

Análisis de la calidad del streaming de vídeo a través de redes OBS

Autor: Campos Rodríguez, Javier

Director: Dr. Guerri Cebollada, Juan Carlos

Resumen: Las redes de conmutación de ráfagas sobre fibra óptica cobran fuerza como la mejor opción para las redes troncales en un futuro a corto medio plazo y, ante el auge de uso del *streaming* de vídeo, se hace necesario un análisis profundo de la calidad de la transmisión de vídeo en estas redes. Para ello se ha simulado el envío de tráfico de vídeo con un ancho de banda agregado suficiente para la transmisión de vídeo en alta definición o varios decenas a cientos de vídeos de baja calidad, a través de una red OBS la cual se simula con los parámetros más usuales en este tipo de redes. Se utilizan los tres protocolos de ensamblado más extendidos en las diversas investigaciones sobre este campo, es decir, ensamblado con límite de tiempo, con límite de tamaño y ensamblado híbrido siendo el objetivo analizar cuál es el que produce una degradación menor en el vídeo transmitido.

Abstract: The optical burst switching networks gain strength as the best option in short-medium term future for backbone networks, address the increase of the video streaming usage, a deep analysis of video transmission quality of these networks becomes necessary. For that purpose diverse video send has been simulated, with an aggregated bandwidth high enough for the transmission of High Definition Video or some dozens or hundreds of low quality video streaming through the OBS network, which is simulated with the more usual parameters in this networks. The three assembly protocols most extended in the researches about this issue are used, namely, time limited, size limited and hybrid assembly protocols, with the objective of analyze which is the one that produce less degradation in the transmitted video.

Autor: Campos Rodríguez, Javier, email: jacamrod@posgrado.upv.es

Director 1: Guerri Cebollada, Juan Carlos, email: jcguerri@dc.com.upv.es

Fecha de entrega: 09-12-09

INDICE

I. INTRODUCCIÓN	3
II. REDES ÓPTICAS DE CONMUTACIÓN DE RÁFAGAS (OBS)	3
II.1. INTRODUCCIÓN.....	3
II.2. ESTRUCTURA Y FUNCIONAMIENTO DE REDES OBS.....	4
III. ALGORITMOS DE ENSAMBLADO DE RÁFAGAS	8
III.1. INTRODUCCIÓN.....	8
III.2. ASPECTOS BÁSICOS DE LOS ALGORITMOS.....	9
IV. TRÁFICO MULTIMEDIA	11
IV.1. INTRODUCCIÓN.....	11
IV.2. PARÁMETROS DE CALIDAD.....	12
V. DESARROLLO DE LOS TEST DE TRANSFERENCIA DE VÍDEO	13
V.1. ENTORNO DE DESARROLLO.....	13
V.2. DESARROLLO DE LA TESINA.....	14
V.2.1. Red dedicada para la transferencia de tráfico multimedia.....	17
V.2.2. Red con tráfico self-similar de background.....	25
VI. CONCLUSIONES	37
VI.1. DESARROLLOS FUTUROS.....	38
Agradecimientos	38
Bibliografía	39
Anexos	41*

*Con numeración propia del artículo (empieza por 1)

I. INTRODUCCIÓN

Las redes ópticas de conmutación de ráfagas (OBS, *Optical Burst Switching*) están llamadas a ser las redes utilizadas para el *backbone* de Internet así como arquitectura alternativa para las redes metropolitanas. Por otro lado, debido al aumento de ancho de banda del que disponen los usuarios y a la comodidad de poder ver lo que quieras cuando quieras sin necesidad de tenerlo almacenado, cada día se tiende más al uso del *streaming* de vídeo y a su vez se está aumentando progresivamente la calidad de estos vídeos. Por otro lado algunas operadoras de telecomunicaciones han entrado en el negocio de la distribución de contenidos multimedia, en concreto televisión y han visto y explotado las posibilidades que ofrecen las redes IP. Debido a esto se ha decidido realizar en esta tesina un análisis para ver las posibilidades que ofrecen este tipo de redes y sus problemas intrínsecos esperando que sirva de referencia como punto de partida para implementar nuevas soluciones específicas para este tipo de redes.

Es posible encontrar diversos estudios relacionados con la transmisión de tráfico TCP sobre redes OBS [1], [2], [3], [4], su problemática y propuestas para solucionar dichos inconvenientes. Al igual que están bastante bien establecidos los parámetros a tener en cuenta en cualquier transferencia de vídeo independientemente de la red para evaluar su calidad [5], [6], [7], [8]. Pero no se ha encontrado ningún estudio en profundidad sobre la transferencia de vídeo sobre redes OBS y como las características de este tipo de redes afectan a la calidad de vídeo, el cual tiene unos requerimientos y características muy distintos a los que puede tener una transferencia TCP.

II. REDES ÓPTICAS DE CONMUTACIÓN DE RÁFAGAS (OBS)

II.1. INTRODUCCIÓN

Para hacer frente a la gran demanda de ancho de banda necesario para dar un servicio de calidad en las redes del futuro, y particularmente en internet, las investigaciones se han encaminado en la búsqueda de la implementación de una red completamente óptica. Para alcanzar este objetivo, destaca el uso de la tecnología de multiplexación por división de longitud de onda, WDM (*Wavelength Division Multiplexing*), como una tecnología que hace posible que el medio físico óptico, como lo es la fibra óptica, pueda ser utilizado para llevar la información de múltiples canales o longitudes de onda en una sola fibra proveyendo de un gran ancho banda a una fibra.

Sobre la base física de una red WDM se están estudiando diversas opciones para la transferencia de los datos como se puede ver en la “Tabla 1”. Entre ellas se encuentra la tecnología *Optical Circuit Switching* (OCS), relativamente fácil de implementar, siendo la usada actualmente, pero con poca flexibilidad respecto a las variaciones del tráfico y el cambio de estado del enlace. Otra opción es el *Optical Packet Switching* (OPS), el cual proporciona multiplexación estadística del tráfico a nivel de paquete en el *backbone*, y posiblemente sea la tecnología dominante en el

futuro, pero requiere ciertas tecnologías, como *buffers* ópticos, los cuales aun no están suficientemente desarrollados a día de hoy. También se puede encontrar una tercera opción, *Optical Burst Switching* (OBS) que combina las ventajas de OCS y OPS y parece la preferida por la industria y los investigadores.

CONMUTACIÓN ÓPTICA	USO DE ANCHO DE BANDA	LATENCIA	BUFFER ÓPTICO	PROC/SYNC OVERHEAD (POR UNIDAD DE DATOS)	ADAPTABILIDAD
CIRCUITOS (OCS)	BAJO	ALTA	NO REQUERIDO	BAJO	BAJA
RÁFAGAS (OBS)	ALTO	BAJA	NO REQUERIDO	BAJO	ALTA
PAQUETES (OPS)	ALTO	BAJA	REQUERIDO	ALTO	ALTA

Tabla 1. Características de los sistemas de conmutación óptica

OBS es la tecnología que será utilizada, a corto-medio plazo, para la transmisión de grandes volúmenes de tráfico, entre el que se encuentra de un modo destacado el tráfico multimedia. Por ello, se hace necesario estudiar su comportamiento frente a este tipo de tráfico y cuáles son las mejores opciones para la transferencia del mismo.

OBS reserva ancho de banda por adelantado a través de una señal de control fuera de banda, eliminando de este modo la necesidad de un *buffer* óptico o *Fiber Delay Lines* (FDL) para conseguir conmutar los datos, de un modo transparente, a través de la red. Y a diferencia de OCS, OBS proporciona multiplexación estadística a nivel de ráfaga en los *core router* sin tener que incurrir en el retardo del establecimiento del circuito.

II.2. ESTRUCTURA Y FUNCIONAMIENTO DE REDES OBS

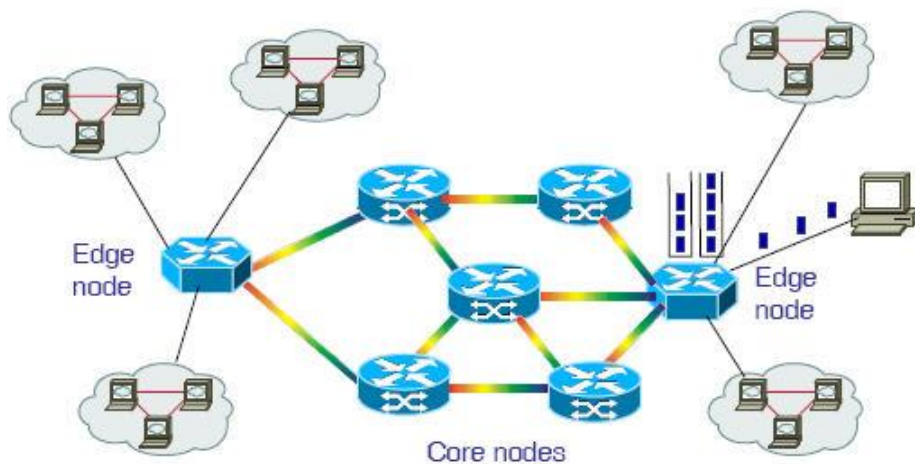


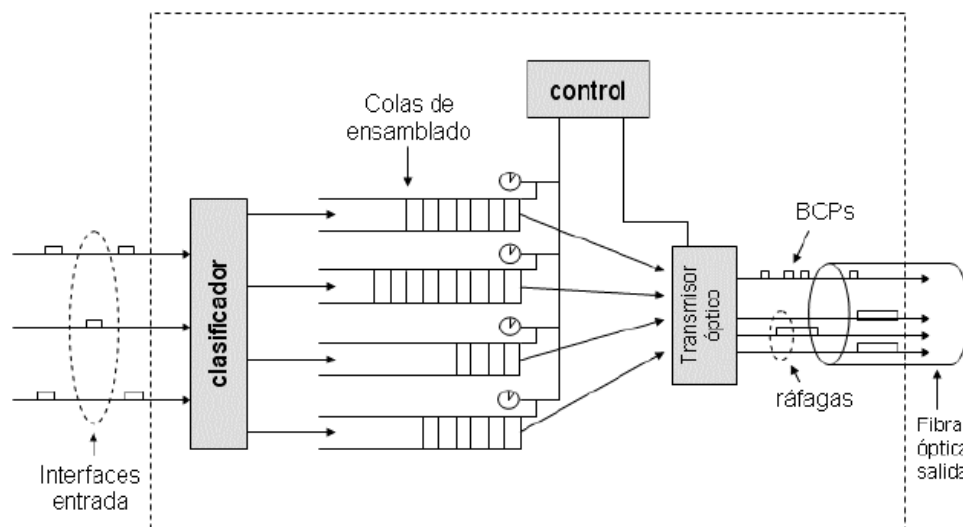
Fig. 1. Estructura de una red OBS

La topología de red considerada es una nube de *core router* ópticos, organizados como una red plana, con *edge router* en los límites de la nube los cuales son responsables de la función de ensamblado y desensamblado de las ráfagas. Los *core router* realizan la función de reenvío de ráfagas dentro de la nube hacia su *edge router* destino y todos estos nodos están interconectados mediante una red óptica WDM “Fig.1”.

La idea básica subyacente es que el *router* separe completamente el canal de datos y el canal de control y, de este modo, poder transmitir los datos de forma totalmente óptica, sin tener que almacenar ni convertir los datos de dominio óptico al dominio electrónico en ningún *core router*. Esto deberá ofrecer una velocidad muy alta, un interfaz directo con la fibra y una buena escalabilidad de la red ya que los subbloques de la red no dependen del bitrate del canal de datos.

La información de control debe viajar en una ventana temporal diferente y en un canal de longitud de onda distinto y debe ser manipulada independientemente de los datos.

Además, los paquetes de datos deben ser de una longitud mínima de varios kilobytes para rentabilizar el tiempo necesario para el procesado de la cabecera y la realización de las funciones de control, incluyendo la decisión de conmutación.



Esquema simple formador ráfagas

Fig. 2. Estructura del *edge router* OBS

En los *edge router*, “Fig. 2”, de la red troncal, es necesario implementar una función que transforme los paquetes IP a ráfagas *OBS* y viceversa.

Las entradas y salidas de los *core router* son fibras ópticas que operan en modo WDM con 16 o más longitudes de onda en orden de garantizar un alto grado de ganancia estadística de multiplexación. La tasa de transmisión por longitud de onda está comprendida en el rango desde 2,5 Gbps (OC48 - STM16) a 10 Gbps (OC192 - STM64) o incluso superior [9].

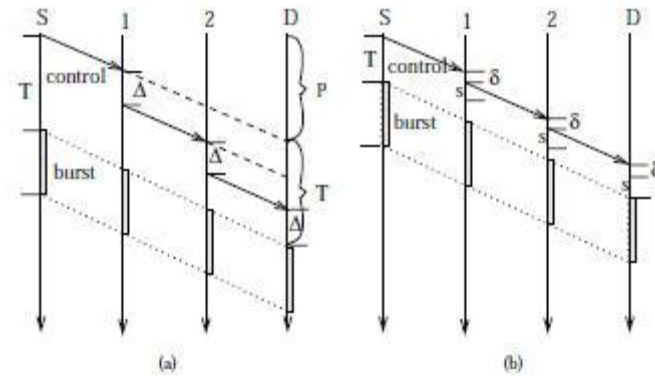


Fig. 3. Protocolo JET en OBS

Para la reserva de recursos, el protocolo más utilizado es el *Just-Enough-Time* (JET), el cual no requiere ningún tipo de almacenamiento óptico o retardo en cada *core router* [10]. Cada paquete de control lleva la información de tiempo de *offset* (T, tiempo entre el paquete de control y su correspondiente ráfaga) y hace una reserva (retardada) para la correspondiente ráfaga. La reserva comienza en el momento esperado de la llegada de la ráfaga, la cual espera en un *buffer* electrónico en el *edge router* origen hasta que se cumple el *offset*. Posibilita hacer una reserva con final acotado. La utilización del protocolo JET permite una reducción en el *offset* y por tanto en el retardo sufrido por los datos contenidos en la ráfaga ya que el cálculo del *offset* no es realizado tomando en cuenta todo el tiempo (Δ) de envío y procesamiento del paquete de control y de las acciones subsecuentes en los *core router* (“Fig. 3. (a)”), sino que sólo tiene en cuenta para su cálculo el tiempo (δ) necesario para el procesado y reenvío del paquete de control y de su procesamiento en el *edge router* destino, como se puede apreciar en la “Fig 3 (b)” dando por hecho que el tiempo (s) de procesar las acciones necesarias para la conmutación de la ráfaga en los *core router* será inferior al *offset* calculado.

El algoritmo de planificación LAUC - *Latest Available Unscheduled Channel algorithm* (Horizon) es el protocolo más utilizado en el estudio sobre redes OBS. En él, para cada longitud de onda, un único horizonte de planificación es mantenido [11]. Sólo los canales en los cuales el horizonte de planificación precede el tiempo de llegada de la nueva ráfaga son considerados disponibles y el que tenga el horizonte de planificación más cercano es escogido como se puede ver en la “Fig. 4”. El horizonte es entonces actualizado después de hacer la reserva para la siguiente ráfaga. La idea principal de este algoritmo es la de minimizar los espacios/vacíos de ancho de banda (gap/voids) creados como resultado de hacer una nueva reserva. Como inconveniente de este algoritmo señalar que desperdicia los espacios/vacíos entre dos reservas existentes, y se generarían muchos vacíos si se implementa FDLs o QoS basada en tiempos de *offset*.

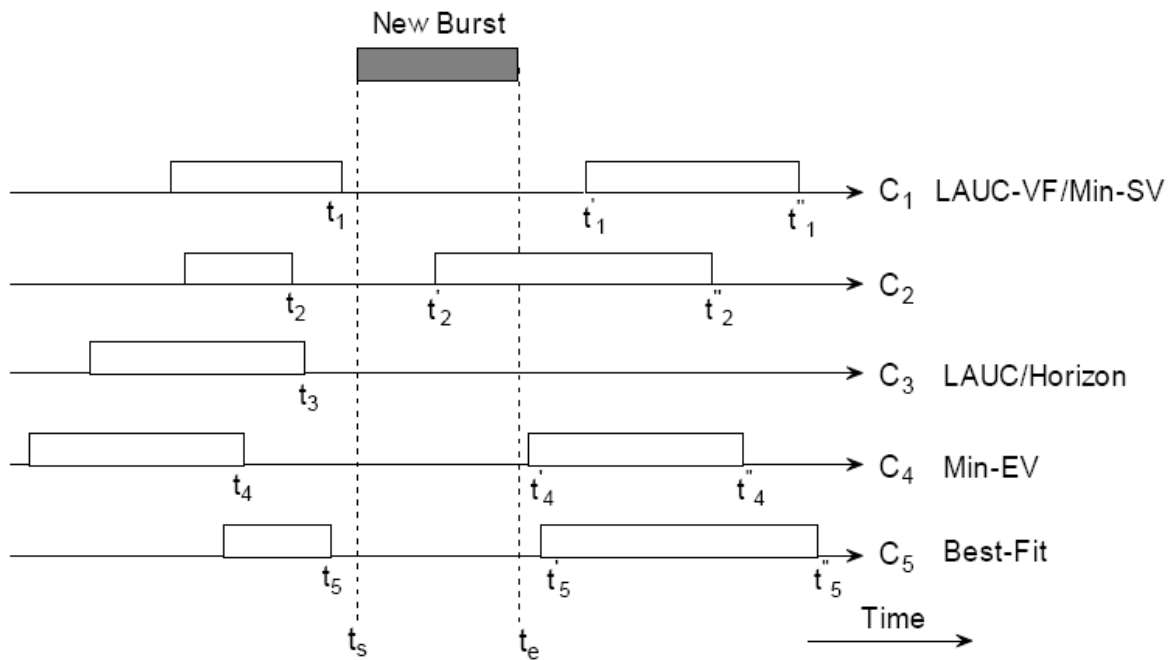


Fig. 4. Algoritmos de planificación

En caso de contienda en los *core router* se opta por la resolución de contiendas mediante la técnica de *Dropping* es decir, en caso de que cuando llegue una ráfaga el *core router* ya este procesando otra, la ráfaga entrante será desechada (*non-preemptive*).

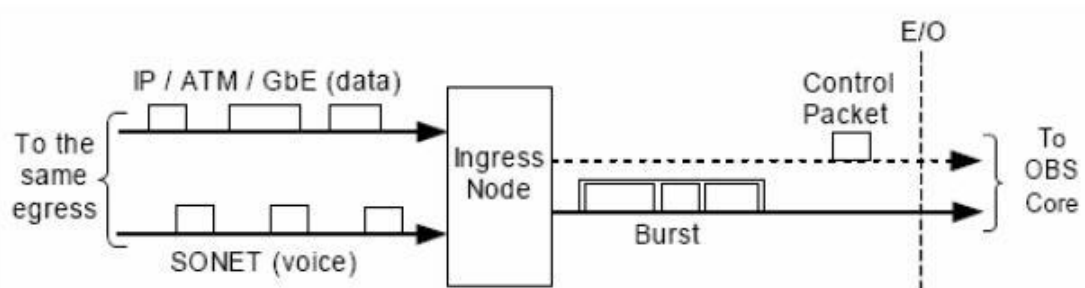


Fig. 5. Ensamblado de ráfagas

En los límites de la nube de conmutación de ráfagas, los *edge router* ensamblan ráfagas “Fig. 5” uniendo paquetes IP y etiquetando cada ráfaga con un *Burst Control Packet* (BCP) el cual contiene la información referente a la fuente, destino, *offset*, etc... En la arquitectura considerada, el BCP es transmitido en una longitud de onda y tasa de transmisión distintas a la utilizada por la ráfaga.

