

Mejora de la Transmisión de Vídeo en Redes Vehiculares mediante Calidad de Servicio

P. Pablo Garrido Abenza, Pablo Piñol Peral, Manuel P. Malumbres, O. López Granado¹

Resumen— La transmisión de vídeo en redes vehiculares puede tener un gran número de aplicaciones. Sin embargo, éstas tienen unas propiedades que las diferencian de las redes cableadas y que dificultan la transmisión de contenido multimedia. El canal inalámbrico es un medio compartido, que tiene un ancho de banda limitado y una naturaleza cambiante a lo largo del tiempo. A todo ello se suma que la transmisión de vídeo requiere de un gran ancho de banda, y que la topología de la red cambia con mucha frecuencia debido a la velocidad a la que se mueven los vehículos. En este trabajo se realizan diversos experimentos mediante simulación utilizando diversas técnicas existentes con objeto de mejorar la transmisión de vídeo en este tipo de redes. Con un entorno de simulación basado en OMNeT++, Veins y SUMO, se analizan algunos modos de codificación con refresco de frames de tipo I, particionado de frames (*tiles*), y calidad de servicio mediante la asignación de prioridad dependiendo del tipo de frame. Los resultados de estos experimentos muestran que mediante la combinación de estas técnicas se consigue que la transmisión de vídeo en redes vehiculares sea más robusta.

Palabras clave— Redes vehiculares, Vídeo, HEVC, QoS

I. INTRODUCCIÓN

Las Redes Vehiculares o Vehicular Ad-hoc Networks (VANETs) tienen un gran potencial puesto que contribuyen a los llamados Intelligent Transportation Systems (ITS), proporcionando una serie de servicios en entornos urbanos e inter-urbanos, para conductores y pasajeros. Entre ellos, podemos mencionar la implementación de sistemas relacionados con la mejora de la seguridad en la conducción, de información de la situación del tráfico o previsión meteorológica, acceso a Internet, y aplicaciones de entretenimiento (*infotainment*).

Sin embargo, las comunicaciones en este tipo de redes tiene una serie de problemas. Por un lado, tenemos los problemas típicos de las comunicaciones inalámbricas, como un ancho de banda limitado, que al ser compartido por todos los dispositivos de la red genera interferencias con las señales de otros dispositivos, o de otras redes inalámbricas que se solapan en su zona de cobertura. Además, se producen diversos fenómenos como la atenuación de la señal con la distancia (*path loss*) o el tiempo (*fading*), la presencia de obstáculos (*shadowing*), y los rebotes debidos a refracción o reflexión (*multipath*). Por otro lado, debido a la movilidad inherente de los nodos de la red, la topología de la red es cambiante, y la alta velocidad de los vehículos limita el tiempo de comunicación. Todo ello se traduce en un aumento del tiempo de espera para el acceso al canal, y un aumento también

del número de paquetes perdidos debido a colisiones, reduciendo su rendimiento en comparación con las redes cableadas [1].

La transmisión de vídeo en redes vehiculares puede tener muchas aplicaciones, como la difusión de anuncios o información turística según nuestra posición, transmisión de vídeo en tiempo real o *video streaming*, videovigilancia, visualización del estado del tráfico en alguna zona, etc. Sin embargo, transmitir vídeo con calidad suficiente es muy complicado, puesto que requiere un gran ancho de banda, bajo retardo o latencia (*delay*) y una variación de retardo (*jitter*) acotada. Con el uso de técnicas de codificación o compresión de vídeo se reduce la cantidad de datos necesaria para su almacenamiento, así como el ancho de banda requerido para la transmisión. En los últimos años han surgido diversos estándares de codificación de vídeo, como High Efficiency Video Coding (HEVC) [2], que mejora las tasas de compresión de su predecesor H.264/AVC (Advanced Video Coding) [3]. Aun así, la calidad de vídeo percibida por el receptor puede verse muy afectada por los problemas ocurridos durante la transmisión, en particular, en escenarios de redes vehiculares. Para tratar de solucionar los problemas en la transmisión de vídeo en redes vehiculares, se suele recurrir a técnicas y mecanismos de control, recuperación y ocultación de errores que tratan de maximizar la calidad del vídeo percibida por el usuario. Estas técnicas, permiten dotar al servicio de transmisión de vídeo de un nivel determinado de calidad de servicio, Quality of Service (QoS). Diversos trabajos como [4] y [5] agrupan estas técnicas en varias categorías: (1) control de admisión y reserva de ancho de banda, (2) QoS a nivel aplicación, (3) diferenciación de tráfico a nivel Medium Access Control (MAC), y (4) adaptación del enlace en la capa física (PHY). Las técnicas de control de errores suelen actuar a nivel de aplicación. En caso de utilizar protocolos de comunicación asíncrona como Automatic Repeat Request (ARQ), el receptor debe enviar un acuse de recibo (ACK) en caso de haber recibido correctamente un paquete, y el emisor retransmitirá aquellos paquetes sin acuse de recibo. Sin embargo, esto no es válido durante el envío de una secuencia de vídeo en tiempo real, en la que se requiere una baja latencia. Otra alternativa son los protocolos de comunicación síncronos, como el uso de Forward Error Correction (FEC), que permiten reconstruir los datos perdidos a partir de la información redundante enviada en otros paquetes, siempre y cuando el número de paquetes perdidos no supere un determinado límite. Estos mecanismos no son adaptativos, por lo que se puede desperdiciar

¹Dpto. Ingeniería de Computadores, Universidad Miguel Hernández, Elche, e-mail: pgarrido, pablop, mels, otoniel@umh.es.

ancho de banda por los paquetes añadidos innecesariamente en caso de que la red no esté saturada, o, por el contrario, en caso de estar saturada podría no ser suficiente para poder restaurar todos los paquetes perdidos.

En cuanto al uso de QoS mediante diferenciación de tráfico a nivel MAC, éste puede realizarse de dos modos: (1) mediante planificación de colas, o (2) mediante la asignación de diferentes niveles de prioridad. Para el caso particular de la transmisión de secuencias de vídeo, este trabajo se centra en priorizar los paquetes en función del tipo de frame al que pertenecen (I, P, B). Se han planificado dos conjuntos de experimentos mediante simulación: (1) priorizando sólo los paquetes pertenecientes a los frames de tipo I, y (2) priorizando todos los paquetes de vídeo (frames I, P, B) respecto al resto de tráfico.

A la vista de los resultados, los modos AI o LPI4, con 6 u 8 tiles por frame, y priorizando todos los paquetes de vídeo (recomendación del estándar) parece ser la alternativa que proporciona los mejores resultados. Utilizando diferenciación de tráfico se obtiene una mejor calidad de vídeo que con el servicio *Best-effort* proporcionado por defecto del IEEE 802.11.

La estructura del artículo es la siguiente. Primero, en la sección II se hace un breve repaso de los estándares de comunicación utilizados en entornos vehiculares, como son el IEEE 802.11, y el IEEE 1609.4. A continuación, en la sección III se presentan algunas trabajos existentes en la literatura relacionados con el uso de QoS en la transmisión de contenido multimedia en redes inalámbricas. En la sección IV se describe en detalle el escenario utilizado para los experimentos, así como la secuencia de vídeo transmitida y los modos de codificación HEVC utilizados. Los resultados de los experimentos se discuten en la sección V. Por último, la sección VI finaliza el artículo y enumera algunos de los trabajos futuros.

II. ESTÁNDARES DE COMUNICACIÓN

En el estándar IEEE 802.11 [6] se define una subcapa Medium Access Control (MAC), así como varias capas a nivel físico (PHY). A pesar de que el IEEE 802.11 es el tipo de red inalámbrica más extendido, no incluye soporte para QoS.

