

**ALGORITMO AAASH PARA EL SERVICIO DE VIDEOSTREAMING SOBRE  
REDES MÓVILES****AAASH ALGORITHM FOR VIDEO STREAMING SERVICE OVER MOBILE  
NETWORKS**

 **PhD. Wilmar Yesid Campo-Muñoz\***,  **Ing. Juan Carlos Imbachi Paz\***  
 **Ing. Andrés Felipe Escobar Zapata\***

\* **Universidad del Quindío**, Facultad de Ingeniería, Programa de Ingeniería Electrónica.  
Carrera 15 con calle 12 norte, Armenia, Quindío, Colombia.  
Tel.: +57 (606) 7359300.  
E-mail: wycampo@uniquindio.edu.co  
E-mail: {afescobarz, jcimbachip}@uqvirtual.edu.co

**Cómo citar:** Campo Muñoz, W. Y., Imbachi Paz, J. C., & Escobar Zapata, A. F. (2023). ALGORITMO AAASH PARA EL SERVICIO DE VIDEOSTREAMING SOBRE REDES MÓVILES. REVISTA COLOMBIANA DE TECNOLOGÍAS DE AVANZADA (RCTA), 1(41), 12–19. Recuperado a partir de <https://ojs.unipamplona.edu.co/index.php/rcta/article/view/2413>

Derechos de autor 2023 Revista Colombiana de Tecnologías de Avanzada (RCTA).  
Esta obra está bajo una licencia internacional [Creative Commons Atribución-NoComercial 4.0](https://creativecommons.org/licenses/by-nc/4.0/).



**Resumen:** Los servicios soportados por la tecnología de videostreaming son los que más tráfico generan sobre las redes móviles. Una de las formas de hacer frente a esta situación, es mediante el uso de protocolos adaptativos, que permiten adecuar el tráfico del servicio a las condiciones de la red para mantener las condiciones de calidad. Por lo que surge el protocolo de transmisión adaptable y dinámica sobre HTTP (DASH - Dynamic Adaptive Streaming over HTTP). En este artículo se aplica el algoritmo adaptativo para streaming HTTP AAASH (Adaptation Algorithm for Adaptive Streaming over HTTP) en los clientes de una red móvil celular. Se presentan las condiciones del escenario de experimentación y el algoritmo se ejecuta en diferentes escenarios incrementando la cantidad de usuarios que consumen video mediante DASH de manera simultánea. AAASH muestra ser adecuado, puesto que se obtienen parámetros de calidad de servicio los cuales satisfacen los requerimientos.

**Palabras clave:** AAASH, QoS, red móvil, videostreaming.

**Abstract:** The services supported by video streaming technology are the ones that generate the most traffic on mobile networks. One of the ways to deal with this situation is through the use of adaptive protocols, which allow the service traffic to be adapted to the network conditions in order to maintain quality conditions. So the Dynamic Adaptive Streaming over HTTP (DASH) arises. We in this paper applies Adaptation Algorithm for Adaptive Streaming over HTTP (AAASH) in clients of a cellular mobile network. The conditions of the experimental scenario are presented and the algorithm is executed in different scenarios increasing the number of users who consume video through DASH simultaneously. AAASH shows to be adequate, since quality of service parameters are obtained which satisfy the requirements.

**Keywords:** AAASH, mobile networks, QoS, video streaming.

## 1. INTRODUCCIÓN

Actualmente los teléfonos inteligentes tienen capacidades equiparables con los de un computador, esto permite que los usuarios tengan acceso desde cualquier lugar a los diferentes servicios, las redes sociales, las plataformas de OTT (over the top), el teletrabajo, la educación virtual y toda clase de distribución de contenidos que ofrece actualmente Internet. Dentro de estos contenidos destacan aquellos que son soportados por videostreaming, por su alta aceptación por parte de los usuarios finales y por el impacto sobre las redes de datos y entre ellas las redes móviles (Gadaleta, Chiariotti, Rossi, & Zanella, 2017), (Solano-Hurtado & Soto-Cordova, 2021). En este orden de ideas, el 3GPP (3rd Generation Partnership Project) en sus releases 12 considera el protocolo estándar DASH (Dynamic Adaptive Streaming over HTTP), para la transmisión de video para las redes móviles celulares (Campo-Muñoz, Andrés, Escobar-Zapata, Juan, & Imbachi-Paz, 2019).

