

Evaluación de un algoritmo de control para video adaptativo empleando enrutamiento tradicional y SDN

Evaluation of an Adaptive Video Control Algorithm using Traditional Routing and SDN

Marlon Gomezcoello Rodríguez¹ <https://orcid.org/0009-0009-5118-2397>,
Justin Picón Barros¹ <https://orcid.org/0009-0008-6319-4750>, Santiago González Martínez¹
<https://orcid.org/0000-0001-6604-889X>

¹Universidad de Cuenca, Departamento de Eléctrica, Electrónica y Telecomunicaciones, Cuenca, Ecuador
xavier.gomezcoello94@ucuenca.edu.ec,
justin.picon@ucuenca.edu.ec, santiago.gonzalez@ucuenca.edu.ec



Esta obra está bajo una licencia internacional Creative Commons Atribución-NoComercial 4.0.

Enviado: 2025/04/11
Aceptado: 2025/05/15
Publicado: 2025/06/30

Resumen

Actualmente, según el Cisco Annual Internet Report, entre el 80% y el 85% del tráfico de datos en Internet es generado por contenido en video. Esto hace imperativo desarrollar métodos o algoritmos que permitan una transmisión eficiente de contenido multimedia a través de la red. En este contexto, el presente artículo propone un algoritmo optimizado para el control de tráfico de video en un entorno SDN, con el objetivo de permitir una transmisión adaptativa en tiempo real. La metodología incluyó la transmisión de un video codificado a un *bitrate* mínimo; posteriormente, se midió el ancho de banda de la red a partir de los cuales se ajustó la tasa de codificación y la calidad del video; y, finalmente, el contenido se reprodujo en el lado del cliente. Los resultados obtenidos mediante el algoritmo de control se compararon con los generados por protocolos de enrutamiento comunes como *Routing Information Protocol* (RIP) y *Open Shortest Path First* (OSPF). El uso del algoritmo de control dio como resultado un retardo del 18.1% menor que RIP y del 94.8% menor que OSPF, con tráfico de fondo se obtuvo que SDN tiene un retardo del 27.23% menor que RIP y del 54.1% menor que OSPF. La principal contribución del trabajo es, mediante la implementación de un *testbed*, realizar el análisis del mecanismo que mejore la calidad de servicio y experiencia (QoS y QoE) que optimice la transmisión de video en tiempo real mediante la selección de la mejor ruta y/o el ajuste dinámico del *bitrate* del video según las condiciones de la red.

Sumario: Introducción, Materiales y Métodos, Resultados y Discusión, Conclusiones.

Como citar: Gomezcoello, M., Picón, J. & González, S. (2025). Evaluación de un algoritmo de control para video adaptativo empleando enrutamiento tradicional y SDN. *Revista Tecnológica - Espol*, 37(1), 264-284. <https://rte.espol.edu.ec/index.php/tecnologica/article/view/1308>

Palabras clave: streaming, adaptable, rip, ospf, redes definidas por software.

Abstract

Currently, according to the Cisco Annual Internet Report, between 80% and 85% of internet data traffic is generated by video content, making it imperative to develop methods or algorithms that enable the efficient transmission of multimedia content over the network. In this context, this article proposes an optimized algorithm for video traffic control in an SDN environment, allowing for adaptive real-time transmission. The methodology consists of the following stages: first, the video is transmitted at a minimum encoded bitrate; then, the available network bandwidth is measured, based on which the encoding rate and video quality are adjusted; and finally, the video is played on the client side. The results obtained using the control algorithm were compared with those obtained using common routing protocols such as RIP (Routing Information Protocol) and OSPF (Open Shortest Path First). The use of the control algorithm resulted in an 18.1% lower delay compared to RIP and a 94.8% lower delay compared to OSPF. Under background traffic conditions, SDN demonstrated a 27.23% lower delay than RIP and a 54.1% lower delay than OSPF. The main contribution of this work is, through the implementation of a testbed, the analysis of a mechanism that improves quality of service (QoS) and quality of experience (QoE) by optimizing real-time video transmission through the selection of the best route and/or the dynamic adjustment of video bitrate according to network conditions.

Keywords: Streaming, adaptable, rip, ospf, software-defined networks.

Introducción

Las Redes Definidas por Software (SDN) representan un paradigma innovador en el campo de las redes informáticas, en el cual los administradores emplean herramientas de software para gestionar sus infraestructuras de red. La SDN optimiza la capacidad de administración al enfocarse en un controlador lógicamente centralizado que proporciona interfaces de configuración de alto nivel y abstracciones de red (Bonaventure et al. 2017).

En el contexto de los proveedores de servicios de Internet (ISP) y las redes empresariales actuales, las redes SDN desempeñan un papel crucial. Esto se debe a que las demandas y requisitos de las redes de los ISP están en constante evolución debido al creciente consumo de contenido multimedia.

Un estudio de Sandvine revela que en 2022 el 65% del tráfico de Internet fue generado por el consumo de video, un aumento del 24% respecto al año anterior (Sam, 2023). Estos datos evidencian el papel crucial del video en las redes, destacando la necesidad de optimizar su gestión. Esto podría lograrse mediante sistemas de control que mejoren la entrega de contenido, ya sea, ajustando la calidad del video según el ancho de banda disponible.

A continuación, se discuten los principales trabajos relacionados disponibles en la literatura sobre el tema. En Makolkina et al. (2017), los autores destacan la importancia de explorar nuevas formas de entregar contenido de video a los usuarios, particularmente mediante redes definidas por software (SDN). En su estudio, implementaron una infraestructura de IPTV sobre SDN para evaluar el impacto de esta tecnología en la calidad de transmisión de video. La topología de prueba consistió en tres switches OpenFlow conectados en anillo. Se analizó el retardo de paquetes en una red con SDN y en una red tradicional sin controlador SDN. Los resultados muestran un retardo promedio de 1.8 ms en la red SDN frente

a 11.2 ms en la red tradicional, lo que sugiere que la arquitectura SDN es una opción eficaz para la distribución de contenido multimedia.

Para ajustar el tráfico de video según la congestión de la red y reducir los tiempos de carga, se emplea el método descrito en Lin (2015), que prioriza las necesidades de los usuarios. A cada usuario se le asigna una prioridad en función de sus requerimientos de calidad de video: por ejemplo, los usuarios que solicitan video en alta definición reciben mayor prioridad que aquellos que solicitan baja definición. El sistema utiliza un algoritmo de enrutamiento multitrayectoria que considera tanto las prioridades de los usuarios como la carga de la red para seleccionar las rutas óptimas para cada flujo de video, además, incorpora un mecanismo de control de congestión que, al detectar saturación, reduce el ancho de banda de los flujos de menor prioridad. Los experimentos muestran que los clientes de mayor prioridad cuentan con una tasa de datos garantizada de 1.5 Mbps, mientras que los de menor prioridad reciben 1 Mbps.

En Jiawei et al. (2018), se propone un algoritmo de enrutamiento multipath dinámico y adaptativo para redes definidas por software (SDN), que considera la pérdida de paquetes, el retardo y el ancho de banda como restricciones. Este algoritmo, llamado *Dynamic and Adaptive Multi-path Routing* (DAMR), emplea la información de estado global del controlador SDN para calcular rutas óptimas, distribuyendo el tráfico a través de múltiples caminos y mejorando así el rendimiento de la red, además, minimiza el impacto de fallas al redirigir rápidamente el tráfico a rutas alternativas. En simulaciones de redes SDN a gran escala, DAMR mostró solo un 10% de pérdida de paquetes frente al 60% del enrutamiento de un solo camino, y un retardo de 0.05 ms frente a 0.4 ms.

En Banchuen et al. (2018) se destaca la relevancia de la videoconferencia en telecomunicaciones y la necesidad de asegurar una alta calidad de servicio. Para optimizar el rendimiento sin modificar la aplicación cliente, se propone el uso de redes SDN, implementando la técnica de par de paquetes para evaluar el ancho de banda y el retardo entre clientes. Esto permite al controlador SDN elegir la mejor ruta, basándose en la topología, puertas de enlace, direcciones de host, latencia y ancho de banda disponible. En las pruebas, la red SDN mostró una latencia de 100 ms, significativamente menor que los 230 ms obtenidos en una red tradicional.

La construcción de árboles de multidifusión es una técnica común en redes tradicionales para implementar multicast. En Sheu et al. (2015), los autores proponen el algoritmo *Maximum Bandwidth Utilization Algorithms with Shortest Path* (MBUA-SP) para transmitir video en entornos SDN. Este algoritmo tiene como objetivo optimizar el ancho de banda máximo disponible entre el origen y cada cliente a lo largo de la ruta con menor latencia, maximizando el rendimiento del enlace servidor-cliente antes de enfrentar problemas de throughput. Los resultados experimentales muestran que el algoritmo reduce la tasa de rechazo en un 20% en comparación con el algoritmo de multicast *Shortest-path tree* (SPT) y aumenta la utilización del enlace en un 80%, es decir, el porcentaje de ancho de banda consumido respecto al total disponible en la red.

