FACULTAD DE INGENIERÍA

Doctorado en Ingeniería

Contribución a la reducción del retardo de encolamiento en un dispositivo intermedio en una conexión punto a punto mediante análisis de patrones de tráfico

Tesis doctoral

Autor:

Albert Giovanny Espinal Santana

Dirigido por:

PhD. Carlos Monsalve Arteaga

PhD. Rebeca Estrada Pico

MENDOZA – ARGENTINA

2021

AGRADECIMIENTO

En primer lugar, a Dios, por brindarme salud, iluminarme y guiarme en este camino de convertirme en investigador, a pesar de las adversidades, pero con su ayuda pude superar cada obstáculo que se presentaba. A mi universidad, la Escuela Superior Politécnica del Litoral, por la oportunidad que me brindó para ser parte del grupo de profesores seleccionados del convenio UNCUYO-ESPOL, y por la ayuda económica recibida para mis estudios. A la Universidad Nacional de Cuyo, a su programa de doctorado en Ingeniería, por el aporte científico de su comité y docentes, que implantaron esa semilla del conocimiento. A la Universidad de Granada, a su departamento DECSAI, quienes nos dieron acogieron de buena forma para poder cumplir con nuestros cursos de formación doctoral.

A mis tutores, el PhD. Carlos Monsalve, y la PhD. Rebeca Estrada, quienes me brindaron su guía en este proceso. Por su tiempo empleado en reuniones, y sobre todo en las revisiones de las publicaciones, y también de esta tesis.

A los investigadores que me brindaron su guía y experiencia, para darle forma a una idea, plasmada en este documento.

DEDICATORIA

A los que ya no están, pero que siempre estuvieron orgullosos de mi carrera profesional y académica: mi padre Diómedes, y Rosita, mi segunda madre.

A las personas que me inspiran día a día para ser una mejor persona, un mejor esposo, un mejor padre: mi esposa Cindy, y mi hijo Sebastián. A ustedes que han estado cerca en todos estos años, quienes han sacrificado el tiempo que merecen, para que yo pueda alcanzar las metas que me he trazado.

A mi madre, Zoila, a mis hermanos: Joffre, Ramiro e Ileana, quienes a lo largo de la vida siempre me han alentado a seguir adelante.

A mis amigo.

"Start by doing what's necessary; then do what's possible; and suddenly you're doing the impossible."

Francis of Assisi

Resumen

El presente trabajo tiene como objetivo proponer un modelo predictivo de Calidad de Servicio (QoS), que contribuya a reducir el retardo de encolamiento con respecto a políticas existentes como el Encolamiento Justo Ponderado basado en Clases (CBWFQ), mediante el análisis de patrones de tráfico en dispositivos intermedios en una conexión punto a punto.

Para su realización, se plantearon 3 etapas de investigación. La primera etapa consistió en determinar las longitudes de paquetes más comunes que utilizan protocolos y aplicaciones actuales, tanto en ambientes alámbricos e inalámbricos, tomando como referencia una red de Campus Universitario, y modelar dicho tráfico mediante distribuciones de Poisson. La segunda fase consistió en analizar los componentes de el retardo de extremo a extremo de una conexión punto a punto, haciendo énfasis en el retardo de encolamiento, y modelar su comportamiento y predictibilidad mediante regresiones polinómicas. En la tercera etapa se presenta un modelo matemático de calidad de servicio basado en los patrones encontrados, y que permite simular mediante MATLAB el retardo de encolamiento, para analizar sí es posible obtener mejores tiempos de retardo en una comunicación de extremo a extremo. Los resultados numéricos obtenidos muestran que el modelo de QoS propuesto ofrece menores valores de retardo. Los resultados de las tres fases de investigación han sido publicados a la comunidad científica.

ABSTRACT

The purpose of this work is to propose a Quality of Service (QoS) predictive model, which helps to reduce the queuing delay with respect to existing policies such as the Class Based Weighted Fair Queuing (CBWFQ), by analyzing traffic patterns in intermediate devices in a point-to-point connection.

For its realization, 3 research phases were proposed. The first stage was to determine the most common package sizes that use current protocols and applications, both in wired and wireless environments, taking as a reference a University Campus network, and modeling the traffic through distributions of Poisson. The second phase consisted of analyzing the components of the end-to-end delay of a connection, emphasizing in the queuing delay, and modeling its predictive behavior by polynomial regressions. In the third stage we present a QoS mathematical model based on the studied patterns, and that allows to simulate the delay by means of MATLAB, to analyze if it is possible to obtain better times in an end-to-end network communication. The numerical results obtained shown that the proposed QoS model has lower values for queuing delay. The results obtained in the different stages have been published to the scientific community.

