el total contratado al operador.

$$\eta = \frac{E(n)}{C(n)} \tag{5.12}$$

Para una mayor comprensión de todo lo expuesto, en la figura 5.1. se representan las condiciones discutidas.

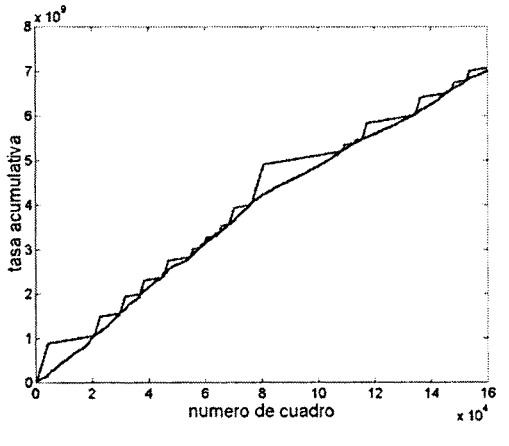


Figura 5.1: Esquema gráfico de las restricciones a cumplir por el sistema propuesto

5.3.2. Número de niveles de contrato.

Tal como se ha expuesto, los dos parámetros más importantes a la hora de realizar un servicio RCBR son los niveles de contrato, tanto su número como su valor, y los puntos de renegociación.

En un sistema *off-line*, la relación entre la tasa codificada y la tasa entregada a la red no tiene sentido, ya que la secuencia es conocida a priori y consecuentemente se pueden aprovechar los recursos de memoria en recepción para "romper" dicha dependencia entre tasas codificada y entregada. Este hecho también se exploró en [SZ98] pero el propio sistema resultaba en un número indeterminado de niveles de contrato, dependientes de la secuencia codificada y los recursos de memoria del cliente y de valor no determinístico. El objetivo que se pretende es encontrar el número mínimo de niveles fijo y de valor determinado que garantice una correcta entrega a la red del vídeo codificado y que cumpla todas las restricciones expuestas anteriormente.

De la observación de la figura 5.1 se deduce que nuestro problema se podría reformular como la aproximación de una función matemática *V(n)* mediante un número no determinado de tramos lineales de distinta pendiente de forma que la distancia entre la función original y su aproximación (*E(n)=C(n)*) sea mínima. Por otra parte, también queremos que el número de pendientes sea el mínimo posible, y siempre bajo las restricciones expuestas inicialmente.

Al estudiar la naturaleza de la función *V(n)* se concluye que se trata de una función monótona creciente con pasos de concavidad a convexidad (y viceversa) producidos por los distintos ritmos de crecimiento que presenta a lo largo de su dominio. El comportamiento de esta función se asemeja al de la función error, en la figura 5.2 se observa que si se pretende aproximar una función de este estilo mediante un solo tramo (una única pendiente) se viola la restricción de inanición de buffer (caso 1) o disminuye la eficiencia final (caso 2). El número mínimo de tramos con el que podemos aproximar una función que siga el comportamiento de *V(n)* cumpliendo las condiciones expuestas es 2. Por tanto el número mínimo de tasas que debemos renegociar con el operador para servir una secuencia de vídeo de este estilo será 2. Nuestro problema ahora se traduce en hallar cuáles son esos dos valores.

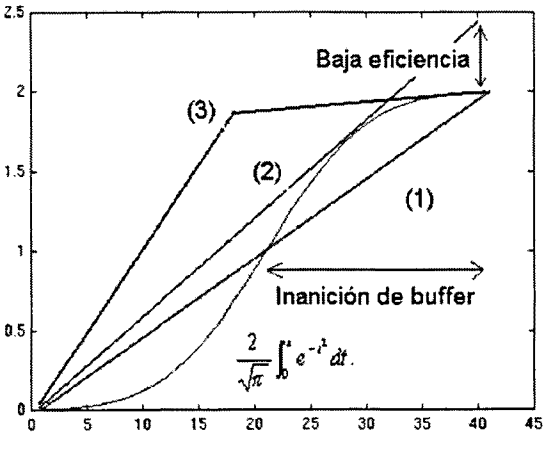


Figura 5.2: Modelado de la función V(n) como función de error

5.3.3. Cálculo de los niveles del contrato

Una vez ya hemos decidido que aproximaremos la función *V(n)* mediante intervalos lineales de dos posibles valores de pendiente debemos hallar cuáles son esos valores. Para ello deberemos estudiar la función *V(n)* con un poco más de detalle. Las expresiones (5.13) y (5.14) se deducen de la misma definición de *V(n)*.

$$V(n+1) - V(n) \leq Max\{v(n)\} = tasa\,max \tag{5.13}$$

$$V(n) - V(n-1) \geq Min\{v(n)\} = tasa\,min \tag{5.14}$$

Para que el mecanismo de renegociación sea lo más general posible, consideremos que cualquier punto de la secuencia *V(n)* puede convertirse en un punto de renegociación. Existirán dos tipos de puntos de renegociación, puntos de buffer máximo y puntos de buffer mínimo. Los primeros se corresponden con la renegociación de nivel alto a nivel bajo mientras que los segundos se corresponden con la renegociación de nivel bajo a nivel alto. Los primeros no son críticos en cuanto a restricción de inanición de buffer ya que precisamente dichos puntos son los que requieren tamaños de buffer mayores. Los puntos de renegociación de nivel bajo a nivel alto, en cambio, sí que son críticos ya que por definición del algoritmo son los que tienen un tamaño de buffer requerido nulo, y por tanto la tasa servida en instantes precedentes o posteriores a dicho punto nos puede llevar al incumplimiento de las restricciones temporales.

Supongamos cualquier punto con índice *i* de la secuencia *V(n)* como punto de renegociación de segundo tipo. Índices precedentes a dicho punto se corresponden con índices pertenecientes a un intervalo de tasa contratada baja, y similarmente, índices posteriores al citado punto se corresponden con índices de un intervalo de tasa contratada alta, ecuaciones (5.15) y (5.16).

$$E(i) - E(i-1) = tasa\,baja \tag{5.15}$$

$$E(i+1) - E(i) = tasa\,alta \tag{5.16}$$

Como habíamos comentado, en estos puntos el tamaño del buffer es nulo (*E(i)=V(i)*). Como además debemos cumplir (5.10) para cualquier valor de n, se pueden plantear las inecuaciones (5.17) y (5.18) de las que se deduce que la tasa alta debe ser la tasa máxima de la secuencia codificada de vídeo y la tasa baja la tasa mínima de la misma secuencia.

$$E(i+1) - E(i) \geq V(i+1) - V(i) \leq tasa\,max \tag{5.17}$$

$$E(i) - E(i-1) \geq V(i) - V(i-1) \leq tasa\,min \tag{5.18}$$

5.3.4. Cálculo de los puntos de renegociación.

Por último sólo nos falta decidir cuáles serán los puntos en los que vamos a renegociar tasas con la red. Recordemos que nuestro objetivo es aproximar la función *V(n)* mediante un número aún indeterminado de tramos lineales de dos posibles pendientes de valores ya determinados. Ya comentamos que de la propia definición de *V(n)* se deduce que es una función monótona creciente con distintos ritmos de crecimiento en su recorrido. Estos cambios en el ritmo de crecimiento producen cambios de concavidad a convexidad y viceversa en la función *V(n)*. Estudiando con detalle dichos pasos de concavidad a convexidad se puede observar que son causados por cambios de complejidad en la codificación de la imagen de vídeo, o lo que es lo mismo, por cambios de escena. Cambios de escena significativos producirán puntos de inflexión pronunciados mientras que cambios de escena no significativos producirán puntos de inflexión poco pronunciados.