Khan (2023) propone el uso de un algoritmo centrado en el usuario, priorizando el contenido que más consume, además, si el usuario prefiere calidades de video altas, el algoritmo entrega dicha calidad, siempre y cuando el ancho de banda lo permita, para lo cual se realiza una medición.

En Abar et al. (2021), los autores aplican el método DASH en SDN para monitorear la calidad del video entregado al usuario, estructurando el enfoque en varias fases: una asignación inicial equitativa del *bitrate*, seguida de la mejora de la calidad visual mediante un algoritmo específico. El controlador SDN gestiona la red, mientras un modelo de procesos de Poisson y técnicas de *machine learning* optimizan la tasa de bits, incluyendo la selección de *bitrate*, cuadros por segundo y resolución del video. Simulaciones realizadas en Weka, usando Random Forest, AdaBoost y Stacking, mostraron que el método Stacking supera a Random Forest, logrando una métrica de recuperación de 0.95 frente a 0.92, por lo que es el más efectivo para predecir recursos óptimos según el comportamiento del usuario.

En Yamansavascular et al. (2020), los autores proponen un plano de datos tolerante a fallos denominado DPQoAP (*Protection with Quality of Alternative Paths*), que considera tanto los fallos de red como la calidad de las rutas alternativas. Para su implementación, se utilizan módulos de detección de congestión basados en DASH, que identifican la mejor ruta alternativa en función de la latencia. Este método establece una ruta primaria y dos de respaldo. Cuando la segunda ruta se satura, el módulo DPQoAP ajusta automáticamente la tercera como secundaria debido a su menor latencia. En caso de fallo en la ruta primaria, ya se ha evaluado la calidad de las rutas alternativas, garantizando una comunicación continua con calidad aceptable. Para validar este enfoque, se comparó con la "protección estática de enlace", donde el controlador busca una nueva ruta ante una falla. En una simulación de fallo en el enlace servidor-cliente, el método propuesto mantuvo el *throughput* cerca de 1000 paquetes/segundo, en contraste con una caída a 250 paquetes/segundo con la protección estática.

El estudio en Sayit et al. (2019) presenta una arquitectura altamente escalable para diferenciar servicios de red en el contexto de DASH sobre SDN. Esta arquitectura incluye dos componentes clave: módulos de control en el controlador SDN y un marco de optimización para establecer rutas de transmisión hacia los clientes DASH. En la capa de control, el controlador gestiona la red, definiendo rutas de flujo entre clientes y servidor y enviando estas reglas a los conmutadores. Una vez establecidas, las rutas se actualizan y envían a los conmutadores, permitiendo el enrutamiento dinámico de los paquetes. Los experimentos mostraron un aumento de hasta un 83% en la tasa de bits promedio y una reducción del 39% en la variación de calidad de video, en comparación con el enfoque de rutas más cortas.

En Abuteir et al. (2016) se describe una nueva técnica para mejorar la entrega de video de alta calidad en redes inalámbricas con múltiples clientes denominado *Wireless Network Assisted Video Streaming* (WNAVS), basado en Redes Definidas por Software (SDN). WNAVS realiza un modelado dinámico del tráfico y asigna ancho de banda en tiempo real a los clientes. En comparación con el método tradicional *Dynamic Adaptive Streaming over HTTP* (DASH), los experimentos demostraron que WNAVS logró una tasa de transmisión de 1000 Kbps para 8 clientes simultáneos, mientras que DASH solo alcanzó 500 Kbps.

En Storck et al. (2020) se implementa el protocolo de Transmisión Adaptativa Dinámica sobre HTTP (DASH, por sus siglas en inglés) en un servidor V2X, el gNodeB recibe las solicitudes de video enviadas por los vehículos a la vez que mide las condiciones de la red, para de esa manera entregar la mejor calidad de video de acuerdo con el ancho de banda disponible.

En la propuesta expuesta en Dac et al. (2021), el video a transmitir se divide en varios segmentos y cada segmento es codificado a diferente tasa, es decir, cada segmento tiene diferente calidad, los segmentos se almacenan en el lado del cliente en el búfer. Si este se mantiene lleno, indica que hay un buen ancho de banda, lo que permite descargar el segmento

de mejor calidad; en cambio, si el búfer no permanece lleno, significa que el ancho de banda ha disminuido, y se opta por descargar segmentos de menor calidad.

La mayoría de los reproductores utilizan algoritmos de tasa de bits adaptativa para elegir dinámicamente la tasa de bits para cada fragmento según la capacidad percibida de la red, lo que mejora la calidad de escenas dinámicas, pero ocupa mayor codificación de la necesaria para escenas estáticas, por lo que, en Li et al. (2023) se propone un algoritmo de tasa de bits adaptativa donde se seleccione una mayor tasa de codificación únicamente para fragmentos dinámicos.

A modo de resumen, en la metodología usada en los trabajos previos se elaboró una tabla que indica si cada trabajo fue implementado mediante simulación o con equipamiento físico. Los trabajos se presentan en el mismo orden que fueron expuestos.

Tabla 1

Metodología usada en los trabajos previos

TRABAJOS RELACIONADOS	SIMULACION	EQUIPAMIENTO
<i>Quality of experience estimation for video service delivery based on sdn core network Makolkina et al. (2017)</i>		X
<i>Research on video traffic control technology based on sdn Lin (2015)</i>	X	
<i>Dynamic and adaptive multi-path routing algorithm based on software-defined network Jiawei et al. (2018)</i>	X	
<i>An sdn framework for video conference in inter-domain network Banchuen et al. (2018)</i>	X	X
<i>Efficient multicast algorithms for scalable video coding in software-defined networking Sheu et al. (2015)</i>	X	
<i>User-centric algorithms: Sculpting the future of adaptive video streaming Khan (2023)</i>	X	
<i>Chapter five - user behavior-ensemble learning based improving qoe fairness in http adaptive streaming over sdn approach Abar et al. (2021)</i>	X	
<i>Fault tolerance in sdn data plane considering network and application based metrics Yamansavascilar et al. (2020)</i>	X	
<i>A scalable network layer service differentiation architecture for dash over sdn Sayit et al. (2019)</i>	X	
<i>An sdn approach to adaptive video streaming in wireless home network Abuteir et al. (2016)</i>	X	
<i>A performance analysis of adaptive streaming algorithms in 5g vehicular communications in urban scenarios Storck et al. (2020)</i>	X	
<i>Qoe-aware video adaptive streaming over http Dac et al. (2021)</i>	X	
<i>An apprenticeship learning approach for adaptive video streaming based on chunk quality and user preference Li et al. (2023)</i>	X	

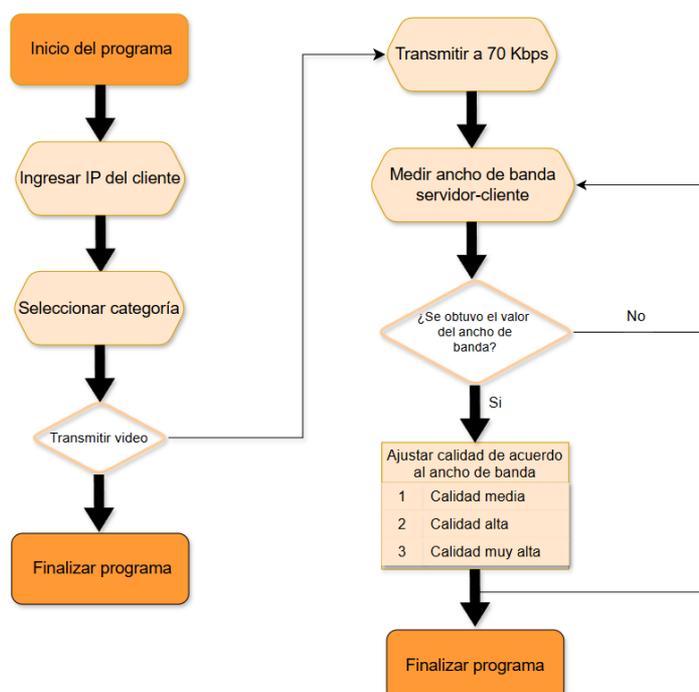
En tal contexto, a partir del análisis de los estudios previos, en el presente trabajo se propone el desarrollo e implementación de un algoritmo de control para la transmisión de video adaptativo en redes SDN, diseñado específicamente para entornos de redes de comunicaciones multimedia. Para evaluar la solución propuesta se implementó un *testbed* el cual consiste en un conjunto de cinco Raspberry Pi para emular una red SDN y comparar su rendimiento frente a esquemas de enrutamiento tradicional, en concreto *Routing Information Protocol (RIP)* y *Open Shortest Path First (OSPF)*. La implementación física de un banco de pruebas se debe a la necesidad de experimentar y analizar los resultados obtenidos en un medio que se asemeje a una red de uso real, ya que, la mayoría de los trabajos previos se centran en simulaciones y no en un banco de pruebas físico.