El protocolo DASH mejora la calidad percibida por el usuario al permitir el acceso para diferentes resoluciones de los contenidos de vídeo a través del protocolo de transferencia de hipertexto (HTTP), de acuerdo a las condiciones de la red (23009-1:2012, 2012). Sin embargo, este es un protocolo genérico no especializado por ejemplo para redes inalámbricas por lo que han surgido sus contrapartes privadas como el protocolo HTTP Smooth Streaming HSS, HTTP Live Streaming – HLS, HTTP Dynamic Streaming – HDS (Van Der Hooft et al., 2016), (Tian & Liu, 2016). En esta investigación se aplica el esquema de adaptación para DASH, mediante el uso del algoritmo adaptativo para streaming HTTP AAASH (Adaptation Algorithm for Adaptive Streaming over HTTP) (Vergados, Michalas, Sgora, & Vergados, 2014) que se aplica directamente sobre DASH y se analiza el comportamiento de los parámetros de calidad de servicio sobre una red inalámbrica.

AAASH, adapta la velocidad de vídeo solicitada por el cliente utilizando un mecanismo de adaptación compleja con múltiples condiciones y parámetros de configuración. El algoritmo se centra en el receptor y en que, el archivo multimedia se divide en segmentos (Vergados et al., 2014). Cada uno de los segmentos se proporciona en múltiples tasas de bits, y el cliente

es capaz de seleccionar la tasa apropiada para cada uno de los segmentos (Miller, Quacchio, Gennari, & Wolisz, 2012). Con este enfoque se pretende: estimar correctamente la dinámica del rendimiento de la red disponible, controlar el nivel de llenado de la memoria intermedia del cliente a fin de evitar subdesbordamientos resultando en interrupciones de reproducción, maximizar la calidad del streaming, mientras que evita las fluctuaciones de calidad innecesarias, y reducir al mínimo el retardo entre la petición del usuario y el inicio de la reproducción (Miller et al., 2012).

El algoritmo comienza por la selección de la representación más baja para el primer segmento a fin de minimizar el retardo entre la petición del usuario para ver el vídeo y el inicio real de la reproducción. Se introduce esta fase de arranque rápido más agresivo, con el fin de aumentar rápidamente hasta la mejor calidad factible. La fase de arranque rápido se procede de la siguiente manera, para cada descarga posterior se selecciona la siguiente representación más alta, mientras su tasa de bits es inferior a un determinado porcentaje del rendimiento, medido en los últimos segundos, el porcentaje varía dependiendo del nivel del búfer en uso (Vergados et al., 2014) (Miller et al., 2012).

En este artículo se analiza el comportamiento del algoritmo AAASH sobre una red móvil emulada sobre la herramienta de investigación NS-3 y el módulo LENA (LTE/EPC Network Simulator) (Sabbah, Jarwan, Issa, & Ibnkahla, 2018). Se obtienen los parámetros de calidad de servicio (QoS) como son el retardo (delay), la fluctuación del retardo (jitter) y la retransmisión de paquetes para escenarios con diferente número de usuarios.

El resto del artículo está organizado de la siguiente forma: en la sección 2 se presentan los trabajos relacionados los cuales permiten evidenciar la importancia de la temática y principalmente evidenciar las diferencias con la investigación aquí presentada. En la sección 3 se exponen las condiciones del escenario de experimentación. En la sección 4 contiene los resultados y su discusión, finalmente en la sección 5 se presentan las conclusiones y trabajos futuros que se derivan de esta investigación.

## 2. TRABAJOS RELACIONADOS

En esta sección se presentan diferentes trabajos que permiten evidenciar la importancia del

videostreaming y su enfoque desde el protocolo DASH. La investigación presentada en este artículo surge como respuesta a uno de los trabajos futuros presentados en (Campo-Muñoz et al., 2019), donde los autores realizan el estudio del algoritmo fuzzy DASH (FDASH) el cual es puesto en funcionamiento en una red LTE, analizan diferentes escenarios con diferente número de usuarios y estudian parámetros de QoS para comprobar que el algoritmo permite cumplir con los valores mínimos de QoS. Igualmente existen otras propuestas como la presentada en (Gadaleta et al., 2017) donde proponen la variante D-DASH, un framework que usa técnicas de aprendizaje profundo y aprendizaje reforzado para optimizar la calidad de la experiencia (QoE) de DASH.

Las tecnologías de transmisión adaptativa se usan para entregar contenido de video a través de redes de datos, pero aún enfrentan problemas en redes con ancho de banda limitado, para videos con definición ultra alta o superior, incluso cuando se usa la codificación de video de alta eficiencia. En (Concolato et al., 2018) los autores proponen una solución a estos problemas basada en tres elementos: una codificación en mosaico con restricción de movimiento; una derivación específica del formato de archivo de medios base ISO (International Organization for Standardization); y su uso en la transmisión adaptativa de contenido de video en mosaico utilizando DASH. Así mismo, en (Solano-Hurtado & Soto-Cordova, 2021) se realiza una evaluación del desempeño del protocolo DASH sobre transmisión de video, pero no proponen mejoras. En (Orozco, Muñoz, & Hoyos, 2021) los autores proponen la caracterización de un servicio de videostreaming soportado por el protocolo DASH para una red LTE. Se basan en entornos emulados donde se construyen diferentes escenarios, se obtienen trazas de tráfico y se encuentran las funciones de densidad de probabilidad que las describe.