Sea *f(t)* una función continua y derivable en todo su dominio. Se definen los puntos de inflexión como los puntos en que la función experimenta un cambio en su sentido de concavidad. Por teorema de continuidad se puede demostrar que estos puntos presentan un valor de la segunda derivada nulo. A su vez, podemos clasificar los puntos de inflexión en dos grupos: fuertes y débiles dependiendo de la pendiente que presente la función *f(t)* en dichos puntos.

Nótese que según el umbral escogido la función *f(t)* presentará más o menos puntos de inflexión fuertes. De este modo se pueden usar los puntos de inflexión como indicativos de los puntos de renegociación de tasa y podremos controlar el número de renegociaciones mediante la variación del umbral de definición de punto de inflexión fuerte.

Para entender mejor la aplicación a nuestro problema de dichas afirmaciones consideraremos continua y derivable la función *V(n)*. Nótese que este cambio no afecta en la generalidad de las afirmaciones expuestas.

Sea *v(t)* la función continua y derivable que describe la cantidad de bits necesarios para codificar una secuencia de vídeo en un determinado instante de tiempo *t*.

Se define *V(t)* como la función continua y derivable que describe la cantidad acumulada total de bits requeridos para codificar una secuencia de vídeo hasta el instante *t*. Dicha función puede expresarse tal como se indica en (5.19), y consecuentemente se puede definir *v(t)* en función de *V(t)* tal como se refleja en (5.20). Finalmente en (5.21) se detalla la relación existente entre la secuencia acumulativa, *V(t)*, la secuencia patrón correspondiente a la codificación del vídeo, *v(t)*, y la función que indica el ritmo de crecimiento y decrecimiento de la secuencia patrón,$\dot{v}(t)$.

$$V(t) = \int_0^t v(s)ds \tag{5.19}$$

$$v(t) = \frac{dV(t)}{dt} \tag{5.20}$$

$$\dot{v}(t) = \frac{dv(t)}{dt} = \frac{d^2V(t)}{dt} \tag{5.21}$$

De estas expresiones se deduce que el cálculo de los puntos de renegociación se reduce a hallar los puntos en que $\dot{v}(t)$ se anula, y se correspondan con un valor de *v(t)* mayor que un determinado umbral. Dichos puntos nos proporcionarán una noción de dónde deben existir intervalos de renegociación, ya que estos puntos dividen la secuencia completa en intervalos que presentan un comportamiento aproximado a un punto de inflexión como el de la figura 5.2. De todos modos, debe decidirse cuáles serán los puntos exactos de renegociación, o lo que es lo mismo, cómo aproximamos ese único punto de inflexión fuerte mediante un par de intervalos lineales.

Por una parte sabemos que debemos empezar a servir el tráfico a la red a tasa máxima. Del cálculo de los puntos de inflexión fuertes sabemos que la función integrativa presentará un salto pronunciado alrededor de dicho punto. Por diseño del sistema sabemos que debemos acabar el servicio en ese intervalo a una tasa mínima y con un tamaño de buffer nulo (*V(n)=E(n)*). Debemos encontrar un punto donde la función acumulativa vuelva a tener ritmos de crecimiento suaves. Ya hemos discutido ampliamente que dichos ritmos de crecimiento se corresponden con los valores de su derivada, es decir, con los valores de la secuencia patrón. Se considerará el punto final del intervalo de inflexión aquel en que el ritmo de crecimiento vuelve a su valor medio. El punto intermedio de renegociación se halla de forma natural de la intersección de los dos tramos lineales.

5.3.5. Algoritmo práctico de cálculo de los puntos de renegociación.

A continuación se presenta una forma práctica de calcular los puntos de renegociación a partir de la secuencia de vídeo conocida. Recordemos que en realidad las funciones de las que disponemos no son continuas, y que inicialmente sólo disponemos de los datos que nos proporciona v(n). Consecuentemente deberemos plantear un algoritmo que se base únicamente en datos conocidos.

1. Calcúlese *V(n)* a partir de *v(n)* según la expresión (5.1) para tantos puntos como longitud tenga *v(n)*.

2. Calcúlese $\dot{v}(n)$ a partir de *v(n)* según la expresión $\dot{v}(n) = v(n) - v(n-1)$ para n tomando valores de *2* a N y con valor inicial nulo, $\dot{v}(1) = 0$.

3. Encuéntrese los puntos de inflexión fuertes como los que cumplan la condición $\dot{v}(n) \geq factor \cdot std\{v(n)\}$.

4. Encuéntrese los puntos inmediatamente posteriores al índice n de los puntos de inflexión fuertes que cumplan $v(n) \leq mean\{v(n)\}$.

5. Hállese los puntos de renegociación pares, según la expresión (5.22).

Todos los puntos calculados en el ítem 4 del algoritmo, junto con *n=1* formarán parte del subconjunto de puntos de renegociación correspondientes a tamaño de buffer mínimo. Este subconjunto se puede expresar como un vector de índices ya que se corresponden con los puntos de renegociación impares. Consideramos siempre el primer elemento de los vectores con índice igual a *1*.

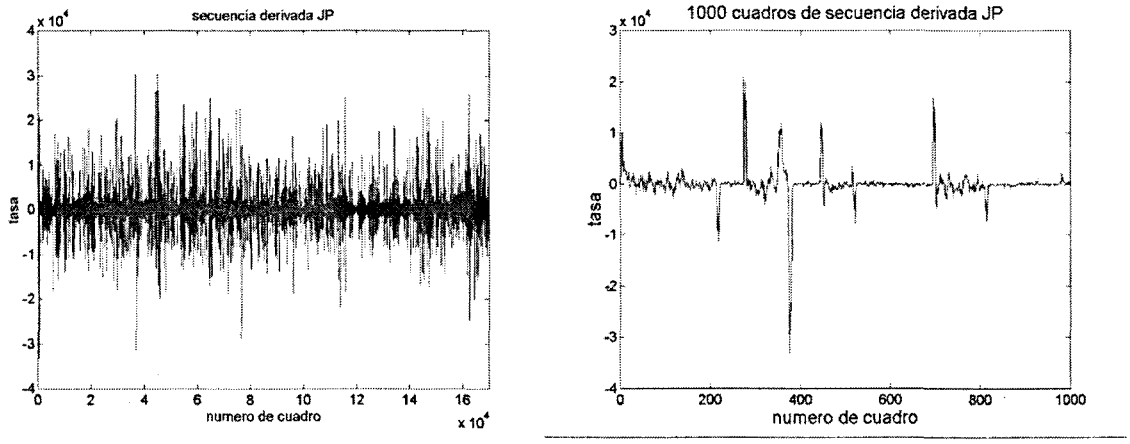
Los tramos comprendidos de índices impares a pares se corresponden con tramos de tasa entregada máxima, los tramos comprendidos de índices pares a impares se corresponden con tramos de tasa entregada mínima.

$$\vec{p}(2n) = \frac{\{V(\vec{p}(2n+1) - V(\vec{p}(2n-1))\} - [tasa\_media\{v\} \cdot (\vec{p}(2n+1) - \vec{p}(2n-1))]}{tasa\_maxima\{v\} - tasa\_media\{v\}} \quad (5.22)$$

5.3.6. Justificación del algoritmo.

Aun quedan por justificar dos aproximaciones realizadas. En primer lugar, para el cálculo de los puntos de inflexión se toman aquellos que presentan un valor positivo y grande de la segunda secuencia $\dot{v}(n)$. Esto se justifica a partir de la observación de la propia secuencia $\dot{v}(n)$ en la figura 5.3.