Materiales y Métodos

Para ilustrar mejor la metodología de la lógica de control se ha desarrollado un diagrama de flujo que se ilustra en la Figura 1. Como se observa en dicha figura, en la interfaz gráfica, se debe ingresar la dirección IP del cliente y seleccionar la categoría del video a transmitir. Inicialmente, el sistema transmite a una tasa de 70 Kbps debido a la falta de conocimiento sobre el ancho de banda disponible entre el servidor y el cliente, dicho valor ha sido seleccionado debido a que, al realizar varios experimentos se observó que 70 Kbps es un valor de codificación en el que se puede obtener una calidad de imagen aceptable sin necesidad de usar mucho ancho de banda, a menores valores la calidad se degradaba mucho por lo que no era pertinente transmitir a dicha calidad. Durante esta transmisión, se mide el ancho de banda y, una vez obtenido, se ajusta la tasa de codificación. Si no se puede medir el ancho de banda, se mantiene la tasa actual y se intenta una nueva medición, asegurando así un control sobre la calidad del video sin exceder la capacidad de la red.

Figura 1

Diagrama de flujo del sistema de control de video



A continuación, se presenta un pseudocódigo para representar el mecanismo que hemos propuesto para la aplicación de nuestro mecanismo de control:

```

Inicializar programa
Ingresar IP del cliente
Seleccionar la categoría de video
Transmitir a una tasa = 70 Kbps y medir ancho de banda
  Si ancho de banda bajo
    Transmitir a 70 Kbps
    Medir ancho de banda
  Si ancho de banda medio
    Transmitir a tasa de codificación media según categoría
    Medir ancho de banda
  Si ancho de banda alto
    Transmitir a tasa de codificación alta según categoría
    Medir ancho de banda
  Si ancho de banda muy alto
    Transmitir a tasa de codificación muy alta según categoría
    Medir ancho de banda
Fin si
  Finalizar transmisión
Finalizar programa

```

Metodología

Una vez expuesta la metodología del sistema de control, es fundamental detallar las herramientas y procesos necesarios para su correcta implementación. Este proceso y las herramientas utilizadas se describen detalladamente en la Figura 2.

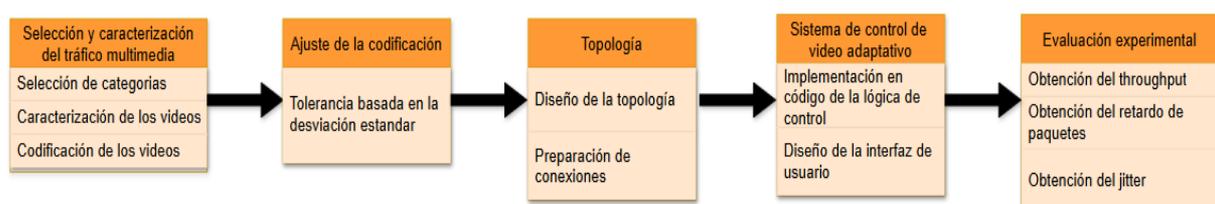
En la subsección de selección y caracterización del tráfico multimedia se describen los criterios definidos para escoger las distintas categorías de video que se va a utilizar, así como los criterios para establecer las tasas de codificación. En la subsección de ajuste de la codificación se presenta el uso de la desviación estándar para la correcta selección de las tolerancias de las tasas de codificación para las distintas categorías.

Luego, en la subsección de topología, se presenta la topología utilizada. En la subsección de preparación del sistema de control de video adaptativo se detalla cómo se implementó el sistema de control de video adaptativo, el mismo que ajusta la calidad del video al tener conocimiento del ancho de banda del enlace con el cliente. Finalmente, en la sección de Resultados y Discusión, se exponen los distintos experimentos que se realizaron.

Todos los comandos, códigos y *scripts* han sido liberados para su uso académico o investigativo en Gomezcoello et al. (2024).

Figura 2

Herramientas y procesos para la implementación del sistema de control



Selección y caracterización del tráfico multimedia

En esta fase se procedió con la selección de las categorías de los siguientes videos que serán transmitidos:

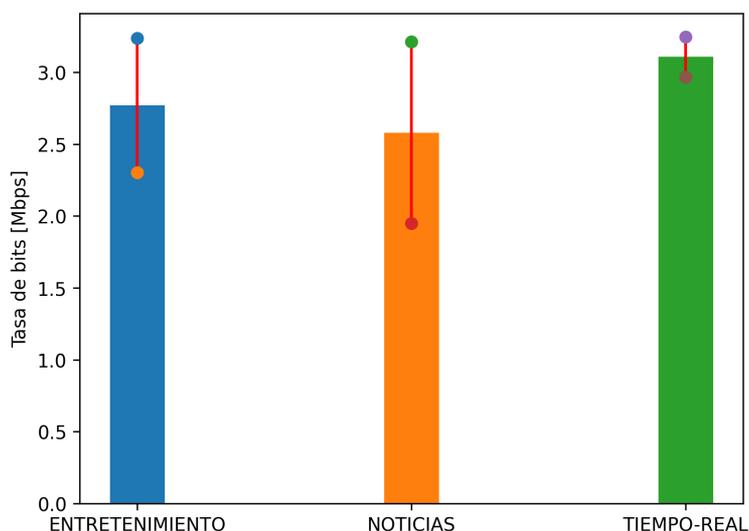
- Noticias
- Entretenimiento
- Tiempo real

Estas categorías fueron seleccionadas a base de la alta demanda actual.

En primer lugar, se capturó el tráfico de transmisiones en vivo según las clases de video previamente seleccionadas. Esto permitió obtener información sobre las tasas de codificación utilizadas en diferentes servicios de *streaming* de video dentro de las distintas categorías seleccionadas. Con estos datos, se puede establecer tasas de codificación realistas para los videos que se va a utilizar, asegurando así que reflejen las prácticas comunes de codificación en Internet. El resultado se presenta en la Figura 3.

Figura 3

Tráfico real de diferentes categorías de servicio



Para ilustrar los resultados de los experimentos realizados se ha utilizado un intervalo de confianza del 95% y barras que muestran los valores promedio.

La figura anterior muestra que la categoría de noticias tiene la tasa de codificación más baja (aproximadamente de 2.5 Mbps), además de contar con un intervalo de confianza más amplio, lo que indica que en algunos instantes la tasa de codificación fue inferior a 2 Mbps o superior a 3 Mbps, las tasas de codificación bajas indican que, al ser una secuencia con pocos cambios o movimientos entre *frames*, el video aprovecha la redundancia espacial y temporal cuando el presentador habla, permitiendo tasas menores sin perder calidad; sin embargo, cuando existen cambios de escenas entre noticias, se usa una codificación más alta, lo cual no sucede con mucha frecuencia, no siendo así el uso de escenas estáticas; en cambio, los contenidos de entretenimiento requieren una tasa más alta debido a los frecuentes cambios de escena, que aumentan la cantidad de información por segundo para mantener la calidad visual.

Por otro lado, los videos en tiempo real tienen una tasa de codificación ligeramente mayor a 3 Mbps, siendo esta, la tasa de codificación más alta de entre todas las categorías, además se observa un intervalo de confianza de menor amplitud, lo cual indica que en la mayor parte de tiempo la tasa de codificación se mantiene elevada (entre 2.8 Mbps y 3.2 Mbps). Esto se debe a la necesidad de procesar y transmitir simultáneamente el contenido, lo que exige una alta tasa para garantizar una transmisión fluida y de calidad, sin interrupciones ni retrasos.

Luego de analizar los resultados obtenidos, se llegó a la conclusión de que la tasa de bits a usar al momento de realizar nuestros experimentos serán diferentes para cada categoría de video, cuyos valores seleccionados serán presentados en la siguiente subsección.