En (Colonnese, Conti, Scarano, Rubin, & Cuomo, 2022), se propone una estrategia de asignación de ancho de banda consciente, para el servicio de video soportado sobre el protocolo DASH, transparente para el cliente. El enfoque, garantiza fluidez a todos los usuarios y brinda servicios basados en prioridades a los usuarios Premium, dicha propuesta puede ser adoptado por cada proveedor de servicios de internet donde sería necesario personalizarlo o adaptarlo a diferentes perfiles lo cual puede resultar mucho más engorroso.

En (Atehortua, Yesid Campo, & Elías Chanchí, 2022) los autores proponen una herramienta que estima la QoE para el servicio de videostreaming basado en el protocolo DASH a partir de parámetros de QoS basada en la norma P.1203 de la UIT. Esta propuesta es interesante como punto de inicio para que las empresas prestadoras de servicio puedan tomar correctivos en tiempo real con el objetivo de mejorar la QoE.

De acuerdo a la literatura existe mucha investigación que involucra al protocolo DASH y como se impactan la QoS y la QoE. A diferencia de dichos trabajos, en este artículo se aplica el algoritmo AAASH implementándolo en el cliente lo que permite que sea extrapolable a otras tecnologías como redes 5G o WiFi. Además, se analizan parámetros de QoS

### 3. ESCENARIO DE EXPERIMENTACIÓN

Como escenario de experimentación se usó la herramienta de investigación NS-3, por ser de libre uso. Como tecnología se escogió la red LTE por ser una tecnología madura y la que mayor tráfico soporta actualmente y que de acuerdo a (Ericsson, 2022) soportará más del 50% del tráfico total hasta el año 2025. Para la construcción del escenario de LTE se recurre a la librería LENA. En tanto que y para la puesta en funcionamiento del algoritmo AAASH, es necesario especificar sobre cada uno de los clientes el tipo de comunicación a establecer, en este caso Sockets TCP, usando la Clase "ns3::TcpSocketFactory" y se procede a definir cada uno de los atributos de la aplicación como son el "targetDt" que corresponden al Target Buffering Time (Tiempo de Búfer Objetivo) que tiene un valor inicial de 35s, y el periodo promedio para la estimación del throughput (Window) para este caso se usan 10 s (Campo-Muñoz et al., 2019). Para la codificación, se emplea el estándar H.264/MPEG-4. La relación señal a ruido más interferencia (SINR) toma valores entre -5 dB y 20 dB. La frecuencia de operación para el enlace de subida es de 1930 MHz y para el enlace de bajada de 2120 MHz y una canalización de 5 MHz. El modelo usado dentro de LENA es el lena-simple-epc (Malinverno et al., 2020), (NS-3 Project, 2023).

Los parámetros de QoS analizados son el retardo (delay), la fluctuación del retardo (jitter) y la pérdida de paquetes (packet loss). El UE accede a un Nodo B evolucionado (eNB) de la red LTE que corresponde a la red de accesos radio terrestre

universal evolucionada EUTRAN (Evolved-UMTS Terrestrial Radio Access Network); y a través de la red troncal EPC (Evolved Packet Core), a un servidor DASH (NS-3 Project, 2023)

Se crean tres escenarios donde para cada uno de ellos se analizan los parámetros de QoS. Para ejecutar el análisis de los tres escenarios se tomó como punto de partida un escenario básico que contiene un solo usuario, a partir del cual se construyen otros escenarios incrementando el número de usuarios.

#### 4. RESULTADOS Y DISCUSIONES

En esta sección se presentan los resultados por escenarios, el primero de los cuales es el básico y está conformado por un solo usuario, el escenario II está conformado por 10 usuarios, el escenario III está conformado por 23 usuarios.

En la Tabla 1 se presentan los valores de los parámetros de retardo y fluctuación de retardo en milisegundos (ms) y el porcentaje de paquetes perdidos (packet loss) (Handbook, 2020), (Cisco, 2017) contra los que se comparan los resultados de cada uno de los escenarios. Cabe resaltar que para el parámetro de paquetes perdidos se compara contra los paquetes retransmitidos, puesto que el protocolo DASH a nivel de capa 4 de la torre OSI trabaja con el protocolo TCP, por lo que aquellos paquetes que no tienen un asentimiento (acknowledge) son retransmitidos.