Para mantener la carga en el canal de control que transporta los BCPs, las ráfagas de datos deben ser mayores de cierto tamaño mínimo (B_{\min}). Por otra parte, reunir los paquetes suficientes para hacer una ráfaga de longitud B_{\min} tomará mucho tiempo si la carga en la línea de entrada es temporalmente muy baja. De este modo deberá imponerse un límite máximo de tiempo de espera de los paquetes para el ensamblado de la ráfaga (T_{\max}).

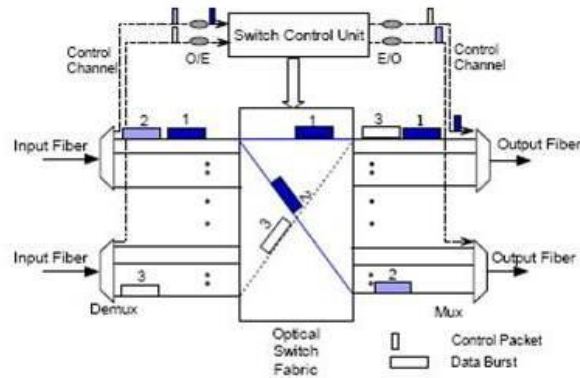


Fig. 6. Core router OBS

Dependiendo de la información en el BCP cada *core router* (“Fig. 6”) a lo largo de la ruta busca un periodo temporal disponible para la longitud de onda de salida deseada y hace una reserva para la ráfaga entrante (*burst scheduling*). De este modo, cuando la ráfaga llega al *core router* es inmediatamente conmutada al puerto de salida sin ningún retardo extra ni conversión óptico-eléctrico-óptica (O/E/O). La ráfaga es desensamblada en el *edge router* de salida y los distintos paquetes llegan a su destino a través de las redes locales.

Al ensamblarse muchos paquetes en una ráfaga, el tráfico deberá tener unas características estadísticas distintas a la del tráfico de los paquetes entrantes, lo que tendrá un impacto diferente en lo que respecta a la tasa de pérdidas, la cual es un aspecto crítico de las redes *OBS*.

III. ALGORITMOS DE ENSAMBLADO DE RÁFAGAS

III.1. INTRODUCCIÓN

Los algoritmos de ensamblado propuestos en la literatura se pueden clasificar en tres tipos: a) basados en tiempo [12], b) basados en longitud de ráfaga [13], c) mixtos basados en tiempo y longitud de ráfaga [14]. Cada uno de ellos presenta ventajas e inconvenientes por lo que al implementar alguno de ellos es necesario tener en cuenta que para valores de tiempo de ráfaga muy largos, el retardo de los paquetes en la frontera de la red puede llegar a ser intolerable, y valores de tiempo de ráfaga muy pequeños ocasiona la generación de muchas ráfagas pequeñas ocasionando sobrecarga de control, por otro lado, algoritmos basados en longitud de ráfagas no garantizan el retardo de ensamblado que experimentarán los paquetes.

En el intento de optimizar el funcionamiento de las redes *OBS*, han sido propuestos los algoritmos *adaptativos*, en estos, el umbral de tiempo, el umbral de longitud, o los dos, son ajustados dinámicamente de acuerdo a las medidas de tráfico de tiempo real. Estos algoritmos proporcionarían un mejor funcionamiento, especialmente con tráfico fuertemente relacionado, pero presentan una mayor complejidad operacional.

III.2. ASPECTOS BÁSICOS DE LOS ALGORITMOS

El **algoritmo basado en tiempo** es una propuesta de algoritmo de ensamblado para *OBS* pensado para reducir la auto-correlación y el grado de auto-similaridad del tráfico y mantener el retardo debido a la creación de ráfagas en los *edge router* de la red.

El algoritmo consiste en una sencilla serie de pasos:

- Se asocia una cola lógica con cada destino en los límites de la nube de conmutación de ráfagas ($[1:M]$)
- Se inicializa un temporizador T_i cada vez que llega un paquete dirigido a la cola i estando esta vacía.
- Cuando T_i alcanza el límite de tiempo T_{max} , se crea una ráfaga y se lanza para transmisión en el canal de datos: si la longitud de la ráfaga es menor de B_{min} , se rellena hasta B_{min} .
- Se resetea T_i a 0 y permanecerá así hasta la llegada del siguiente paquete a la cola i .

Este algoritmo es muy sencillo, de todos modos, cumple el requisito de tamaño mínimo de las ráfagas y pone una cota superior en el retardo experimentando por los paquetes debido al ensamblado de ráfagas.

Sus principales inconvenientes son que, en caso de baja carga en la línea, los paquetes se envían con muy poca información, pudiendo llegar el caso de estar enviando más bits de relleno que de datos. En el caso de carga alta, el tiempo de espera para el envío de los datos será sensiblemente superior que si se utiliza un algoritmo basado en longitud.

Además, el algoritmo de ensamblado de ráfagas conforma el tráfico, haciéndolo más aleatorio que los paquetes IP que componen la ráfaga, reduciendo su grado de auto-similaridad.

En estudios anteriores [15], se considera que el tráfico de internet tiene características fractales y puede ser descrito en términos de procesos estocásticos auto-similares. Esta característica supone una grave desventaja en términos de rendimiento de la cola. Simulaciones y análisis concluyen que el rendimiento de este tipo de tráfico es mucho peor que el obtenido con tráfico aleatorio.

Por lo tanto, esta reducción de la auto-similaridad de las ráfagas es una característica muy favorable, ya que supone que la red *backbone* de conmutación de ráfagas podrá asumir una congestión limitada, y podría ser suficiente con *buffers* pequeños para garantizar una buena calidad de servicio (*Quality of Service*, QOS).

El **algoritmo basado en tamaño** (*Threshold-based scheme*) introduce el parámetro de tamaño máximo de ráfaga (MBS) mediante el cual delimita el tamaño umbral para el envío de las ráfagas.

Su modo de funcionamiento es el siguiente, cuando llega un paquete de tamaño b en bytes destinado al puerto i , es dirigido a la cola i . Si esta cola está vacía $q[i]=0$, se actualiza el tamaño de ráfaga $q[i]=b$. Para que un paquete muy largo no tenga que esperar en la cola hasta que llegue el próximo paquete, se debe comprobar si el tamaño del paquete actual b excede el MBS. En caso

afirmativo, se generará una traza de tamaño b . Si la cola no está vacía cuando llegue el paquete, habrá que decidir si el tamaño de la ráfaga excede el MBS al añadirle el nuevo paquete. Existen tres casos posibles:

- $q[i]+b > \text{MBS}$, en este caso, se genera una ráfaga con un tamaño de $q[i]$ enviándolo a i y actualizamos el tamaño de ráfaga a $q[i]=b$
- $q[i]+b < \text{MBS}$, en este caso sólo hay que actualizar el tamaño de la ráfaga a $q[i]=q[i]+b$
- $q[i]+b = \text{MBS}$, en este caso se generara una ráfaga del tamaño del MBS y se resteara el tamaño de cola a 0.

El inconveniente principal de este tipo de algoritmo, es que tiene una mayor varianza que los algoritmos basados en tiempo afectando al rendimiento de los *core router*.

Algoritmos híbridos tiempo/longitud: Es preferible tener un algoritmo de ensamblado que rinda bien bajo todas las condiciones de carga, para eso el híbrido tiempo/longitud usa los tres limites anteriores: el temporal T_{\max} , el tamaño máximo MBS y el mínimo B_{\min} .

El algoritmo híbrido sigue el siguiente esquema:

- Se asocia una cola lógica con cada destino en los límites de la nube de conmutación de ráfagas ($[1:M]$)
- Se inicializa un temporizador T_i cada vez que llega un paquete de tamaño b dirigido a la cola i estando está vacía y se actualiza su tamaño $q[i]=b$ siempre que b sea menor que el MBS, en caso de ser mayor se envía directamente una ráfaga con este paquete.
- Cada vez que se recibe un nuevo paquete, se actualiza el tamaño de la cola y se compara al igual que en el caso de tamaño. En caso de alcanzarse o superarse el tamaño del MBS se envía la ráfaga con el contenido de la cola y se reinicia el temporizador T_i de dicha cola.
- Cuando T_i alcanza el límite de tiempo T_{\max} , se crea una ráfaga y se lanza para transmisión en el canal de datos: si la longitud de la ráfaga es menor de B_{\min} , se rellena hasta B_{\min} .
- Se resetea T_i y $q[i]$ a 0 y permanecerá así hasta la llegada del siguiente paquete a la cola i .

El algoritmo híbrido, permite obtener las ventajas tanto del algoritmo de tiempo, es decir un retardo controlado en condiciones de baja carga, como del de tamaño, es decir un tamaño de ráfaga acotado en situaciones de alta carga.

Algoritmo basado en límites dinámicos (Adaptativo) [16], [17], [18], [19]: el límite temporal T_{\max} , el de tamaño MBS o ambos son ajustado dinámicamente según las mediciones y predicciones sobre el tráfico real. Estos algoritmos dinámicos son más adaptativos a las condiciones del tráfico y pueden obtener un mejor rendimiento con tráfico altamente correlado como TCP o *long-range-dependent traffic* en redes OBS. El inconveniente de este tipo de algoritmos es que requieren una complejidad computacional mucho más alta debido a la necesidad de predecir y ajustar los límites.

Uno de los puntos más interesantes de este tipo de algoritmos es cómo predecir la longitud de la siguiente ráfaga.

Un punto a tener en cuenta es la predicción de la longitud de la siguiente ráfaga. Con esta predicción, el BCP de una ráfaga puede ser enviado a la red *OBS* antes de que se cree la ráfaga, y de este modo reducir el retardo de pre transmisión introducido por el ensamblado de ráfagas. La predicción es especialmente importante en el caso del algoritmo basado en límites dinámicos ya que los límites de longitud y/o temporales de la siguiente ráfaga deben ser establecidos a un valor de acuerdo a la predicción del tráfico entrante.

IV. TRÁFICO MULTIMEDIA

IV.1. INTRODUCCIÓN

Hoy en día, los proveedores de servicios de internet ya ofrecen servicios “*Triple Play*”, en los cuales se incluye la transmisión de *streaming* de vídeo, transmitidos a través de redes tradicionales como ATM/MPLS/SONET.

Por otro lado, el tráfico de *streaming* en Internet es cada vez mayor dado el éxito de la filosofía Web 2.0, donde los usuarios son los proveedores de contenidos, y especialmente a través de páginas como Youtube, que copan ya alrededor del 10-15% del tráfico en Internet y se espera que estas cifras sigan aumentando [20].

Conforme el número de usuarios y los requerimientos de los flujos aumente (vídeo de alta definición), las capacidades ofrecidas por las redes *OBS* se volverán esenciales.

En la tesina se ha utilizado el códec MPEG4 de la familia MPEG una de las más utilizadas para la codificación de vídeos para *streaming*.

El estándar MPEG define 3 tipos de *frames*: *frames* I, P y B. Los *frames* I pueden ser descodificados por ellos mismos, en cambio los *frames* P dependen del *frame* I o P anterior y los *frames* B dependen tanto del *frame* I o P anterior como del posterior. Esta dependencia es debido a que los *frames* P y B no contienen toda la información por sí mismos, sino tan sólo la variación de la imagen con respecto a los *frames* de los que dependen, lo que supone una reducción importante del tamaño de los *frames* manteniendo la calidad.

Los *frames* I, P y B se agrupan en grupos de imágenes (*Group of Pictures*, GoPs). Un GoP, “Fig. 7” es una secuencia de *frames* la cual empieza por un *frame* I y termina con el siguiente *frame* tipo I. Cuando mayor número de *frames* contenga un GoP, es decir cuando mayor sea el número de *frames* tipo P y sobretodo tipo B respecto al número de *frames* tipo I, menor será el tamaño del flujo de datos a transmitir, pero los errores, en caso de coincidir con la transmisión de un *frame* I, serán más graves y perjudicarán en mayor grado la calidad del vídeo recibido.

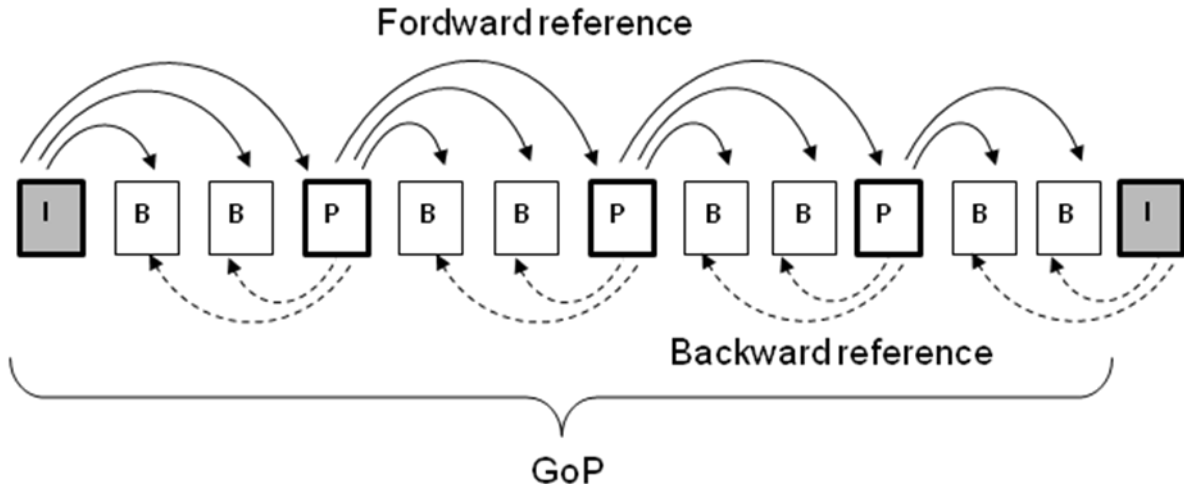


Fig. 7. Modelo de GOP

IV.2. PARÁMETROS DE CALIDAD

En la transmisión de vídeo se consideran dos tipos de parámetros de calidad, los parámetros objetivos y los parámetros subjetivos.

La manera más confiable de medir la calidad de una imagen o un vídeo es la evaluación subjetiva, realizada por un conjunto de personas que opinan acerca de su percepción. La opinión media, obtenida mediante el **MOS** (*Mean Opinion Score*) es la métrica generalmente aceptada como medida de la calidad. Para ello, los experimentos subjetivos controlados continúan siendo actualmente los métodos de medida reconocidos en la estimación perceptual de la calidad del vídeo.

Los métodos subjetivos son costosos, difíciles de realizar, e impracticables en aplicaciones de tiempo real. Por esto se hace necesario el uso de métodos objetivos y automáticos, que puedan predecir con fiabilidad la calidad percibida, en base a medidas objetivas tomadas en algún punto del sistema.

Se entiende por parámetros de calidad objetivos, aquellos que miden de un modo matemático los aspectos relativos al vídeo que afectan a su calidad, como pueden ser **la tasa pico señal a ruido (PSNR)**, la cual mide la calidad de la secuencia de transmisión del vídeo. El PSNR es una de las métricas objetivas más comúnmente utilizadas para evaluar la calidad del vídeo y se puede definir con esta ecuación:

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) \quad (1)$$

$$MSE = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [I_{org}(m,n) - I_{dec}(m,n)]^2$$

Donde I_{org} es la imagen original y I_{dec} es la imagen descodificada; M , N es el tamaño de la imagen y MSE es el error cuadrático medio.

Una **interrupción** se observa cuando uno o más *frames* consecutivos no pueden ser decodificados debido a la pérdida de algunos paquetes de vídeo. La naturaleza del sistema de visión humano hace muy difícil para un espectador percibir distorsión alguna si sólo se pierde una pequeña cantidad de *frames* consecutivos. Cuando el número de *frames* se incrementa por encima de un límite, se percibe la distorsión. La importancia de una interrupción depende de la duración de la misma, siendo mayor cuantos más *frames* consecutivos no puedan ser reproducidos. Así, las interrupciones pueden clasificarse según su gravedad como interrupciones menores o mayores.

Se asume que una interrupción se considera menor cuando dura menos de 1 segundo, mientras que una interrupción mayor durará más de 1 s. Las interrupciones se miden de acuerdo al número de *frames* consecutivos que no pueden ser visualizados. Este parámetro ofrece una medida objetiva de calidad del vídeo más cercana a la percibida por el usuario, dado que refleja de forma directa los fallos en la reproducción del vídeo, percibidos por el espectador.

Otro parámetro objetivo importante a evaluar es el **Throughput**, es decir el porcentaje de paquetes recibidos correctamente en destino respecto el total de paquetes enviados. Relacionado con este, también se observa el **Frame Loss Ratio** que indica el ratio de pérdida de tramas.

Por último también se tendrá en cuenta el **Frame Delay** como parámetro importante, ya que si es excesivo puede provocar cortes en la recepción del *streaming*, dado que los *frames* que llegan después de su instante esperado de reproducción se consideran perdidos, menguando de este modo su calidad.

V. DESARROLLO DE LOS TEST DE TRANSFERENCIA DE VÍDEO

V.1. ENTORNO DE DESARROLLO

En el desarrollo de la tesina, es necesario realizar simulaciones para obtener resultados que se acerquen a un comportamiento, lo más real posible, de una red de comunicaciones de conmutación óptica. Este tipo de redes ópticas está aún en fase de pruebas y sólo existen algunos prototipos a nivel de laboratorio en algunos institutos de investigación a nivel tanto europeo como mundial. Particularmente, en la Universidad Politécnica de Valencia, aún no se ha desarrollado prototipos de redes con tecnología exclusivamente óptica. Debido a esto se ha decidido utilizar el software de simulación NS-2 (*Network Simulator 2*), un software de simulación de eventos discretos utilizado para la simulación de redes en tareas de investigación, y al que se hace referencia en diversos artículos como el simulador utilizado para obtener resultados. Este software, al igual que todos los descritos más adelante son gratuitos y trabajan sobre Linux, aunque algunos de ellos

también tengan su versión para Windows, en concreto se ha utilizado el *Ubuntu 8.10 Intrepid* como sistema operativo y la versión de simulador NS-2 2.33 para realizar esta tesina.

Debido a que la red sobre la que se va a trabajar es una red de *Optical Burst Switching*, en adelante OBS, y no una red *Ethernet*, se le ha tenido que incorporar el módulo *obs-ns*, el cual modifica los ficheros necesarios para otorgar las características propias de una red OBS al simulador NS-2.

Para poder crear los scripts con los que trabaja el simulador NS-2 se utilizan los lenguajes TCL/TK, en concreto se ha utilizado la versión 8.4.18 que incluye el simulador NS-2 2.33. A través de estos scripts es como se configuran los parámetros de la red tales como topología, capacidad de los enlaces, etc..., así como el tipo y características del tráfico de la red.