El grupo de trabajo IEEE 802.11e [7] definió algunas extensiones al estándar IEEE 802.11 para proporcionar QoS a nivel de la capa MAC. Habilitando diferenciación de tráfico a nivel MAC es posible dar soporte a tráfico de distintas aplicaciones dependiendo de sus restricciones de QoS, como pueden ser las llamadas Voice over IP (VoIP), la videoconferencia, la videovigilancia, y cualquier otra aplicación en tiempo real. El estándar IEEE 802.11e introdujo la Hybrid Coordination Function (HCF), la cual define dos nuevos mecanismos de acceso que sustituyen a Distributed Coordination Function (DCF) y Point Coordination Function (PCF) del IEEE 802.11: el HCF Controlled Channel Access (HCCA) y el Enhanced Distributed Channel Access (EDCA). Este último, el EDCA, es el que deben implementar los

TABLA I: Correspondencia entre UP y AC.

User Priority (UP)	Access Category (AC)	Nombre
1	AC_BK o AC(0)	Background
2	AC_BK o AC(0)	Background
0	AC_BE o AC(1)	Best Effort
3	AC_BE o AC(1)	Best Effort
4	AC_VI o AC(2)	Video
5	AC_VI o AC(2)	Video
6	AC_VO o AC(3)	Voice
7	AC_VO o AC(3)	Voice

nodos de una red inalámbrica Ad hoc (sin infraestructura). A los paquetes a transmitir se les asigna una prioridad denominada User Priority (UP) a nivel aplicación, cuyo valor está comprendido entre 0 (la menor) y 7 (la mayor). Cuando la capa MAC recibe un paquete de las capas superiores, éste se clasifica según una de las cuatro categorías de acceso, o Access Categories (AC), es decir, que a cada AC se le asignan dos UP, tal como se muestra en la Tabla I. La categoría AC(3) es la de mayor prioridad, y la AC(0) la de menor prioridad.

Los nodos que implementan EDCA tienen 4 colas de servicio en la capa MAC (una para cada AC), con objeto de clasificar los paquetes de acuerdo a su prioridad. Si existen paquetes a transmitir en varias de las colas se produce una colisión interna (virtual), seleccionándose la cola de mayor prioridad. Además, la capa MAC también tratará de forma diferente a los paquetes a la hora de acceder al medio dependiendo de la prioridad asignada, en función de los límites inferior y superior de la ventana de contención o Contention Window (CW_{min} y CW_{max}), el Arbitration Inter-Frame Space Number (AIFSN), y la Transmission Opportunity (TXOP).

El parámetro AIFSN es un tiempo adicional que se debe esperar antes de transmitir una vez que se detecte que el medio está libre, y es específico para cada una de las colas de las ACs. En caso de que el canal esté ocupado, el dispositivo tendrá que iniciar un proceso de *backoff*, en el que tendrá que esperar un tiempo proporcional a un valor aleatorio comprendido en el rango $[0..CW]$, donde el valor inicial de la ventana de contención es CW_{min} . Si pasado ese tiempo el medio sigue ocupado, la ventana de contención se incrementará de forma exponencial, hasta llegar al valor máximo CW_{max} . Las colas de mayor prioridad tienen valores menores de CW y AIFSN para acceder al canal. Por último, el parámetro TXOP define un intervalo de tiempo en el que el nodo puede transmitir sin que exista competencia con otros nodos. Esto permite obtener un mayor rendimiento, que será mayor para la cola que tenga un mayor valor de TXOP, así como un incremento en la ocupación global del canal.

Sin embargo, las redes vehiculares, además de las características propias de las redes inalámbricas, tienen el problema de la alta movilidad de los vehículos, que provoca que la topología de la red cambie frecuentemente y las comunicaciones no puedan du-

TABLA II: Valores por defecto de los parámetros EDCA para IEEE 802.11p.

AC	CW _{min..max}	AIFSN	TXOP _{limit}
AC_BK	15..1023	9	0 ms
AC_BE	15..1023	6	0 ms
AC_VI	7..15	3	0 ms
AC_VO	3..7	2	0 ms

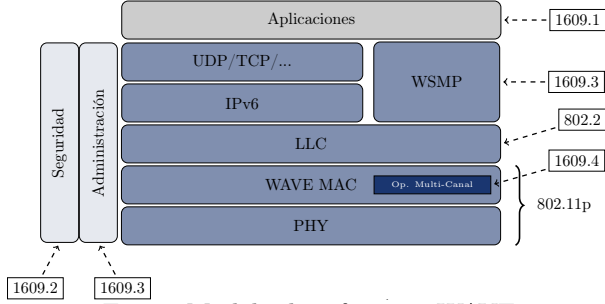


Fig. 1: Modelo de referencia WAVE

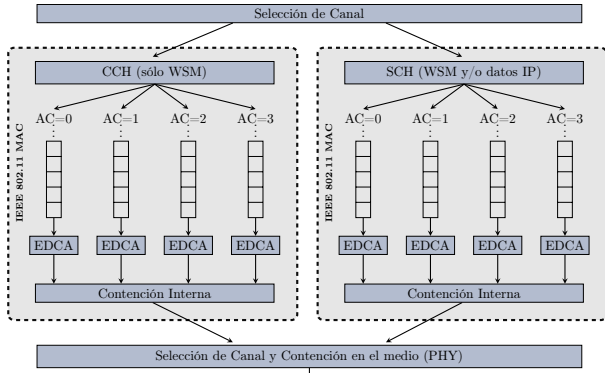


Fig. 2: Capa MAC de la arquitectura WAVE

rar mucho tiempo. Por ello, se propuso el conjunto de estándares IEEE 1609, conocido como Wireless Access in Vehicular Environments (WAVE), que proporciona una pila de protocolos de comunicación optimizada para entornos vehiculares (Fig. 1). En concreto, el IEEE 1609.4 [8] especifica las extensiones que es necesario realizar a la capa MAC del IEEE 802.11, para la necesaria coordinación entre el uso de un canal de control, Control CHannel (CCH) y un canal de servicio, Service CHannel (SCH), incluyendo una capa MAC para cada uno de ellos (Fig. 2). Las características de esta capa MAC específicamente diseñada para entornos de redes vehiculares se definen en el estándar IEEE 802.11p [9], que está basado en el IEEE 802.11e con ciertas modificaciones. En concreto, modifica ligeramente los valores por defecto de los parámetros EDCA de la subcapa MAC (Tabla II). La capa física (PHY) del IEEE 802.11p es similar a la del estándar IEEE 802.11a, pero soporta como máximo una tasa de transmisión de hasta 27 Mbps, es decir, la mitad que aquel.

WAVE soporta tanto transferencias de datos IP como no-IP. Las transferencias no-IP consisten en el envío de mensajes cortos o WAVE Short Messages (WSMs), definidos en el estándar IEEE 1609.3 o WAVE Short Message Protocol (WSMP). Otros estándares incluidos en la arquitectura WAVE son el

IEEE 1609.2, que especifica los servicios de seguridad, y el IEEE 1609.0, que describe la arquitectura y operaciones de WAVE (Fig. 1).

III. TRABAJOS RELACIONADOS

Existen distintas propuestas existentes en la literatura que asignan distinta prioridad en función del tipo de frame, bien de forma estática [10] [11] [12], o bien, de forma dinámica como el Dynamic Frame Assignment Algorithm (DFAA) [13] y otros [14] [15].

Por ejemplo, en [10] se realiza una partición de datos a nivel aplicación (*slices*), a los cuales se les asigna una prioridad (AC) a nivel MAC IEEE 802.11e en función del tipo de partición, de forma estática y utilizando todas las AC.

En [16] se propone un mecanismo adaptativo denominado Adaptive Mapping Mechanism (AMM) para mejorar la calidad de vídeo H.264 transmitido sobre redes inalámbricas (WLAN) basadas en IEEE 802.11e, mediante la asignación de distinta prioridad en función de la estructura del vídeo codificado (tipo de frame), importancia del frame, y la carga de cada Access Category (AC). El mecanismo AMM se compara con el mecanismo EDCA del IEEE 802.11e, así como con otras propuestas estáticas y dinámicas.