*Tabla 1: Parámetros de QoS*

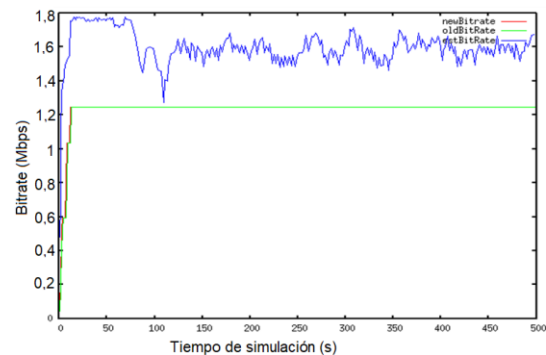
Parámetro	
Retardo (Delay)	150 ms
Fluctuación del retardo (Jitter)	30 ms
Paquetes perdidos (Packet Loss)	< 1 %

##### 4.1 Escenario I

En el primer escenario, un usuario adquiere el servicio de videostreaming con un tamaño del búfer de 35 segundos, pero con las características especificadas anteriormente en el escenario de experimentación.

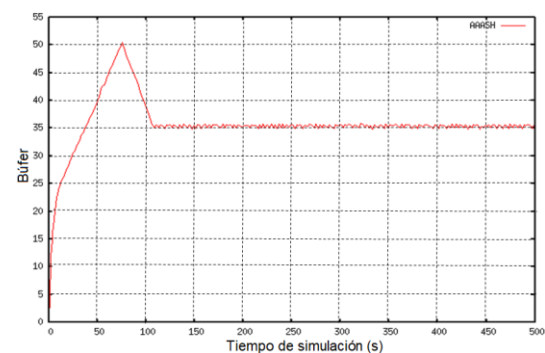
El tiempo de simulación es de 500 segundos que es suficiente para evaluar el comportamiento del servicio sobre el protocolo AAASH. En la Fig. 1 la curva denominada NewBitrate y OldBitrate representan la tasa de bits del usuario en dos instantes, según como trabaja el algoritmo AAASH. El rendimiento no corresponde a una resolución en particular, puesto que lo que hace el

algoritmo es cambiar las resoluciones con tal de mantener precisamente el rendimiento el cual corresponde a la curva denominada estBitRate. Donde el rendimiento del segmento se calcula en el cliente y corresponde a un cociente entre un producto y una diferencia. El producto se obtiene multiplicando la tasa de bits del segmento por su tiempo de duración, mientras que la diferencia se obtiene, restando el tiempo en el que todo el segmento ha sido recibido por el cliente menos el tiempo en que el segmento inicia la descarga. Respecto al rendimiento se observa que el protocolo requiere 12 segundos para estabilizar la tasa de bits en 1,21 Mbps



*Fig. 1. Nivel de Throughput de un usuario con algoritmo AAASH*

En la Fig. 2, se observa el comportamiento del búfer para el algoritmo AAASH, donde, en los primeros segundos de simulación este incrementa rápidamente hasta alcanzar un valor aproximado de 25 s de búfer durante los primeros 10 s de reproducción, luego alcanza un valor de 50 s de búfer entre los 1 s y los 70 s de reproducción, posteriormente desciende a los 35 s de búfer en los 110 s de reproducción, cuyo valor permanece estable observándose un rizado en su comportamiento.



*Fig. 2. Nivel del búfer de un usuario con algoritmo AAASH*

### 3.1.1 Retardo para un usuario

Para el cálculo del retardo promedio, de acuerdo a la documentación de NS-3 y su FlowMonitor, se calcula mediante el cociente entre el delaySum y los paquetes recibidos (rxPackets) cuyos valores se observan en la Fig. 3. Así, una vez realizados los cálculos se obtiene que, el valor promedio del retardo es de 15,05 ms. En la Fig. 4 se observa el comportamiento del retardo asociado al número de paquetes recibidos cuya suma coincide con el valor de los paquetes recibidos (rxPackets) de la Fig. 3.

```
<Flow timesForwarded="0" lostPackets="382" rxPackets="150700" txPackets="151089"
rxBytes="86412410" txBytes="86636068" lastDelay="+15064648.0ns"
jitterSum="+77657631353.0ns" delaySum="+2268284396931.0ns"
timeLastRxPacket="+499986999999.0ns" timeLastTxPacket="+499971935351.0ns"
timeFirstRxPacket="+284999999.0ns" timeFirstTxPacket="+271930314.0ns" flowId="2">
```

Fig. 3. Monitorización del flujo

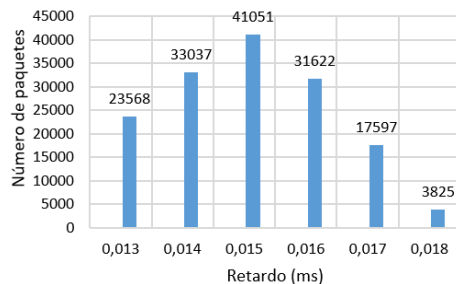


Fig. 4. Histograma del retardo para un usuario

Comparando los valores del retardo obtenidos para el escenario I con los especificados en la Tabla 1 el valor de este parámetro cumple para este escenario, es decir, cumple con el parámetro de QoS, puesto que los paquetes que más retardo presentan es de 0,018 ms menor a los 50 ms.