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

LISTA DE ABREVIATURAS

QoS	Calidad de Servicio
WFQ	Encolamiento justo ponderado
FIFO	Primero en llegar, Primero en salir
OWD	Retardo de una vía
CBWFQ	Encolamiento justo ponderado basado en clases
LLQ	Encolamiento de baja retardo
TCP/IP	Protocolo de Control de Transmisión / Protocolo de Internet
LTE	Red celular de evolución a largo plazo
IDS	Sistema de detección de intrusos
IPS	Sistema de prevención de intrusos
TCP	Protocolo de Control de Transmisión
UDP	Protocolo de Datagrama del Usuario
RFC	Solicitud de comentarios, documento estándar de IETF
IEFT	Fuerzas de Tareas de Ingeniería para Internet
IPv4	Protocolo de Internet versión 4
IPv6	Protocolo de Internet versión 6
ISOC	Sociedad de Internet
SSH	Shell seguro
SMTP	Protocolo de transporte de correo electrónico
HTTP	Protocolo de transferencia de hipertexto
HTTPS	Protocolo seguro de transferencia de hipertexto
SSL	Capa de conexión segura
TLS	Protocolo de Transporte seguro
DNS	Servicio de nombres de dominio
MPLS	Protocolo de conmutación de etiquetas
DHCP	Protocolo de configuración dinámica de host
NETBIOS	Sistema básico de entrada y salida de red
QUIC	Protocolo de conexión rápida de internet sobre UDP
SSDP	Protocolo de descubrimiento de servicios

MDNS	Servicio de nombres de dominio sobre Multicast
NIC	Tarjeta de interfaz de red
ICMP	Protocolo de control y mensajería de IP
RTT	Tiempo de ida y vuelta de un paquete
IPPM	Medición de rendimiento del protocolo IP
LAN	Red de área local
WAN	Red de área amplia
NTP	Protocolo de Sincronización en Red
GPS	Sistema de Posicionamiento Global
PTP	Protocolo de precisión de tiempo (IEEE 1588)
RF	Radio Frecuencia
CPU	Unidad central de procesamiento
NAT	Traducción de direcciones de red
BestEffort	Modelo QoS de Máximo Esfuerzo
IntServ	Modelo QoS de Servicios Integrados
DiffServ	Modelo QoS de Servicios Diferenciales
NBAR	Reconocimiento de aplicaciones basado en Red
ToS	Tipo de Servicio
DSCP	Punto de código de servicios diferenciados
ECN	Notificación de congestión extendida
FTP	Protocolo de Transferencia de Archivos
MAC	Dirección física de una interfaz de red
PQ	Prioridad estricta de cola
WRED	Detección temprana aleatoria ponderada
MAC	Control de Acceso al Medio
CSMA/CA	Sensor de portadora en un medio de múltiple acceso que evita colisiones
RIPE	Centro de coordinación de redes europeas
LTE-A	LTE Avanzado
PHY	Capa física
SDN	Redes definidas por software

NGN	Redes de próxima generación
QPSS	Esquema de planificación de paquetes orientado a QoS
CDMA	Red celular de acceso múltiple por división de código heterogéneo
VoIP	Voz sobre protocolo de internet
WiFi	Tecnología de comunicación inalámbrica para LAN (IEEE 802.11)
WiMAX	Tecnología de comunicación inalámbrica para WAN (IEEE 802.16)
MOS	Puntuación de opinión media
VLAN	Red de área local virtual
WLC	Controlador de red inalámbrica
MIMO	Múltiple entrada múltiple salida (IEEE 802.11ac)
MU-MIMO	MIMO Multi usuario
PCAP	Formato de archivos de captura de paquetes
SPAM	Analizador de puerto de Switch
IS-IS	Sistema intermediario a sistema intermediario
IGP	Protocolo de enrutamiento de interior
PPS	Paquetes por segundo
WRR	Round Robin por peso / prioridad
LPQ	Encolamiento por longitud de paquete