Sin embargo, los trabajos anteriores no están orientados a redes vehiculares (IEEE 802.11p), y transmiten secuencias de vídeo codificadas con H.264. Además, la transmisión se realiza en modo unidifusión (*unicast*), no en multidifusión (*multicast*), más propio de redes vehiculares. De ahí que sea de interés analizar el comportamiento de la transmisión de vídeo en escenarios concretos de redes vehiculares utilizando flujos de vídeo codificados con la última generación de compresores de vídeo.

IV. ENTORNO DE PRUEBAS

Con objeto de evaluar la transmisión de vídeo se ha diseñado un entorno vehicular y una serie de secuencias de simulaciones. Esta sección describe el escenario urbano utilizado, la secuencia de vídeo transmitida por un servidor de vídeo o Road-Side Unit (RSU), y la configuración de los distintos experimentos. Todo ello se ha realizado con la ayuda de un entorno de trabajo denominado Video Delivery Simulation Framework over Vehicular Networks (VDSF-VN) [17], que facilita la creación de los archivos necesarios para realizar simulaciones utilizando el simulador OMNeT++ v4.6 [18], junto con el framework Veins (Vehicles In Network Simulation) v4.4 [19], y el simulador de tráfico SUMO (Simulation of Urban Mobility) v0.25.0 [20]. Por otro lado, también permite realizar las tareas previas de codificación de secuencias de vídeo y generación de trazas de vídeo que se utilizarán durante la simulación, así como la posterior evaluación de las métricas de la calidad percibida por un receptor tras las simulaciones.

A. Escenario

El escenario consiste en un área rectangular de la ciudad de Kiev (Ucrania), de tamaño $2000 \times 2000 m^2$.

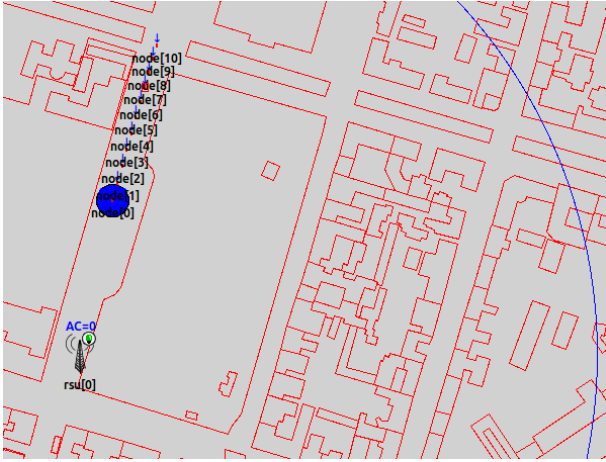


Fig. 3: Ciudad de Kiev - Zona de interés

Se ha colocado una única antena fija o RSU ($rsu[0]$) en medio de una avenida, que transmite una secuencia de vídeo de forma cíclica. El rango de comunicación se ha establecido aproximadamente en 500 m, que es el valor por defecto en Veins, mostrado parcialmente como círculo de color azul en la Figura 3. Los principales parámetros de la simulación se resumen en la Tabla III, y los parámetros de las tarjetas de red se muestran en la Tabla IV.

Durante la simulación, un vehículo cliente de vídeo ($node[0]$) viaja a lo largo de la citada avenida, recibiendo la secuencia de vídeo. Siguiendo a éste, otros 10 vehículos ($node[1..10]$) transmiten datos que se consideran tráfico de fondo. Cada uno de ellos transmite de forma constante una cantidad de paquetes por segundo (pps) de 512 bytes, que se incrementa de un experimento a otro para comprobar el efecto de distintos niveles de tráfico: $\{0, 12, 25, 50, 75\}$ pps, sumando un total de $\{0, 120, 250, 500, 750\}$ pps entre todos los vehículos, equivalente a $\{0, 491, 1024, 2048, 3072\}$ Kbps.

La simulación dura 340s, tiempo suficiente para que los vehículos, que se mueven a una velocidad máxima de 14 m/s (50 km/h), recorran toda la avenida. Sin embargo, nuestra zona de interés está en las proximidades de la antena, en concreto consideramos el intervalo comprendido entre $t=[180..190]$ s, tiempo necesario para la recepción de una secuencia de vídeo completa (10s), que se describe a continuación.

TABLA III: Parámetros de simulación.

Parámetro	Valor
Area de simulación	$2000 \times 2000 \text{ m}^2$
Tiempo de simulación	340s
N. RSUs	1
N. vehículos cliente	1
N. vehículos tráf. fondo	10
Tasa tráf. fondo	$\{0, 12, 25, 50, 75\}$ pps
Velocidad máxima	14 m/s (50 km/h)

B. Secuencia de vídeo

La secuencia de vídeo transmitida por el RSU es 'BasketballDrill', que pertenece al HEVC Common Test Conditions Set [21]. Como se muestra en la Tabla V, tiene una resolución de 832×480 píxeles, una longitud de 250 frames, y una tasa de 25 frames por segundo (fps), representando 10 segundos de vídeo. La secuencia original tiene 500 frames de longitud a una tasa de 50 fps, pero ha sido submuestreada a 25 fps con objeto de reducir el ancho de banda requerido. La secuencia fue codificada con el estándar HEVC, utilizando el software de referencia HEVC Test Model (HM) [21].

Se han utilizado tres modos de codificación para los experimentos (Tabla VI), variando la estructura de cada grupo de imágenes, o Group of Pictures (GoP). En el modo All Intra (AI), todos los frames de la secuencia de vídeo se codifican como frames de tipo I, esto es, ningún frame referencia a otro. Como todos los frames son codificados de forma independiente, el modo AI es el modo de codificación de vídeo más robusto. En el otro extremo está el modo Low-delay P (LP), en el que el primer frame se codifica como un frame de tipo I, y el resto como frames de tipo P, utilizando cuatro frames previamente codificados como referencia. Este modo es muy eficiente en cuanto a compresión debido al uso de estimación de movimiento y compensación, pero es muy sensible a pérdida de paquetes debido a la interdependencia entre frames. Por ello, se utilizan otros modos en los que se inserta periódicamente frames de tipo I (*intra-refresh*) con objeto de evitar la propagación de error temporal. Además del modo AI, en este trabajo se han utilizado los modos de codificación IP7 y LPI4. El modo IP7 es similar al modo LP pero cada frame de tipo I va seguido de siete frames de tipo P. Por último, el modo LPI4 también es una variación del modo LP en el que se inserta un frame de tipo I cada cuatro frames.

Mediante el valor Quantization Parameter (QP) se puede controlar el nivel de calidad o el ancho de banda (*bitrate*) necesario para la transmisión del vídeo codificado. Cuanto mayor valor de QP, mayor compresión se conseguirá, requiriendo un menor espacio de almacenamiento y menor *bitrate* para su transmisión, a costa de una calidad menor. Para cada uno de los modos se estableció el valor de QP para el que se consiguiera, de forma aproximada, la misma calidad de la secuencia de vídeo codificada ($PSNR \approx 36$

TABLA IV: Parámetros PHY/MAC.

Parámetro	Valor
Frecuencia portadora	5.890 GHz
Modelo de propagación	SimpleObstacleShadowing
Bitrate	18 Mbps
Potencia TX	20 mW
Sensibilidad RX	-89 dBm
Rango comunicación	510.87 m
Tamaño colas MAC	0 (infinito)

TABLA V: Secuencia de vídeo.

Parámetro	Valor
Nombre	BasketballDrill
Resolución	832 × 480 pixels
Duración	10 s
Longitud	250 frames
Tasa	25 fps

TABLA VI: Modos de codificación.