Al no ser la función continua, los puntos en los que dicha función se anulan no se corresponden con índices naturales de la función sino a segmentos de extrapolación. Así, los puntos de inflexión no son valores nulos de la segunda derivada sino transiciones de valor negativo a positivo. Además existen numerosos de estos puntos que no indican cambios bruscos. Los puntos buscados son cambios de ritmo bruscos y positivos en la secuencia patrón. Dichos puntos son aquellos en que la primera derivada de la secuencia patrón toma valores positivos elevados, es decir, $\dot{v}(n) > umbral$.



(a) Secuencia entera

(b) 1000 primeras muestras

Figura 5.3: Secuencia incremental de $\dot{v}(n)$

5.3.7. Resultados obtenidos.

A continuación se presentan los resultados obtenidos de la simulación del algoritmo 2-RCBR sobre 4 secuencias de vídeo previamente conformadas. Las secuencias son: *Jurassic Park* y *Concert by America Band*, PAL y de 170000 y 34000 cuadros de longitud respectivamente, y *Blade Runner* y *Concert by Neil Young*, NTSC y de 156000 y 47000 cuadros cada una.

La tabla 5.1 muestra el número de renegociaciones para cada una de las secuencias en función de distintos umbrales de punto de inflexión fuerte. Se observa una reducción drástica cuando el umbral se toma alrededor de la desviación típica de la secuencia incremental. Este valor reducido en cuanto a número de renegociaciones conlleva un incremento de buffer en recepción, efecto no deseado, por lo que se debe hallar un compromiso entre número de renegociaciones y tamaño de buffer del cliente. Valores de umbral cercanos a 0.75 veces la desviación típica ofrecen del orden de 1 renegociación cada 2500 cuadros, es decir, una media de 1 renegociación cada minuto y medio. Umbrales cercanos a la mitad de la desviación típica ofrecen valores del orden de 1 renegociación cada 750 cuadros, o 30 segundos en media. Éstos a su vez, presentan un comportamiento de buffer de cliente siempre inferior a los 100 Mbytes.

	Std	$0{,}75 \cdot std$	$0{,}5 \cdot std$	$0{,}25 \cdot std$
J. Park	13	71	211	1078
Neil Y.	4	6	16	309
Blade R.	91	217	551	2019
America	5	23	104	541

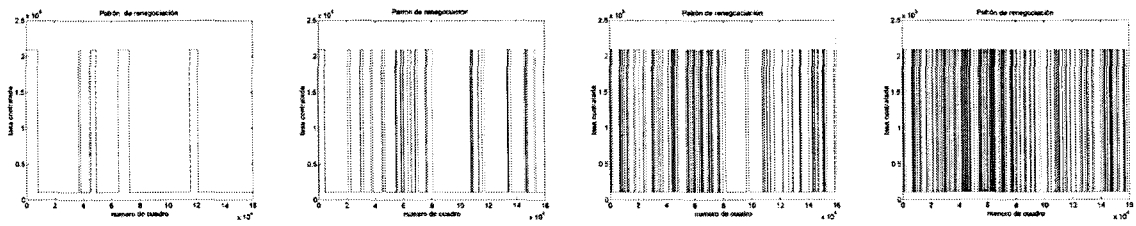
Tabla 5.1: Renegociaciones necesarias para distintas secuencias de vídeo y distintos umbrales

En la figura 5.4 se pueden ver los patrones de renegociación para distintos umbrales, mientras que en la figura 5.5 se observa lo precisa que es la aproximación mediante tramos lineales de

la función *V(n)*. Finalmente en la tabla 5.2 y la figura 5.6 se detallan los tamaños de buffer de cliente necesarios para no producirse saturación.

	Std	0,75 · *std*	0,5 · *std*	0,25 · *std*
J. Park	150 MB	90 MB	41 MB	14,5 MB
Neil Y.	42 MB	38 MB	12 MB	3,5 MB
Blade R.	230 MB	114 MB	67 MB	22,5 MB
America	23 MB	12 MB	8 MB	2,5 MB

Tabla 5.2: Tamaños de buffer para distintas secuencias y umbrales



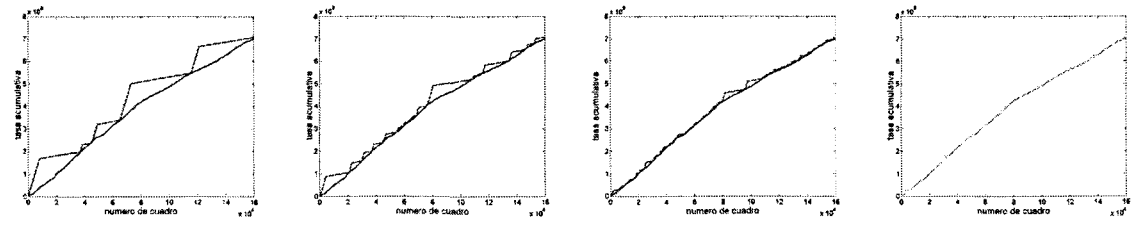
(a) umbral std (b) umbral 0,75std (c) umbral 0,5std (d) umbral 0,25std

Figura 5.4: Patrones de los intervalos de renegociación para la secuencia *Jurassic Park* suavizada con distintos umbrales de punto de inflexión fuerte

5.4. Segmentación basada en el modelo MMFP bidimensional

Otra de las técnicas empleadas para la segmentación de las secuencias de vídeo se basa en la identificación de conjuntos de imágenes consecutivas que presentan un nivel de complejidad semejante. Los algoritmos utilizados para la identificación de estos conjuntos de imágenes se fundamentan en el análisis del contenido de la imágenes, o en los bits resultantes de la codificación de las imágenes. El primer tipo de algoritmos requieren de un alto coste computacional y se aplican durante el proceso de codificación con el fin de alcanzar mayores niveles de compresión [MPE99]. Para el proceso de transmisión y dado el alto coste computacional de estos algoritmos, en la práctica se recurre a un simple análisis del tamaño de las imágenes codificadas. En este trabajo se emplearán estos mecanismos analizando el valor del número de bits por GoP. Así, se considerará que un segmento está formado por grupos de GoPs consecutivos que presentan niveles de compresión semejantes. La segmentación resultante al aplicar estos algoritmos permite realizar una clasificación de los segmentos, agrupándolos en función de la tasa media requerida para ser transmitidos. La elección del número de clases definidas se ha realizado en otros trabajos de forma heurística. Por ejemplo en [Ros97] se propone utilizar un rango entre 7 y 10 clases diferentes. En [SRS03] se realiza una experimentación con 7 clases que resulta suficiente para derivar modelos escénicos markovianos de las secuencias de vídeo.

A diferencia de los trabajos mencionados donde se aplica un proceso de segmentación para derivar un modelo escénico markoviano, en el presente trabajo se parte de un modelo



(a) umbral std (b) umbral 0,75std (c) umbral 0,5std (d) umbral 0,25std

Figura 5.5: Aproximaciones mediante tramos lineales de la secuencia *Jurassic Park* con cálculo de intervalos de renegociación para distintos umbrales de punto de inflexión fuerte

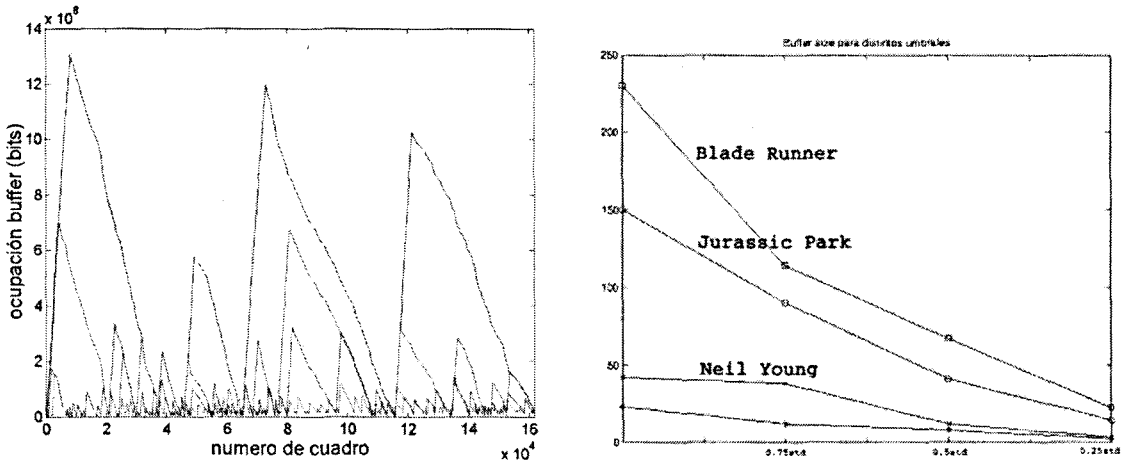
markoviano que define el proceso de segmentación a realizar, el cual será empleado para la transmisión de la secuencia de vídeo.