LISTA DE SÍMBOLOS

Nd	Retardo unidireccional de origen a destino
n	Número de saltos en la ruta
Sdi	Retardo de serialización por cada salto
Pdi	Retardo de propagación por cada salto
Fdi	Retardo de reenvío o retardo de procesamiento por cada salto
Qdi	Retardo de encolamiento por cada salto
Ti	Retardo en los enlaces
Qi	retardo en las colas
Sd	Retardo de Transmisión o Serialización
Pd	Retardo por Propagación
Qd	Retardo de encolamiento
Lq	Longitud de la cola
Ra	Unidades de paquetes
L	Longitud de un paquete
Tr	Velocidad de transmisión
N	Número de paquetes simultáneos
μ	Tasa de servicio
λ	Tasa de llegada
$\lambda 1$	Ocurrencia promedio en el intervalo 1 de distribución de Poisson
$\lambda 2$	Ocurrencia promedio en el intervalo 2 de distribución de Poisson
P1	Probabilidad que un paquete siga la primera distribución
P2	Probabilidad que un paquete siga la segunda distribución
L_{PPG}	Longitud promedio de un paquete de longitud grande
L_{PPM}	Longitud promedio de un paquete de longitud mediana
L_{PPP}	Longitud promedio de un paquete de longitud pequeña
P_{rPG}	Probabilidad de arribo de un paquete de longitud grande
P_{rPM}	Probabilidad de arribo de un paquete de longitud mediana
P_{rPP}	Probabilidad de arribo de un paquete de longitud pequeña

REMPG	Retardo de encolamiento medio de un paquete de longitud grande
REMPG	Retardo de encolamiento medio de un paquete de longitud mediana
REMPP	Retardo de encolamiento medio de un paquete de longitud pequeña
ABET	Ancho de banda total del enlace
TEPG	Tasa estimada de paquetes grandes que ingresan al sistema de colas
TEPM	Tasa estimada de paquetes medianos que ingresan al sistema de colas
TEPP	Tasa estimada de paquetes pequeños que ingresan al sistema de colas
TAE	Tasa de arribos esperada total al sistema de colas
LCPG	Longitud de la cola de paquetes grandes
LCPM	Longitud de la cola de paquetes medianos
LCPP	Longitud de la cola de paquetes pequeños
RSPG	Retardo de servicio para un paquete grande
RSPM	Retardo de servicio para un paquete mediano
RSPP	Retardo de servicio para un paquete pequeño
RS	Retardo de servicio promedio
UcPG	Utilización de la cola de paquetes grandes
UcPM	Utilización de la cola de paquetes medianos
UcPP	Utilización de la cola de paquetes pequeños
RRPG	Retardo de residencia/encolamiento para un paquete de longitud grande
RRPM	Retardo de residencia para un paquete de longitud mediana
RRPP	Retardo de residencia para un paquete de longitud pequeña
RR	Retardo de residencia promedio

CAPÍTULO I: INTRODUCCIÓN

1.1. Antecedentes

La convergencia de servicios y aplicaciones en las redes modernas ha generado cambios en los requerimientos de la infraestructura disponible. Por ejemplo, se diseñan e implementan mecanismos que permitan mejorar la eficiencia en cuanto al tránsito de paquetes, reduciendo tiempo de procesamiento, tiempos de consultas, análisis de paquetes y esquemas de conmutación de estos.

Entre estos requerimientos se encuentran los mecanismos de calidad de servicio, cuya funcionalidad es priorizar el despacho de paquetes en los dispositivos intermedios de conectividad, como por ejemplo los ruteadores [1]. Estos poseen dos tipos de colas para el despacho de paquetes: una física y una lógica. La primera está ligada a la interfaz física y despacha los paquetes de forma simple: el primero que llega es el primero que sale (esquema FIFO); el segundo tipo de cola trata de clasificar al tráfico por protocolos o requerimientos (marcación), y de esta forma priorizar el despacho de paquetes hacia la cola física, con la finalidad de evitar su agotamiento o incrementos en el retardo de encolamiento [2].

El hecho de aplicar marcación y clasificación, en los mecanismos tradicionales de calidad de servicio previo al despacho de paquetes en un dispositivo intermedio, genera retardo, el misma que puede ser reducido o compensado si se aplican mecanismos alternativos [3], en las fases de marcación y clasificación o en su defecto, reemplazarlos por otro tipo de comportamiento como el análisis predictivo, basado en los patrones reales de tráfico.