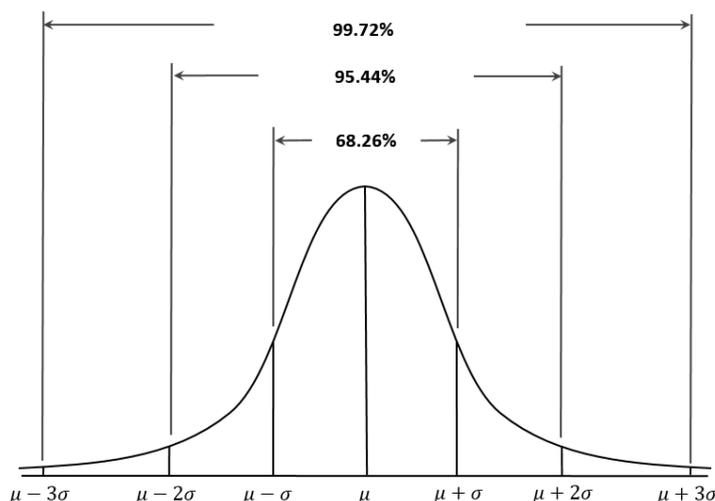
Ajuste de la codificación

De acuerdo con la teoría del límite central, independientemente de la forma de la población, la distribución muestral de la media se aproxima a una curva normal si el tamaño de la muestra es, estadísticamente, significativo en Witte et al. (2017); por lo tanto, al tener claro este concepto, se modeló las muestras obtenidas del tráfico real como si tuvieran una distribución normal.

La regla empírica de probabilidad expone que, cada dato que se puede observar caerá bajo tres desviaciones estándar diferentes de la media en una distribución normal, de acuerdo con la regla, el 68% de los datos caerán en la primera desviación estándar, el 95% caerán en la primera y la segunda desviación y el 99,7% de los datos caerán en las tres desviaciones Team (2020). Dicho concepto se ilustra en la Figura 4.

Figura 4

Regla empírica



Luego de exponer los conceptos anteriores, se procedió a establecer los valores de codificación para las distintas calidades de acuerdo con la regla empírica; es decir, el valor para la codificación de la calidad media será restando una desviación estándar del valor promedio ($\mu - \sigma$); el valor para la calidad alta será el valor del promedio (μ); y el valor para la calidad muy alta será sumando una desviación estándar al valor promedio ($\mu + \sigma$). Esto se debe a que dichos valores tienen altas probabilidades de aparecer en una transmisión, logrando así un modelado más realista de cómo se codifican las distintas categorías. En todas las categorías, el único valor de codificación constante es de 70 Kbps, establecido debido a la necesidad de una codificación baja para iniciar la transmisión de video.

Los valores de codificación para las distintas categorías se exponen en la siguiente Tabla 2.

Tabla 2

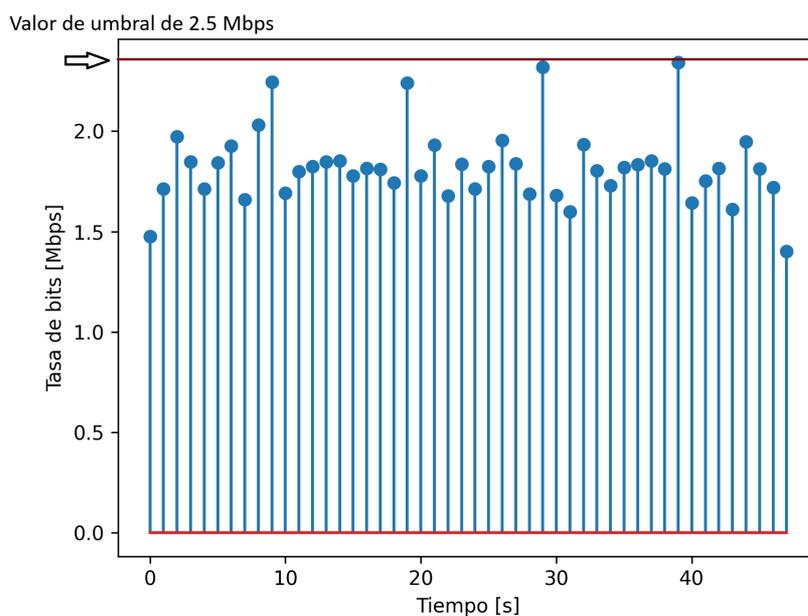
Tasas de codificación para las distintas categorías y calidades

CATEGORIA	BAJA	MEDIA	ALTA	MUY ALTA
Noticias	70 Kbps	1703 Kbps	2580 Kbps	3457 Kbps
Entretenimiento	70 Kbps	1829 Kbps	2770 Kbps	3712 Kbps
Tiempo real	70 Kbps	2051 Kbps	3107 Kbps	4164 Kbps

Con las codificaciones previas, se pudo obtener información relevante sobre el perfil de tráfico. Con los resultados obtenidos de los perfiles de tráfico, se procedió a configurar una tasa de codificación personalizada para cada categoría de tráfico y establecer los umbrales de ancho de banda para seleccionar el valor de codificación de acuerdo con el valor pico del perfil de tráfico; por ejemplo, para la categoría de noticias el perfil de tráfico para la calidad media cuenta con picos alrededor de 2.5 Mbps, como se observa en la Figura 5 por lo que, un valor de umbral adecuado debe ser superior a dicho pico.

Figura 5

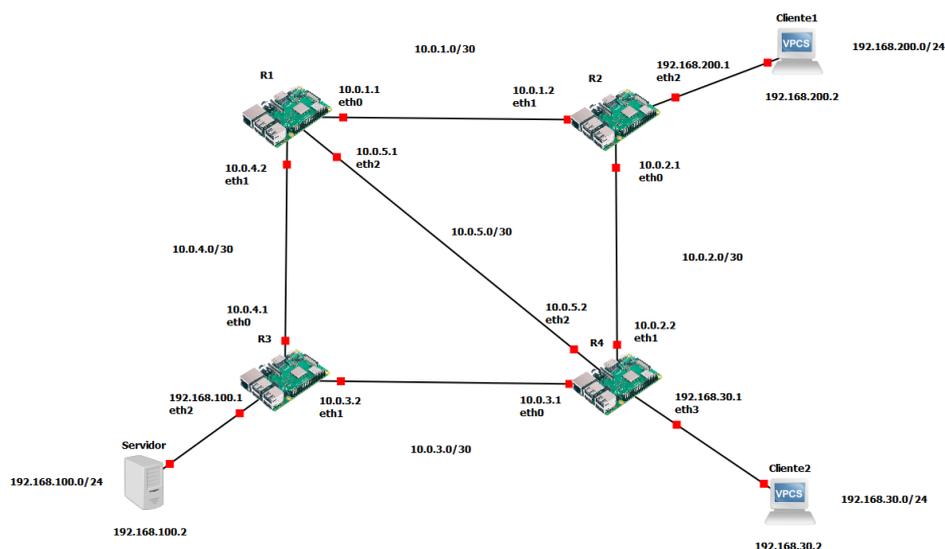
Perfil de tráfico de la categoría noticias en calidad media



En la siguiente subsección se expondrá la topología diseñada para la implementación de nuestro *testbed*.

Topología

La topología implementada se expone en la Figura 6.

Figura 6*Topología del testbed*

Como se puede observar en la Figura 6, la topología consta de cuatro Raspberry Pi que funcionan tanto como enrutadores como Open vSwitch. Se eligió esta topología debido a su redundancia y múltiples caminos. Se pueden apreciar todas las direcciones IP que se ha implementado al usar las Raspberry Pi como enrutadores. Al utilizar SDN, simplemente se configuran las direcciones IP de los clientes y del servidor, los cuales deben estar en la misma red. Luego de diseñar la topología a utilizar, se procedió con la implementación física de la misma, realizando todas las conexiones necesarias utilizando los adaptadores USB a RJ45 y cables Ethernet, como se ilustra en la Figura 7.

Figura 7*Implementación física del testbed*

Preparación del sistema de control de video adaptativo

El sistema de control de video adaptativo se lo realizó siguiendo la metodología expuesta en la Figura 1. Para su implementación se ha escrito un código en lenguaje Python, el mismo que se encuentra en Gomezcoello y Picón (2024).

El código implementado para el algoritmo de control de video adaptativo sigue una serie de pasos. Primero, se creó un diccionario para almacenar las rutas de los videos de las distintas categorías a transmitir; luego, se implementaron funciones para cargar y seleccionar estos videos. A continuación, se desarrolló una función para medir el ancho de banda utilizando la herramienta iPerf en Jon Dugan y Prabhu (2024). Basándose en los resultados de esta medición, se implementó una función que ajusta la tasa de codificación para cada categoría de video.

Posteriormente, se implementó la función de transmisión de video utilizando la herramienta FFmpeg en FFmpeg (2024). Si no se ha realizado la medición del ancho de banda, se transmite a una tasa de codificación de 70 Kbps. Una vez obtenido el valor del ancho de banda disponible, se ajusta la tasa de codificación según los umbrales establecidos.