Una vez preparada la infraestructura virtual, y debido a que esta tesina se basa en el estudio del tráfico multimedia en una red *OBS*, ha sido necesario el uso de software específico para la manipulación de los vídeos.

Para la codificación de vídeos se ha utilizado el software *ffmpeg* con el cual obtenemos el vídeo en formato *mpeg4* con los parámetros que se hayan decidido, el cual será transmitido por la red virtual, y con esta misma aplicación, decodificado en destino.

Para poder transmitir este vídeo por la red virtual y analizar sus datos, se ha dispuesto del software *Evalvid*, el cual incluye un pequeño grupo de aplicaciones a través de las cuales se consiguen las trazas resultantes del envío del vídeo a través de la red y la regeneración del vídeo en destino. Además también permite obtener algún dato interesante como el *Peak Signal to Noise Ratio* (PSNR) del vídeo recibido como medida objetiva de calidad.

Para automatizar el paso de parámetros de configuración y realizar pruebas para diversos casos, se ha hecho uso de *scripts* de *shell*. Estos también se han utilizado para automatizar la recolección de los diversos resultados a fin de ser analizados mediante *Octave*.

Para el análisis de los resultados obtenidos y la generación de las correspondientes gráficas se ha utilizado el *Octave 3.2.2*, software similar al conocido *Matlab*. Creando *scripts* *.m* con las instrucciones necesarias para el análisis y representación de los datos y su almacenamiento a modo de ficheros de imagen.

V.2. DESARROLLO DE LA TESINA

Una vez el entorno preparado, se ha realizado el estudio de los algoritmos más comunes de ensamblado sobre una misma topología de red de 2.5 Gb de capacidad. En ambos escenarios se ha introducido un tráfico compuesto por flujos de vídeo real de 11 segundos de duración con tamaño CIF y 359 frames, codificado usando el estándar *mpeg4*.

Para comprobar cómo afectan los distintos parámetros (tamaño de ráfaga, tiempo de ráfaga, entre otros) a la transmisión del vídeo se ha decidido simular el mismo tráfico en dos entornos distintos:

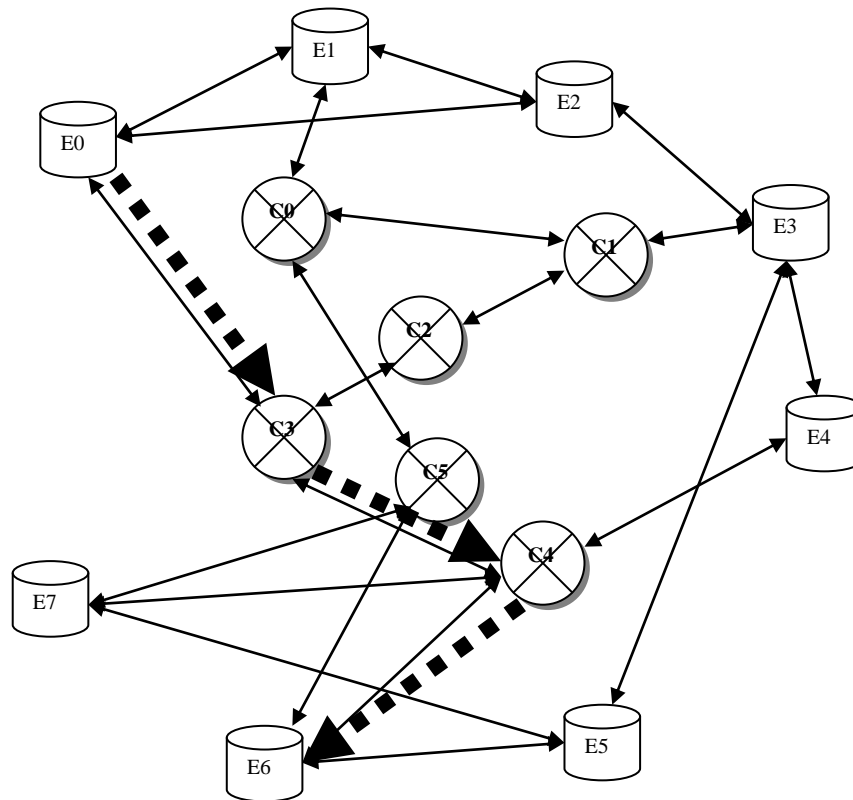


Fig. 8. Red OBS de la simulación con 8 *edge router* y 6 *core router*

En el primero de ellos, se transmiten 10 ráfagas de vídeo simultáneas con *bitrate* bajo a través de una red OBS “Fig. 8” siguiendo la ruta indicada por la flecha punteada, sin ningún otro tráfico que pueda interferir en ningún sentido, ni desde el mismo nodo ni desde otros, por lo tanto se puede observar cómo afecta directamente la red a la transferencia de vídeo permitiendo así establecer unos parámetros óptimos iniciales los cuales se comprobarán en el escenario 2.

El modelo de red OBS de 14 nodos conecta ciudades de estados diferentes sobre la geografía de Estados Unidos, cada enlace tiene una distancia diferente y por esta razón también introduce un retardo diferente. Los enlaces son *Duplex*, luego el retardo es el mismo para los dos enlaces tanto el de subida como el de bajada. En la “Tabla 1” se muestra el retardo asignado para cada enlace.

En el Escenario 2 se transmite un mayor número de flujos de vídeo, 100 flujos de vídeo a través de la misma ruta “fig. 8” y con un *bitrate* de 500 Kb y un GoP de 16, con el fin de generar un flujo de vídeo de características similares a la transferencia de vídeo de alta definición, estos vídeos se han distanciado entre ellos unos milisegundos para reducir las colisiones en el *edge router* de entrada. Para simular el tráfico de background que se puede encontrar en una red de estas características, se transmite un tráfico self-similar entre todos los *edge router*, incluidos los nodos origen y destino del vídeo, con el fin de ver el comportamiento de los *core router* y como afectan

las colisiones en estos a la transmisión de vídeo. Para obtener datos más fiables se analiza diez iteraciones de cada caso de los vistos en el Escenario 1, y se calculan las medias de los 100 vídeos y de los diversos escenarios ya que debido a la aleatoriedad del tráfico self-similar al igual que del tráfico de una red troncal, es el mejor método con el que modelar la transmisión de vídeo resultante.

Enlace	Retardo (ms)
E(0) – E(1)	700
E(0) – E(2)	1000
E(0) – C(3)	16
E(1) – E(2)	400
E(1) – C(0)	5
E(2) – E(3)	1200
E(3) – E(4)	400
E(3) – E(5)	1800
E(3) – C(1)	8
E(4) – C(4)	4
E(5) – E(6)	700
E(5) – E(7)	1200
E(6) – C(4)	5
E(6) – C(5)	5
E(7) – C(4)	5
E(7) – C(5)	2
C(0) – C(1)	2
C(0) – C(5)	4
C(1) – C(2)	5
C(2) – C(3)	2
C(3) – C(4)	1

Tabla 2. Retardo introducido en cada enlace

Los parámetros del tráfico self-similar introducido en cada uno de los *edge router* son los presentes en la “Tabla 3”. Estos parámetros se han seleccionado teniendo en cuenta otros estudios [21], [22]. El parámetro Hurst indica la auto-similaridad del tráfico self-similar generado y suele variar entre 0.5 y 1 siendo 0.9 un valor típico para el modelado del tráfico de red.

Tasa	1000	2000	2500
Tamaño de Paquete (Bytes)	1600	1600	1600
Tamaño de Batch (Bytes)	1600	1600	1600
Hb	-0.5	-0.5	-0.5
Tiempo de Desviación Estándar	$1.0e^{-4}$	$5.0e^{-5}$	$4.0e^{-5}$
Hurst	0.9	0.9	0.9
Carga de Tráfico generada	15%	30%	35%

Tabla 3. Parámetros del tráfico self-similar

Para comprobar cómo afecta la carga a la transferencia de vídeos se simula el Escenario 2 con tres tasas distintas de 1000, 2000 y 2500 paquetes por segundo, las cuales generan una carga de tráfico media del 15%, el 30% y el 35% respectivamente. Adicionalmente se utilizará, para comprobar cómo afecta la carga, una carga de tráfico del 70% producto de la agregación de *edge router* conectados al *core router C3* manteniendo una tasa de 2500 paquetes por segundo.

No se han generado cargas más altas debido, en primer lugar, a las limitaciones en cuanto a hardware disponible para realizar las simulaciones y, en segundo lugar, debido a que con esta carga de red ya se pueden observar unas pérdidas importantes.

V.2.1. Red dedicada para la transferencia de tráfico multimedia

V.2.1.1. Algoritmo de ensamblado limitado por tamaño

En primer caso se ha analizado el algoritmo de ensamblado por tamaño considerando tamaños de ráfaga de entre 20kb y 320kb en saltos de 20kb. Con estos valores se han obtenido los siguientes resultados.

Retardo:

En las gráficas de la “Fig. 9” se puede observar que para tamaños máximos de ráfaga pequeños, se obtienen retardos máximos del rango de 1-2 segundos, pero el problema grave se tiene cuando se quiere agrupar con tamaños de ráfagas mayores, obteniendo retardos incluso superiores a los 15-20 segundos, esto provocaría que la transmisión multimedia fuera inviable. Otro punto a tener en cuenta es la varianza temporal en la transmisión de ráfagas, lo que complica cualquier intento de prever cuando se transmitirá la siguiente ráfaga y, por lo tanto, el uso de sistemas de reserva de recursos predictivos.

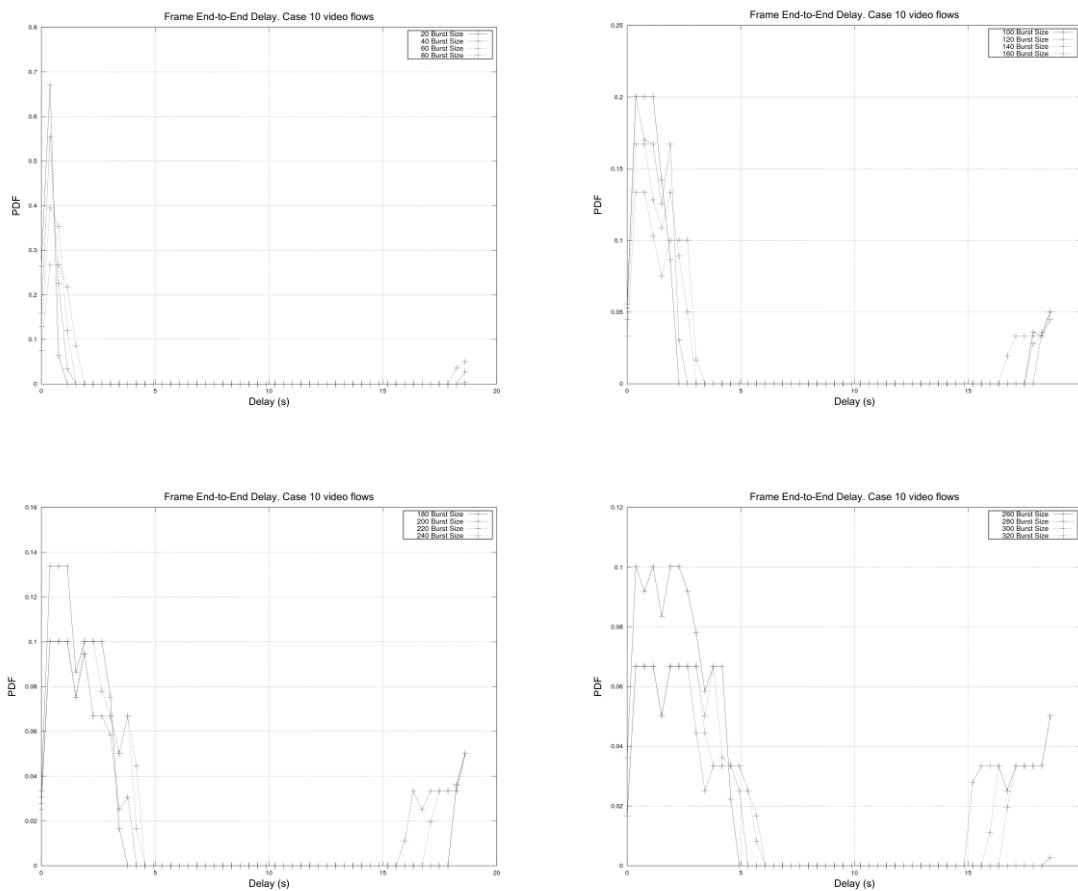


Fig. 9. Retardo en la red OBS para MBS de tamaño comprendido entre los 20 y los 320 KB

Estos resultados también se deben a que la prueba se ha realizado con el envío únicamente de las ráfagas de vídeo, por lo que se podría considerar como de baja densidad de tráfico, y para estos entornos no es una buena opción el uso de ensamblado por tamaño como aquí se demuestra. En caso de usarse este tipo de algoritmo de ensamblado se debería usar un MBS del rango de los 60-80 KB ya que es el mayor tamaño de ráfaga el cual aún mantiene acotado el retardo introducido, como se puede apreciar en las gráficas sólo introducen una muestra con mayor retardo, la correspondiente al final de la transmisión que nunca alcanzará el MBS.

Pérdidas:

Como se puede observar en la gráfica “Fig. 10” sólo se detectan pérdidas en el caso de ráfagas muy pequeñas (20-40 KB), estas pérdidas se deben al sistema de *buffer* implementado en el cual, en caso de que se tengan suficientes paquetes de datos para formar una ráfaga antes de que la anterior haya sido formada y enviada, el simulador descarta la ráfaga, esto implica que en tamaños pequeños de ráfaga y momentos altos de carga se produzcan pérdidas en los *edge router*, no apareciendo estas en tamaños mayores de ráfaga, ya que el tiempo de formar y transmitir la ráfaga es menor que el tiempo necesario para reunir bastantes datos para la siguiente ráfaga.

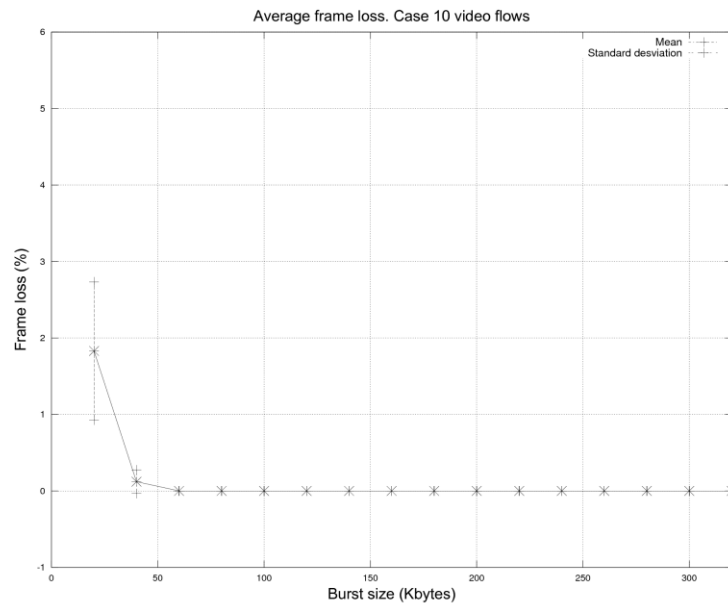


Fig. 10. Tasa de pérdidas de frame respecto al MBS utilizado.

PSNR:

Estas pérdidas en los *edge router* no se produce con tamaños superiores a 60 KB donde se estabiliza la media siendo la misma que la obtenida mas adelante con la simulación del algoritmo de tiempo. Así se considerarían óptimos valores de ráfaga de 60-80 KB.

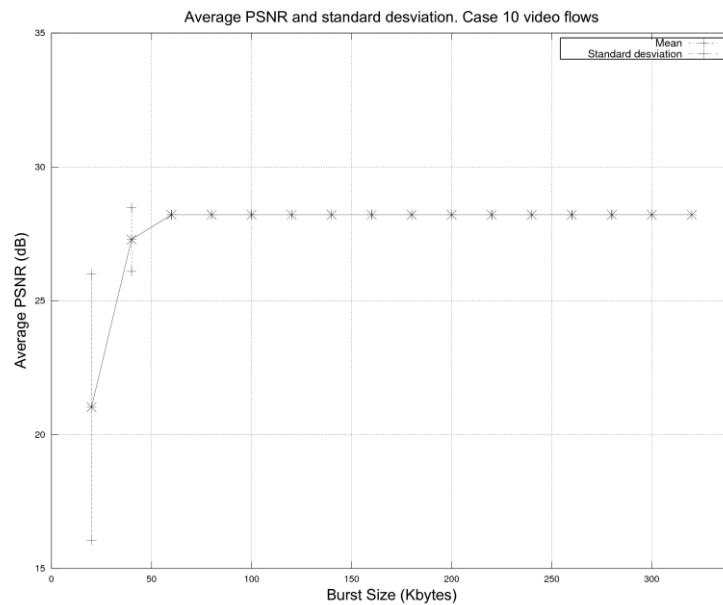


Fig. 11. PSNR y desviación estándar respecto al MBS utilizado.

V.2.1.2. Algoritmo de ensamblado limitado por tiempo

En segundo caso se ha estudiado el algoritmo de ensamblado basado en tiempo, primero se ha realizado un análisis de amplio rango para observar que valores son los óptimos a la hora de trabajar con *streaming* de vídeo, para ello se han analizado potencias negativas de 10 desde millonésimas hasta décimas de segundo como tiempo de ráfaga. Se ha comprobado para todos estos valores de límite temporal sus estadísticas de PSNR, Retardo, pérdidas, tamaños de ráfagas y, debido a que ninguno de los valores anteriores produce pérdidas, ni degradación del PSNR, se ha utilizado el retardo como factor para elegir el rango a simular.

Retardo para rango amplio:

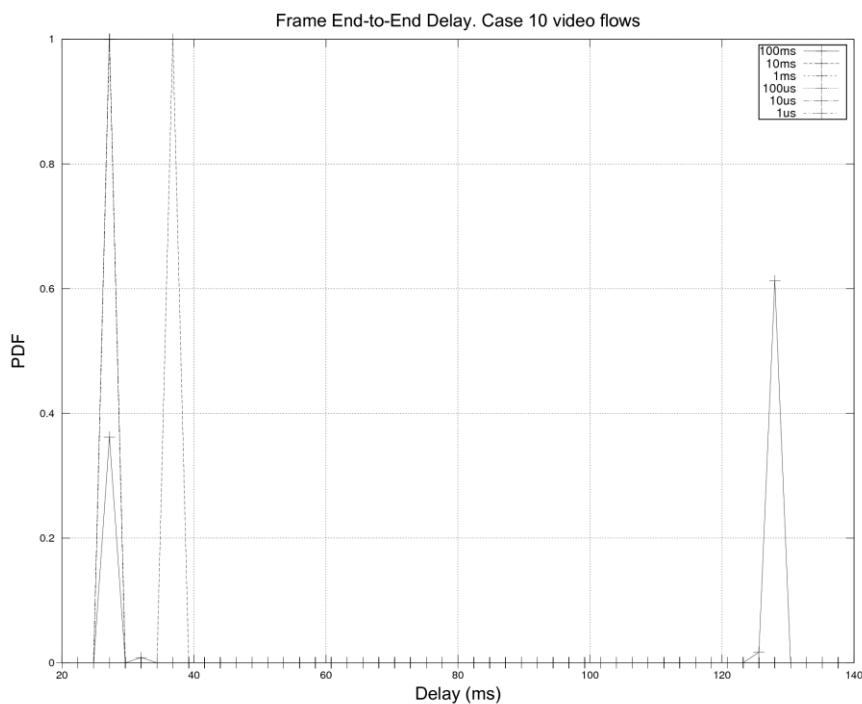


Fig. 12. Retardo en las ráfagas para distintos límites de tiempo

Como se aprecia en la gráfica, "Fig. 12", los valores más favorables para la codificación y transmisión de tráfico multimedia, son los próximos al rango de los milisegundos ya que el retardo introducido por el ensamblado de la ráfaga y su transmisión hace poco beneficioso el transmitir ráfagas con límite temporal inferior al milisegundo, ya que se obtiene un retardo extremo a extremo muy similar al obtenido en el rango de 1-10 milisegundos. Otros estudios también han llegado a la misma conclusión [9], por lo tanto se decide tomar como valores a simular 1ms, 5ms y 10ms.