#### 4.1.2 Fluctuación del retardo para un usuario

La fluctuación promedio del retardo de acuerdo a la documentación de NS-3 y su FlowMonitor se calcula mediante el cociente entre el jitterSum y los paquetes recibidos menos uno (rxPackets) cuyos valores se observan en la Fig. 3. Así, una vez realizados los cálculos se obtiene que, el valor de la fluctuación promedio es de 0,51 ms, es decir que la mayoría de los paquetes muestran un valor de fluctuación del retardo cercano a cero como se observa en la Fig. 5. En la Fig. 5 se observa el comportamiento de la fluctuación del retardo asociado al número de paquetes recibidos cuya suma coincide con el valor de los paquetes recibidos (rxPackets) de la Fig. 3. En la Fig. 5 no se observa una barra para la fluctuación del retardo de 0,005 s por la escala del número de paquetes y ya que dicho número de paquetes asociado a este valor es de apenas 16.

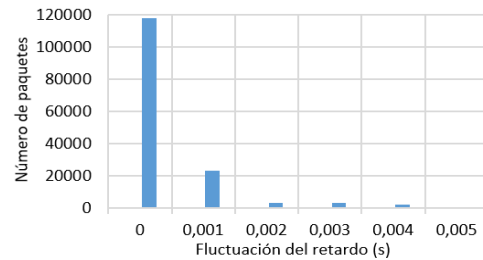


Fig. 5. Fluctuación del retardo para un usuario

De acuerdo a la Tabla 1, donde para el video establecen valores entre 30ms y 50 ms, la fluctuación del retardo para este escenario (0,51 ms) no representa problema, es decir cumple con los parámetros de QoS.

#### 4.1.3 Retransmisión de paquetes

Para este escenario la retransmisión de paquetes corresponde a lo que LENA denomina lostPackets. Cabe recordar que la transmisión se realiza a nivel de capa 4, mediante el protocolo TCP el cual en la cabecera IP se nota como protocolo 6 (protocol="6" ver Fig. 6, es decir los paquetes que se pierden luego son retransmitidos, dicho valor es igual a 382 como se puede observar en la Fig. 3. Así, este valor representa el 0,25% del total de paquetes transmitidos (txPackets). Por tanto, cumple con el parámetro de QoS de acuerdo a la Tabla 1.

```
<Ipv4FlowClassifier>
<Flow flowId="2" destinationPort="49153" sourcePort="80" protocol="6"
destinationAddress="7.0.0.2" sourceAddress="1.0.0.2"/>
```

Fig. 6. Transmisión bajo el protocolo TCP

## 4.2 Escenario II con 10 usuarios

En esta sección se muestra el comportamiento de los parámetros de QoS para 10 usuarios con el algoritmo AAASH consumiendo de manera simultánea un servicio de videostreaming.

### 4.2.1 Retardo para 10 usuarios

En la Fig. 7, se observa el retardo para cada uno de los 10 usuarios, dicho parámetro representa el valor promedio de los flujos de cada uno de los usuarios. El usuario 3 es quien presenta el mayor valor con 27,7 ms y el usuario 5 quien presenta el menor valor con 27,3 ms. Por tanto, el retardo que presenta cada uno de los usuarios para este escenario es muy similar. Además, de acuerdo a la Tabla 1, el retardo para el escenario II, cumple con los parámetros de QoS.



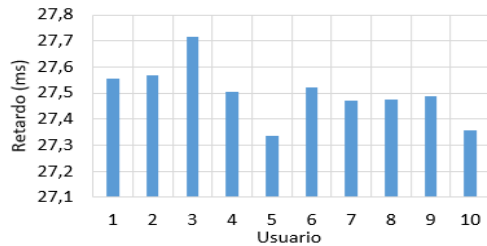


Fig. 7. Retardo para 10 usuarios

#### 4.2.2 Fluctuación del retardo para 10 usuarios

En la Fig. 8, se observa la fluctuación del retardo promedio para cada uno de los 10 usuarios. El usuario 4 presenta el mayor valor de fluctuación de retardo alcanzando los 0,53 ms. De acuerdo a la Tabla 1, la fluctuación del retardo para este escenario cumple con los parámetros de QoS.