La convergencia del tráfico de redes de comunicaciones, datos, voz y video, en una misma infraestructura de red, introdujo nuevos retos, como, por ejemplo, el tratar de brindar o garantizar recursos a las comunicaciones establecidas, tal como ancho de banda, minimizar incidencia en el rendimiento tal como el retardo, fluctuación del retardo y

paquetes perdidos [4], esquema conocido como mecanismos o técnicas de calidad de servicio (Ver figura 1.1).

En las comunicaciones convergentes, deben garantizarse ciertos requerimientos, como el retardo de extremo a extremo, conocido también como retardo unidireccional (OWD), para evitar un comportamiento inesperado de las comunicaciones. Por ejemplo, en una llamada de Voz sobre IP, el retardo unidireccional debe estar por debajo de 150 milisegundos [5], de lo contrario escucharíamos una comunicación entrecortada, o con problemas de comprensión, debido a la pérdida de paquetes, o a la fluctuación del retardo de estos.

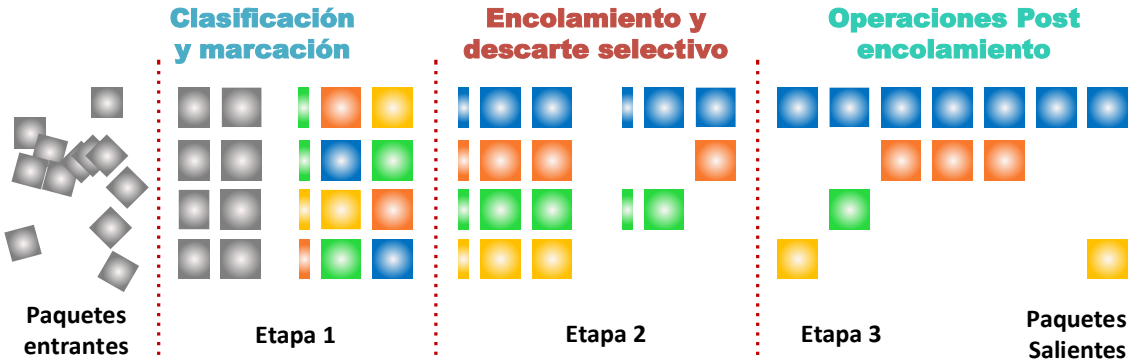


Figura 1.1. Operación de la Calidad de Servicio [Fuente: Cisco Systems]

En los dispositivos intermedios, como los ruteadores, se tiene una única cola de hardware disponible (por interfaz) para el despacho de los paquetes, la que debe optimizarse previamente por medio de la creación de colas de paquetes basados en software, que se encargan de clasificar a los paquetes que previamente han sido marcados, para poder diferenciar sus prioridades o requerimientos [6].

Los mecanismos de marcación y clasificación de paquetes en colas, como por ejemplo CBWFQ (encolamiento justo ponderado basado en clases) o LLQ (encolamiento de bajo retardo) han sido diseñados e implementados pensando en reducir el retardo en los enlaces de comunicación [7] [8]. Sin embargo, el comportamiento no predictivo del tráfico de red sigue generando retardo en los dispositivos. Las aplicaciones de hoy, especialmente de misión crítica, requieren de mejores tiempos de respuesta para trabajar de forma aceptable.

Actualmente, una forma de solucionarlo es incrementando el ancho de banda de los enlaces, lo que a veces puede involucrar incluso un cambio de medio de transmisión, dejando de lado alguna estrategia que permita compensar el retardo y la pérdida de paquetes, y que permitan mejorar el comportamiento de la calidad de servicio. Adicionalmente, esto genera costos por incremento del ancho de banda del canal de comunicación.

Por este motivo, se buscan nuevos mecanismos o estrategias con el fin de analizar la manera de reducir el retardo en el tránsito de paquetes [9], que, como puede observarse en la figura 1.2, puede deberse a:

1. Retardo de procesamiento: define el tiempo que el dispositivo intermedio toma en analizar y procesar la información de los paquetes para tomar decisiones de reenvío de estos, hacia las interfaces de salida.
2. Retardo de encolamiento: es el tiempo que toma el análisis y despacho de un paquete en las colas asociadas a la interfaz de salida. Depende de las políticas de despacho de las colas, basadas en las técnicas de calidad de servicio.
3. Retardo de propagación: se debe a la transmisión de los paquetes (codificación de bits a señales) y propagación de las señales, dependientes del tipo de medio de comunicación que se utilice, como una fibra óptica, un medio de cobre o un radioenlace.