Una ventaja muy importante del algoritmo limitado en tiempo es el hecho de que no introduce las pérdidas observadas en el caso del algoritmo basado en tamaño, esto se debe a que la generación de ráfagas no depende del tráfico que le llegue, por lo que, teniendo un *buffer*

suficientemente grande, no se ve afectada por los picos de tráfico entrante que producen pérdidas en el algoritmo de ensamblado por tamaño, en donde intenta ensamblar una ráfaga antes de finalizar con la anterior. En el algoritmo de tiempo, el tiempo entre ensamblado de ráfagas es determinado por el límite temporal impuesto y siempre es mayor al tiempo requerido para el ensamblado de la ráfaga.

Tamaño de ráfagas:

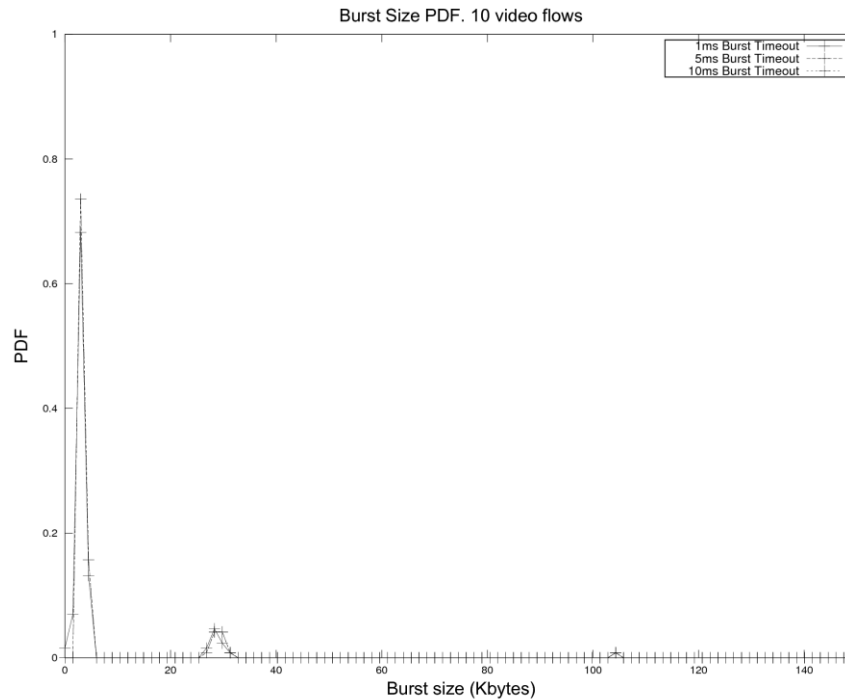


Fig. 13. Tamaño de ráfagas en el algoritmo de tiempo

En la “Fig. 13” se puede observar cómo, en el caso de 1ms de *Timeout*, aparece un cierto porcentaje, en torno al 8% de ráfagas de tamaño inferior a 3KB, en cambio este porcentaje se distribuye mayormente en ráfagas de 3KB y, en menor medida, en ráfagas de unos 30 KB tanto para el caso de 5ms como para el de 10ms, resultado lógico debido a que al tener el *Timeout* más elevado se producen ráfagas de mayor tamaño. Esto se debe a que se está transmitiendo únicamente 10 ráfagas de vídeo sin más tráfico que siga la misma ruta, como se podrá observar en el sistema completo con tráfico de *background*, las ráfagas serán mayores y más aleatorias.

La distribución del tamaño de ráfaga es peor que en el caso de ensamblado por límite de tamaño, ya que no es tan uniforme lo cual complica el tratamiento de los datos y el uso de sistemas de conmutación con *slots* predefinidos a cierto tamaño, teniendo que indicar el paquete de cabecera el tamaño exacto y el *core router* realizar los cálculos temporales para la conmutación y transferencia de dichos datos, sin poder tener unos parámetros preestablecidos.

Retardo:

En el caso del retardo, la “Fig. 14” muestra como el retardo es mayor lógicamente cuando mayor es el *Timeout*, siendo la diferencia entre ellos la misma que la relativa a los distintos

Timeouts ya que sólo se transmite el tráfico del vídeo sin más tráfico que pueda producir variaciones en el retardo sufrido.

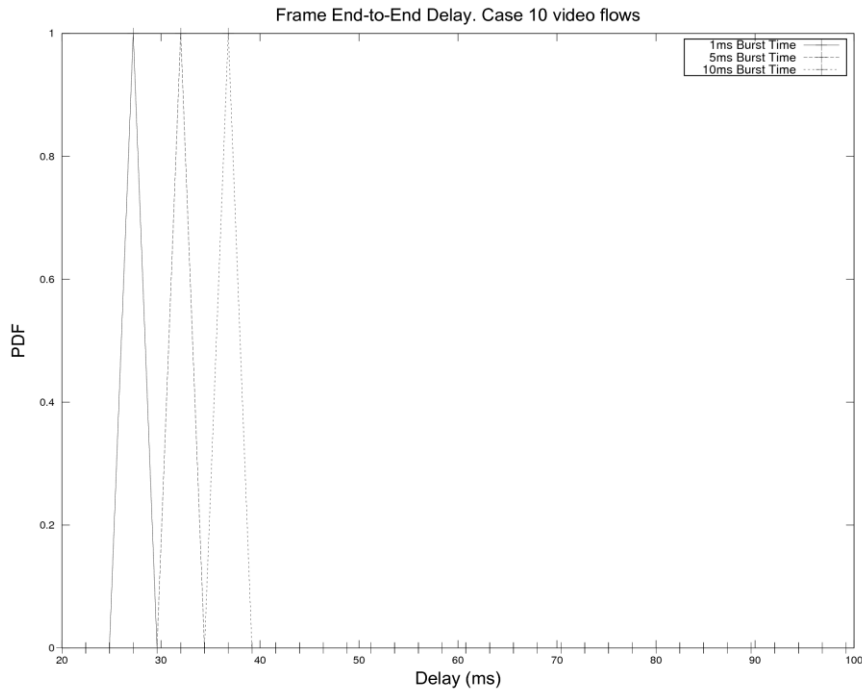


Fig. 14. Retardo sufrido para distintos límites de tiempo

PSNR:

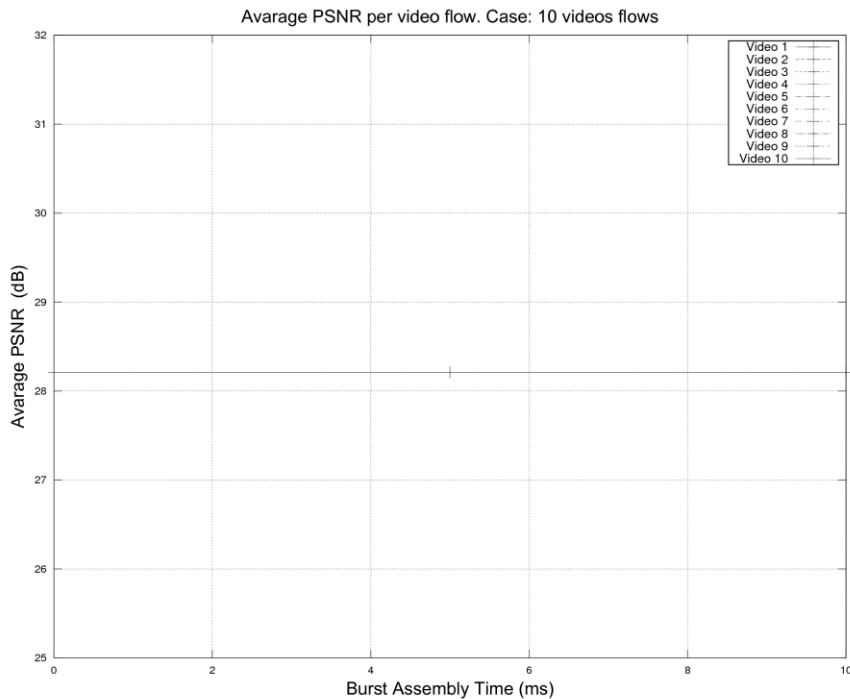


Fig. 15. PSNR en el algoritmo de ensamblado limitado en tiempo

El PSNR, como se aprecia en la “Fig. 15”, se mantiene constante para los 10 vídeos y los distintos timeouts introducidos a diferencia del algoritmo de tamaño, en el cual se produce cierta degradación del PSNR en los valores de ráfaga más pequeños.

V.2.1.3. Algoritmo de ensamblado híbrido

Para solucionar los distintos inconvenientes de los dos algoritmos de ensamblado anteriores, se presenta como primera opción un algoritmo híbrido tiempo/tamaño el cual, como esta explicado en el punto III.2, permite el envío de tramas sin superar un T_{\max} (*timeout*), es decir con un retardo máximo controlado y sin superar cierto tamaño MBS en casos de alta densidad de tráfico, evitando ráfagas demasiado largas las cuales tienen mayor probabilidad de colisión durante su transmisión dentro de la red *OBS*. Para mantenerse en el rango de las simulaciones anteriores que son teóricamente las óptimas para este tipo de redes, se ha decidido simular en el rango de 1-10ms y con tamaño máximo de ráfaga de 20-100KB, ya que este rango es el que se ha observado como el más probable en el rango temporal visto.

Tamaño de ráfagas:

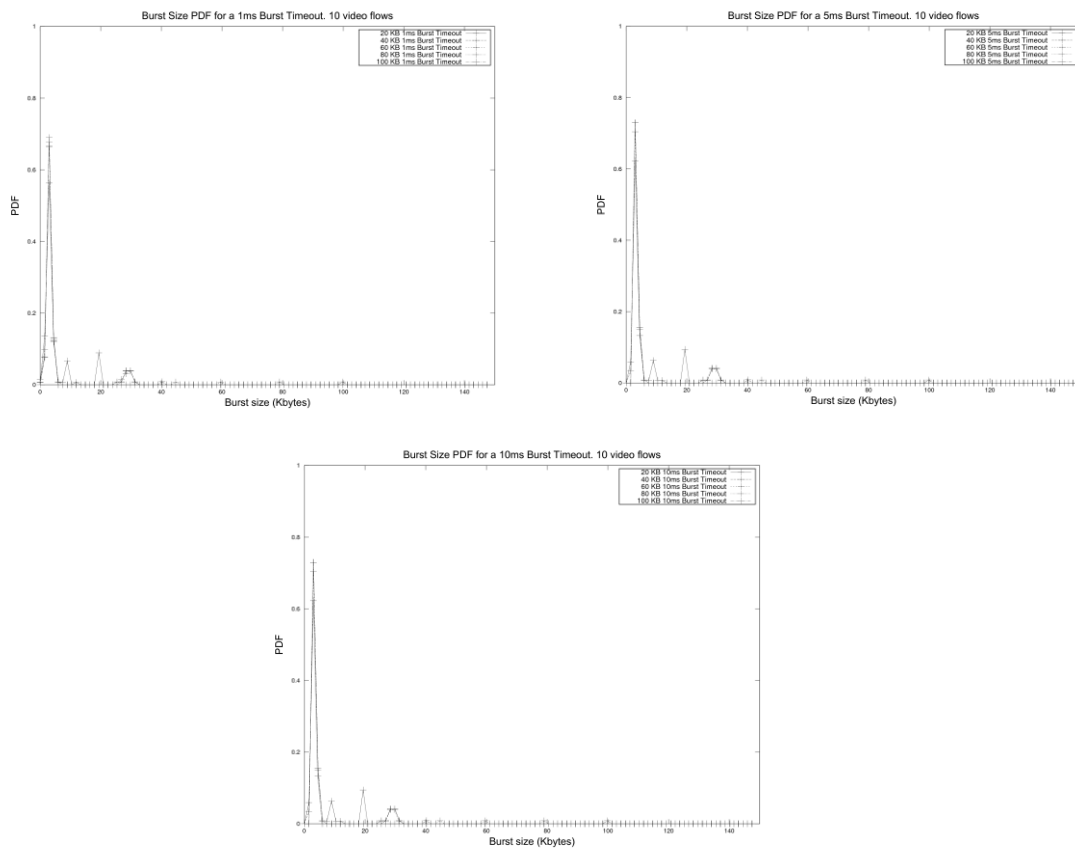


Fig. 16. Tamaño de ráfaga para 1,5 y 10ms, y rango de MBS de 20 a 100 KB

Con el algoritmo híbrido se consigue un gran porcentaje de paquetes de tamaño reducido debido a la limitación en tiempo y unos pocos paquetes de tamaño máximo controlado según la limitación en tamaño impuesta, como se observa en la “Fig. 16” en ningún caso los paquetes exceden estos tamaños.

Retardo:

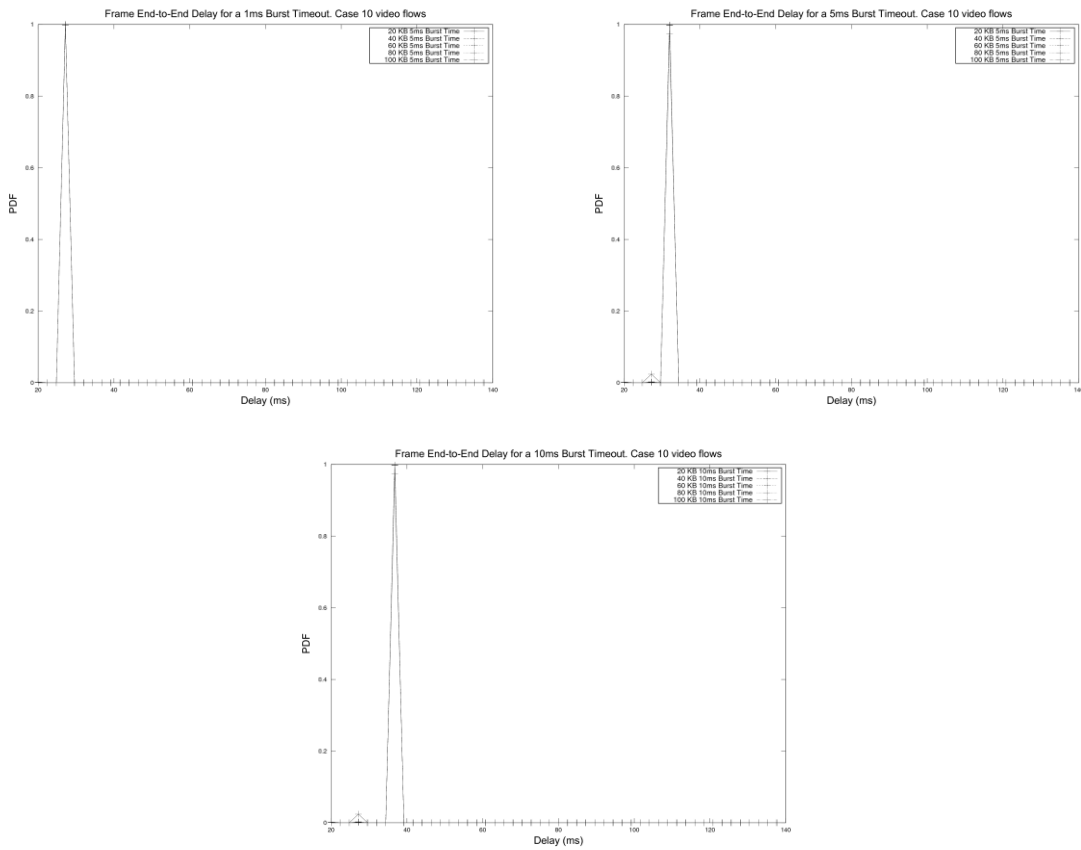


Fig. 17. Delay sufrido por las ráfagas para 1,5 y 10ms, y rango de MBS de 20 a 100 KB

En la “Fig. 17”, se puede comprobar cómo el retardo sufrido con el uso del algoritmo híbrido es igual al sufrido en el algoritmo de tiempo o incluso mejor en los casos de tamaño máximo más pequeños. Además resulta obvio que es muy inferior al retardo que se produce con estos mismos tamaños de ráfaga en el algoritmo limitado en tamaño.

PSNR:

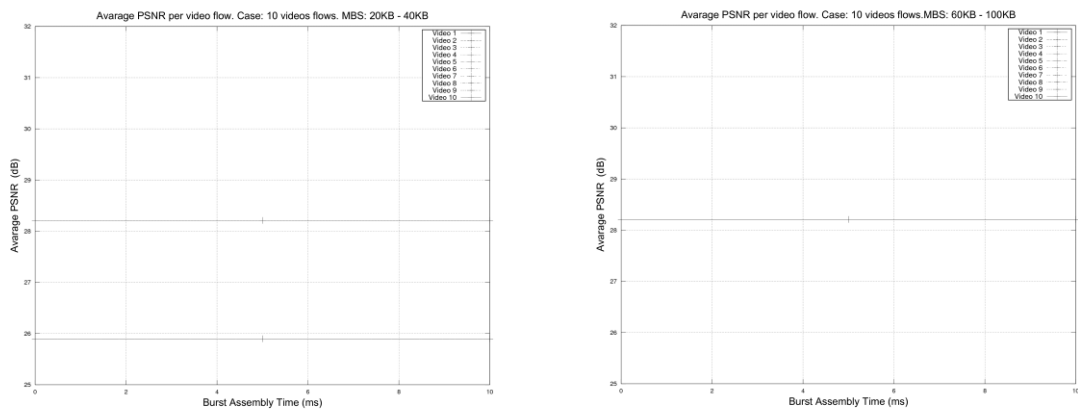


Fig. 18. PSNR sufrido con tamaños inferiores a 60 KB y de 60 KB o superiores en el algoritmo híbrido

Como se observa en la “Fig. 18”, algunos vídeos sufren cierta degradación del PSNR para tamaños máximos de ráfaga de 40 KB o inferiores, degradación que no se produce con tamaños de 60 KB o superiores con los cuales se obtiene el mismo valor de PSNR óptimo que con el algoritmo limitado en tiempo.

Vistos todos los resultados inferiores se pueden establecer como parámetros óptimos para la transferencia de vídeo en una red dedicada los valores de límite temporal de 1ms y de tamaño máximo de ráfaga de 60 KB ya que de este modo se evitan las pérdidas debidas a la creación de ráfaga en periodos de alta carga del *edge router* emisor las cuales se han observado para tamaños de MBS de 20 KB y 40 KB, y al mismo tiempo se mantiene controlado que no se produzcan ráfagas excesivamente grandes, al mismo tiempo se controla el retardo limitando el T_{\max} a 1ms, permitiendo de este modo una buena calidad en el vídeo recibido reduciendo el retardo al máximo y evitando interrupciones mayores.