La estructura del modelo MMFP bidimensional desarrollado en el capítulo 4 permite identificar una clasificación de los segmentos de la secuencia. La estructura del modelo presenta tres grados de actividad asociados con la dimensión vertical. Por lo tanto, de forma natural quedan definidas tres clases de segmentos vinculados a los tres niveles de actividad. Estos segmentos quedan caracterizados estadísticamente a través del conjunto de estados que pertenecen al mismo nivel horizontal o, equivalentemente, al mismo nivel de actividad. Las tasas de generación de los estados limítrofes de un mismo nivel de actividad determinan los márgenes de variación dentro de un mismo segmento. Estos valores limítrofes determinan el inicio y final de cada segmento dentro de la secuencia. Tras el proceso de segmentación se puede realizar una clasificación en los niveles definidos en el capítulo 4, técnica que fue empleada para validar el ajuste final de los modelos obtenidos. La validación se fundamentó en el análisis estadístico de los segmentos clasificados dentro de un mismo nivel de actividad, teniendo en cuenta las tres posibilidades siguientes:

- Actividad regular
- Actividad elevada
- Gran actividad

La transición entre niveles de actividad simula los posibles cambios significativos de actividad que se producen en las secuencias de vídeo. En la figura 5.7 se muestra la segmentación realizada de 490 grupos de imagen (GoP) de la película *Las normas de la casa de la sidra.* En el margen izquierdo de la figura se detallan mediante flechas y números los umbrales de decisión así como las zonas de actividad. Mediante líneas de puntos verticales se marcan los puntos de renegociación.

El proceso de segmentación de una secuencia de vídeo se inicia ajustando el modelo MMFP bidimensional a la secuencia para obtener los umbrales de decisión que determinan los tres niveles de actividad. El siguiente paso es la determinación del nivel de actividad al que pertenece cada grupo de imágenes de la secuencia. Finalmente se realiza un filtrado, ya que debido a la gran variabilidad que sigue mostrando la secuencia de vídeo, aparecen segmentos en los



(a) Distintos umbrales

(b) Distintas secuencias

Figura 5.6: Tamaños de buffer

que solo hay un GoP o bien un número reducido de GoPs lo que obligaría a realizar una renegociación con la red para la transmisión de ese conjunto reducido de datos. Un ejemplo de este filtrado puede observarse en la figura 5.7; antes y después del GoP 12600 se observan dos picos que pasan por encima del umbral de decisión del nivel de Actividad Elevada (1) al de Gran Actividad (2), sin embargo, no se lleva a cabo el cambio de nivel de actividad. El proceso de filtrado determina el nivel de actividad en función de un tiempo de permanencia mínimo en el nuevo nivel. El valor del tiempo de permanencia mínimo se escoge de forma heurística. El tiempo mínimo de permanencia en el nivel de Actividad Regular es de 40 GoPs, el del nivel de Actividad Elevada es de 12 GoPs y en el de Gran Actividad es de 6 GoPs. La selección de diferentes tiempos de permanencia se debe a que las probabilidades de cada nivel son también diferentes; en general, utilizando este modelo, la probabilidad del nivel de Actividad Regular es del 80 %, el del nivel de Actividad Elevada es del 15 % y el de Gran Actividad es del 5 %. Parece lógico pensar que en función de esas probabilidades, existirán más transiciones aisladas al nivel de Actividad Regular que al nivel de Actividad Elevada y al nivel de Gran Actividad. El mismo razonamiento se aplica a los siguientes niveles de actividad.

Una vez realizada la segmentación de la secuencia de vídeo, se procede a calcular los descriptores de tráfico de cada segmento que se utilizarán para negociar los nuevos recursos con la red. En este caso y dado que este trabajo se enmarca dentro de redes IP con arquitectura IntServ, se calculan los parámetros del TSpec: la tasa media r, el tamaño del bucket b y la tasa de pico p.

5.4.1. Umbrales de segmentación y percentiles

Una pregunta que nos hacemos, es si existe alguna relación entre los umbrales de decisión obtenidos para diferentes factores de cuantificación de una misma secuencia. Para ello comparamos los umbrales de decisión de la secuencia *"el graduado"* con factores de cuantificación de 4, 6 y 8 (ver ajustes en el apéndice B) mostrados en la tabla 5.3

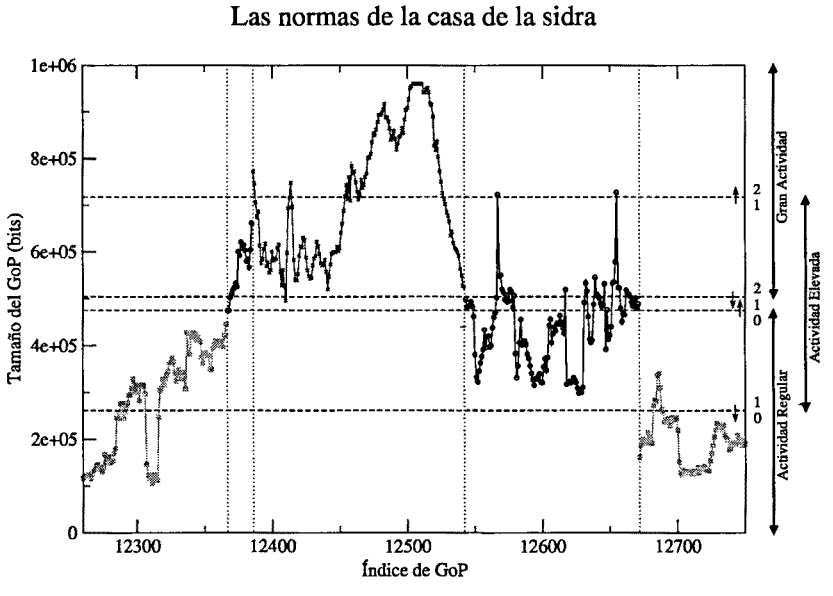


Figura 5.7: Segmentación de *Las Normas*

El graduado	U_{10}	U_{01}	U_{21}	U_{12}
Q4	266239	467378	512550	713689
Q6	198211	363433	376495	541717
Q8	141362	237133	262795	358566

Tabla 5.3: Percentiles de la secuencia de *el graduado* para valores de cuantificación 4, 6 y 8

Los valores de los umbrales son muy diferentes porque el valor máximo de cada codificación también son diferentes. Para poder compararlos de forma efectiva, normalizamos los umbrales de cada codificación respecto al valor máximo obtenido en cada secuencia codificada. El valor máximo para Q4 es 960000 bits/GoP, el de Q6 es de 720000 bits/GoP y el de Q8 es de 480000 bits/GoP y los valores obtenidos están mostrados en la tabla 5.4