Modo	Estructura	Descripción
AI	IIIIIIII ...	Cada imagen es un I-frame (All Intra)
I7P	IPPPPPPP ...	I-frame seguido de siete P-frames
LPI4	IPPP IPPP ...	Similar a LP pero insertando un I-frame cada cuatro frames

TABLA VII: Valores QP para obtener PSNR≈36 dB.

Modo	QP	Tiles	Bitrate (Kbps)	PSNR (dB)
AI	31	1	3437	35.863
AI	31	16	3656	35.862
I7P	29	1	1467	36.071
I7P	29	16	1604	36.064
LPI4	29	1	1626	36.045
LPI4	29	16	1786	36.034

dB). La Tabla VII muestra el *bitrate* requerido para los casos extremos de cada modo (1 y 16 tiles por frame). Como se puede observar, para una misma calidad percibida, el modo AI es el que requiere un mayor ancho de banda para su transmisión, seguido del LPI4, y, por último, el modo I7P. Esto es debido a que los frames de tipo I son los que requieren almacenar más información al no depender de otros frames, por lo que cuanto mayor proporción de frames de tipo I, mayor ancho de banda será necesario.

Puesto que un frame puede ser mayor que el tamaño máximo de paquete de datos que la red puede transmitir o Maximum Transmission Unit (MTU), para su transmisión a través de la red es necesario encapsular cada frame en una serie de paquetes del protocolo Real-time Transport Protocol (RTP) [22] [23]; a este proceso se le denomina *paquetización* (Fig. 4). En los experimentos realizados, todos los paquetes de un mismo frame tienen la misma prioridad; por ejemplo, si un frame es elegido para tener una prioridad AC(2), todos los paquetes en los que se dividirá para su transmisión tendrán asignada esa prioridad.

En un trabajo previo [24] se utilizaron 9 modos de codificación distintos, estudiando el efecto de dividir los frames de vídeo en fragmentos (*tiles*). En dichos experimentos no se utilizó calidad de servicio (QoS), es decir, en todos los casos se asignó la prio-

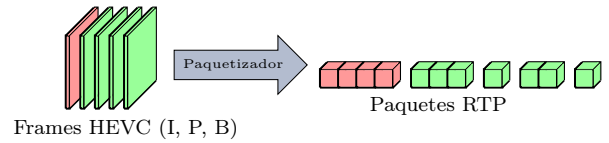


Fig. 4: Paquetización de frames

ridad AC(0) a todos los paquetes, ya sea de vídeo o de tráfico de fondo. En este trabajo se han seleccionado los tres modos mencionados (AI, I7P, LPI4), con objeto de comprobar el resultado de aplicar calidad de servicio (QoS), asignando la prioridad AC(2) a ciertos paquetes de vídeo, y estudiar cómo afecta todo ello al resto de tráfico no prioritario (tráfico de fondo).

En los experimentos se han considerado una gran cantidad de combinaciones de parámetros, dependiendo de los modos de codificación y valores QP seleccionados, el número de *tiles* por frame, y el nivel de tráfico de fondo. Todo ello se resume en los siguientes puntos:

- Modos de codificación (×3): {AI, I7P, LPI4} (ver Tabla VI).
- Valores QP (×1): para cada modo se utiliza el valor de QP fijo que permite obtener la calidad de vídeo deseada (PSNR ≈ 36 dB) cuando se usa 1 tile por frame (ver Tabla VII).
- Número de tiles por frame (×7): siete valores se han considerado para codificar cada *bitstream*: {1, 2, 4, 6, 8, 10, 16}. El tamaño de estas particiones se puede especificar con el número de filas y columnas (uniforme), o con el número de Coding Tree Units (CTUs) por cada fila y columna (no-uniforme). En este trabajo se han considerado los siguientes patrones uniformes de tiles: {1 × 1, 1 × 2, 2 × 2, 2 × 3, 2 × 4, 2 × 5, 4 × 4}, respectivamente.
- Tasa de tráfico de fondo (×5): {0, 12, 25, 50, 75} pps de 512 bytes.

Por tanto, todas estas combinaciones hacen un total de 21 *bitstreams* diferentes (3×1×7), con los diferentes niveles de tráfico de fondo (×5), haciendo un total de 105 simulaciones por cada experimento. Se han realizado un total de 10 experimentos, que pueden clasificarse en dos grupos:

1. Grupo 1: asignando una mayor prioridad AC(2) a un porcentaje de los frames de tipo I únicamente (×5): P={0, 25, 50, 75, 100} %; al resto (frames P y B) se le asigna la misma prioridad que el tráfico de fondo, AC(0).
2. Grupo 2: asignando una mayor prioridad AC(2) a un porcentaje de todos los paquetes de vídeo, independientemente del tipo de frame al que corresponda (×5): P={0, 25, 50, 75, 100} %.

El primer grupo de experimentos está motivado en el hecho de que la pérdida de un frame de tipo I es más grave que la pérdida de otros tipos de frame, debido a las interdependencias existentes entre frames. En este caso, puesto que hay paquetes de diferentes

prioridades, tendremos paquetes que al llegar a la capa MAC se clasificarán en distintas colas, y los de menor prioridad tendrán que esperar más que otros paquetes de frames posteriores en el tiempo de reproducción, pero que tengan mayor prioridad. Por ello, es posible que el receptor reciba los paquetes desordenados, por lo que se almacenarán en una memoria intermedia (*buffer*) antes de su reproducción. Sin embargo, también es posible que los paquetes de los frames de baja prioridad puedan sufrir de latencias importantes, por lo que en aplicaciones en tiempo real o sensibles al retardo se establece un retardo máximo para descartar aquellos paquetes que hayan sido recibidos con un retardo mayor.

En la Sección V se muestran exclusivamente los resultados de los casos $P=\{0, 100\}$ % por cuestiones de espacio, y a que las otras probabilidades son casos intermedios. Por tanto, y con objeto de facilitar la explicación de resultados, destacamos los resultados de 3 experimentos:

- Experimento 1: asignando la prioridad AC(0), tanto a los paquetes de vídeo como a los de tráfico de fondo (sin QoS).
- Experimento 2: asignando la prioridad AC(2) a todos los frames de tipo I únicamente ($P=100$ %), y asignando al resto (frames P y B, y tráfico de fondo) la prioridad AC(0).
- Experimento 3: asignando la prioridad AC(2) a todos los paquetes de vídeo ($P=100$ %), independientemente del tipo de frame al que correspondan. Este caso sería equivalente al mecanismo EDCA del IEEE 802.11e, que especifica que a todos los paquetes de vídeo se les debe asignar la prioridad AC(2).

V. RESULTADOS

Para comparar el rendimiento de los tres experimentos definidos se han ejecutado las correspondientes simulaciones, en las que se ha recogido, entre otras, estadísticas a nivel aplicación (APP), como el número de paquetes transmitidos (Load), los paquetes recibidos (Throughput o Goodput), los paquetes perdidos, el ratio de paquetes enviados respecto a los recibidos o Packet Delivery Ratio (PDR), el retardo o End-to-End Delay (EED), la variación del retardo o *jitter*; también a nivel MAC y físico (PHY), como la ocupación de las colas de cada Access Category (AC), el ratio de ocupación del canal, o el número de colisiones en el medio. Por último, también se calculan métricas objetivas para evaluar la calidad de las secuencias de vídeo recibidas, como el Peak Signal-to-Noise Ratio (PSNR).