Finalmente, se desarrolló la función de transmisión *demo*, que permite la transmisión a tres tasas de codificación diferentes: primero se transmite a la calidad más baja, luego a una calidad media y, finalmente, a una calidad alta. La última parte del código se encarga de la configuración para la implementación de la interfaz gráfica.

Resultados y Discusión

El estudio realizado comparó el rendimiento de la transmisión de video en redes tradicionales (RIP y OSPF) frente a una red definida por software (SDN), en escenarios con y sin tráfico de fondo.

Los resultados mostraron diferencias claras en términos de *throughput*, retardo y *jitter*. Dichas métricas son fundamentales para garantizar una transmisión de video eficiente y de alta calidad. El *throughput* asegura que se transmitan suficientes datos para evitar almacenamiento en búfer, mientras que el *jitter*, que mide la variabilidad en la llegada de paquetes, es crucial para evitar interrupciones en la reproducción, por último, un bajo *delay* es esencial para aplicaciones en tiempo real, ya que permite una sincronización adecuada entre audio y video. El análisis de estas métricas permite diseñar algoritmos que se adapten dinámicamente a las condiciones de la red, optimizando así la experiencia del usuario.

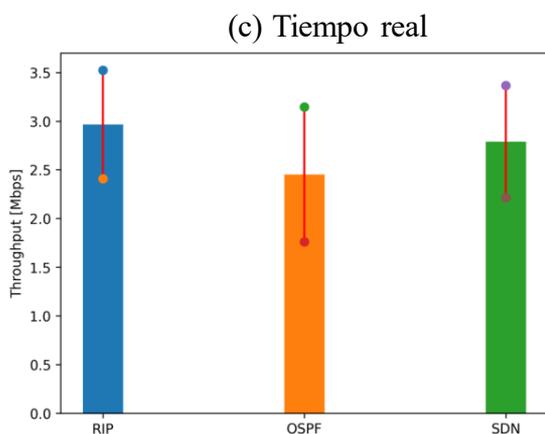
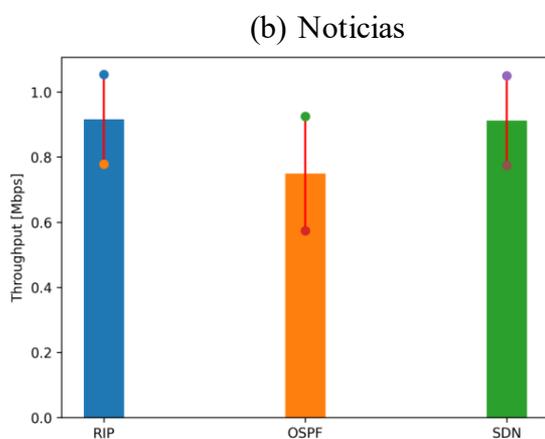
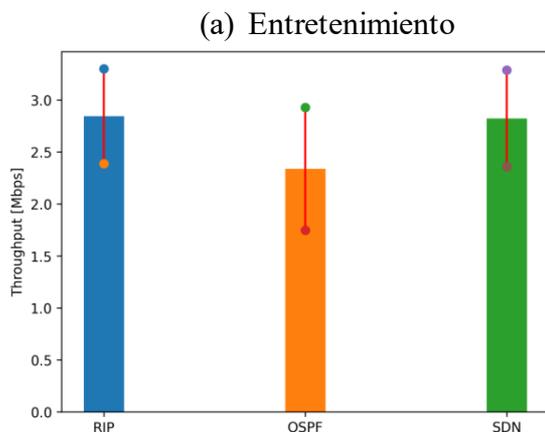
Resultados obtenidos con sistema de control sin tráfico de fondo

Throughput

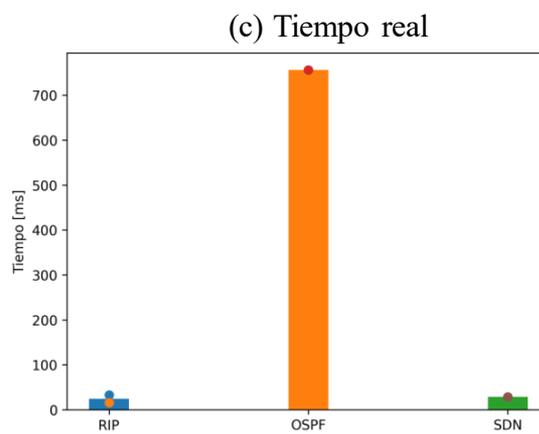
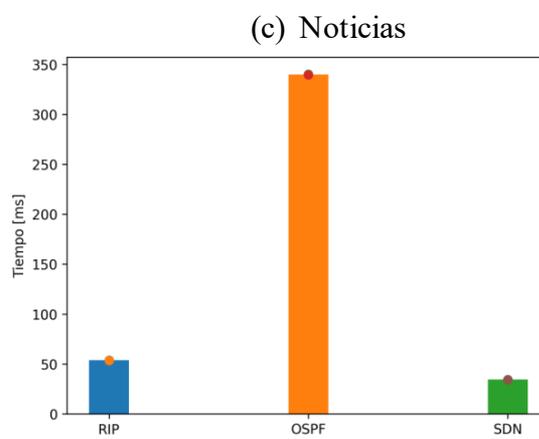
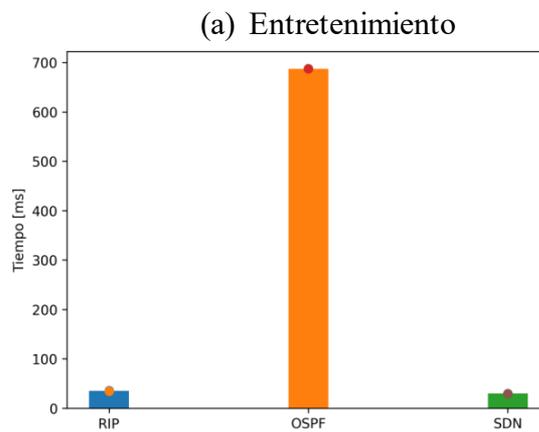
En la Figura 8(a) que corresponde a la categoría de entretenimiento se observa que, el peor rendimiento en *throughput* lo obtiene OSPF con velocidad promedio de 2.3 Mbps y un intervalo de confianza amplio lo que indica una inestabilidad en la tasa de transmisión; por otro lado, RIP y SDN tienen un *throughput* (2.7 Mbps) y un intervalo de confianza similares las cuales son más estables, lo cual indica que OSPF no selecciona la ruta con mayor ancho de banda, no así RIP y SDN. El mismo análisis anterior aplica para las Figura 8(b) y 9(c).

Figura 8

Throughput obtenido para las diferentes categorías de video

**Figura 9**

Retardo obtenido para las diferentes categorías



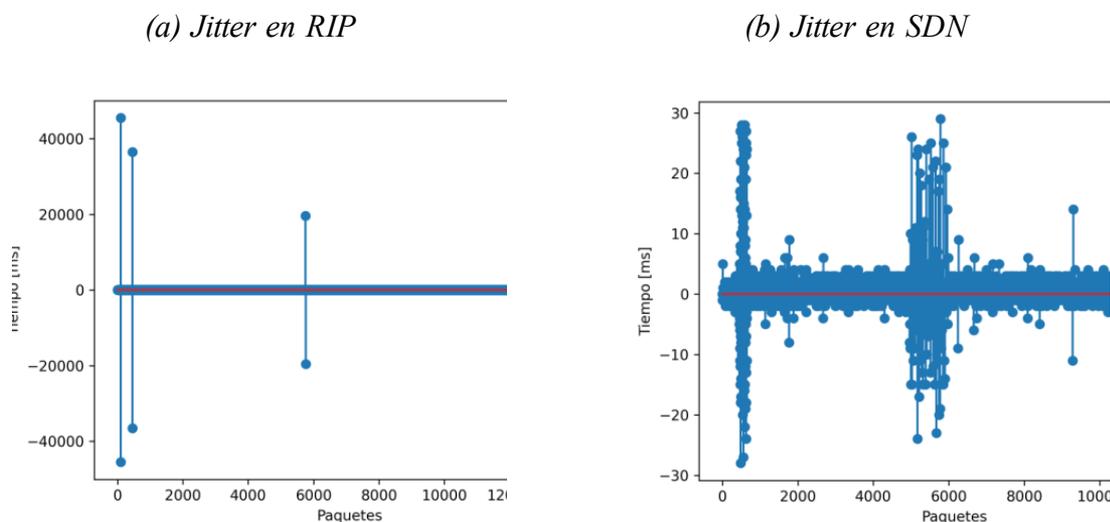
Retardo

Las Figura 9(a), (b) y (c) evidencian que SDN y RIP cuentan con el menor retardo en la transmisión de video, seguido por OSPF, dichos resultados indican que SDN y RIP seleccionaron la ruta con mayor ancho de banda para la transmisión de video en todas las categorías; sin embargo, en la categoría de tiempo real, RIP tuvo un retardo 0.4 ms menor, pero con gran inestabilidad debido a que enviaba todo el tráfico por una única ruta. SDN mostró retardo más estable, garantizando que la ruta principal no se sature, mientras que OSPF seleccionó una ruta sin importar el ancho de banda.