V.2.2. Red con tráfico self-similar de background

V.2.2.1. Algoritmo de ensamblado limitado por tamaño

En primer caso se ha analizado el algoritmo de ensamblado por tamaño considerando tamaños de ráfaga de entre 20KB y 320KB en saltos de 20KB. Del análisis del Escenario 1 se presuponen pérdidas para tamaños pequeños de ráfaga por colisiones cuando coincide la llegada de diversas tramas (*frames*) intra de vídeo en el mismo instante, para reducir este factor se ha distanciado temporalmente varios milisegundos el inicio de los 100 vídeos a transferir, aunque esto no elimina completamente las pérdidas por colisión en el *edge router* de entrada, si que las reduce de un modo considerable.

En este análisis debido a las previsibles pérdidas se han analizado también las interrupciones, es decir cuando varios *frames* consecutivos no llegan al destinatario o no pueden ser reproducidos debido a la falta del *frame* de referencia, como parámetro objetivo de calidad. Este parámetro no se ha tenido en cuenta en el Escenario 1 debido a que todas las pérdidas producidas en el mismo eran de una sola trama por colisión en el *edge router* de entrada con la misma trama de otros vídeos simultáneos.

Retardo medio:

El retardo sufrido por el vídeo es mucho menor que el observado en el Escenario 1, quedando en el rango de milisegundos, debido a que se genera un volumen de tráfico muy superior entre el *edge router* origen E(0) y el *edge router* destino E(6) del vídeo debido tanto al mayor número de flujos como, principalmente, al volumen de tráfico de background introducido para simular el funcionamiento habitual de una red troncal de fibra óptica. Esta es una de las razones del uso de dos escenarios diferenciados, ya que este parámetro al que en el Escenario 2 se podría dar poca

importancia y no tener en cuenta al modelar las características de la red, en el Escenario 1 se demuestra importante para entornos de escaso tráfico o para *edge router* dedicados a la provisión de vídeo exclusivamente como pudiese darse el caso de una compañía que ofreciese servicio de vídeo a través de cable (Imagenio, ONO,...) que usase un *edge router* sólo para la transferencia del vídeo desde su central principal a las distintas centrales provinciales las cuales a su vez podrían estar recibiendo y enviando tráfico diverso.

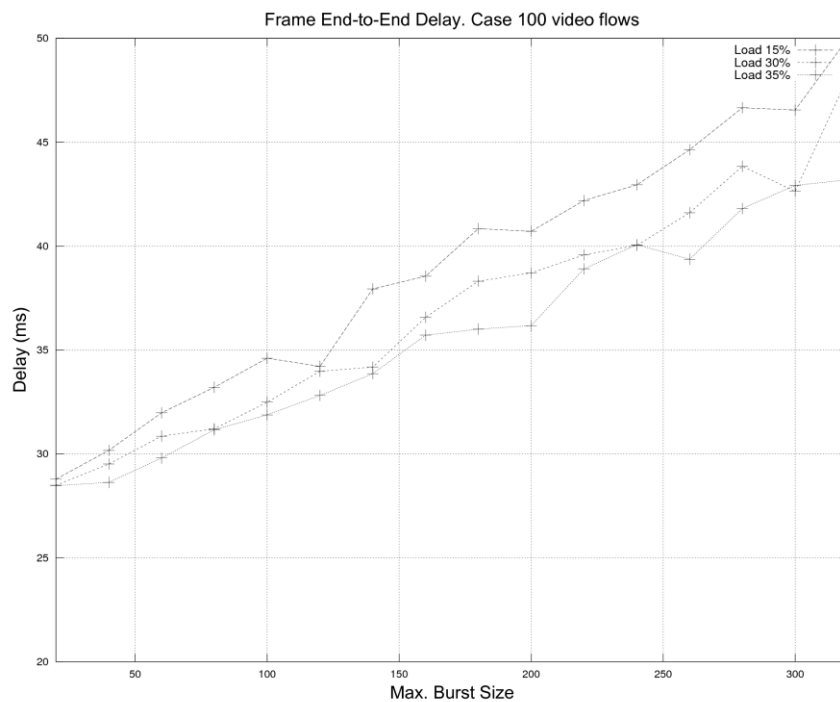


Fig. 19. Retardo de Trama medio para distintas cargas de tráfico y MBS

Como se puede apreciar en la “Fig. 19” el retardo sufrido por las tramas aumenta conforme aumenta el tamaño máximo de las ráfagas llegando, con un tamaño de ráfaga de 320 KB a prácticamente duplicar el retardo sufrido por las mismas tramas enviadas con un tamaño de ráfaga máximo de 20 KB. También se muestra como la carga de la red influye, principalmente la carga de tráfico entre el mismo *edge router* origen y el mismo *edge router* destino, reduciendo el retardo sufrido por las tramas conforme aumenta la carga de tráfico, lo que es lógico debido a que al generarse más paquetes el tiempo necesario para llenar el *buffer* de entrada es menor y por lo tanto se reduce el retardo de los paquetes que de otro modo deben esperar un mayor tiempo a ser enviados.

Con todo esto sería interesante utilizar un tamaño de ráfaga con un retardo reducido, se debería utilizar un MBS del rango 20 KB a 80 KB

Pérdidas:

Para poder ver como varían las pérdidas dependiendo del tráfico de background introducido a la red *OBS*, en concreto de la carga, se ha decidido representar el valor medio de las pérdidas para

cada tamaño de ráfaga entre 20KB y 320 KB calculando éste como la media de todas las pérdidas en cada uno de los 100 vídeos de entrada y para las diversas iteraciones realizadas, de modo que se obtiene un sólo valor medio por tamaño máximo de ráfaga y valor de Hurst.

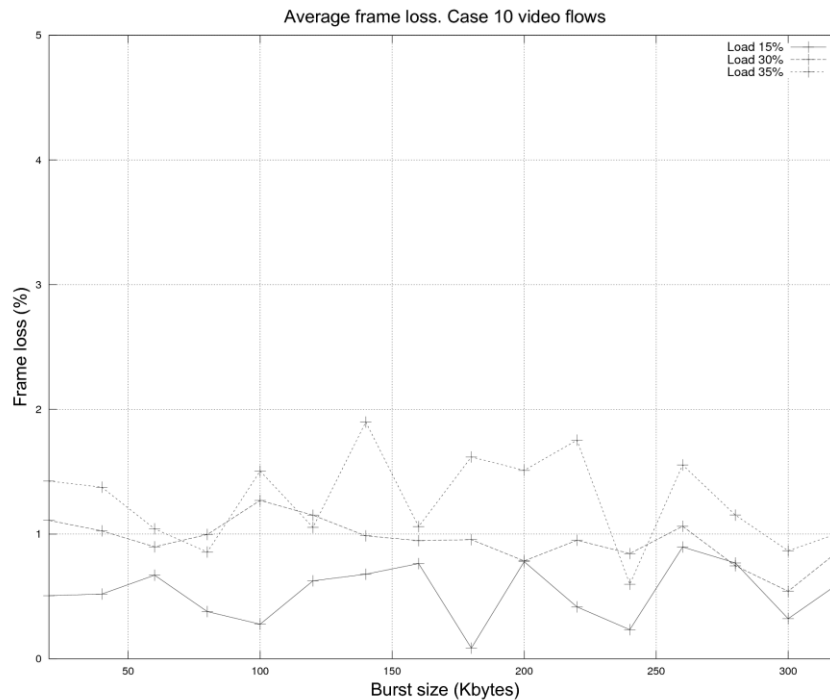


Fig. 20. Pérdidas sufridas para distintas cargas de tráfico y MBS

La “Fig. 20” debería mostrar como disminuyen las pérdidas medias conforme aumenta el tamaño de ráfaga debido a que un mayor tamaño de ráfaga implica un menor número de paquetes en la red por lo que disminuye la probabilidad de colisión en los *core router*. Contrariamente, cuando los tamaños de los paquetes aumentan hacen que vuelvan a aumentar las pérdidas, ya que la pérdida de un paquete implica la pérdida de mayor cantidad de datos, también influye que el *core router* tardará más en procesar un paquete al ser de tamaño mayor, por lo que si el retardo que introduce el protocolo JET no es suficiente, cuando se envíe la ráfaga, puede que aun este procesándose la anterior y el paquete de control no haya podido encontrar ningún horizonte disponible para su reserva y procesamiento de la ráfaga entrante.

Por otro lado se observa como el porcentaje de tramas perdidas aumenta conforme aumenta el carga del tráfico de background, esto se debe principalmente a que, al haber mayor numero de ráfagas disputando los mismos recursos, en concreto las fibras disponibles de los *core router*, será mayor el número de veces en las cuales, a su llegada, el paquete de control no encuentre un horizonte disponible, según el algoritmo *LAUC/Horizon*, para reservar la fibra para la ráfaga entrante. Es decir se produce un mayor número de “colisiones” en los *core router*. Al mismo tiempo también aumentan las colisiones en los *Edge router* para tamaños de ráfaga pequeños ya

que al aumentar la carga se generaran mas tramas pudiendo colisionar como se vio en el Escenario 1.

Tamaño medio de las interrupciones:

Las interrupciones representan bloques de frames consecutivos que se pierden en una transmisión de vídeo, se ha analizado este parámetro ya que es posiblemente la medida de calidad objetiva más perceptible en la apreciación subjetiva del observador del vídeo resultante.

En este caso se ha tenido en cuenta el tamaño medio de las interrupciones para cada tamaño máximo de ráfaga y para diversas tasas de tráfico.

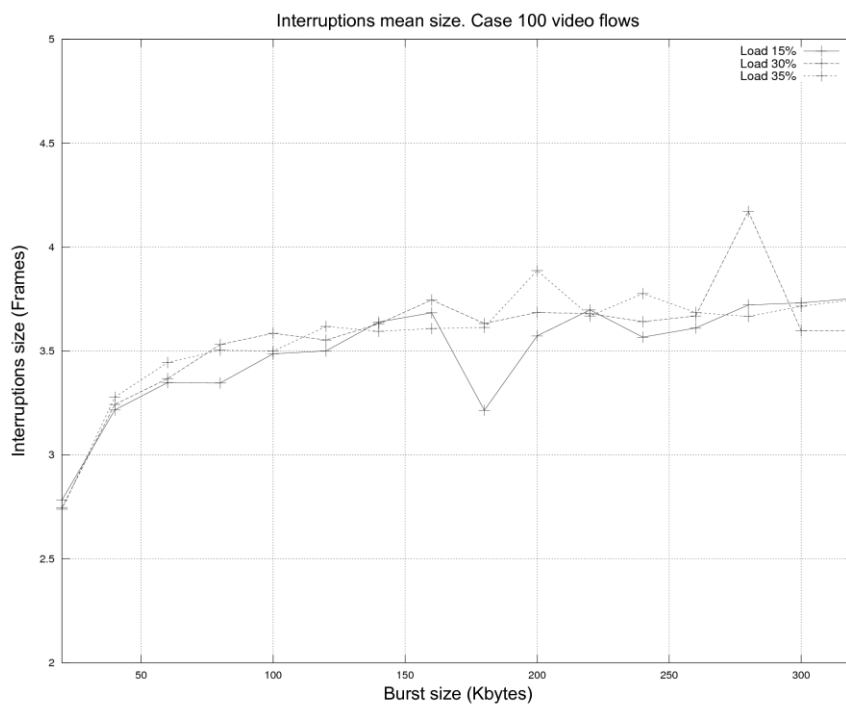


Fig. 21. Tamaño medio de las interrupciones para distintas cargas y MBS

La “Fig. 21” muestra como el tamaño medio de las interrupciones aumenta significativamente desde los tamaños de ráfaga más pequeños para luego aumentar ligeramente para tamaños superiores a los 120 KB, el significado de esto es que, aunque el número de pérdidas es mayor para tamaños pequeños de ráfaga, la importancia de estas pérdidas observadas individualmente es menor. El tamaño medio de las interrupciones observado en la “Fig. 21” hace suponer que muchas de estas interrupciones serán perceptibles para el destinatario del vídeo, debido a esto se debería intentar al menos mantener el tamaño máximo de ráfaga en valores inferiores a 80KB para intentar minimizar el tamaño de las interrupciones.

En lo que respecta a la carga de la red parece no afectar en exceso siendo de un tamaño similar para las distintas cargas estudiadas.

Si se tienen en cuenta todos los parámetros aquí analizados, en caso de usar el algoritmo de ensamblado limitado en tamaño se elegiría un tamaño máximo de ráfaga de 60 KB u 80 KB ya que son los que muestran un mejor resultado conjunto, al igual que ya lo hiciesen para el Escenario 1.

PSNR:

El último parámetro a tener en cuenta es el del PSNR el cual puede ser superior al del Escenario 1 debido a la mejora en la calidad de origen del vídeo a transmitir, es decir al aumento significativo en la tasa hasta los 500 KB y a la reducción del tamaño del GoP de 30 a 16 imágenes.

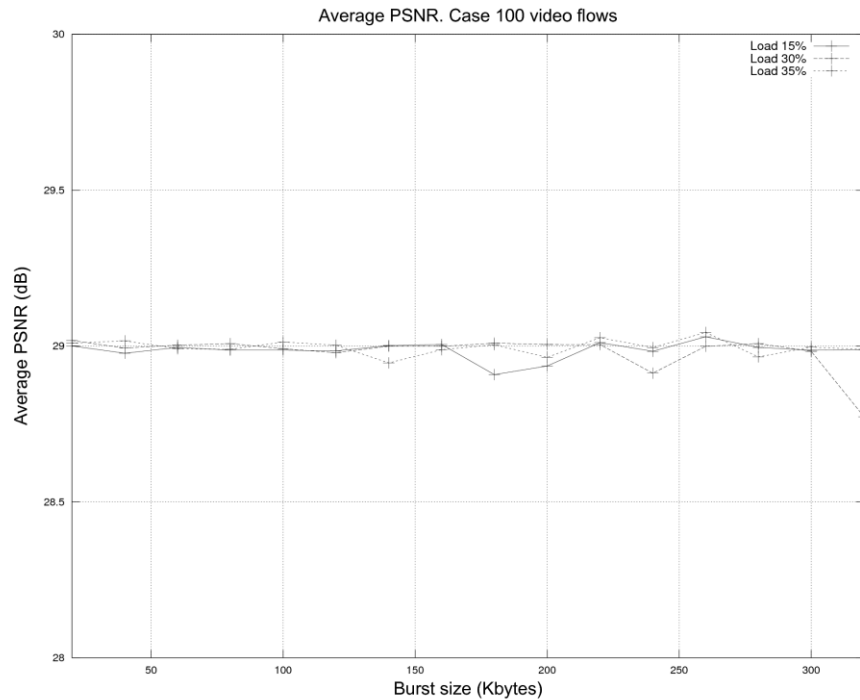


Fig. 22. PSNR para distintas cargas y MBS

En la “Fig.22” se muestra como el PSNR medio apenas varía para los distintos tamaños máximos de ráfaga, tan sólo cayendo puntualmente sobre todo para cargas más elevadas.

V.2.2.2. Algoritmo de ensamblado limitado por tiempo

Pérdidas:

Al igual que en el caso del algoritmo limitado por tamaño, se ha calculado las pérdidas realizando la media sobre los datos obtenidos de 100 vídeos transmitidos simultáneamente entre los *edge router* E(0) y E(6) con un tráfico de *background self-similar* con cargas de tráfico del 15%, 30% y 35%. A la vez estas simulaciones se han realizado diez veces para obtener datos más veraces calculando su media.

El porcentaje de tramas perdidas en el algoritmo de tiempo se espera sea inferior al porcentaje obtenido con el algoritmo de tamaño si tenemos en cuenta los cálculos realizados en el Escenario 1, donde el algoritmo de tiempo no producía ninguna pérdida, y no así el de tamaño. En este caso

como las pérdidas se deben a las “colisiones” en los *core router*, es decir, a que el paquete de control no encuentre un horizonte libre y cuando llegue el paquete de datos este deba ser descartado, no debe haber tanta diferencia con respecto a las pérdidas sufridas, debiendo ser estas similares a la de los tamaños de ráfaga medios obtenidos para cada Timeout estudiado.

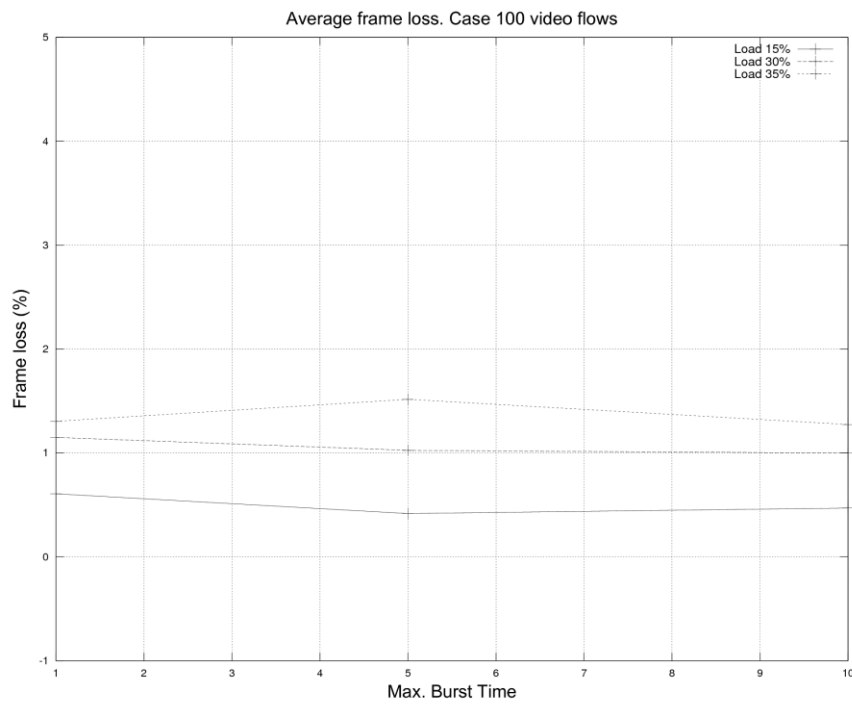


Fig. 23. Media de tramas perdidas para distintas cargas de tráfico

En la “Fig. 23” se puede apreciar como el porcentaje de pérdidas es inferior, pero de valores similares a las pérdidas registradas con el algoritmo de ensamblado limitado en tamaño, la diferencia de pérdidas se hace mayor cuando mayor numero de fuentes hay, puesto que la aleatoriedad del tráfico facilita al algoritmo *LAUC/Horizon* encontrar un horizonte para la ráfaga a enviar. Como era de prever, conforme aumenta la tasa del tráfico de background aumentan las pérdidas, aunque en menor medida que la observada con la misma variación de parámetros en el algoritmo de ensamblado por límite en tamaño. También se puede observar como las pérdidas se reducen conforme se aumenta el *Timeout* del algoritmo de ensamblado por límite de tiempo.