El graduado	U_{10}	U_{01}	U_{21}	U_{12}
Q4	0,2773323	0,4868521	0,5339063	0,7434260
Q6	0,2752931	0,5047681	0,5229097	0,7523847
Q8	0,2945042	0,4940271	0,5474896	0,7470125

Tabla 5.4: Percentiles normalizados de la secuencia *el graduado* para valores de cuantificación 4, 6 y 8

Los valores tan próximos de los umbrales de decisión para cada una de las secuencias codificadas indican que la función de probabilidad acumulada debe tener una forma muy parecida, salvo por un factor de escalado en el eje x. De forma ilustrativa representamos las funciones de probabilidad acumulada de las tres secuencias, normalizando el eje x . El resultado obtenido era previsible dado que las funciones de probabilidad acumulada se obtienen de

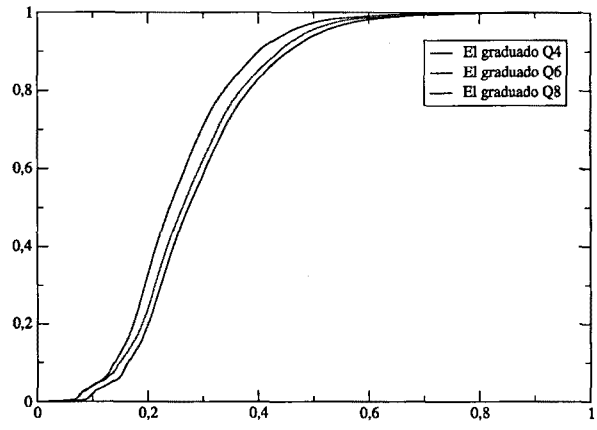


Figura 5.8: Funciones de probabilidad acumulada del la secuencia *"el graduado"* para factores de cuantificación 4, 6 y 8. (Eje x normalizado a 1)

la misma secuencia donde se han aplicado diferentes codificaciones que introducen distintos niveles de degradación. Por lo tanto, las variaciones de complejidad dependen de la secuencia y no de la degradación introducida en el proceso de codificación.

Extendemos el estudio realizado a la secuencia *"City of Angels"*, cuyos resultados se presentan en la tabla 5.5

City of Angels	U_{10}	U_{01}	U_{21}	U_{12}
Q4	0,2753781	0,4889948	0,5308802	0,7444969
Q6	0,3130083	0,4269847	0,5995153	0,7134917
Q8	0,2986375	0,4822250	0,5575250	0,7411125

Tabla 5.5: Percentiles normalizados de la secuencia *City of Angels* para valores de cuantificación 4, 6 y 8

Y en la figura 5.9 se muestran las funciones de probabilidad acumulada de las tres codificaciones. Nuevamente, se observa la propiedad de invarianza descrita. Asimismo, se puede comprobar que el aspecto ofrecido por las curvas de probabilidad acumulada es similar en ambas secuencias. Este aspecto se caracteriza por un rápido ascenso entre valores normalizados de bits por GoP que van de 0,1 a 0,4. Tras estos valores, la curva reduce sustancialmente su pendiente y suavemente se aproxima a su máximo. Después de este análisis, se sugiere la experimentación con varias secuencias con el fin de realizar un estudio comparativo de los valores de los umbrales normalizados de cada una de ellas. Los valores obtenidos para un conjunto de cuatro secuencias y para un paso de cuantificación de 4, se presentan en la tabla 5.6

En la figura 5.10 se muestran las funciones de densidad acumuladas de las cuatro secuencias con el eje de abscisas normalizado. Como se observa, las distintas secuencias tienen valores muy próximos lo que nos lleva a concluir que existen unos valores invariantes de los umbrales normalizados.

Finalmente, para determinar la genericidad de estos valores se analiza la invarianza de estos umbrales normalizados respecto a la resolución de la secuencia codificada. A priori, la distribución de la complejidad será la misma para distintas codificaciones, por lo que cabe

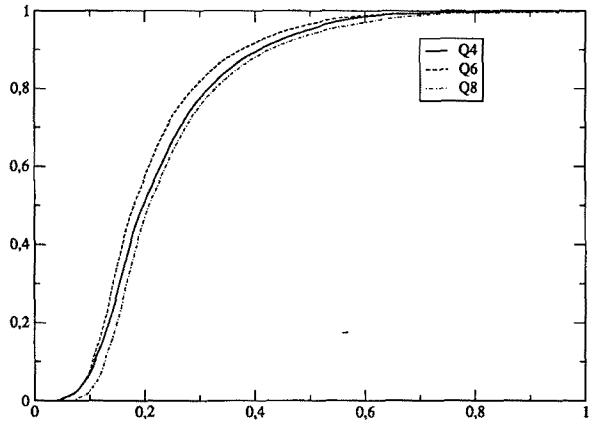


Figura 5.9: Funciones de probabilidad acumulada del la secuencia *"City of Angels"* para factores de cuantificación 4, 6 y 8. (Eje x normalizado a 1)

Q4	U_{10}	U_{01}	U_{21}	U_{12}
El graduado	0,2773323	0,4868521	0,5339063	0,7434260
City of Angels	0,2753781	0,4889948	0,5308802	0,7444969
Las Normas	0,2782510	0,4832479	0,5366271	0,7416240
La Boda	0,2724990	0,4947521	0,5251229	0,7473760

Tabla 5.6: Comparación de percentiles

esperar nuevamente que la codificación aplicada no influya sustancialmente en los resultados esperados. En la tabla 5.7 se muestran los resultados obtenidos para la codificación de una secuencia con distintas resoluciones, corroborando la hipótesis realizada.

Los Angeles Confidencial (Q6)	U_{10}	U_{01}	U_{21}	U_{12}
352x288	0,2642359	0,5198730	0,5042994	0,7599365
640x288	0,2621972	0,5545597	0,4849181	0,7772806
720x320	0,2711458	0,5296906	0,5063010	0,7599365

Tabla 5.7: Percentiles de la secuencia *Los Angeles Confidential* para diversas resoluciones y valor de cuantificación 6

En el estudio realizado se han desarrollado tres análisis ortogonales que han demostrando la existencia de un conjunto invariante de umbrales normalizados para las secuencias de vídeo. Se ha observado que los valores de estos umbrales normalizados, derivados del modelo bidimensional propuesto en el capítulo anterior, mantienen un valor similar para diversas secuencias de vídeo y no se ven alterados por el nivel de degradación introducido en la codificación ni por la resolución empleada. La genericidad de estos valores nos facilita la propuesta de una nueva técnica de segmentación de secuencias de vídeo basada en estos umbrales normalizados.