Jitter

En cuanto al *jitter*, SDN mostró una variabilidad más estable (± 30 ms) como se muestra en la Figura 10(b) mientras que RIP presentó picos significativos, lo que confirma que SDN es superior en términos de estabilidad del retardo. Dichas fluctuaciones se muestran en la Figura 10(a).

Figura 10



Resultados obtenidos con sistema de control con tráfico de fondo

En estos experimentos, el escenario incluyó tráfico de fondo mediante la descarga de un archivo de video de aproximadamente 200 MB.

Throughput

OSPF presentó el mayor *throughput* (hasta 2.5 Mbps), mientras que SDN tuvo el menor rendimiento debido al balance de carga aplicado al descargar el archivo simultáneamente, llegando máximo hasta 1.5 Mbps. Los resultados obtenidos se presentan en la Figura 11.

En la Figura 11(a) se observan tasas de transmisión promedio de 2.1 Mbps, 2.4 Mbps y 1.5 Mbps para los protocolos RIP, OSPF y SDN, respectivamente, con intervalos de confianza de amplitud similar, lo que indica una variabilidad comparable entre los tres protocolos. OSPF presentó el mayor *throughput*, mientras que SDN registró el menor rendimiento, debido principalmente al mecanismo de balanceo de carga aplicado durante la descarga simultánea del archivo.

Este mismo análisis es aplicable a las Figura 11(b) y 11(c), ya que los resultados obtenidos son consistentes con los de la Figura 11(a).

Retardo

En las Figura 12(a) y 12(b), SDN presentó los menores valores de retardo en las categorías de entretenimiento y noticias, con aproximadamente 20 ms, mientras que RIP y OSPF registraron retardos cercanos a 40 ms y 50 ms, respectivamente. En la Figura 12(c), los retardos observados fueron de 28 ms para RIP, 58 ms para OSPF y 37 ms para SDN. En este caso particular, se evidenció un cambio de ruta en RIP que permitió una reducción del retardo, mientras que SDN experimentó un leve incremento debido al volumen de datos que debió gestionar mediante balanceo de carga. Este mecanismo, si bien sacrificó ligeramente el *throughput*, logró mejorar el retardo; por otro lado, OSPF y RIP presentaron mayores pérdidas

en la calidad del video, atribuidas a la congestión del único enlace disponible en las categorías de entretenimiento y noticias.

Figura 11

Throughput obtenido para las diferentes categorías de video

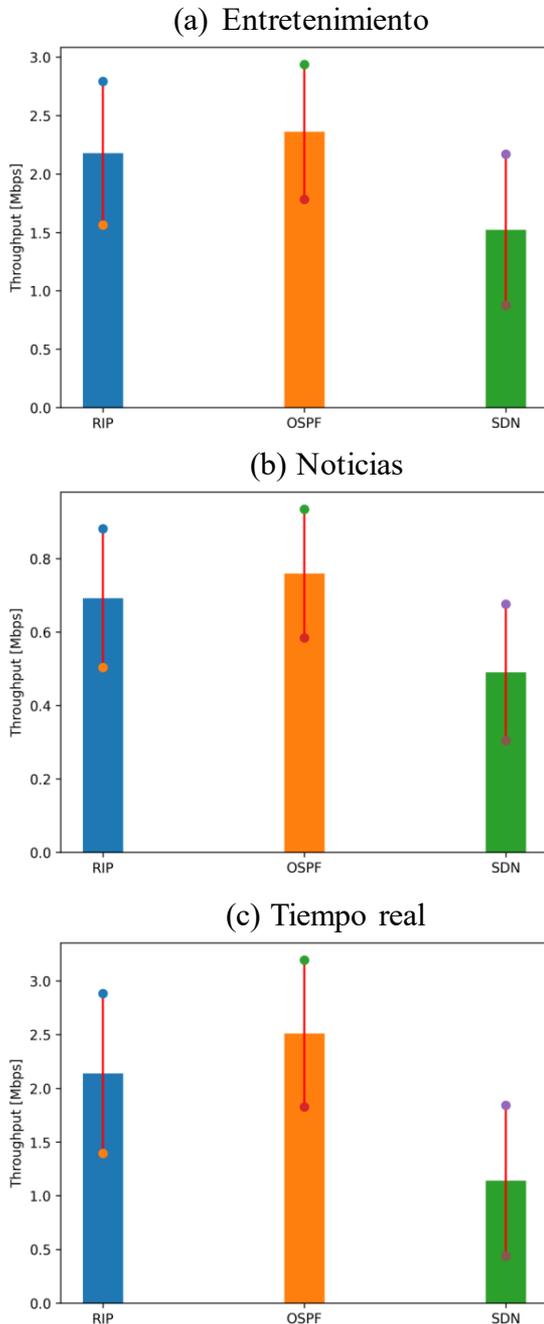
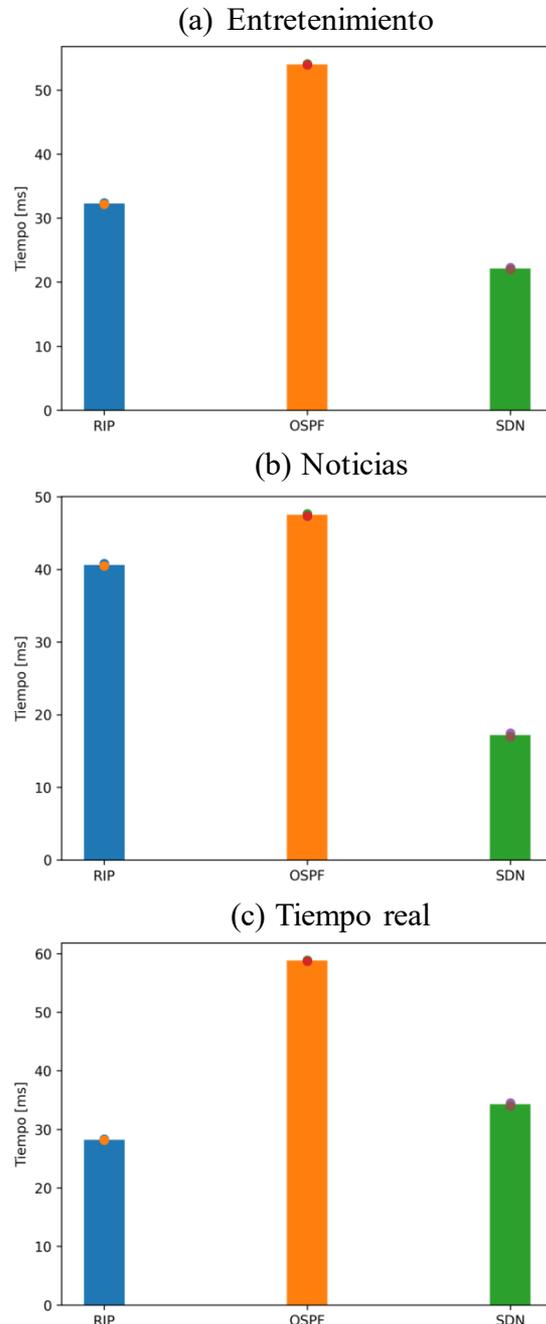


Figura 12

Retardo obtenido para las diferentes categorías de video



Calidad del video

Las Figuras 13(a), 13(b) y 13(c), correspondientes a las categorías de noticias, entretenimiento y transmisión en tiempo real, respectivamente, evidencian una disminución en la calidad del video ante movimientos en los protocolos OSPF y RIP, mientras que en SDN la calidad se mantuvo más alta y estable. Los fotogramas analizados muestran que SDN conserva una mejor calidad visual en comparación con OSPF y RIP, incluso en escenarios con tráfico

de fondo. Es importante destacar que estas conclusiones se obtuvieron a partir de un proceso de evaluación subjetiva.