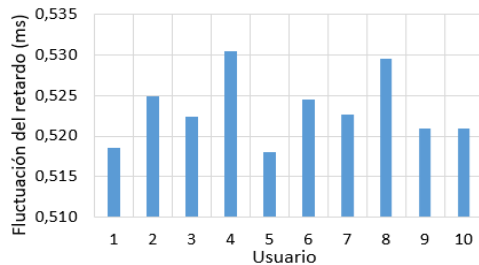


Fig. 8. Fluctuación del retardo para 10 usuarios

#### 4.2.3 Retransmisión de paquetes

La Fig. 9 presenta el porcentaje de paquetes perdidos, paquetes retransmitidos y paquetes perdidos. Así el usuario 10 es quien necesita se le retransmitan el mayor número de paquetes cuyo valor es de 1056, sin embargo, este número de paquetes apenas representa el 0,46% de los mismos. Mientras que, por ejemplo, al usuario 4 se le retransmiten apenas 816 paquetes, pero representan el 0,45% de los paquetes. En todo caso, de acuerdo a la Fig. 9 el número de paquetes que cada usuario necesita que se le retransmitan es inferior al 1%, por lo que cumple con el parámetro de QoS de acuerdo a la Tabla 1.

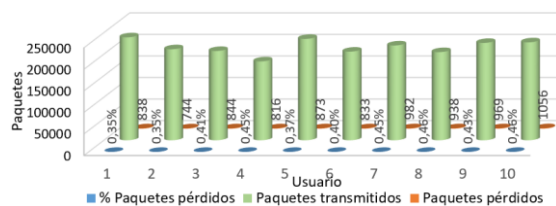


Fig. 9. Paquetes perdidos para 10 usuarios

### 4.3 Escenario III con 23 usuarios

En esta sección se muestra el comportamiento de los parámetros de QoS para 23 usuarios con el algoritmo AAASH consumiendo de manera simultánea un servicio de videostreaming. El número de 23 usuarios en el escenario III se escoge ya que, éste es el máximo número que permite la herramienta NS-3, puesto que al superar dicho valor la herramienta presenta fallas.

#### 4.3.1 retardo para 23 usuarios

En la Fig. 10, se observa como el retardo para 23 usuarios se ha incrementado en cada uno de ellos. En el peor de los casos se ha incrementado un 3,4 ms respecto al escenario con 10 usuarios. Así, se observa que el menor valor es de 30,52 ms para el usuario 4, mientras que el mayor valor es de 30,77 para el usuario 14. Por tanto, el retardo que presenta cada uno de los usuarios para este escenario es muy similar. Además, de acuerdo a la Tabla 1 el retardo para este escenario cumple con los parámetros de QoS, ver Tabla I.

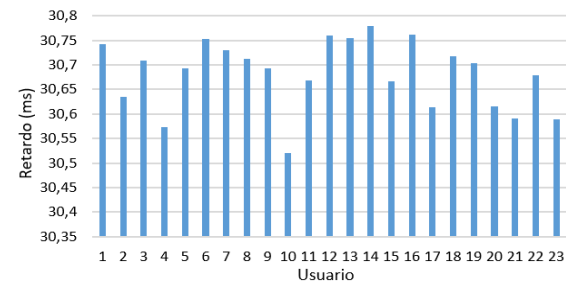


Fig. 10. Retardo para 23 usuarios

#### 4.3.2 Fluctuación del retardo para 23 usuarios

En la Fig. 11 se observa la fluctuación del retardo para cada uno de los 23 usuarios, el cual se ha incrementado en todos los usuarios. En el peor de los casos se ha incrementado 0,05 ms respecto a los valores presentados para 10 usuarios. El valor máximo lo presenta el usuario 16 con un valor de 0,582 ms. De acuerdo a la Tabla 1, la fluctuación del retardo para este escenario cumple con los parámetros de QoS, ver Tabla I.

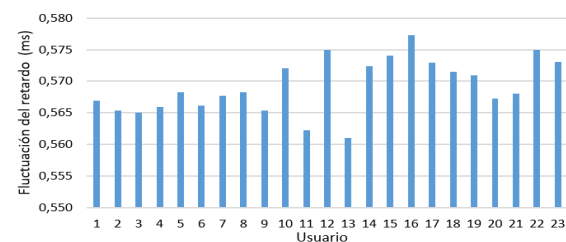


Fig. 11. Fluctuación del retardo para 23 usuarios

### 4.3.3 Retransmisión de paquetes

La Fig. 12, presenta el porcentaje de paquetes perdidos, paquetes retransmitidos y paquetes recibidos. Así el usuario 21 es quien necesita se le retransmitan el mayor número de paquetes cuyo valor es de 1391, este número de paquetes representa el 1,47% de los mismos. Comparando las Fig. 9, del escenario II con 10 usuarios versus la Fig. 12, del escenario III con 23 usuarios se observa el aumento en el número de retransmisiones y el porcentaje que representan, esto es, a mayor número de usuarios se observa como los parámetros de QoS van empeorando.