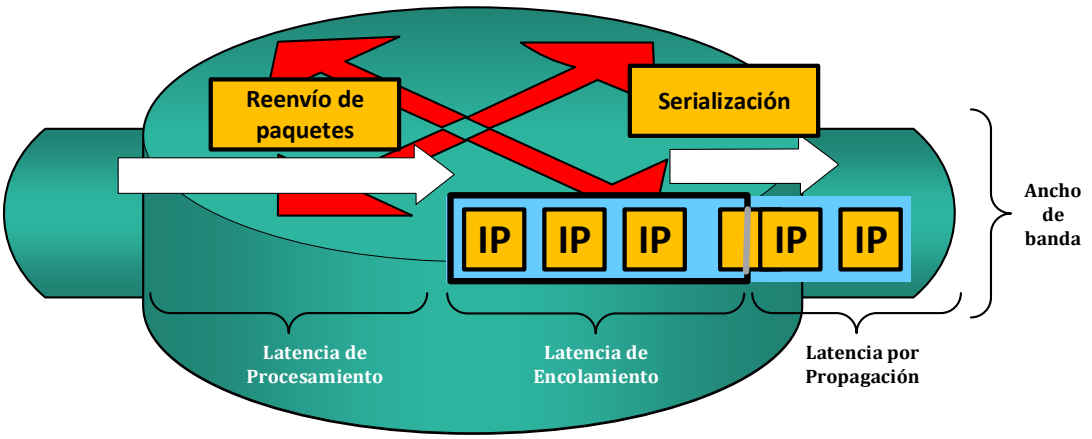


Figura 1.2. Tipos de retardo [Fuente: Cisco Systems]

De estos tipos de retardo, lo que deseamos contribuir es en la reducción del retardo debido a los procesos de encolamiento, para lo cual plantearemos la utilización de mecanismos de predicción, basados en la longitud de los paquetes, que permitan minimizar el tiempo que los mismos pueden pasar en una cola de despacho basada en software.

1.2. Descripción del problema

El proyecto de investigación tiene como objetivo presentar un novel modelo de calidad de servicio que permita reducir el retardo de encolamiento en un dispositivo intermedio, basado en el análisis y estimación de patrones reales de tráfico en una comunicación TCP/IP de un entorno de red convergente de campus.

El estudio demostrará la factibilidad de disminuir el retardo asociado al tránsito de paquetes IP en dispositivos intermedios, con capacidad de enrutamiento, donde actualmente los paquetes son encolados, marcados y clasificados, y finalmente despachados en base a una política que prioriza el factor ancho de banda como parámetro de rendimiento.

En lugar de una fase de marcación y clasificación previo al despacho de paquetes, se utilizarán métodos de análisis de patrones (predictivos) en función de las longitudes de paquetes, que contribuyan a disminuir el retardo agregado en un dispositivo intermedio. Este modelo matemático se lo simulará con la finalidad de medir o determinar tiempos de retardo en un escenario lo más cercanos a la realidad, para obtener muestras confiables y compararlas con estimaciones de retardo asociados a otros métodos de QoS como CBWFQ (encolamiento justo ponderado), y comprobar que la solución propuesta es más eficiente en términos de retardo.

1.3. Objetivo general del estudio

Formular un novel modelo de calidad de servicio que permita reducir el retardo de encolamiento en una comunicación TCP/IP basado en la distribución de la longitud de

los paquetes que se envían a través de una red de tráfico convergente, mediante el análisis y modelamiento de los patrones del tráfico real de protocolos y aplicaciones, y que sea más eficiente en cuanto al retardo comparado con mecanismos estándares como el CBWFQ.

1.4. Objetivos específicos

1. Analizar y modelar el tráfico real TCP/IP sobre una red convergente, alámbrica e inalámbrica, a nivel de protocolos y aplicaciones, en base a los patrones de longitud de paquetes, con el fin de estimar su comportamiento estocástico mediante distribuciones de Poisson.

2. Realizar un estudio sobre el retardo de encolamiento, y caracterizarlo mediante un enfoque predictivo basado en modelos con Regresión Polinómica.

3. Proponer y simular un modelo matemático de QoS basado en el análisis de los patrones por longitud de paquetes, que sea más eficiente que mecanismos existentes como CBWFQ, en función del retardo de encolamiento.