Figura 13

Fotogramas al usar SDN con tráfico de fondo y sistema de control

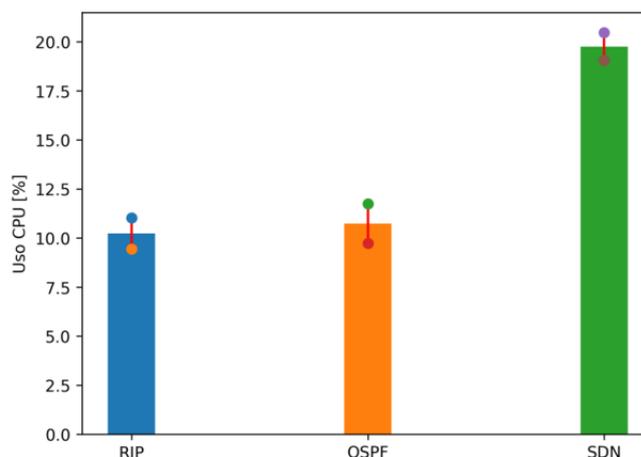
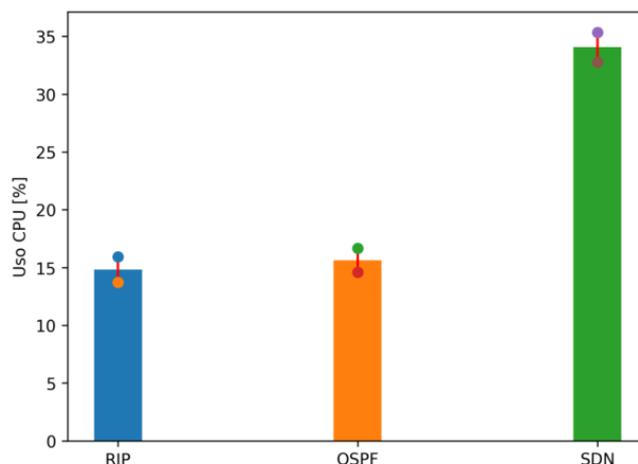


Resultados obtenidos de carga de CPU

En este experimento se midió la carga del CPU del Raspberry Pi utilizado como *gateway* del servidor de video.

Carga de CPU sin tráfico de fondo

Con el sistema de control activo, se observó que SDN requiere una mayor carga de procesamiento en comparación con los protocolos tradicionales. Las mediciones de uso de CPU fueron de 10% para RIP, 11% para OSPF y 20% para SDN. Este incremento en el uso de CPU por parte de SDN se debe al mecanismo de balanceo de carga, mediante el cual los datos se distribuyen simultáneamente a través de múltiples rutas, a diferencia de RIP y OSPF, que establecen una única ruta fija para la transmisión de datos; no obstante, este aumento en la carga de procesamiento en SDN no resulta significativo y no genera degradación perceptible en los niveles de Calidad de Servicio (QoS) ni en la Calidad de Experiencia (QoE). Estos resultados se ilustran en la Figura 14.

Figura 14*Comparación de carga de CPU***Figura 15***Carga del CPU****Carga de CPU con tráfico de fondo***

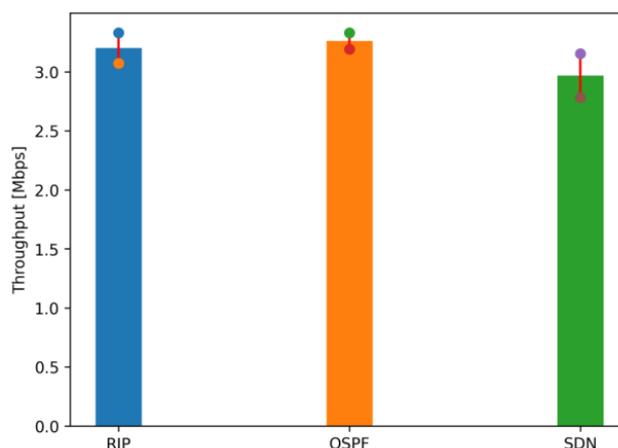
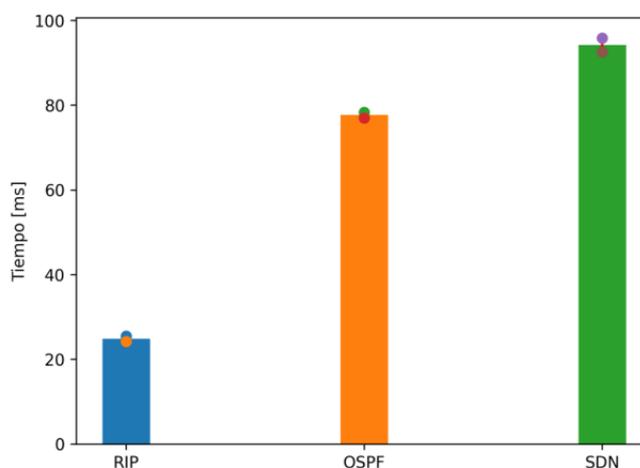
El tráfico de fondo aumenta aún más la carga del CPU, especialmente en el modo Open vSwitch utilizado por SDN, lo que confirma que SDN es más exigente en términos de procesamiento. Esto nos llevó a obtener los resultados de la Figura 15.

Resultados obtenidos sin sistema de control y sin tráfico de fondo***Throughput***

Los tres protocolos (SDN, RIP y OSPF) presentaron valores promedio de *throughput* de hasta 3 Mbps.

Retardo

RIP mostró el mejor rendimiento en términos de retardo, seguido por OSPF y SDN. Sin tráfico de fondo, los protocolos tradicionales son más eficientes, ya que envían todo el tráfico por un solo enlace, mientras que SDN presenta mayor retardo debido al balanceo de carga. Las métricas analizadas en este caso se presentan en las Figuras 16 y 17.

Figura 16*Throughput de la categoría noticias sin tráfico y sin control***Figura 17***Retardo de la categoría noticias sin tráfico y sin control*

Presentaremos un consolidado de los resultados obtenidos donde compararemos los aspectos más relevantes de nuestro estudio en la Tabla 3.

Tabla 3*Comparación de resultados*

Categoría	Métrica	Con tráfico	Sin tráfico
Entretenimiento	Throughput (Mbps)	1.5 (SDN) < 2.2 (RIP) < 2.4 (OSPF)	2.4 (OSPF) < 2.7 (SDN) < 2.71 (RIP)
	Delay (ms)	21 (SDN) < 31 (RIP) < 55 (OSPF)	40 (SDN) < 50 (RIP) < 680 (OSPF)
	Jitter (ms)	±32 (SDN) vs. ± 40000 (RIP y OSPF)	± 30 (SDN) vs. ± 40000 (RIP y OSPF)
	Carga CPU (%)	15 (RIP) < 16 (OSPF) < 34 (SDN)	10 (RIP) < 11 (OSPF) < 20 (SDN)
Noticias	Throughput (Mbps)	0.5 (SDN) < 0.7 (RIP) < 0.75 (OSPF)	0.75 (OSPF) < 0.9 (SDN) < 0.91 (RIP)
	Delay (ms)	18 (SDN) < 40 (RIP) < 47 (OSPF)	40 (SDN) < 51 (RIP) < 340 (OSPF)
	Jitter (ms)	±31 (SDN) vs. ± 40000 (RIP y OSPF)	± 35 (SDN) vs. ± 40000 (RIP y OSPF)
	Carga CPU (%)	14 (RIP) < 15 (OSPF) < 30 (SDN)	9 (RIP) < 10 (OSPF) < 22 (SDN)

Categoría	Métrica	Con tráfico	Sin tráfico
Tiempo real	Throughput (Mbps)	1.2 (SDN) < 2.1 (RIP) < 2.5 (OSPF)	2.47 (OSPF) < 2.75 (SDN) < 2.9 (RIP)
	Delay (ms)	28 (RIP) < 25 (SDN) < 58 (OSPF)	40 (RIP) < 41 (SDN) < 750 (OSPF)
	Jitter (ms)	±29 (SDN) vs. ±40000 (RIP y OSPF)	± 33(SDN) vs. ± 40000 (RIP y OSPF)
	Carga CPU (%)	15 (RIP) < 18 (OSPF) < 40 (SDN)	11 (RIP) < 12 (OSPF) < 19 (SDN)

Como se puede analizar en la Tabla 3, con tráfico de fondo, SDN muestra bajo ancho de banda efectivo; sin embargo, mantiene fluida la reproducción de los mismos gracias al mecanismo implementado. Siendo este un protocolo eficaz para el tipo de tráfico estudiado.

Conclusiones

Al emplear este sistema de control sin tráfico de fondo, se observó que SDN presenta menores valores de retardo. Las categorías de video en las que más destacó fueron noticias y entretenimiento. Promediando los valores obtenidos de las tres categorías, SDN obtuvo un retardo de 30.89 ms, de igual manera, el protocolo RIP presentó un retardo de 37.72 ms, finalmente OSPF obtuvo un valor de 594.57 ms. Lo que significa que SDN tiene un retardo 18.1 % menor que RIP y 94.80 % menor que OSPF, de esta manera, se concluye que SDN tiene un mejor rendimiento al emplear nuestro sistema de control, con menores retardos, lo cual es muy importante debido al alto consumo multimedia actual, especialmente en lo que respecta al contenido en *streaming*.