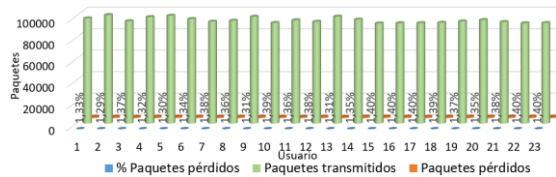


Fig. 12. Paquetes perdidos para 23 usuarios

En principio si comparamos el porcentaje de pérdida de paquetes de cada uno de los 23 usuarios observamos que se supera el valor del 1% de referencia de la Tabla 1, por tanto, ninguno de los usuarios estaría cumpliendo con dicho parámetro de QoS. Sin embargo, cabe aclarar que se está trabajando con el protocolo DASH que a nivel de capa 4 de la torre OSI trabaja con el protocolo TCP por tanto lo que existe en realidad es una retransmisión de paquetes y no una pérdida. Lo anterior explica la importancia del búfer de AAASH de 35 segundos, es decir que en realidad no existe una pérdida de paquetes y que los efectos de la retransmisión de los mismos deben ser absorbida por el búfer. Adicional a lo anterior cabe recordar que los experimentos se realizan sobre un canal de 5 MHz que puede soportar velocidades pico entre 1,6 Mbps y 21 Mbps (Agusti et al., 2010) por tanto cada uno de los 23 usuarios podría transmitir el video a una tasa muy baja de entre 69,5 Kbps y 913 Kbps, inferior al valor ideal de 1,21 Mbps calculado para el escenario I.

## 5. CONCLUSIONES

En este artículo se presenta el algoritmo AAASH en el lado del cliente de una red móvil celular a fin de analizar su comportamiento ante el aumento de usuarios consumiendo un servicio de videostreaming bajo el protocolo DASH. Los resultados muestran que el algoritmo permite cumplir con los parámetros de QoS a pesar del aumento de usuarios consumiendo de manera simultánea el servicio.

Cabe destacar el comportamiento del parámetro de fluctuación del retardo puesto que de acuerdo a la Tabla 1 tiene un valor máximo permitido de 30 ms, sin embargo, el máximo alcanzado fue de apenas 0,58 ms esto representa apenas un 1,9% del valor máximo. Asimismo, en el caso del retardo el valor máximo permitido es de 150 ms, sin embargo, el valor máximo alcanzado fue de 30,75%, esto representa apenas un 20% del valor máximo.

La pérdida de paquetes es el parámetro más crítico pues en el escenario con 23 usuarios no se cumple lo especificado en la Tabla 1. Sin embargo, se debe saber interpretar este comportamiento desde dos puntos de vista: i) cabe recordar que precisamente el protocolo DASH evita la pérdida de paquetes al transportarse sobre TCP. Así, surge la necesidad de que el equipo de usuario tenga la suficiente capacidad en memoria como para soportar un búfer de 35 s, lo cual con las capacidades de los dispositivos actuales parece un requerimiento fácil de cumplir, ii) el escenario III excede las capacidades puesto que no soporta el mínimo requerido de una tasa de bits de 1,21 Mbps calculado para el escenario I.

De acuerdo a los resultados se puede concluir que el algoritmo AAASH a ejecutarse sobre el cliente, es una opción válida para soportar videostreaming en una red móvil celular.

Como trabajo futuro se sugiere estudiar la QoE para estos mismos escenarios y así conocer el efecto de la retransmisión de paquetes y saber si se presentan evento de estancamiento.

## RECONOCIMIENTO

Los autores del artículo agradecen a la universidad del Quindío por el apoyo prestado para la realización de esta investigación.

## REFERENCIAS

- 23009-1:2012, I. (2012). ISO - ISO/IEC 23009-1:2012 - Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats. Retrieved August 3, 2022, from <https://www.iso.org/standard/57623.html>
- Agusti, R., Bernardo, F., Casadevall, F., Ferrús, R., Pérez-Romero, J., & Sallent, O. (2010). LTE: NUEVAS TENDENCIAS EN COMUNICACIONES MÓVILES