1.5. Hipótesis del trabajo

La aplicación de modelos predictivos del tráfico de red en los mecanismos de calidad de servicio, basados en la variable longitud de paquete, permite reducir los tiempos de retardo de encolamiento en un dispositivo intermedio de una conexión TCP/IP, en comparación con los obtenidos con métodos comunes como CBWFQ que incluyen fases como la marcación y clasificación de paquetes.

1.6. Metodología de la investigación

El proceso se investigación se divide en tres etapas:

1. Análisis y modelamiento de los patrones de tráfico
2. Estimación del retardo de encolamiento
3. Formulación y simulación de modelo propuesto

En el escenario propuesto de topología de pruebas que se muestra en la figura 1.3, la idea principal es generar tráfico convergente aleatorio (voz, datos y video), clasificar y modelar este tráfico por longitud de paquetes, y determinar y estimar el comportamiento del retardo de encolamiento para mecanismos de calidad de servicio comunes; finalmente compararlo con el modelo matemático propuesto en la tercera etapa utilizando un modelo de predictibilidad a partir de los datos obtenidos en la primera fase. La topología de prueba es diseñada pensando en los componentes de tráfico convergente en una red de campus moderna, en los protocolos y aplicaciones más utilizados.

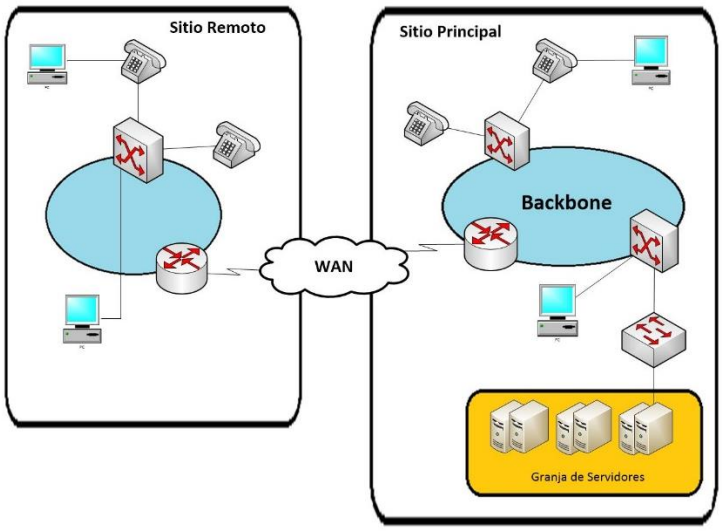


Figura 1.3. Escenario general propuesto como topología de prueba

En la primera etapa se realiza un estudio detallado de la variable longitud de paquete, tanto a nivel de protocolos como a nivel de aplicativos más comunes, sobre dispositivos conectados a una red alámbrica e inalámbrica, y en una red LTE (Long Term Evolution), con el objetivo de caracterizar y modelar dicha variable. Para obtener los datos necesarios, se implementan escenarios de red basadas en la topología de red convergente propuesta. Se recopilan millones de paquetes de datos en cada escenario, y posteriormente se analizan los mismos, y se estima su comportamiento estadístico.

En una segunda etapa, se utilizan los modelos y estimaciones de la fase uno con la finalidad de realizar un estudio del retardo unidireccional y del retardo de encolamiento. Este estudio sobre OWD nos muestra datos sobre retardo de transmisión, retardo de propagación, retardo de procesamiento y retardo de encolamiento. Nuestro estudio se centra en el último tipo de retardo, ya que es el que se debe al tratamiento de los paquetes de datos en las colas de salida, ligadas o no, a una política de calidad de servicio. Como resultado de esta fase, se modela el comportamiento predictivo del retardo de encolamiento, utilizando el método de Regresión Polinómica, el mismo que es comparado con otros métodos de regresión para determinar cuál representa une mejor relación de las variables dependiente e independiente. Esto permite definir una línea base de los tiempos esperados para el retardo de encolamiento, y sobre el cual se espera contribuir a su reducción.