En cuanto al experimento con sistema de control y tráfico de fondo, los valores de *throughput* fueron ligeramente inferiores a los de otros protocolos; sin embargo, promediando los valores de retardo de las tres categorías SDN obtuvo un retardo de 24.53 ms, RIP de 33.71 ms y OSPF de 53.43 ms, lo que significa que SDN tiene un *delay* 27.23 % menor que RIP y 54.1 % menor que OSPF. SDN emplea balance de carga, utilizando varias rutas para enviar los datos. Esto reduce el *throughput*, pero también disminuye el retardo, por tanto, SDN es una opción recomendada para redes en las que se puede tolerar una disminución de *throughput* pero no un aumento en el retardo, como en servicios de tiempo real, tales como videoconferencias o intervenciones quirúrgicas remotas.

El uso de SDN conlleva un costo relacionado con el procesamiento necesario en cada nodo para llenar sus tablas de flujo. Presentó un valor promedio más alto que los otros dos protocolos, lo que se traduce en un mayor consumo energético; por lo tanto, implementar SDN en un centro de datos implica un aumento en los gastos de climatización y energía eléctrica.

Cuando no está presente el sistema de control y no hay tráfico de fondo, el *throughput* fue similar. En términos de retardo, los protocolos tradicionales mostraron un mejor rendimiento, ya que, al no existir congestión en el enlace, el retardo es menor, el valor de retardo obtenido para la categoría de noticias en SDN fue de 94.21 ms, para RIP de 24.85 ms y para OSPF de 77.65 ms, lo que se traduce en que RIP tiene un retardo 73.62 % menor que SDN y OSPF un 17.57 % menor que SDN, cabe destacar que dichos valores fueron similares para el resto de las categorías.

A pesar de utilizar un *testbed*, se obtuvieron resultados significativos que demuestran que SDN en conjunto con un sistema de control es considerablemente superior a los protocolos tradicionales; por lo tanto, su implementación en redes complejas y congestionadas en una opción a considerar.

En la actualidad, implementar SDN es fundamental para mejorar la calidad de servicio y la calidad de experiencia en sistemas de transmisión multimedia, ya que permite una gestión centralizada e inteligente de la red, capaz de adaptarse dinámicamente a las condiciones cambiantes del tráfico. Dado que el video representa más del 80% del tráfico global en Internet, y exige bajos niveles de retardo, *jitter* y pérdida de paquetes, las redes tradicionales con protocolos como RIP u OSPF resultan insuficientes; en cambio, SDN posibilita priorizar flujos críticos, asignar ancho de banda de forma eficiente y enrutar el contenido por trayectorias óptimas, lo que se traduce directamente en una experiencia de usuario más fluida y estable frente al creciente consumo de contenidos como *streaming*, videollamadas, telemedicina y plataformas interactivas.

Reconocimientos

Los autores agradecen al Departamento de Eléctrica, Electrónica y Telecomunicaciones de la Facultad de Ingeniería de la Universidad de Cuenca por el apoyo brindado durante el desarrollo del presente trabajo de investigación.

Los autores declaran la contribución y participación equitativa de roles de autoría para esta publicación.

Referencias

- Abar, T., Ben Letaifa, A., & El Asmi, S. (2021). Chapter five - user behavior-ensemble learning based improving qoe fairness in http adaptive streaming over sdn approach. In A. R. Hurson (Ed.), (Vol. 123, pp. 245-269). Elsevier. Recuperado a partir de <https://www.sciencedirect.com/science/article/pii/S0065245821000292>
- Abuteir, R. M., Fladenmuller, A., & Fourmaux, O. (2016). An sdn approach to adaptive video streaming in wireless home networks. 2016 International Wireless Communications and Mobile Computing Conference (IWCMC), 321-326. Recuperado a partir de <https://api.semanticscholar.org/CorpusID:12044836>
- Banchuen, T., Kawila, K., & Rojviboonchai, K. (2018). An sdn framework for video conference in inter-domain network. 2018 20th International Conference on Advanced Communication Technology (ICACT), 600-605. Recuperado a partir de <https://api.semanticscholar.org/CorpusID:4370122>
- Bona venture, O., Hartert, R., Lebrun, D., Schaus, P., Tilmans, O., & Vissicchio, S. (2017). Robust and scalable software defined networks. Recuperado a partir de <https://api.semanticscholar.org/CorpusID:41750697>
- Dac, C. T., Tran, H. T. T., Huong, T. T., Tran, S., Thanh, N. H., Ngoc Nam, P., & Thang, T. C. (2021). Qoe-aware video adaptive streaming over http. In 2020 IEEE Eighth International Conference on Communications and Electronics (ICCE) (pp. 117-122). <https://doi.org/10.1109/ICCE48956.2021.9352077>
- FFmpeg. (2024). A complete, cross-platform solution to record, convert and stream audio and video. Recuperado a partir de <https://www.ffmpeg.org/>
- Gomezcoello, M., & Picón, J. (2024). Desarrollo e implementación de un algoritmo de control de video adaptativo en entornos de redes definidas por software (SDN). Recuperado a partir de <https://github.com/marlonpikis/trabajo-titulacion>
- Jiawei, W., Xiuquan, Q., & Guoshun, N. (2018). Dynamic and adaptive multi-path routing algorithm based on software-defined network. International Journal of Distributed Sensor Networks, 14(10), 1550147718805689. <https://doi.org/10.1177/1550147718805689>
- Jon Dugan, B. A. M. J. P., Seth Elliott, & Prabhu, K. (2024). iperf - the ultimate speed test tool for tcp, udp and setp. Recuperado a partir de <https://iperf.fr/>

- Khan, K. (2023). User-centric algorithms: Sculpting the future of adaptive video streaming. *International Transactions on Electrical Engineering and Computer Science*. Recuperado a partir de <https://api.semanticscholar.org/CorpusID:268910394>
- Li, W., Huang, J., Wang, S., Wu, C., Liu, S., & Wang, J. (2023). An apprenticeship learning approach for adaptive video streaming based on chunk quality and user preference. *IEEE Transactions on Multimedia*, 25, 2488-2502. <https://doi.org/10.1109/TMM.2022.3147667>
- Lin, Z. (2015). *Research on video traffic control technology based on sdn*. Recuperado a partir de <https://api.semanticscholar.org/CorpusID:55442197>
- Makolkina, M., Muthanna, A., & Manariyo, S. (2017). *Quality of experience estimation for video service delivery based on sdn core network*. In *Next Generation Teletraffic and Wired/Wireless Advanced Networking*. Recuperado a partir de <https://api.semanticscholar.org/CorpusID:43433926>
- Sam, G. (2023). *In 2022, 65% of all internet traffic came from video sites*. Recuperado a partir de <https://www.tubefilter.com/2023/01/20/sandvine-video-data-bandwidth-internet-traffic-report-streaming-video-youtube-netflix/>
- Sayit, M., Cetinkaya, C., Yildiz, H. U., & Tavli, B. (2019). *Dash-qos: A scalable network layer service differentiation architecture for dash over sdn*. *Computer Networks*, 154, 12-25. <https://doi.org/10.1016/j.comnet.2019.02.015>
- Sheu, J.-P., Chang, C., & Chang, Y.-C. (2015). *Efficient multicast algorithms for scalable video coding in software-defined networking*. 2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), 2089-2093. Recuperado a partir de <https://api.semanticscholar.org/CorpusID:12455700>
- Storck, C. R., & Duarte-Figueiredo, F. (2020). *A performance analysis of adaptive streaming algorithms in 5g vehicular communications in urban scenarios*. In *2020 IEEE Symposium on Computers and Communications (ISCC)* (pp. 1-7). <https://doi.org/10.1109/ISCC50000.2020.9219682>
- Team, D. S. (2020). *Regla empírica*. Recuperado a partir de <https://datascience.eu/es/matematica-y-estadistica/regla-empirica-que-es/>
- Witte, R., & Witte, J. (2017). *Statistics*. Wiley. Recuperado a partir de <https://books.google.com.ec/books?id=KcxjDwAAQBAJ>
- Yamansavascular, B., Baktir, A. C., OZgovde, A., & Ersoy, C. (2020). *Fault tolerance in sdn data plane considering network and application based metrics*. *Journal of Network and Computer Applications*, 170, 102780. <https://doi.org/10.1016/j.jnca.2020.102780>