Retardo Medio:

El retardo medio sufrido por el algoritmo de ensamblado limitado en tiempo no dependerá tanto del tráfico de *background* como en el caso del algoritmo de ensamblado limitado en tamaño, ya que en este último el tráfico de *background* afecta en gran medida a cuando se completa la ráfaga y por tanto al tiempo que pasan los datos en el *buffer* electrónico del *edge router* antes de ser ensamblados en una ráfaga y enviados a su destino.

Como muestra la “Fig. 24”, el retardo sufrido por el algoritmo de ensamblado limitado en tiempo es inferior al observado en la “Fig. 19” para la mayoría de tamaños de ráfaga, quedando en

orden similar al retardo sufrido por ráfagas del orden de 20 KB a 60 KB. También hay que destacar que la variación de este retardo es mucho menor, simplificando una previsión acotada de la llegada de la ráfaga al destino. En lo que respecta a cómo afecta la variación de la tasa de tráfico de background, el efecto apreciado en la gráfica es muy pequeño, al contrario de lo observado en el algoritmo de ensamblado limitado en tamaño, donde afecta directamente al tiempo en que tarda en reunir los datos necesarios para formar la ráfaga.

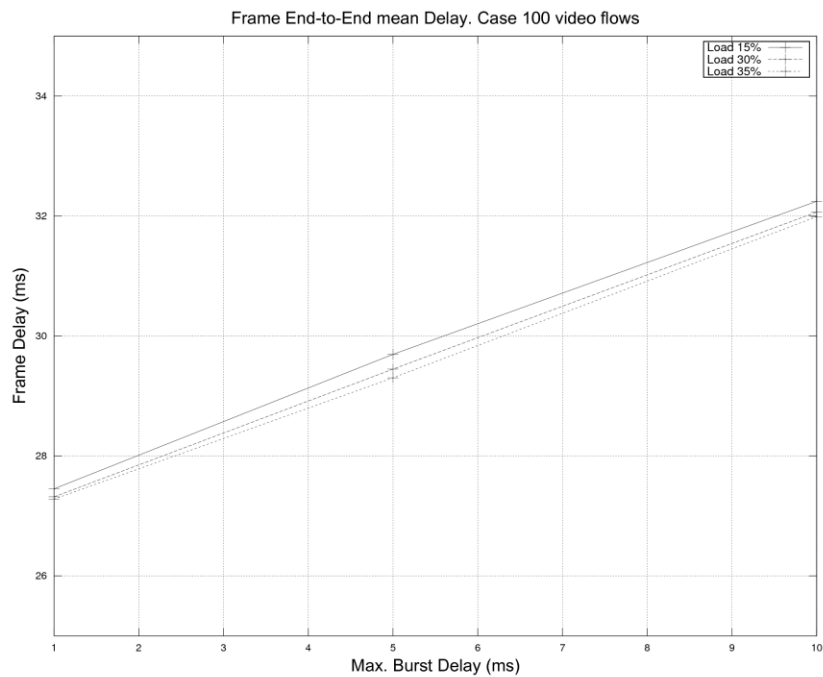


Fig. 24. Retardo medio de trama para distintas cargas de tráfico

Tamaño medio de las interrupciones:

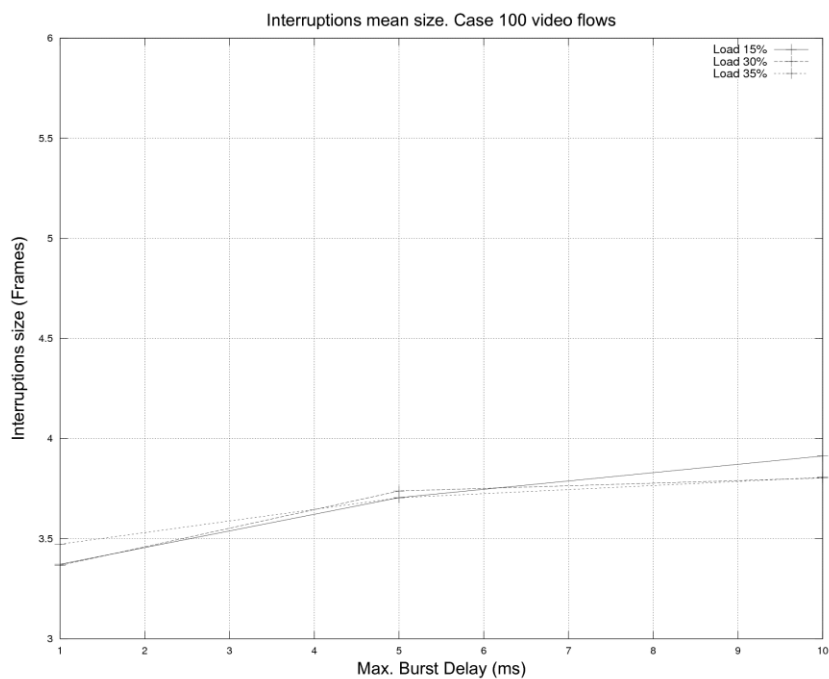


Fig. 25. Tamaño medio de las interrupciones para distintas cargas de tráfico

El tamaño medio de las interrupciones debe aumentar conforme aumenta el límite temporal del algoritmo de ensamblado limitado en tiempo, ya que, al igual que ocurre con el algoritmo de ensamblado limitado en tamaño, un límite mayor, implica un tamaño mayor de ráfaga y por lo tanto el volumen de datos perdidos con la pérdida de una ráfaga es mayor, y además estos datos forman varias tramas consecutivas, que es lo que se mide mediante las interrupciones.

En la “Fig. 25” se aprecia como la variación del tamaño medio de las interrupciones tiene una variación similar a la percibida en el algoritmo de ensamblado limitado en tamaño en la “Fig. 21”, creciendo conforme aumenta el límite temporal y estando en el rango comprendido entre los 3 y 4 *frames* de media. La variación con respecto a las distintas tasas de tráfico de *background* analizadas no es muy importante al igual que en los parámetros anteriores de retardo y pérdidas ya que no tiene un efecto tan directo como en el caso del algoritmo de ensamblado limitado por tamaño.

Tamaño medio de ráfaga:

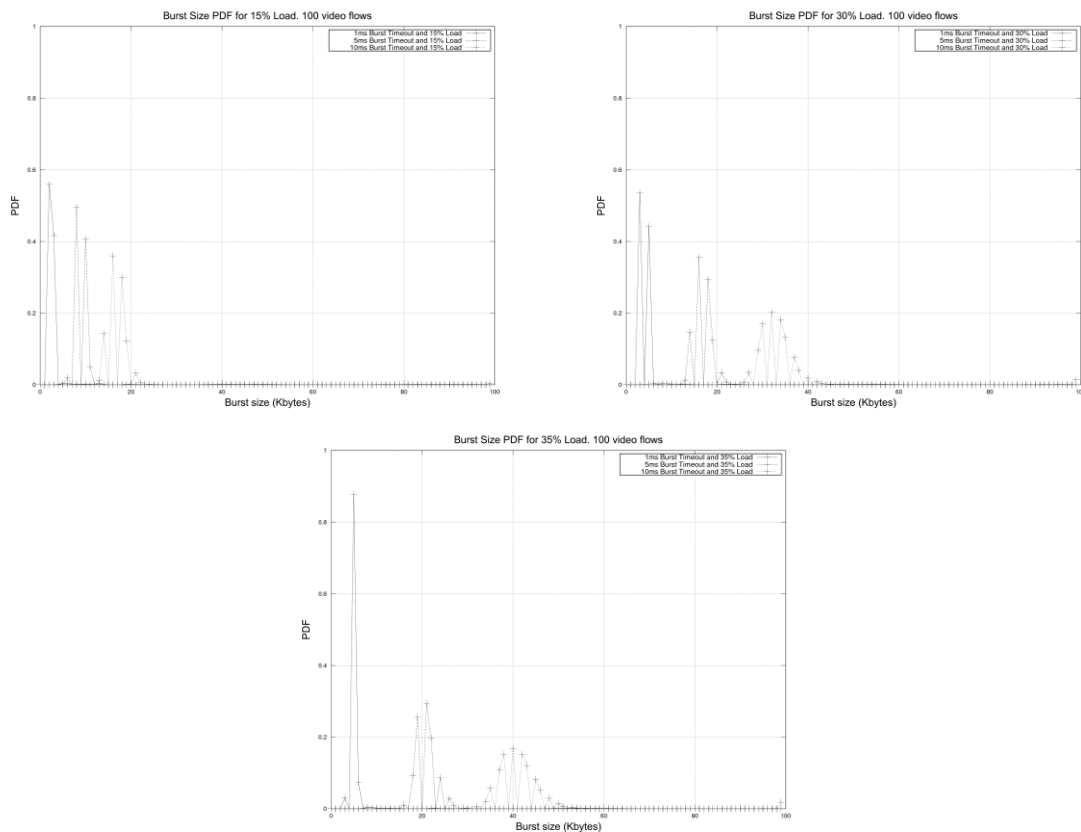


Fig. 26. Tamaño medio de ráfagas para distintas tasas y límites de tiempo

El tamaño de las ráfagas aumentan según lo esperado conforme aumenta el valor del límite temporal al igual que ocurre en el Escenario 1. Adicionalmente se comprueba fácilmente en la “Fig. 26” como el aumento de carga afecta directamente al tamaño de las ráfagas enviadas en el

algoritmo de tiempo siendo estas mayores cuando aumenta la carga llegando prácticamente a duplicar su tamaño en el cambio de una carga del 15% a una carga del 35%.

V.2.2.3. Algoritmo de ensamblado híbrido

Retardo Medio:

En el caso del algoritmo de ensamblado híbrido, el retardo sufrido debe ser inferior al sufrido tanto en el caso del algoritmo de ensamblado limitado en tiempo como el del algoritmo de ensamblado limitado en tamaño ya que, en caso de una carga de tráfico elevada se transmitirá la ráfaga cuando el *buffer* alcance el tamaño máximo de ráfaga, y en caso de que la carga de tráfico sea baja, se transmitirá la ráfaga cuando se cumpla el tiempo establecido en el límite temporal.

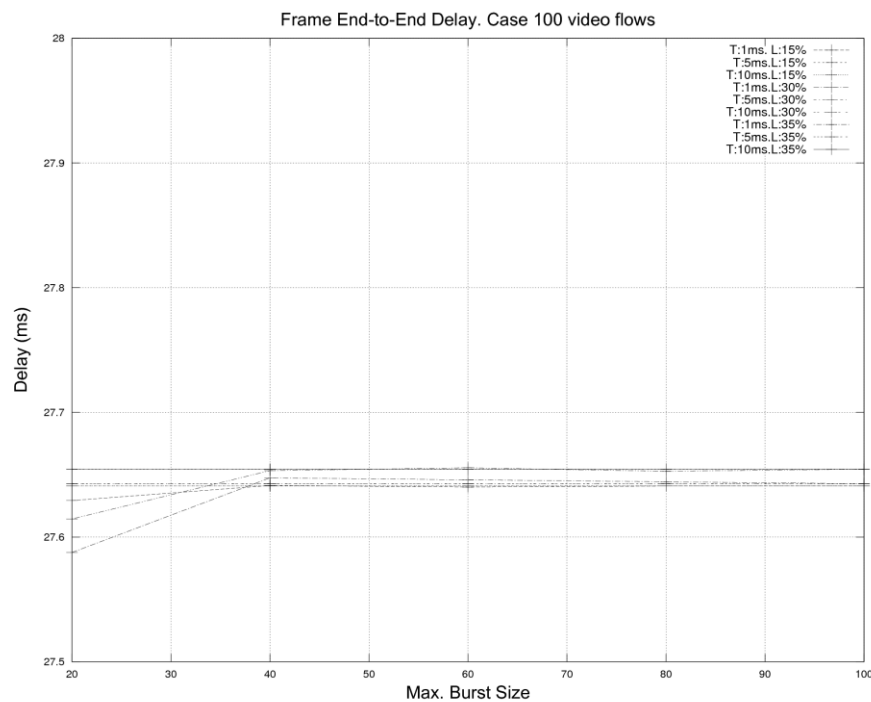


Fig. 27. Retardo medio extremo a extremo en el algoritmo híbrido para diversas cargas de tráfico

Como se aprecia en la “Fig. 27”, el retardo sufrido sólo se ve reducido con los tiempos máximos de ráfaga de 1ms y tamaños de ráfaga de 20KB, siendo más bajo este retardo conforme mayor es la carga en la red. Para valores mayores de tamaño máximo de ráfaga se mantiene bastante constante ya que las ráfagas se envían mayoritariamente al vencer el límite temporal, y por lo tanto siempre con el mismo retardo.

Pérdidas:

En el apartado de pérdidas, como se puede observar en la “Fig. 28”, lógicamente estas aumentan con la carga en la red siendo este el parámetro que más variación provoca. Debido a esto se ha incluido en la gráfica una simulación con más carga para ver como aumentan las pérdidas. En

segundo lugar el tiempo límite produce variaciones importantes también ya que en lo que se refiere al tamaño máximo de ráfaga sólo es apreciable la diferencia en los valores más bajos de 20KB y 40 KB entre los cuales se reduce conforme mayor es el tamaño ya que se reducen las pérdidas, sobretodo en el *edge router*. Para valores más elevados de tamaño de ráfaga, las pérdidas serán aproximadamente las relativas al protocolo de ensamblado limitado en tiempo.

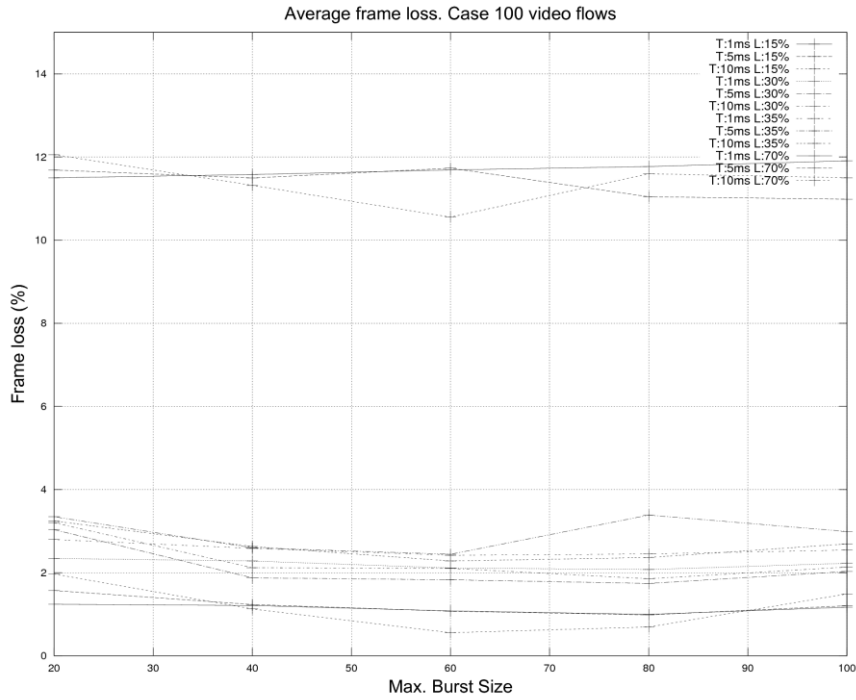


Fig. 28. Porcentaje de tramas perdidas en el algoritmo híbrido

Tamaño medio de las interrupciones:

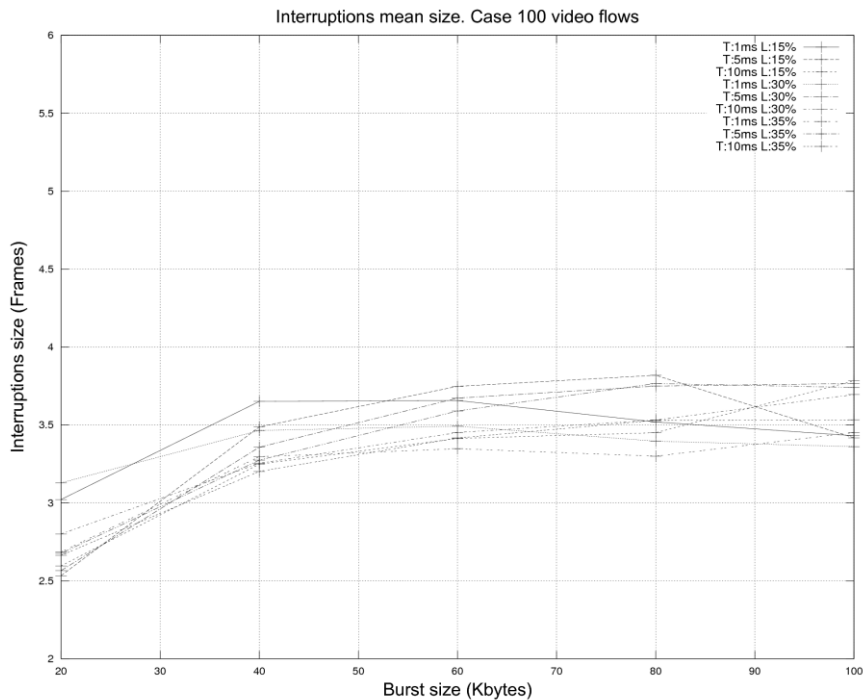


Fig. 29. Tamaño medio de las interrupciones en el algoritmo híbrido

Se ha visto en los otros algoritmos que el tamaño medio de las interrupciones aumenta conforme aumenta el tamaño y conforme aumenta el límite de tiempo respectivamente en los algoritmos limitados en tamaño y en tiempo. Por tanto en el algoritmo híbrido debe ir aumentando conforme ambos parámetros vayan aumentando. En la “Fig. 28” se aprecia como esto se cumple mayormente aunque a partir de 40 – 60 KB parece estabilizarse el tamaño. Esto se debe principalmente a que, debido a la carga de tráfico de background, y como se podrá comprobar en el siguiente apartado, el tamaño de las ráfagas enviadas en muy pocos casos supera los 40 KB, por lo tanto a partir de este tamaño, el parámetro de tamaño máximo de la ráfaga pierde importancia a favor del límite de tiempo impuesto.

Tamaño de ráfaga:

El tamaño de ráfaga variará según los tres parámetros, el principal por más restrictivo, y por tanto el que más afecta en este estudio, es la variación del límite de tiempo, en segundo lugar la variación del tamaño máximo de ráfaga y en tercer lugar la variación de la carga en la red.