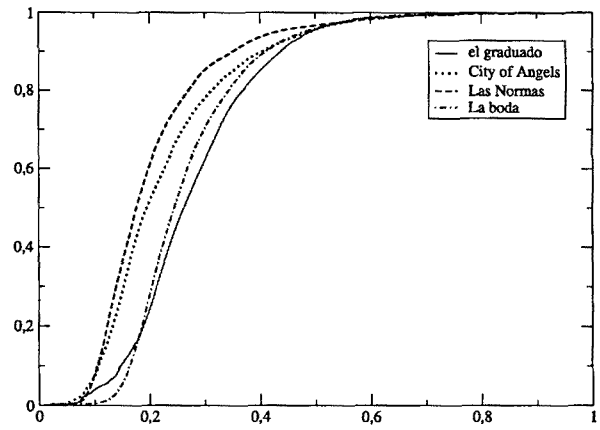


Figura 5.10: Funciones de densidad acumulada con el eje de abscisas normalizado a 1

Para la elección de los valores genéricos de los percentiles de segmentación y dada su reducida dispersión, se ha considerado válida la elección del valor medio de los resultados de la tabla 5.6, obteniéndose los siguientes percentiles:

U_{10}	U_{01}	U_{21}	U_{12}
0,2758651	0,4884617	0,5316341	0,7442307

Un análisis realizado de sensibilidad de los umbrales de segmentación demostró que la variación del umbral U_{01} es el que tiene una mayor incidencia sobre el proceso de segmentación. Dado que se está proponiendo utilizar unos umbrales genéricos en este proceso de segmentación, es interesante determinar el error introducido por el uso de los umbrales genéricos frente a los umbrales obtenidos a través del proceso de modelización. En la tabla 5.8 se muestra el error obtenido para varias secuencias. Las cuatro primeras secuencias corresponden a las utilizadas para el cálculo de los percentiles genéricos y como puede observarse el error máximo cometido no excede del 1,3 %. Para el resto de secuencias, se puede destacar que el error cometido en el cálculo de los percentiles de segmentación es pequeño y es básicamente la secuencia *Embrujo* la que muestra una mayor desviación respecto a sus umbrales propios. Sin embargo, aunque el error cometido al usar los percentiles genéricos es pequeño, no nos indica cuál será el impacto sobre el proceso de segmentación. Por ello se ha realizado un estudio comparativo del proceso de segmentación utilizando los percentiles obtenidos mediante el proceso de modelización y los percentiles genéricos para el mismo conjunto de secuencias anterior. Los resultados de esta comparación se presentan en la tabla 5.9, donde el parámetro de eficiencia mostrado en dicha tabla, se introduce en el punto 5.4.2, sin embargo la interpretación de la bondad de la aproximación es sencilla, ya que si con ambos conjuntos de umbrales se consigue el mismo número de segmentos y la misma eficiencia, nos indica que ambos procesos de segmentación dan como resultado exactamente el mismo conjunto de segmentos. Finalmente, puede observarse que tanto para la secuencia *Embrujo* como para *Grease* se produce la mayor diferencia en el número de segmentos obtenido mediante ambos conjuntos de percentiles, y ambas secuencias presentan los valores mayores de error en el percentil U_{01} corroborando que este percentil presenta la mayor sensibilidad respecto al proceso de segmentación.

Q4	$\varepsilon(U_{10})$	$\varepsilon(U_{01})$	$\varepsilon(U_{21})$	$\varepsilon(U_{12})$
El graduado	0,53 %	0,33 %	0,42 %	0,11 %
City of Angels	0,17 %	0,11 %	0,14 %	0,04 %
Las normas	0,85 %	1,01 %	0,93 %	0,35 %
La boda	1,23 %	1,27 %	1,23 %	0,42 %
Harts War	2,72 %	2,83 %	2,79 %	0,94 %
Embrujo	5,71 %	5,99 %	6,08 %	2,05 %
Medianoche	1,89 %	1,96 %	1,91 %	0,65 %
Empalme_A[1]	0,19 %	0,31 %	0,25 %	0,10 %
Flores de Otro Mundo	1,00 %	0,94 %	0,86 %	0,31 %
Grease	1,51 %	5,24 %	2,99 %	1,66 %

Tabla 5.8: Errores de los umbrales de segmentación

Q4	N° segmentos	Eficiencia	N° segmentos	Eficiencia
	Umbrales del modelo		Umbrales genéricos	
El graduado	81	0,786850	81	0,786004
City of Angels	81	0,673034	81	0,673061
Las Normas	60	0,634831	62	0,638005
La boda	89	0,778784	91	0,785160
Harts War	41	0,513575	41	0,539784
Embrujo	138	0,784134	152	0,803508
Medianoche	168	0,818856	170	0,820979
Empalme_A	418	0,690402	415	0,690442
Flores de Otro Mundo	93	0,794171	95	0,795138
Grease	179	0,833676	158	0,815946

Tabla 5.9: Comparación de la segmentación utilizando los umbrales propios y los umbrales genéricos

5.4.2. Eficiencia y retardo máximo de la segmentación

Para poder comparar los resultados obtenidos a la hora de transmitir una secuencia de vídeo utilizando un cierto esquema de renegociación es necesario definir algunas funciones matemáticas que cuantifiquen la bondad de dicho esquema. En sí mismo, el obtener unos ciertos puntos de renegociación a través de un algoritmo de segmentación no nos aporta información. La reserva de los recursos de red para la transmisión del conjunto de segmentos identificados nos indicará si los recursos de red reservados se adaptan de forma efectiva a la información transmitida. Es por ello que la definición de un parámetro de eficiencia vendrá dado por los recursos de red reservados frente al uso real de esos recursos a la hora de transmitir los datos. Una forma de obtener los recursos necesarios para la transmisión de un flujo de datos, es calcular los parámetros de una *token bucket* (r,b) que mejor se ajustan a esos datos. Con estas condiciones, los datos totales transmitidos en un cierto instante de tiempo T_o deben ser

[1]Empalme_A corresponde a la concatenación de las secuencias Flores de otro mundo, Hartswar, La boda, Las normas, El graduado y City of Angels.

menores que $r \cdot T_o + b$. Si $x(n)$ es el tráfico transmitido por unidad de tiempo, la cantidad de datos transmitidos después de un cierto tiempo T_o, que corresponde a un índice n_o, será $\sum_0^{n_o} x(n)$ y la eficiencia del uso de recursos se definirá como:

$$\eta = \frac{\sum_0^{n_o} x(n)}{r \cdot T_o + b}$$

La eficiencia calculada corresponde al número total de bits transmitidos dividido por el valor máximo de bits que la reserva realizada nos permitiría transmitir. En primera instancia, el valor de r se escoge como el valor medio de la tasa de transmisión de la secuencia de vídeo, $x(n)$, o del conjunto de GoPs que forman un segmento, $x_i(n)$; y para ese valor se calcula el tamaño máximo necesario de bucket (b). En la figura 5.11 se muestra la integral del tráfico

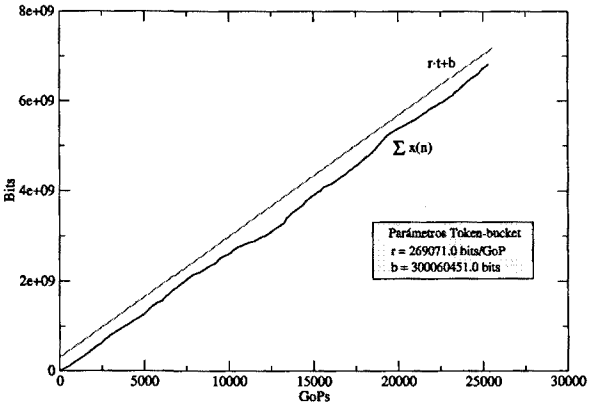


Figura 5.11: Integral del tráfico de la secuencia de "El graduado" y curva $r \cdot t + b$

de la secuencia de *"el graduado"* y se compara con el valor de la curva $r \cdot t + b$ donde r se ha calculado como el valor medio de la secuencia. En este caso los valores de r y de b obtenidos son $269071{,}0\,bits/GoP$ y $300060451{,}0\,bits$, con una eficiencia de

$$\eta = \frac{6814473076{,}0}{7114552597{,}0} = 0{,}957822$$

El problema que presentan estos valores es el retardo máximo alcanzado en cada nodo en servicios de carga controlada, o extremo a extremo, en un servicio garantizado, ya que si se hiciese una reserva con estos valores de r y b sería de $b/r = 1115{,}172\,GoPs$ que pasado a segundos corresponde a $267{,}64\,seg$. Este valor es inadmisible cuando hablamos de transmisión de vídeo ya que corresponde a casi 5 minutos de retardo máximo en el nodo. Para solucionar el problema del retardo, se puede incrementar el valor de la tasa media (r) y calcular el nuevo valor del bucket (b) de forma iterativa hasta alcanzar el retardo deseado. Para un retardo de 2,5 segundos los valores de r y b son $662239{,}0\,bits/GoP$ y $6824849{,}0\,bits$ que corresponde a un retardo de $2{,}473\,seg$ y un valor de eficiencia de

$$\eta = \frac{6814473076{,}0}{16778689763{,}0} = 0{,}406139$$

La figura 5.12 muestra la comparación de la integral del tráfico frente a la curva $r \cdot t + b$ con los nuevos valores obtenidos.