## AUTORES.

- Atehortua, J., Yesid Campo, W., & Elías Chanchí, G. (2022). Proposal of a tool for the automatic estimation of quality of experience in MPEG-DASH scenarios. *Przeład Elektrotechniczny*.  
<https://doi.org/10.15199/48.2022.08.26>
- Campo-Muñoz, W. Y., Andrés, Escobar-Zapata, F., Juan, & Imbachi-Paz, C. (2019). Análisis del servicio de video streaming basado en el algoritmo FDASH sobre LTE. *Ciencia e Ingeniería Neogranadina*, 29(1), 67–80.  
<https://doi.org/10.18359/RCIN.3122>
- Cisco. (2017). Video Quality of Service (QoS) Tutorial - Cisco. Retrieved April 1, 2023, from  
<https://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/qos-video/212134-Video-Quality-of-Service-QoS-Tutorial.html>
- Colonnese, S., Conti, F., Scarano, G., Rubin, I., & Cuomo, F. (2022). Premium quality or guaranteed fluidity? Client-transparent DASH-aware bandwidth allocation at the radio access network. *Journal of Communications and Networks*, 24(1), 59–67.  
<https://doi.org/10.23919/JCN.2021.000046>
- Concolato, C., Le Feuvre, J., Denoual, F., Mazé, F., Nasser, E., Ouedraogo, N., & Taquet, J. (2018). Adaptive Streaming of HEVC Tiled Videos Using MPEG-DASH. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(8), 1981–1992.  
<https://doi.org/10.1109/TCSVT.2017.2688491>
- Ericsson. (2022). Past and present reports - Mobility Report - Ericsson. Retrieved April 1, 2023, from  
<https://www.ericsson.com/en/reports-and-papers/mobility-report/reports>
- Gadaleta, M., Chiariotti, F., Rossi, M., & Zanella, A. (2017). D-DASH: A Deep Q-Learning Framework for DASH Video Streaming. *IEEE Transactions on Cognitive Communications and Networking*, 3(4), 703–718.  
<https://doi.org/10.1109/TCCN.2017.2755007>
- Handbook, C. Q. (2020). Cisco QoS Handbook 2nd Edition.
- Malinverno, M., Mangués-Bafalluy, J., Casetti, C. E., Chiasserini, C. F., Requena-Esteso, M., & Baranda, J. (2020). An edge-based framework for enhanced road safety of connected cars. *IEEE Access*, 8, 58018–58031.  
<https://doi.org/10.1109/ACCESS.2020.2980902>
- Miller, K., Quacchio, E., Gennari, G., & Wolisz, A. (2012). Adaptation algorithm for adaptive streaming over HTTP. *2012 19th International Packet Video Workshop, PV 2012*, 173–178.  
<https://doi.org/10.1109/PV.2012.6229732>
- NS-3 Project. (2023). ns-3: src/lte/examples/lena-simple-epc.cc Source File. Retrieved April 1, 2023, from  
[https://www.nsnam.org/docs/release/3.19/xygen/lena-simple-epc\\_8cc\\_source.html](https://www.nsnam.org/docs/release/3.19/xygen/lena-simple-epc_8cc_source.html)
- Orozco, H. F. B., Muñoz, W. Y. C., & Hoyos, E. A. (2021). Revista Ingeniería. *Ingeniería*, 26(3), 381–400.  
<https://doi.org/10.14483/23448393.17960>
- Sabbah, A., Jarwan, A., Issa, O., & Ibnkahla, M. (2018). Enabling LTE emulation by integrating CORE emulator and LTE-EPC Network (LENA) Simulator. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, 2017-October*, 1–6.  
<https://doi.org/10.1109/PIMRC.2017.8292642>
- Solano-Hurtado, L. R., & Soto-Cordova, M. M. (2021). A Study on Video Streaming Quality of DASH Scheme in Multimedia Services. *2021 7th Congreso Internacional de Innovación y Tendencias En Ingeniería, CONIITI 2021 - Conference Proceedings*.  
<https://doi.org/10.1109/CONIITI53815.2021.9619720>
- Tian, G., & Liu, Y. (2016). Towards Agile and Smooth Video Adaptation in HTTP Adaptive Streaming. *IEEE/ACM Transactions on Networking*, 24(4), 2386–2399.  
<https://doi.org/10.1109/TNET.2015.2464700>
- Van Der Hooft, J., Petrangeli, S., Wauters, T., Huysegems, R., Alfaced, P. R., Bostoen, T., & De Turck, F. (2016). HTTP/2-Based Adaptive Streaming of HEVC Video over 4G/LTE Networks. *IEEE Communications Letters*, 20(11), 2177–2180.  
<https://doi.org/10.1109/LCOMM.2016.260187>
- Vergados, D. J., Michalas, A., Sgora, A., & Vergados, D. D. (2014). A fuzzy controller for rate adaptation in MPEG-DASH clients. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, 2014-June*, 2008–2012.  
<https://doi.org/10.1109/PIMRC.2014.7136501>