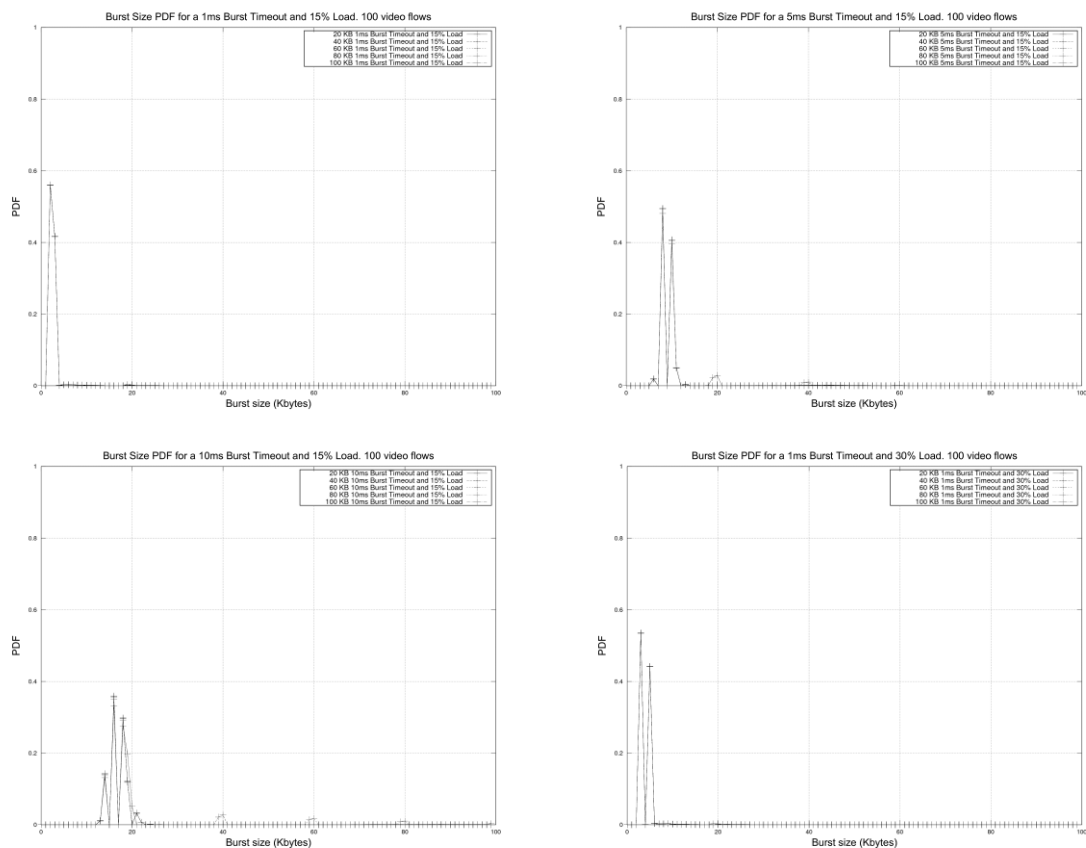


Fig. 30. Tamaño de ráfaga para el algoritmo de ensamblado híbrido con diversas cargas de tráfico

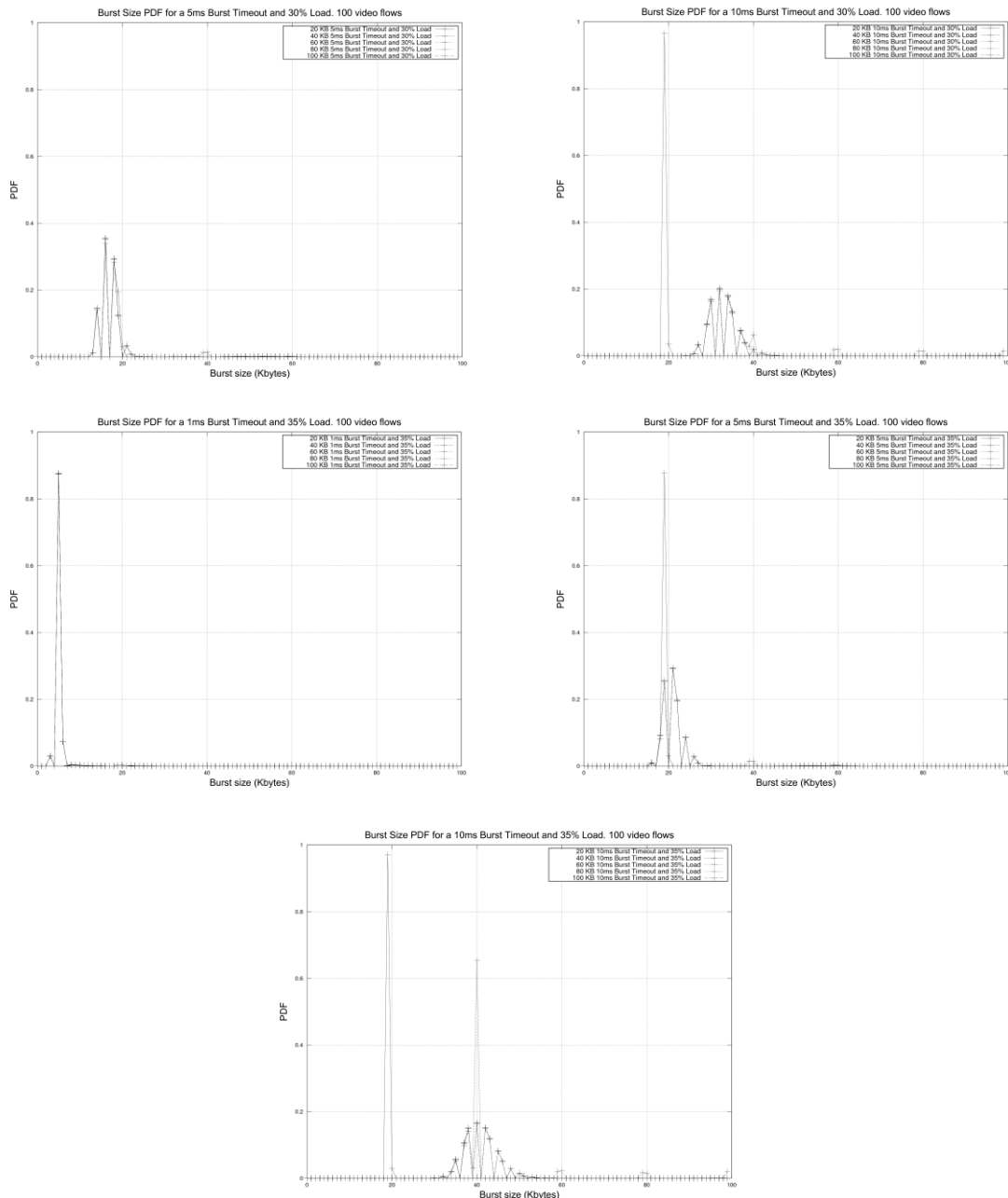


Fig. 31. Tamaño de ráfaga para el algoritmo de ensamblado híbrido con diversas cargas de tráfico

Como se puede apreciar en “Fig. 30” y “Fig. 31”, el tamaño de las ráfagas enviadas depende mayoritariamente del límite temporal impuesto. En el caso del límite de 1 ms, las ráfagas se centran en tamaños inferiores a los 10 KB quedando sólo alguna ráfaga residual de mayor tamaño. Para un límite temporal de 5ms se puede apreciar que la carga se sitúa mayoritariamente en torno a los 20 KB de tamaño para cargas de 30% y 35% quedándose la carga de 15% en valores en torno a los 10 KB. Por último, para un límite de 10 ms se observa la misma progresión, llegando esta vez a situarse las ráfagas en torno a los 30-40 KB de tamaño para las cargas más elevadas mientras que la carga del 15% de tráfico apenas llega a los 20 KB. Estos resultados son lógicos debido a que cuando más tráfico se transmite mayor será el tamaño de ráfaga a enviar cuando venza el *timeout*.

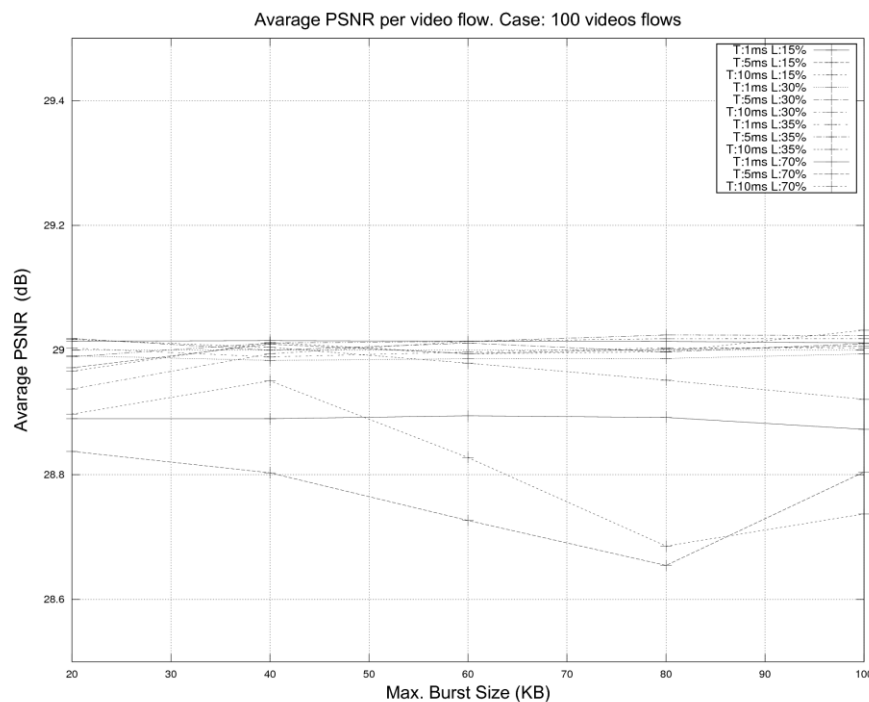
PSNR:

Fig. 32. PSNR medio obtenido en el algoritmo de ensamblado híbrido

El PSNR medio obtenido apenas sufre variaciones para valores inferiores a 40 KB, es decir en los valores que son enviados debido a un límite de tamaño máximo de ráfaga y para valores de timeout elevados, en las ráfagas que se envían debido al límite de tiempo el PSNR se mantiene prácticamente constante como se aprecia en la “Fig. 32”. Para detectar alguna variación de apenas unas décimas ha sido necesario introducir un tráfico de *background* mucho mayor. Los parámetros óptimos vistos los datos obtenidos son el uso de límite temporal de 1ms o de 5ms con tamaño máximo de ráfaga no inferior a 40 KB.

VI. CONCLUSIONES

Tras el análisis de más de 35.000 vídeos, habiendo simulado y analizado los resultados obtenidos mediante un pc Intel Duo 1.73GHz con 2GB de RAM a lo largo de más de 1300 horas de procesado, y a pesar de que debido a estas limitaciones técnicas no se han podido simular cargas de red mayores, se puede concluir que la transferencia de *streaming* de vídeo a través de redes *OBS* con protocolo *LAUC/Horizon* ofrece una calidad baja, ya que con una carga en un *core router* del 35% la transmisión de video sufre unas pérdidas en torno al 3% o 4% subiendo hasta un 12% al añadirle otro *edge router* produciendo una carga en el *core router* de aproximadamente un 70% de la capacidad del enlace (2.5 Gb).

Aunque sea capaz de mantener unos valores de PSNR elevados, presenta unas pérdidas concentradas en interrupciones de tamaños medios de 3 o 4 *frames*, lo que produce una pérdida de calidad subjetiva importante, ya que se pueden apreciar en muchos casos pequeños cortes en los

vídeos. La naturaleza de este tipo de pérdidas, es decir, el porqué muestra este tipo de interrupciones de tamaño elevado, es la naturaleza misma del protocolo *OBS* ya que al reunir una gran cantidad de paquetes en una ráfaga, la pérdida de una sola ráfaga influye en varios *frames* consecutivos. Ante este problema no sirve el uso de los sistemas habitualmente utilizados para mejorar la calidad de los vídeos como es la transmisión de vídeos por capas, ya que las distintas capas serían con gran probabilidad encapsuladas en la misma ráfaga produciendo el mismo tipo de resultados.

Entre las distintas opciones para el encapsulado de vídeo analizadas, la que demuestra una mejor calidad en los vídeos recibidos es la opción del algoritmo híbrido, ya que unifica las virtudes de ambos algoritmos eliminando la mayor parte de sus defectos. En concreto los parámetros óptimos para la transferencia de vídeo a través de una red OBS como la utilizada, es el uso de un *Timeout* de 1ms o, a lo sumo, de 5ms con tamaño máximo de ráfaga de 60 KB para mantener cierto compromiso entre las pérdidas sufridas y el retardo sufrido por los vídeos.

VI.1. DESARROLLOS FUTUROS

Una opción para solucionar los problemas aquí demostrados, sería el uso de vídeo por capas separadas entre ellas cierto timeout, lo que implicaría que las distintas capas se transferirían en distintas ráfagas, por lo que posibilitaría una recuperación de un vídeo, sin cortes, con el algoritmo de regeneración de vídeo adecuado en destino.

La otra vía de trabajo es el uso de otros protocolos alternativos al LAUC/Horizon que optimicen la conmutación de ráfagas en los *core router*, ya que este es el causante de la mayor parte de las pérdidas registradas en la red.

AGRADECIMIENTOS

Este trabajo ha sido realizado dentro del programa de Máster en Tecnologías y Redes de Comunicación del Departamento de Comunicaciones de la Universidad Politécnica de Valencia, el desarrollo del mismo ha tenido el apoyo del Grupo de Comunicaciones Multimedia del Instituto de Telecomunicaciones y Aplicaciones Multimedia.

He colaborado en el artículo: T.R. Vargas, J. Campos, J.C. Guerri, and S. Sales. "*Performance Evaluation of OBS Networks to provide QoE for Video Transport*". Workshop Red Temática Multilayer. IP sobre Redes de Transporte. Vilanova i la Geltru, España. 14 y 15 diciembre, 2009.

Quiero agradecer al Dr. Juan Carlos Guerri por dirigirme la tesina, a Tito R. Vargas por sus consejos los cuales fueron de gran ayuda para la puesta a punto del entorno, y a todos los que me han tenido que soportar durante el desarrollo de esta tesina.

En especial quiero dar las gracias a mi familia y amigos sin cuyo apoyo moral y económico no hubiese sido posible llegar a este punto.

Bibliografía:

- [1] Xiang Yu, Jikai Li, Xiaojun Cao, Yang Chen, and Chunming Qiao *Traffic Statistics and Performance Evaluation in Optical Burst Switched Networks*. JOURNAL OF LIGHTWAVE TECHNOLOGY, VOL. 22, NO. 12, DECEMBER 2004.
- [2] J. Padhye et al. *Modeling TCP Reno Performance: A Simple Model and Its Empirical Validation*. IEEE/ACM Transactions on Networking,
- [3] Xiang Yu., Chunming Qiao., *TCP Performance over OBS Networks with Multiple Flows Input* Broadband Communications, Networks and Systems, 2006. BROADNETS 2006. 3rd International Conference on, vol., no., pp.1-10, 1-5 Oct. 2006
- [4] Gowda, S., Shenai, R.K., Sivalingam, K.M., Cankaya, H.C., *Performance evaluation of TCP over optical burst-switched (OBS) WDM networks*, Communications, 2003. ICC '03. IEEE International Conference on , vol.2, no., pp. 1433-1437 vol.2, 11-15 May 2003.
- [5] C. Atici and M. O. Sunay, *High data-rate video broadcasting over 3G wireless systems*, IEEE Transactions on Broadcasting, vol. 53, no. 1, pp. 212–223, Mar. 2007.
- [6] Kyungtae Kang, Won J. Jeon, Kyung-Joon Park Roy H. Campbell, and Klara Nahrstedt. *Cross-layer Quality Assessment of Wireless Video Transmission over Mobile Broadcast Networks*. <http://www.ideals.illinois.edu/bitstream/handle/2142/10841/UIUCDCS-R-2009-3041.pdf?sequence=2>
- [7] Kevin James Wiebe, Anup Basu. *Improving image and video transmission quality over ATM with foveal prioritization and priority dithering*.
- [8] J.C. Guerri, P. Arce, P. Acelas. *Mechanisms for improving video streaming in mobile ad hoc networks*. Waves 2009, pp 20-30.
- [9] An Ge, Franco Callegati, and Lakshman S. Tamil. *On Optical Burst Switching and Self-Similar Traffic*. IEEE COMMUNICATIONS LETTERS, VOL. 4, NO. 3, MARCH 2000
- [10] Chunming Qiao, Myungsik Yoo. *Optical Burst Switching (OBS) A New Paradigm for an Optical Internet*, Lab for Advanced Network Design, Evaluation and Research (LANDER)
- [11] Vargas Hernández, Tito Raúl. *Simulación de servicios de tiempo real sobre redes OBS*. Tesina Final de Master en Tecnologías, Sistemas y Redes de Comunicación (Universidad Politécnica de Valencia)
- [12] Vinod M.Vokkarane, Karthik Haridoss, and Jason P.Jue. *Threshold-Based Burst Assembly Policies for QoS Support in Optical Burst-Switched Networks*. Center for Advanced Telecommunications Systems and Services. The University of Texas at Dallas
- [13] Kostas Ramantas, Kyriakos Vlachos, Óscar González de Dios, Javier Aracil and Carla Raffaelli, *Timer-based burst assembly algorithms and their effect in TCP Congestion window*, OSA J. Optical Networks, vol. 7, Iss. 5, pp. 487-495, May 2008
- [14] Pastor A. *Burst formation strategies in OBS* . U. Navarra. <https://www.tlm.unavarra.es/investigacion/seminarios/>
- [15] W. Leland, M. Taqqu, and D.Wilson, *On the self-similar nature of Ethernet traffic (extended version)*. IEEE/ACM Trans. Networking, vol. 2, pp. 1–15, Feb 1994.
- [16] Burak Kantarci, Sema Oktuk. *Adaptive threshold based burst assembly in OBS Networks*. 1-4244-0038-4 2006 IEEE CCECE/CCGEI, Ottawa, May 2006

- [17] J. N. T Sanghavi, H. Elbiaze and M.F. Zhani. *Adaptive Burst Assembly Mechanism for OBS Networks Using Control Channel Availability*. LTIR Lab, Computer Science Department, University of Quebec at Montreal, Montreal, Quebec, Canada.
- [18] Yuan Chi, Huang Junbin, Li Zhengbin, Xu Anshi. *A Novel Burst Assembly Algorithm for OBS networks Based On Data-length Time-lag Product*. National Laboratory on Fiber-Optic Communication Networks & Advanced Optical Communication Systems Peking University
- [19] Angeliki Sideri and Emmanouel A. Varvarigos. *New Assembly Techniques for Optical Burst Switched Networks Based on Traffic Prediction*. Computer Engineering and Informatics Dept, University of Patras, Greece.
- [20] <http://www.ipoque.com/news-and-events/news/ipoque-internet-study-2008-2009-finds-web-and-streaming-outgrows-p2p-traffic.html>
- [21] Óscar González de Dios, Ignacio de Miguel, Víctor López. *Performance evaluation of TCP over OBS considering background traffic*.
- [22] Burak Kantarci a, Sema F. Oktug a,*, Tu` lin Atmaca. *Performance of OBS techniques under self-similar traffic based on various burst assembly techniques*. Computer Communications 30 (2007) 315–325

Performance Evaluation of OBS Networks to provide QoE for Video Transport

T.R. Vargas, J. Campos, J.C. Guerri, and S. Sales.

Abstract—Optical Burst Switching (OBS) has been proposed to be a technology for implementing the next generation optical Internet. In this architecture, burst assembly algorithms have an important influence in the traffic pattern that characterizes this sort of optical networks. On the other hand, traffic coming from new multimedia applications that have real time and bandwidth constraints has been experimented a rapid increment. Consequently, we consider important to evaluate the performance of the real time traffic over OBS networks. In this paper, we evaluate the video transmissions performance over OBS networks, for different burst assembly strategies. Through event driven simulation, we conclude about the optimal configuration values of the assembly algorithm in order guarantee high quality video transmissions.