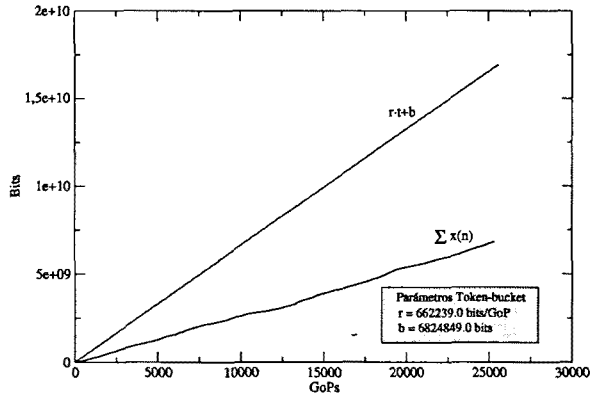


Figura 5.12: Integral del tráfico de la secuencia de "El graduado" y curva $r \cdot t + b$ con retardo máximo limitado a 2,5 segundos

Si se realiza una segmentación de una secuencia y se obtienen k segmentos, y para cada segmento se calculan los valores correspondientes de r_i y b_i con $i = 1 \ldots k$, el tráfico total que se puede transmitir será

$$\sum_{i=1}^{k} (r_i \cdot T_i + b_i)$$

donde T_i es la duración de la escena i-ésima, y la eficiencia se puede calcular de la siguiente forma

$$\eta = \frac{\sum_1^k (\sum_0^{n_k} x_i(n))}{\sum_1^k (r_i T_i + b_i)} = \frac{\sum x(n)}{\sum_1^k (r_i T_i + b_i)}$$

que en este caso corresponde al número total de bits transmitidos en toda la secuencia dividido por la suma del máximo número de bits que se pueden transmitir en cada una de las reservas correspondientes a cada segmento. Los valores que se obtienen para la secuencia de *"el graduado"* con 81 segmentos obtenidos del modelo MMFP ajustado es de una eficiencia del 94,02 % sin acotar el retardo máximo y del 81,6575 % con un retardo máximo de 2,5 segundos por segmento. Queda patente que la imposición de un retardo máximo por segmento influye decisivamente sobre la eficiencia en la explotación de los recursos. Para poder realizar una comparación de los retardos calculamos el retardo máximo ponderado de la segmentación de una secuencia de la siguiente manera:

$$D_{mp} = \sum_{i=1}^{k} D_i \frac{T_i}{T_{tot}} = \sum_{i=1}^{k} \frac{b_i}{r_i} \frac{T_i}{T_{tot}}$$

Que corresponde a la suma de los retardos máximos de cada segmento ponderados por la duración de los segmentos respecto a la duración total de la secuencia, en un único nodo de red. En la tabla 5.10 se muestran los valores de eficiencia y retardo máximo ponderado para la secuencia de *"el graduado"*, para un solo punto de renegociación (el segmento corresponde a toda la secuencia) y para 81 puntos de renegociación obtenidos a partir del modelo MMFP bidimensional y diversos valores para la cota del retardo máximo. Estos resultados revelan que si no se fija el retardo máximo, las eficiencias obtenidas en ambos casos son muy elevadas, sin embargo al fijar el retardo se pone de manifiesto los beneficios de la renegociación.

Retardo máx.	No acotado		2,5 *seg*		1,6 *seg*		1 *seg*	
N° segmentos	η	D_{mp}	η	D_{mp}	η	D_{mp}	η	D_{mp}
1	95,78 %	1115,172	40,61 %	10,306	38,98 %	6,099	35,74 %	4,094
81	94,02 %	85,662	81,66 %	9,122	78,68 %	5,928	75,21 %	3,738

Tabla 5.10: Comparación de eficiencia y retardo máximo ponderado de la secuencia *"el graduado"*

Llegados a este punto, nos preguntamos cuán bueno es este sistema de segmentación basado en el modelo MMFP. Para ello, compararemos los resultados obtenidos con otros esquemas de segmentación.

5.5. Otros esquemas de segmentación

5.5.1. Método del coeficiente de variación

En [Ros97], el autor propone un sistema de segmentación de una secuencia de vídeo basándose en el coeficiente de variación. Este mecanismo es semejante al empleado en el algoritmo desarrollado en 5.3.5. La idea principal de este algoritmo de segmentación es ir añadiendo GoPs a una secuencia hasta que el coeficiente de variación ponderado de los tamaños de los GoPs de la secuencia tenga un cambio mayor que un cierto ϵ. El último GoP añadido determina el inicio de la siguiente secuencia.

Realizando un barrido de los valores de ϵ se obtienen diferentes resultados de segmentación que representados en una gráfica permiten la observación de las variaciones de la eficiencia frente al número de renegociaciones. En las figuras 5.13, 5.14 y 5.15 se muestra el valor de la eficiencia frente al número de renegociaciones obtenido para un barrido de ϵ de 0,2 hasta 2,0 de la secuencia de *"el graduado"* con paso de cuantificación de 4 y el valor de retardo máximo limitado a 1, 1,6 y 2,5 segundos. En la mismas gráficas se destaca el valor obtenido por el modelo MMFP para los mismos parámetros y como puede observarse se mejora levemente la eficiencia de transmisión.

5.5.2. Método de los retardos

Los mecanismos de segmentación que se han presentado necesitan de una corrección cuando se acota el retardo máximo permitido, y este proceso de corrección lleva a un esquema de reserva de recursos más ineficiente (ver punto 5.4.2). El proceso de corrección se basa en incrementar la tasa (r) y recalcular el tamaño del buffer (b) de forma iterativa hasta que el cociente b/r, que representa el retardo máximo, disminuya hasta el valor deseado. El hecho de aumentar el valor de la tasa media lleva a que se produzca esta ineficiencia, como se puede observar comparando las figuras 5.11 y 5.12. Por lo que se acaba de exponer, parece lógico pensar en un algoritmo de segmentación que vaya añadiendo GoPs a una secuencia de GoPs consecutivos, buscando los parámetros TSpec de la secuencia, hasta que el retardo de la secuencia supere un cierto valor especificado. El último GoP añadido determina el inicio de la siguiente secuencia. El valor de la tasa (r) se corresponde con el valor medio de los GoPs de la secuencia, y el valor del tamaño del buffer (b) se ha establecido haciendo que la relación b/r esté limitada a un cierto valor máximo.

El algoritmo descrito permite establecer el retardo máximo adecuado para la transmisión de este tipo de tráfico, obteniéndose una mejora sensible en la eficiencia de los recursos utilizados ya que la tasa media de transmisión no debe ser modificada para ajustarse a los requerimientos de retardo, tal y como se realiza en el método del coeficiente de variación o utilizando los percentiles. En las figuras 5.13, 5.14 y 5.15 se muestra el valor de la eficiencia frente al número de renegociaciones obtenido, para un barrido en el retardo de 0,1 a 50 segundos, de la secuencia de *"el graduado"* con paso de cuantificación de 4 y el valor de retardo máximo limitado a 1, 1,6 y 2,5 segundos. Como se puede observar, mientras el retardo del barrido se mantiene por debajo de retardo máximo impuesto, la eficiencia en muy alta. Cuando el retardo del barrido es superior al retardo máximo, empieza a actuar el algoritmo de corrección de los parámetros de la *token bucket* disminuyendo la eficiencia.