Comenzamos mostrando el Throughput a nivel aplicación (Goodput) alcanzado con los tres modos para los tres experimentos. En general, cuando no se utiliza QoS (Exp. 1), el tráfico de vídeo es afectado en gran medida al aumentar el tráfico de fondo, siendo especialmente significativo para el modo AI (Fig. 5a), que, para el mayor nivel de tráfico de fondo, baja un 44.28 % (unos 1.5 Mbps de pérdida). En cuanto al tráfico de fondo, también baja un porcentaje similar

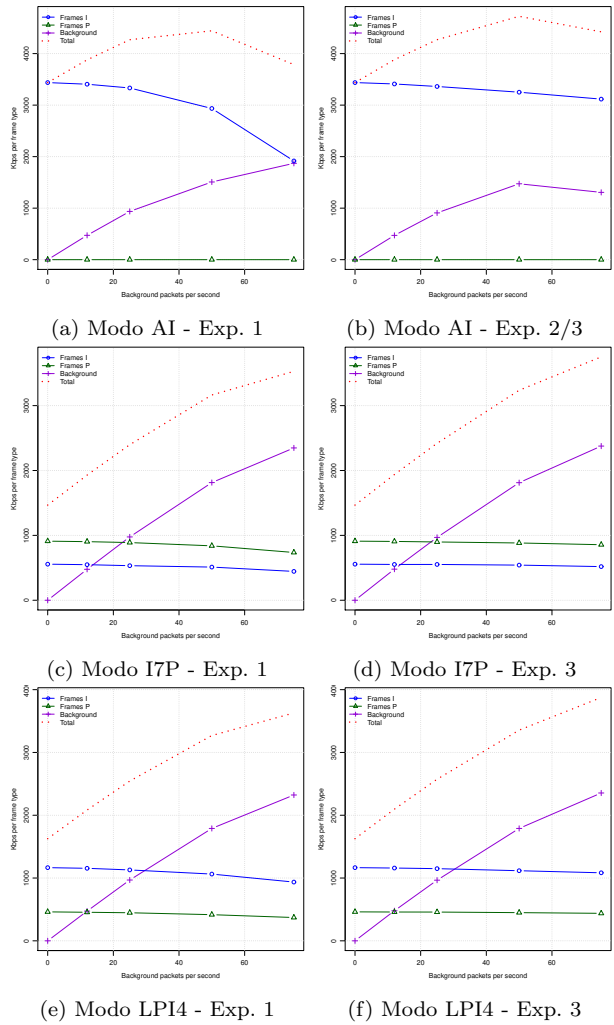


Fig. 5: Throughput para 1 tile y tráfico de fondo

(39.13 %), ya que ambos tipos de tráfico, al tener la misma prioridad, compiten en el acceso al medio en igualdad de condiciones. En los Exp. 2 y 3, que para el modo AI son idénticos puesto que sólo hay frames de tipo I (Fig. 5b), puede apreciarse que el tráfico de vídeo se ve mucho menos afectado por tener una mayor facilidad a la hora de acceder al canal, teniendo sólo un 9.34 % de pérdida (casi 35 puntos menos), en detrimento del tráfico de fondo, que en el peor caso baja un 57.49 % (13 puntos más que cuando no se utiliza QoS). Vemos también que según se aumenta el tráfico de fondo se va haciendo un mayor uso del canal (sumando el Throughput obtenido tanto por

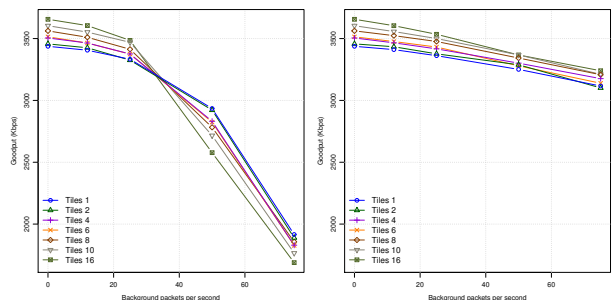


Fig. 6: Throughput (todos los tiles) - Modo AI

el flujo de vídeo como por el tráfico de fondo de todos los vehículos), llegando al punto máximo para 50 pps; a partir de ahí el canal se satura, y el intentar transmitir un mayor número de paquetes consigue el efecto contrario por aumentar el número de colisiones y periodos de espera para acceso al canal (*backoff*). También se aprecia que al utilizar QoS aumenta la ocupación máxima del canal en ese punto, pasando de los 4.44 Mbps sin QoS a un máximo de 4.72 Mbps (6% de incremento). Este incremento es aún mayor para 75 pps, pues se alcanza un total de 3.78 Mbps sin QoS, frente a 4.42 Mbps con QoS (16.83% de incremento).

En cuanto a los modos I7P y LPI4, las pérdidas son aproximadamente la mitad que con el modo AI debido a la menor tasa necesaria para la transmisión de la secuencia de vídeo codificada en este modo, estando el canal más libre. En la Figura 5, el tráfico de vídeo se muestra desglosado en los dos tipos de frames que contienen, I y P. Para el caso de I7P y de mayor nivel de tráfico de fondo (75 pps), en el Exp. 1 (Fig. 5c) el Throughput correspondiente a ambos tipos de frames cae aproximadamente lo mismo (~19%), valor parecido al del tráfico de fondo (23.6%). En el Exp. 2 (no mostrado) los frames de tipo I, al tener más prioridad, apenas sufren una bajada de Throughput (4.68%), mientras que los frames de tipo P siguen teniendo pérdidas similares al caso anterior (18%). En el Exp. 3 (Fig. 5d) ambos tipos de frames se mantienen bastante estables, bajando un 6.83% y 5.93%, respectivamente, para el

mayor nivel de tráfico de fondo. Los resultados del modo LPI4 son muy similares al I7P en todos los experimentos y valores que se acaban de detallar (Fig. 5e y 5f). En cuanto al tráfico de fondo, las pérdidas para los dos modos I7P y LPI4 en ese punto también se mantienen en valores muy parecidos en los 3 experimentos, en torno al 22%..24%. Por tanto, vemos que la asignación de prioridad a los distintos frames de vídeo tiene efectos muy positivos a nivel de estadísticas de red, sin perjudicar demasiado al resto de tráfico de menor prioridad existente en la red.

A continuación analizamos el efecto de fragmentar los frames (*tiles*); cuanto mayor número de tiles por frame se utilicen a la hora de codificar la secuencia de vídeo, mayor *bitrate* será necesario (Tabla VII), debido a que se pierde algo de eficiencia en la codificación. Por ejemplo, para el modo AI, el *bitrate* necesario para 1 tile es 3437 Kbps, mientras que para 16 tiles es de 3656 Kbps. Esto provoca que el canal se sature antes según se aumenta el tráfico de fondo, y en caso de no utilizar prioridad, el tráfico de vídeo se ve bastante afectado debido a las colisiones que se producen. En la Figura 6a se puede observar este efecto para el modo AI, ya que, a partir de un determinado nivel de tráfico de fondo, se invierte totalmente el orden de las curvas correspondientes al Throughput de los distintos casos, y las pérdidas llegan hasta un 44.25% para 1 tile, y 53.84% para 16 tiles. Sin embargo, al utilizar QoS las curvas se mantienen ordenadas al reducirse los paquetes perdidos hasta una media de 10.36% para cualquier número