Index Terms—Optical burst switching, burst assembly algorithm, video streaming.

I. INTRODUCTION

Network operators are willing to implement new architectures and technologies in the different network segments, access, metropolitan and backbone, in order to reduce operational costs, so that they can offer broadband services using 100-Mbps access. Reducing capital expenditures and operational expenditures ensures network operators to bear the overall cost of maintaining high speed network without increasing cost of the broadband access service.

Regarding to the metropolitan segment there is a great interest in Optical Burst Switching because OBS has arise as a switching proposal [1] that could be implemented in short term with the current photonics technology advances and, it could take advantage of the SDH/SONET links deployed in the current metropolitan network. Also, OBS allows a decrease of the number of physical ports required per node to support different traffic channels, or lambdas, as well as reducing opto-electronic conversions. The aggregation of traffic at the network edge nodes enables a significant reduction in cost, and that is why manufacturers and operators are leading projects for the development and implementation of OBS metropolitan networks [2].

An OBS network is formed by edge nodes interconnected by core nodes through Wavelength-Division Multiplexed (WDM) optical links. In the ingress nodes of the OBS network packets are buffered in assembly queues and then

aggregated into a burst based in a burst assembly criteria. The bursts generated are optically switched by the core nodes to reach the egress node following a path reserved by a Burst Header Packet (BHP) previously sent with an offset time. The burst size or the burst time depends on the burst assembly algorithm used in the ingress node.

In the case of OBS metropolitan networks, each node has a tunable transmitter and a receiver fixed at a certain wavelength. If a node wants to communicate with another node that simply tune its transmitter to the appropriate wavelength. To avoid collisions between traffic flows with different sources but share the same destination, a protocol medium access control is used, so that each node in the network knows the time slot and wavelength to transmit. These allows a virtual mesh interconnection between different nodes in the metropolitan network using a single lambda. Another important feature is that nodes could route not only metropolitan traffic, but also interconnection traffic, destined for the backbone segment, using few channels.

With the next generation OBS metropolitan network and an access network supported by fiber to the home (FTTH) network, operators can enhance their service offerings multimedia based communications such as video conferencing, VoIP, IPTV, among others, as well as improving its basic data transfer services.

In a converged services network of voice, data, and video, video transmission has a significant influence on the network design, due to their needs like high bandwidth, real-time constraints and highly demanded from end users. Taking into account the need of a high speed metropolitan networks, with low operational cost, and the characteristics of multimedia traffic, it is important to study the quality of video transmissions over broadband IP OBS networks in order to establish the optimal parameters in terms of video coding, as well as the optimal configuration parameters for the communications network to guarantee delay, loss rate, jitter and, high quality reproduction of the video streams.

This paper presents an evaluation of video transmissions over OBS networks taking into account mesh architecture, concluding on the optimal parameters of the assembly algorithm used in the edge node of the OBS network. Length, time and hybrid burst assembly proposals are evaluated.

II. VIDEO QUALITY EVALUATION OVER OBS

Video quality measurements must be based on the

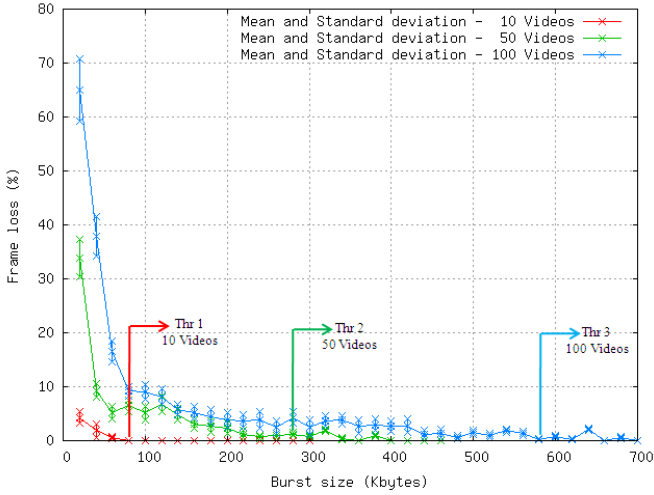


Fig. 1. Average frame losses for different cases of number of video flows and burst size threshold values.

perceived quality of the actual video being received by the end users. In order to make the quality assessment of video transmission over OBS networks we take into account two approaches to measure digital video quality [3], subjective quality measures and objective quality measures.

The subjective quality metrics are used to obtain the impression of a human user watching a video. Since, subjective metrics provide important information and their determination is costly as real humans have to watch videos, objective metrics have been developed to emulate the quality impression of a video.

MOS (Mean Opinion Score) is the (subjective) human impression of video quality as described in Table 1, and it is normally assigned by a group of experts, but it can be approximated with PSNR calculations on video frames. PSNR score express the degree of noise introduced through burst losses in the OBS network. PSNR calculations are performed by comparing reference video frames with reconstructed video frames. The conversion table used to obtain MOS metric from PSNR values is the following:

TABLE I. PSNR to MOS conversion table

PSNR [dB]	MOS	Impairments
> 37	5 (Excellent)	Imperceptible
31- 37	4 (Good)	Perceptible, not annoying
25 – 31	3 (Fair)	Slightly annoying
20 – 25	2 (Poor)	Annoying
< 20	1 (Bad)	Very annoying

Besides PSNR and MOS to make a complete quality assessment is possible to obtain results from parameter such as, frame delay and frame losses. The frame losses affect the video quality, producing big or small local distortions per video sequence depending of what kind of video frame is lost (I, P or B frame). Losing one I frame or one P frame could yield a video distortion that is propagated through all the group of pictures (GOP).

An MPEG video consists of many video sequences, each of which has many groups of pictures in it [4]. Typically a GOP

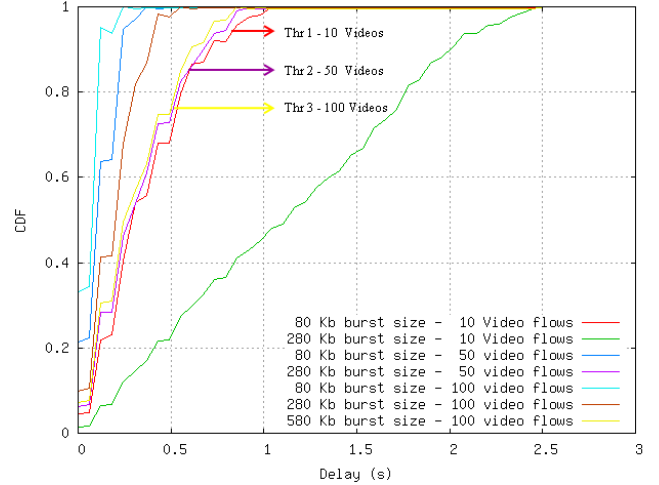


Fig. 2. Cumulative distribution function of the frame delay using static length-based assembly for different amounts of video flows.

has a single Intra frames (I frame) and many Predictive frames and Bidirectional frames (P frames and B frames). Although there is no limit on the size of GOP, values around 15 is common. An MPEG-4 encoder generates these three types of frames. I frames contain information from encoding a still image. P frames are encoded from the previous I-frames or P frames. B frames are encoded bi-directionally from the preceding and the following I and P frames.

Taking into account video quality assessment parameters and the OBS network architecture and protocols we have configured an OBS simulation environment with possibility of video stream quality evaluation. The video specific metrics were obtained with Evalvid framework [3], which is a set of tools for the generation of video traffic, and the evaluation of video transmissions. Combined with an ns-based OBS simulator it results in a powerful video-over-OBS modeling tool. The final simulator architecture consists of the following modules:

- An OBS network module for ns-2 simulator implemented in C++ and partly based on OIRC obs-ns simulator.
- An MPEG-4 encoder and decoder (ffmpeg open source tool).
- A video sender utility, which generates a video trace file with the intrinsic details of the mpeg4 file (specifically video frames, packets within frames, and frame transmission times).
- A video traffic generator, which generates video traffic according to the aforementioned video trace file.
- A set of tools that evaluate some perceptual-based metrics of video transmission, and specifically MOS score.

III. BURST ASSEMBLY ANALYSIS

Different burst assembly techniques could be implemented in the ingress node of an OBS network: length-based, time-based and hybrid schemes. Using a length-based burst assembly technique, when a constant burst size threshold is reached, the incoming packets buffered in the assembly queue are assembled into a burst. Then, the burst is forwarded to an

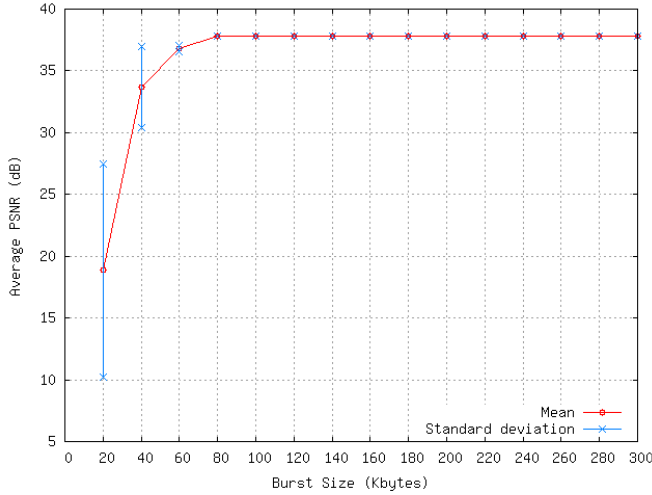


Fig.3. Average PSNR and standard deviation. Case 10 video flows.

output queue and immediately the BHP is send to do the reservation in the down link to the corresponding burst. In the time-based aggregation strategy the burst formation time is constant and it is established beforehand. When a packet arrives at a burst assembly queue, a time counter is started. Then, incoming packets are buffered in the burst assembly queue until the timeout is reached. Then, the burst is created with the content of the burst assembly queue, and then the burst is queued for transmission. The time counter is restarted to zero and remains in this value until the next packet arrives to the queue. Hybrid schemes use time and size thresholds, where the burst is assembled when any of the thresholds is reached.

In these work burst assembly algorithms were evaluated through experimentation for video transmission, in the video-over-OBS Network Simulator 2 modeling tool. We modeled an OBS network over the 14-node NSF network topology. The data links between nodes are bidirectional at 2.5 Gbps with two channels available per link. One wavelength is dedicated to data channel and the other one to control channel.

With each burst assembly scheme, different simulations are done varying the number of traffic sources, 10, 50 and 100 video flows, to make an evaluation that take into account the amount of video traffic. Other parameter that is dynamically modified through the simulation is the maximum burst size or the burst time that could be reached. So, it is possible to find the optimal burst size or time parameter that assures quality video transmissions.

In the evaluation the focus lies in finding if these basic assembly strategies fulfill quality video transmission requirements like frame delay and frame losses. Result of the different scenarios is presented in the next subsections. During each simulation cycle, we obtain detailed log files of burst transmissions as well as burst and frame losses. At the end of the simulation cycle, these are used to calculate the loss ratio, as well as the video perceptual quality. In the video quality calculation phase, we focus on a specific video stream, reconstructing the reference video file by taking into account frame losses that occurred during transmission over the OBS

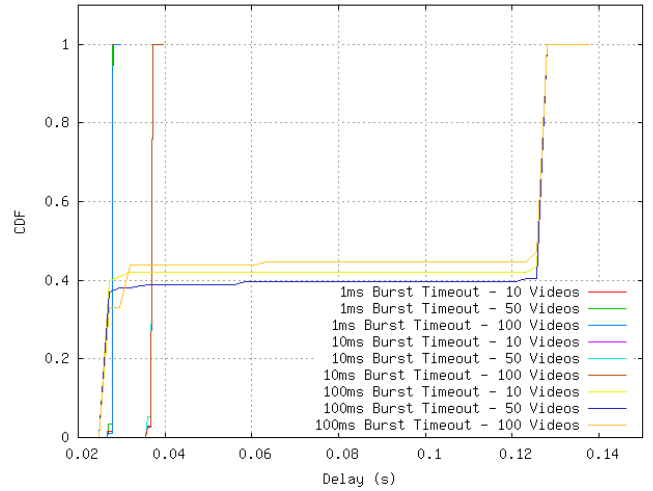


Fig.4. CDF of the frame end-to-end delay, assembly times 1ms, 10ms and 100ms. Cases 10, 50, 100 video flows.

network.

A. Length based burst assembly algorithm

With a length-based burst assembly algorithm, we have found that the quality of the received videos get worse when the number of active video flows increase and at the same time the burst size threshold remains static. Fig. 1 shows video frame losses, as it could be observed frame losses increment when video traffic flows increment because the burst size threshold value is constant. Also, the Fig. 1 shows that there is a threshold value at which, depending of the number of active video flows, higher frame losses can be avoided. These valid burst size thresholds are 80, 280 and 580 Kbytes for 10, 50 and 100 videos, so it is necessary that the burst size threshold must be adaptive. However, as shown in the Fig. 2, there is a tradeoff between the burst size threshold and the number of video flows to have a constant frame delay and lower frame losses. If it is used as optimal burst size the threshold presented in the Fig. 1, 80, 280 and 580 Kbytes, the frame losses are lower and it is possible to guarantee that 70 % of the frames have 500 ms average delay in each case of number of video flows.

Frame losses are related to video quality of the video in reception as we could be observed in Fig. 3 that represents the PSNR and its standard deviation of the received video.

B. Time burst assembly algorithm

When evaluating video transmission with a time-based burst assembly strategy, we have found that time based scheme improves the length-based strategy in terms of PSNR and frame losses. The average PSNR values obtained were constants for all videos in each case of traffic load with the different timeouts evaluated. The average PSNR of the video frames values were 37.17 dB. This PSNR values indicates high quality video transmissions.

Implementing a time-based burst assembly strategy the frame end-to-end delay is guaranteed. The delay through the entire path (propagation delay plus burst switching times) without the burst assembly time is approximately 0.027 s. Fig.

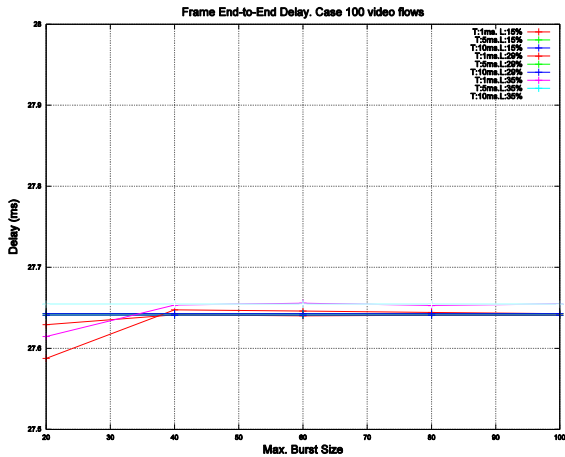


Fig. 5. Average frame end-to-end delay. 15%, 29% and 35 % background traffic load

4 represent the CDF of the frame end-to-end delay. It shows the delay improvement using time-based strategy. As we could see in the figures, the frame delay varies from time propagation plus burst timeout, (for the frame packets that arrive at the beginning of the burst formation) to a value near the propagation time (for UDP packets at the burst tail). For instance, in the case of 100 ms timeout the frame delay value is in the range of 0.027s to 0.127 s. The improvement in terms of PSNR, frame losses and end-to-end frame delay using a time-based assembly strategy is due mainly to:

- The burst timeout guaranteed limited delay.
- The burst timeout makes possible the formation of burst with optimal sizes to carry out UDP packets of variable size video frames, minimizing frames losses and maintaining the PSNR constant.

C. Hybrid burst assembly algorithm with background traffic

In the evaluation using a hybrid burst assembly algorithm we complete the simulation scenario adding background traffic using self-similar traffic sources. Hurst parameter equal to 0.9, 1600 bytes packet size and, different packets rates per second, 1000 2000 and 2500 that correspond an average traffic load of 15 %, 29 % and 39 % respectively.

In this scenario, the delay is below compare with the case of the time based and the length based burst assembly algorithms. This is because in the case of a high traffic load the burst is transmitted when the buffer reaches the maximum burst size, and if the traffic load is low, the burst is transmitted when the time limit is fulfilled, see Fig. 5.

Regarding to video quality assessment the videos at reception presented PSNR acceptable values. In the Fig. 5 we present results using the objective quality metric call interruptions. The interruptions represent consecutive frame that have been lost during the video transmission. The interruptions are higher when the burst generated are larger because high burst time or burst length, a large burst could carry a large number of video frames. In the Fig. 5 is shown

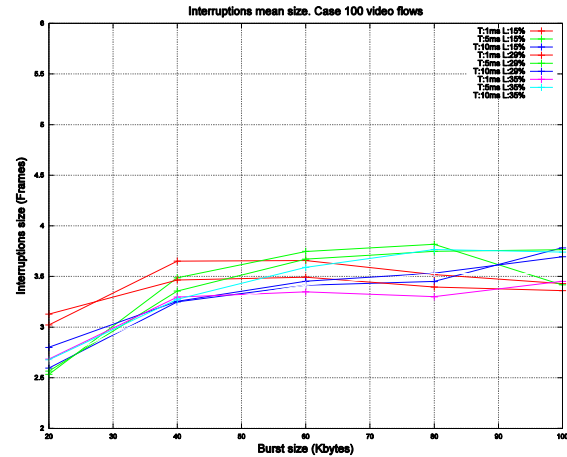


Fig. 6. Video interruptions mean size. 15%, 29% and 35 % load.

that fewer interruptions occur when the burst are not greater than 60 Kbytes. Using the hybrid scheme the delay is reduced but we still have to configure the maximum burst size should be kept around 80 Kbytes to avoid large interruptions and 1 ms burst time out to ensure a bounded delay.

IV. CONCLUSION

In this paper, we presented a video-over-OBS modeling tool that makes possible the study of the quality of video transmission over OBS networks, obtaining results of video quality metrics and network performance metrics. Also, burst assembly algorithms were evaluated using the simulation framework. The length based, time based and hybrid scheme were evaluated to present the possible optimal configuration for digital video transmissions over OBS.

ACKNOWLEDGMENT

The work described in this paper was carried out with the support of Spanish Ministry of Science and Innovation under contract TEC2008-02552-E/TEC, IPoTN.

REFERENCES

- [1] Yang Chen, et al., "Optical burst switching: a new area in optical networking research," *Network*, IEEE, vol.18, no.3, pp. 16-23, June 2004.
- [2] Dunne, J.; Farrell, T.; Shields, J., "Optical Packet Switch And Transport: A new metro platform to reduce costs and power by 50% to 75% while simultaneously increasing deterministic performance levels," *Transparent Optical Networks*, 2009. ICTON '09. 11th International Conference on , vol., no., pp.1-5, June 28 2009-July 2 2009.
- [3] J. Klaue, B. Rathke, and A. Wolisz, "EvalVid - A Framework for Video Transmission and Quality Evaluation", In Proc. of the 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation, Urbana, Illinois, USA, September 2003.
- [4] A. Lo, G. Heijenk, I. Niemegeers, "Performance Evaluation of MPEG-4 Video Streaming over UMTS Networks using an Integrated Tool Environment", International Symposium on Performance Evaluation of Computer and Telecommunication Systems, Philadelphia, PA, USA, July 24--28, 2005.