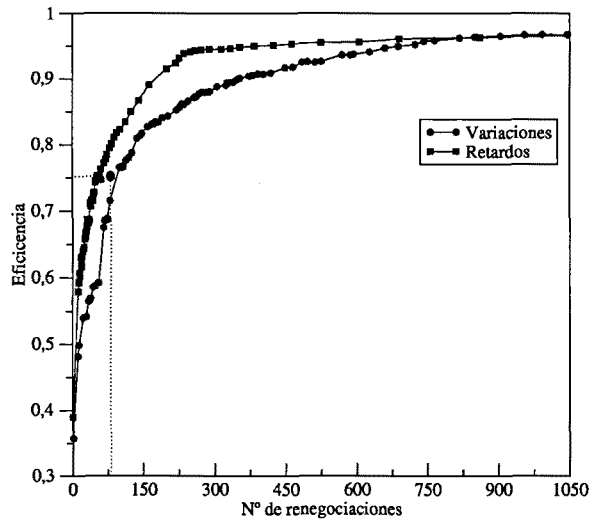


Figura 5.13: Eficiencia frente a renegociaciones de la secuencia de *"el graduado Q4"* con el retardo limitado a 1 segundo

5.6. Conclusiones

En este capítulo se ha realizado un análisis de diferentes métodos de segmentación para la transmisión de vídeo en redes IP con QoS y con mecanismos de renegociación y asignación dinámica de recursos. Basándonos en técnicas de *work-ahead buffering*, se ha desarrollado un algoritmo que utiliza dos tasas distintas para la transmisión de toda la sesión de vídeo. Los instantes de transición de tasa se han determinado a través de una técnica de segmentación basada en el ajuste del tráfico de vídeo por segmentos lineales de dos pendientes diferente que establecen las dos tasas resultantes. Los valores de tasa de transmisión elegidos buscan un compromiso entre el número de renegociaciones y el tamaño de buffer en el cliente. Este mecanismo es especialmente adecuado para sistemas con gran capacidad de almacenamiento primario, como estaciones de trabajo y set-top-box, y además presenta un nivel de explotación máximo de los recursos reservados.

Como alternativa, se ha derivado una nueva técnica de segmentación basada en el modelo MMFP bidimensional estudiado en el capítulo anterior. Esta técnica emplea los valores extre-

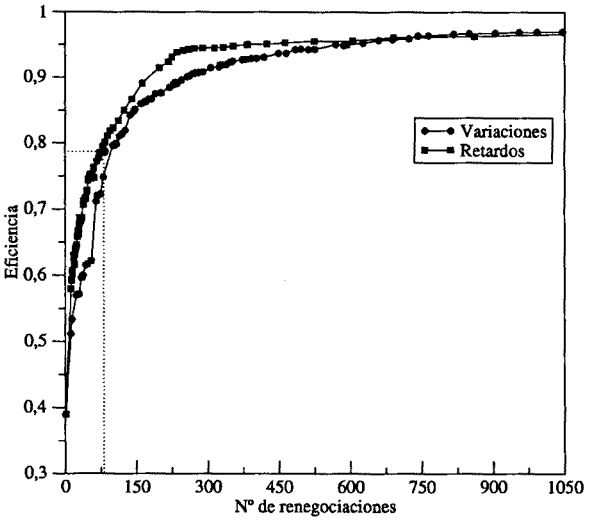


Figura 5.14: Eficiencia frente a renegociaciones de la secuencia de *"el graduado Q4"* con el retardo limitado a 1,6 segundos

mos de cada uno de los niveles de actividad identificados en la estructura bidimensional, para definir un conjunto de umbrales de decisión. Estos umbrales determinan un ciclo de histéresis que facilita la identificación de GoPs consecutivos del mismo segmento. Esta técnica se ha aplicado sobre distintas secuencias y se ha hallado que los umbrales normalizados de cada una de las secuencias presentan una fuerte similitud que nos lleva a definir un conjunto de umbrales invariantes para todas ellas. La independencia de los umbrales se constata tanto con el valor de resolución espacial como con el nivel de calidad de imagen utilizado en la codificación. Asimismo, este estudio se realiza sobre diferentes secuencias en las cuales no existe semejanza en la forma de las funciones de probabilidad acumulada. Sin embargo, los umbrales normalizados de decisión hallados en cada una de ellas siguen estando sumamente próximos, constatando nuevamente su independencia de la secuencia utilizada.

Los resultados obtenidos se han contrastado con los derivados del trabajo de O. ROSE, cuya propuesta de segmentación ha sido ampliamente referenciada en los trabajos relacionados. Este análisis ha revelado que el esquema de segmentación propuesto presenta un buen compromiso entre la eficiencia conseguida y la sobrecarga de señalización. Por otro lado, el tiempo medio de renegociaciones resultante se sitúa en el orden de operación definido en la especificación del protocolo de señalización RSVP. La especificación recomienda la generación de mensajes de PATH o RESV cada 20 segundos mientras que los tiempos medios de renegociación obtenidos están alrededor de esos 20 segundos.

Considerando la especificación dinámica de recursos que se realiza a lo largo de la transmisión y teniendo en cuenta que se desea fijar una cota máxima del retardo extremo a extremo durante el servicio de *video streaming*, se ha propuesto una nueva técnica de segmentación basada en el retardo aceptado. Esta técnica desarrolla un esquema de segmentación, el cual identifica los grupos de GoPs consecutivos que requieren de un determinado TSpec, garantizando el retardo especificado en una red que emplea como protocolo de señalización RSVP. Un análisis comparativo de esta técnica frente al algoritmo del coeficiente de variación revela un incremento de eficiencia notable para el mismo número de renegociaciones. Esta técnica de

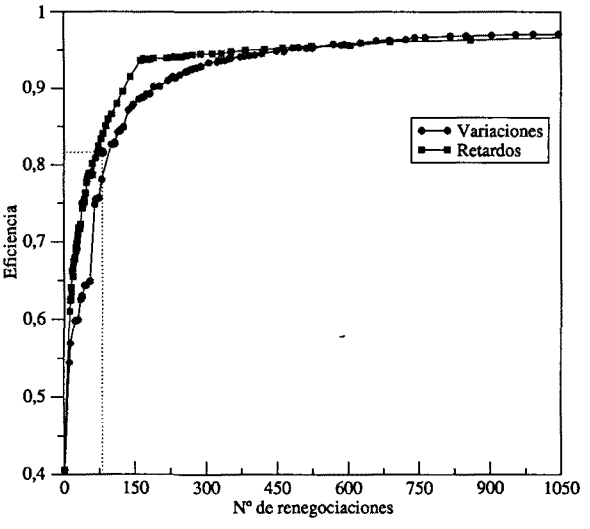


Figura 5.15: Eficiencia frente a renegociaciones de la secuencia de *"el graduado Q4"* con el retardo limitado a 2,5 segundos

segmentación permite observar que el punto de trabajo definido por el método de los umbrales normalizados es un valor adecuado para la operatividad del sistema con un elevado nivel de eficiencia.

Finalmente, de lo expuesto anteriormente, se puede concluir que el método de los umbrales genéricos permite desarrollar una segmentación de una secuencia de vídeo de forma extraordinariamente simple con un elevado nivel de eficiencia. El empleo de esta técnica de segmentación define un perfil del tráfico bien caracterizado y modelado. Este conocimiento del comportamiento estadístico del tráfico de vídeo facilita el desarrollo de técnicas específicas de control de admisión y de ubicación de recursos.