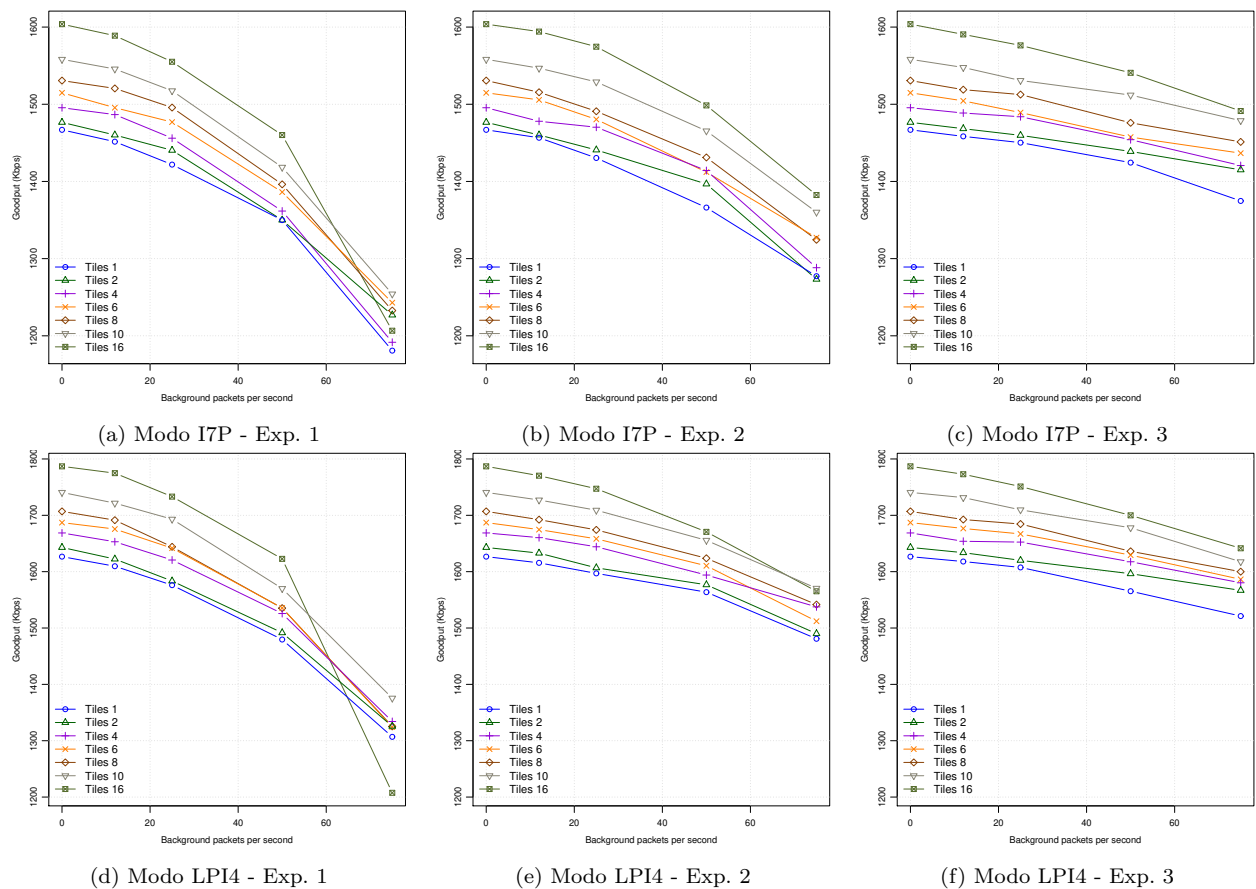


Fig. 7: Throughput (todos los tiles) - Modos I7P y LPI4

de tiles (Fig. 6b).

En cuanto a los modos I7P y LPI4 se observa algo parecido, pero debido a que estos modos requieren un menor *bitrate* que el modo AI, la red sólo se satura para el caso de utilizar 16 tiles por frame. En el modo I7P las pérdidas para el caso de mayor tráfico de fondo se quedan entre 19.56 % para 1 tile y 24.75 % para 16 tiles (Fig. 7a). Igual que para el caso AI, al utilizar QoS (Exp. 2 y 3) las curvas se mantienen ordenadas y tienen similar tendencia, es decir, que el aumento del tráfico de fondo les afecta por igual. Para el modo I7P, en el Exp. 2 las pérdidas están en torno al 13.4 % para cualquier número de tiles por frame utilizado (Fig. 7b), mientras que en el Exp. 3, se quedan en torno al 6.5 % (Fig. 7c). Por último, en el modo LPI4 las pérdidas sin QoS para 1 tile son muy similares al modo I7P (19.67 %), pero para 16 tiles sube hasta 32.46 % debido a la mayor proporción de frames de tipo I y, por tanto, su mayor *bitrate* (Fig. 7d). Al utilizar las dos modalidades de QoS, nuevamente las curvas se mantienen ordenadas, y las pérdidas bajan bastante. En este caso, las pérdidas se quedan en torno al 10.7 % para el Exp. 2 (Fig. 7e) y entorno al 7.3 % para el Exp. 3 (Fig. 7f), siempre algo menores para 1 tile, y algo mayores para 16 tiles.

Para analizar el porcentaje de paquetes entregados correctamente se utiliza el valor del Packet Delivery Ratio (PDR). Las Figuras 8a y 8b muestran el PDR para el modo AI; a partir de ellas se observa cómo baja el número de paquetes entregados, tanto al aumentar el tráfico de fondo, como al aumentar el número de tiles por frame, debido al incremento en el ancho de banda necesario para su transmisión. Sin embargo, al incrementar el número de tiles por frame, a pesar de tener un mayor número de pérdidas, el porcentaje de tiles perdidos es menor, lo que implica que la calidad del vídeo reconstruido sea mayor. Por ejemplo, en el modo AI, las pérdidas de paquetes son especialmente notables en el Exp. 1 (sin QoS), donde las curvas del PDR muestran una fuerte bajada cuando aumenta el tráfico de fondo; para el caso de 1 tile por frame se produce una pérdida de paquetes de un 44 %, y para el caso de 16 tiles, un 53 % (Fig. 8a). Estas pérdidas tienen como consecuencia que no sea posible decodificar un 95.2 % de los tiles para 1 tile por frame, mientras que en el caso de utilizar 16 tiles por frame, tan sólo un 13.5 % (Fig. 8c). Esto es debido a que la pérdida de un único paquete en un tile hace que el tile completo no pueda decodificarse, que en el caso de 1 tile por frame significa el frame completo. El resultado en ese caso es un vídeo recibido de una calidad muy pobre, con un valor PSNR de menos de 20 dB, mientras que si se utilizan 10 ó 16 tiles, la calidad supera el umbral considerado como aceptable (28 dB) en todos los casos; en concreto, para 16 tiles supera los 31 dB (Fig. 8e). Siguiendo con el modo AI, en el Exp. 2 (o 3) apenas hay diferencia en los valores del PDR para cualquier número de tiles por frame. Sin embargo, el número de tiles perdidos y la calidad final del vídeo reconstruido sí que muestran una gran diferencia en ambos casos. Con 1 tile se pierde

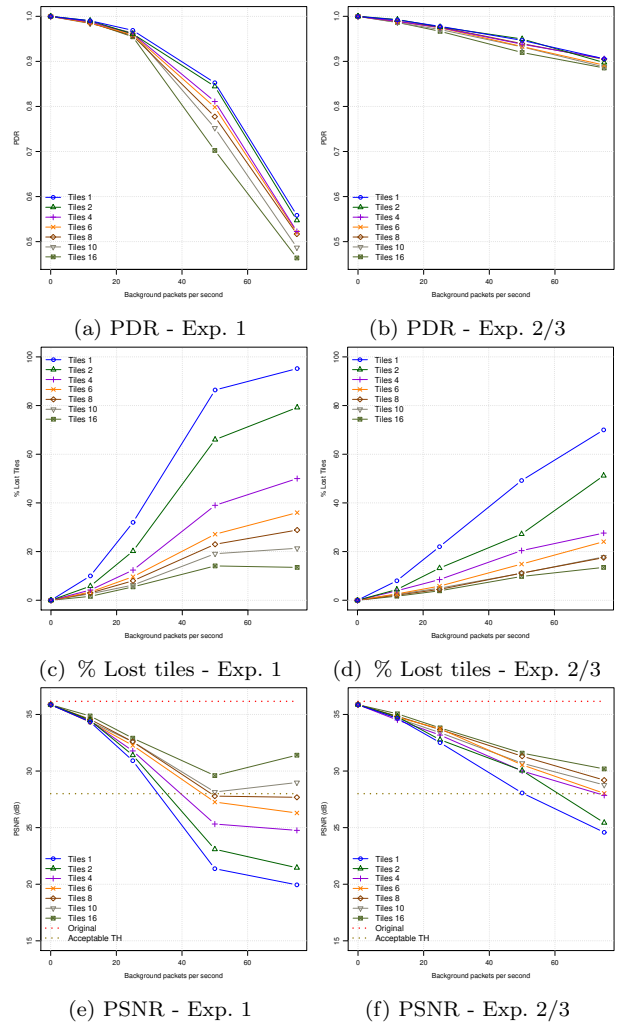


Fig. 8: Modo AI - PDR (arriba), % de paquetes perdidos (centro), y PSNR (abajo)

el 9.44 % de los paquetes (Fig. 8b), que se traduce en una pérdida del 70 % de los tiles (Fig. 8d), y se obtiene un PSNR de 24.58 dB (Fig. 8f). Sin embargo, con 16 tiles por frame, a pesar de que se pierde un porcentaje ligeramente mayor de paquetes (11.46 %), eso conlleva la pérdida de sólo el 13.5 % de los tiles, obteniendo una calidad final de 30.19 dB.

La Tabla VIII muestra el valor PSNR obtenido con el modo AI para todas las combinaciones de tráfico de fondo (BGT pps) y número de tiles por frame (de 1 a 16), para el Exp. 1 (arriba) y 2/3 (abajo), corres-

TABLA VIII: Modo AI - PSNR Exp. 1 y 2/3 (dB).

BGT	1	2	4	6	8	10	16
0	35,86	35,87	35,86	35,87	35,86	35,86	35,86
12	34,34	34,49	34,41	34,33	34,57	34,36	34,87
25	30,91	31,40	31,78	32,26	32,59	32,59	32,91
50	21,37	23,09	25,32	27,26	27,78	28,15	29,60
75	19,95	21,46	24,76	26,29	27,68	28,99	31,40

BGT	1	2	4	6	8	10	16
0	35,86	35,87	35,86	35,87	35,86	35,86	35,86
12	34,65	34,80	34,51	34,75	34,84	34,66	35,05
25	32,51	32,76	33,18	33,73	33,66	33,39	33,81
50	28,06	30,04	30,00	30,54	31,31	30,69	31,58
75	24,58	25,43	27,85	28,03	29,20	28,77	30,19

TABLA IX: Modo I7P - PSNR Exp. 1, 2 y 3 (dB).

BGT	1	2	4	6	8	10	16
0	36,07	36,08	36,08	36,08	36,07	36,07	36,06
12	30,53	32,57	33,76	33,39	34,31	33,63	33,71
25	25,62	28,10	29,69	31,81	31,55	30,54	30,42
50	20,77	21,01	23,44	24,22	24,28	25,25	25,19
75	16,77	17,61	19,23	20,33	21,38	20,26	21,63

BGT	1	2	4	6	8	10	16
0	36,07	36,08	36,08	36,08	36,07	36,07	36,06
12	33,41	32,21	33,39	34,50	34,01	34,12	34,29
25	29,46	29,09	30,49	30,96	31,11	31,98	31,35
50	23,90	25,11	25,96	26,51	27,48	26,72	26,15
75	19,34	20,92	22,38	22,85	24,20	23,45	24,04

BGT	1	2	4	6	8	10	16
0	36,07	36,08	36,08	36,08	36,07	36,07	36,06
12	33,85	34,20	35,00	34,22	34,38	34,52	34,60
25	32,41	34,02	33,56	31,43	32,57	32,68	32,35
50	26,48	29,34	30,40	28,82	29,60	30,53	29,14
75	22,83	26,92	26,91	26,43	26,86	28,04	26,28

pondientes a las Figuras 8e y 8f, respectivamente. Si el valor de PSNR es mayor de 32 dB (valor muy bueno) se marca en color verde, si es mayor de 28 dB (valor aceptable) se marca en amarillo, si está por debajo de ese valor y hasta 24 dB (valor deficiente) en naranja, y para valores inferiores (muy deficientes) en rojo.

En cuanto al modo IP7, al ser el de menor refresco Intra, no es muy robusto. A partir de un nivel de tráfico de fondo moderado los resultados ya no son aceptables (Tabla IX). Por último, el modo LPI4 es más robusto que el I7P, obteniendo unos resultados bastante aceptables (Tabla X), muy parecidos a los del modo AI (Tabla VIII).

Para terminar, analizamos la calidad del vídeo de los tres modos en cada uno de los tres experimentos para el caso de mayor tráfico de fondo (75 pps) variando el número de tiles por frame (Fig. 9). Se puede deducir que al incrementar el número de tiles hasta un número de 6 u 8 la calidad experimenta una mejora significativa en todos los modos, compensando el exceso de ancho de banda requerido para codificar la secuencia de vídeo. Sin embargo, para valores mayores las mejoras son mínimas. Los mejores resultados se obtienen en el experimento 3, en el que los modos AI y LPI4, y para un número de tiles igual o superior a 6 tiles alcanzan una calidad aceptable; en cuanto al modo I7P, aunque el uso de tiles también proporciona una mejora, no es suficiente cuando hay niveles moderados o altos de tráfico de fondo.

VI. CONCLUSIÓN Y TRABAJOS FUTUROS

Se han realizado varios experimentos combinando diversas técnicas con objeto de mejorar la calidad del vídeo transmitido en redes vehiculares, como son el refresco de frames de tipo I (*intra-refresh*), el número de tiles por frame, y la calidad de servicio. Se han utilizado varios modos de codificación variando la frecuencia de refresco de frames de tipo I, llegando a la conclusión de que es fundamental el uso de frames de tipo I frecuentemente, como en los modos

TABLA X: Modo LPI4 - PSNR Exp. 1, 2 y 3 (dB).

BGT	1	2	4	6	8	10	16
0	36,05	36,05	36,05	36,05	36,05	36,05	36,03
12	32,90	33,22	34,10	34,52	34,51	34,09	34,77
25	27,32	28,65	31,06	31,71	30,78	32,99	32,58
50	20,89	23,41	25,89	25,66	26,56	26,45	27,29
75	17,14	19,65	20,57	21,69	21,89	22,61	23,90

BGT	1	2	4	6	8	10	16
0	36,05	36,05	36,05	36,05	36,05	36,05	36,03
12	34,12	34,66	34,94	35,08	34,93	34,71	34,58
25	31,34	31,12	33,15	32,52	32,84	33,22	32,35
50	28,59	28,48	28,94	30,05	29,05	29,34	28,81
75	21,36	24,79	26,46	25,35	26,08	26,43	25,55

BGT	1	2	4	6	8	10	16
0	36,05	36,05	36,05	36,05	36,05	36,05	36,03
12	34,06	35,24	33,50	34,67	34,57	35,11	34,45
25	32,03	32,87	33,84	33,92	33,50	33,56	32,96
50	27,40	29,82	30,96	30,54	30,56	30,95	29,58
75	22,28	27,50	28,43	28,38	28,63	27,62	28,17

AI y LPI4, descartando el uso del modo I7P. Usar un mayor número de tiles por frame incrementa la calidad del vídeo reconstruido, siendo 6 u 8 un número que mejora notablemente la calidad, sin incrementar demasiado el ancho de banda necesario para su transmisión. Además, todo ello se ha combinado con el uso de la calidad de servicio (QoS) mediante la asignación de más prioridad a ciertos frames de la secuencia de vídeo. Sin QoS, solamente el modo AI supera el umbral de calidad aceptable a partir de 10 tiles por frame; los modos I7P y LPI4 no lo alcanzan en ningún caso. Mediante el uso de QoS, asignando la prioridad AC(2) a los frames de tipo I, los resultados indican que aunque la calidad mejora, no es suficiente para llegar al umbral aceptable en esos modos. En cambio, asignando la prioridad AC(2) a todos los frames de vídeo, el modo LPI4 también llega a dicho umbral, no siendo así para el modo I7P, que, aunque mejora su calidad, sigue sin ser suficiente. Como conclusión final, el uso de los modos AI o LPI4, con 6 u 8 tiles por frame, y asignando una prioridad AC(2) a todos los frames de vídeo es la alternativa que proporciona los mejores resultados.

Como continuación de este trabajo se propone el uso técnicas adaptativas, que tengan en cuenta el nivel de saturación de la red, y el tamaño de las distintas colas del MAC, así como el uso de otras técnicas de protección de errores.

AGRADECIMIENTOS

Este trabajo ha sido financiado por el Ministerio de Ciencia, Innovación y Universidades (Ref. RTI2018-098156-B-C54) cofinanciado con fondos FEDER (MINECO/FEDER/UE), así como por la Convocatoria de Ayudas y Bolsas de viaje para la difusión de resultados de investigación en el marco del programa de doctorado TECNIT (RR 2426/18 UMH).

REFERENCIAS

- [1] Ali Tufail, Syed Ali Khayam, Muhammad Taqi Raza, Amna Ali, and Ki-Hyung Kim, "An Enhanced Backbone-Assisted Reliable Framework for Wireless Sensor Net-

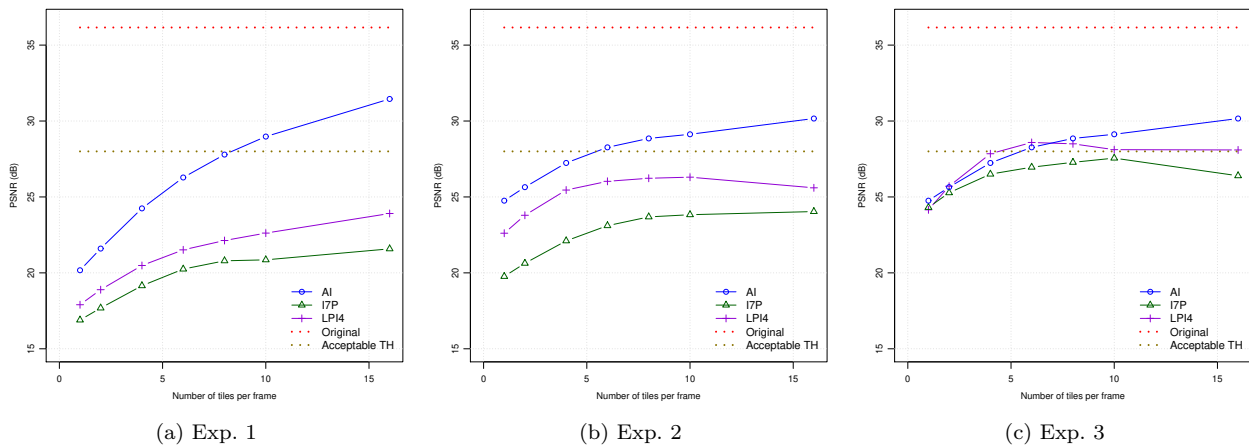


Fig. 9: PSNR Todos los modos - Variando número de tiles por frame

- works,” *Sensors (Basel, Switzerland)*, vol. 10, pp. 1619–51, 03 2010.
- [2] Joint Collaborative Team on Video Coding (JCT-VC), “High Efficiency Video Coding (HEVC),” ITU-T Recommendation H.265, 2013.
- [3] “Advanced Video Coding (AVC) for generic audiovisual services,” ITU-T Recommendation H.264, 2003.
- [4] D. B. Rawat and D. C. Popescu and, “Performance enhancement of EDCA access mechanism of IEEE 802.11e wireless LAN,” in *2008 IEEE Radio and Wireless Symposium*, Jan 2008, pp. 507–510.
- [5] Hua Zhu, Ming Li, , Imrich Chlamtac, and B. Prabhakaran, “A survey of quality of service in IEEE 802.11 networks,” *IEEE Wireless Communications*, vol. 11, no. 4, pp. 6–14, Aug 2004.
- [6] “IEEE Standard for Information technology–Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,” *IEEE Std 802.11-2016*, pp. 1–3534, Dec 2016.
- [7] “IEEE Standards for Information Technology – IEEE Std 802.11e – Draft Supplement to Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Quality of Service Enhancements,” online, 2005.
- [8] “IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Multi-Channel Operation,” *IEEE Std 1609.4-2016 (Revision of IEEE Std 1609.4-2010)*, pp. 1–94, March 2016.
- [9] “IEEE Standard for Information technology – Local and metropolitan area networks – Specific requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments,” *IEEE Std 802.11p-2010*, pp. 1–51, July 2010.
- [10] A. Ksentini, M. Naimi, and A. Gueroui, “Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture,” *IEEE Communications Magazine*, vol. 44, no. 1, pp. 107–114, Jan 2006.
- [11] R. MacKenzie, D. Hands, and T. O’Farrell, “QoS of Video Delivered over 802.11e WLANs,” in *2009 IEEE International Conference on Communications*, June 2009, pp. 1–5.
- [12] I. Ali, M. Fleury, S. Moiron, and M. Ghanbari, “Enhanced prioritization for video streaming over wireless home networks with IEEE 802.11e,” in *2011 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, June 2011, pp. 1–6.
- [13] Zheng Wan, N. Xiong, N. Ghani, Min Peng, A. V. Vasilakos, and Liang Zhou, “Adaptive scheduling for wireless video transmission in high-speed networks,” in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2011, pp. 180–185.
- [14] Naveen Chilamkurti, Serali Zeadally, Robin Soni, and Giovanni Giambene, “Wireless multimedia delivery over 802.11e with cross-layer optimization techniques,” *Multimedia Tools and Applications*, vol. 47, no. 1, pp. 189–205, Mar 2010.
- [15] C.-H. Lin, C.-K. Shieh, C.-H. Ke, N. K. Chilamkurti, and S. Zeadally, “An adaptive cross-layer mapping algorithm for MPEG-4 video transmission over IEEE 802.11e WLAN,” *Telecommunication Systems*, vol. 42, no. 3, pp. 223, Jul 2009.
- [16] Xin-Wei Yao, Wan-Liang Wang, Shuang-Hua Yang, Yue-Feng Cen, Xiao-Min Yao, and Tie-Qiang Pan, “IPB-frame Adaptive Mapping Mechanism for Video Transmission over IEEE 802.11e WLANs,” *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 2, pp. 5–12, Apr. 2014.
- [17] P. Pablo Garrido Abenza, Pablo Piñol Peral, Manuel P. Malumbres, and O. López-Granado, “Simulation Framework for Evaluating Video Delivery Services over Vehicular Networks,” in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, August 2018, pp. 1–5.
- [18] Andrés Varga and Rudolf Hornig, “An Overview of the OMNeT++ Simulation Environment,” in *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*, ICST, Brussels, Belgium, 2008, Simutools ’08, pp. 60:1–60:10, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [19] Christoph Sommer, Reinhard German, and Falko Dressler, “Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis,” *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011.
- [20] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker, “Recent Development and Applications of SUMO - Simulation of Urban MObility,” *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128–138, December 2012.
- [21] Joint Collaborative Team on Video Coding (JCT-VC), “HEVC reference software HM (HEVC Test Model) and Common Test Conditions,” <https://hevc.hhi.fraunhofer.de>, [Accessed May 11, 2018].
- [22] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications,” RFC 3550, 2003.
- [23] Ye Wang, Yago Sanchez, Thomas Schierl, Stephan Wenger, and Miska Hannuksela, “RTP Payload Format for High Efficiency Video Coding,” RFC 7798, 2016.
- [24] Pedro Pablo Garrido Abenza, Manuel P. Malumbres, Pablo Piñol, and Otoniel López-Granado, “Source Coding Options to Improve HEVC Video Streaming in Vehicular Networks,” *Sensors*, vol. 18, no. 9, 2018.