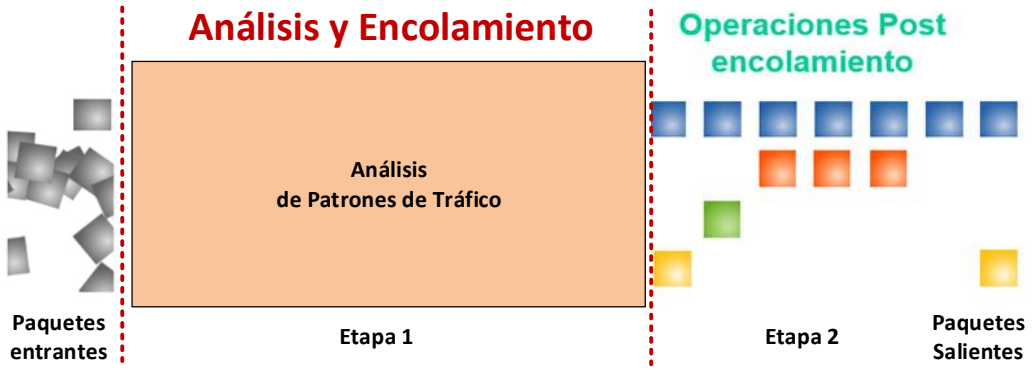


Figura 1.4. Modelo predictivo basado en análisis de patrones de tráfico

Una vez obtenidos los tiempos de respuesta de nuestro escenario propuesto de topología de prueba, se procede en una tercera etapa, al diseño y formulación de un modelo matemático que representa un modelo de calidad de servicio que contempla los resultados obtenidos en las fases 1 y 2, y que nos permita obtener mediante su simulación, un mejor comportamiento o reducción del retardo de encolamiento con respecto a lo identificado en la línea base. Los datos y resultados obtenidos mediante la simulación del modelo son utilizados para validar nuestras hipótesis. La representación de dicho modelo puede apreciarse en la figura 1.4.

Para cumplir con estas etapas se utilizan herramientas de simulación que permiten comprobar y validar el comportamiento deseado, y posteriormente, compararlos con los obtenidos en el mismo escenario de prueba donde se tomaron los datos de los diferentes

comportamientos de los mecanismos de calidad de servicio existentes. La idea de esta comparación de técnicas de calidad de servicio existentes es tener una línea base de referencia en cuanto al retardo de encolamiento medido para el mismo escenario, lo que nos ayuda a determinar qué técnica puede ser mejor según los tipos de tráfico de protocolos y/o aplicaciones.

Este estudio no busca correlacionar variables, ni demostrar que el retardo depende de otras posibles variables. Nos basamos en validar que podemos reducir el retardo de encolamiento utilizando métodos alternativos en el tratamiento de las colas de paquetes, analizando su longitud.

Para las simulaciones de tráfico se utilizan herramientas de software libre, bajo ambiente Linux (por mejor optimización de recursos). Para las mediciones reales de los datos de línea base, se utilizan equipos de redes que permiten configurar calidad de servicio en su sistema operativo.

1.7. Organización de la tesis

En el capítulo I se presenta el enfoque general de este proyecto de tesis. En el capítulo II se exponen el marco teórico, así como el estado del arte analizado en cuanto a clasificación de tráfico, retardo unidireccional y de encolamiento, y calidad de servicio. El análisis de resultados y modelamiento de los patrones de tráfico de red alámbrico e inalámbrico por protocolos y aplicaciones estudiados se presentan en el capítulo III. En el Capítulo IV se presentan los resultados y modelos estimados para el retardo unidireccional y de encolamiento en función de los patrones por longitud de paquetes que se presentan en el capítulo III. El modelo matemático que se propone para un esquema de QoS basado en despacho por longitud de paquetes, y los resultados de su simulación, se presentan en el capítulo V. Finalmente, las conclusiones y perspectivas de trabajo futuro son expuestas en el capítulo VI, así como la referencia de las publicaciones realizadas.

CAPÍTULO II: MARCO TEÓRICO Y ESTADO DEL ARTE

2.1. Clasificación y modelamiento del tráfico TCP / IP

La clasificación de tráfico consiste en una serie de métodos automatizados para analizar y clasificar conjuntos de paquetes de una red de datos, basados en características observadas de forma activa o pasiva en el tráfico, de acuerdo con objetivos de clasificación específicos.

2.1.1 La importancia de la clasificación del tráfico

La clasificación de tráfico tiene importancia en varios aspectos relacionados a las redes de datos, tales como: diseño, aprovisionamiento de recursos, registro de eventos, monitoreo y seguridad. Otras aplicaciones pueden ser para análisis de tendencia del tráfico de las aplicaciones; y finalmente en la conformación del tráfico para la asignación de técnicas de QoS.

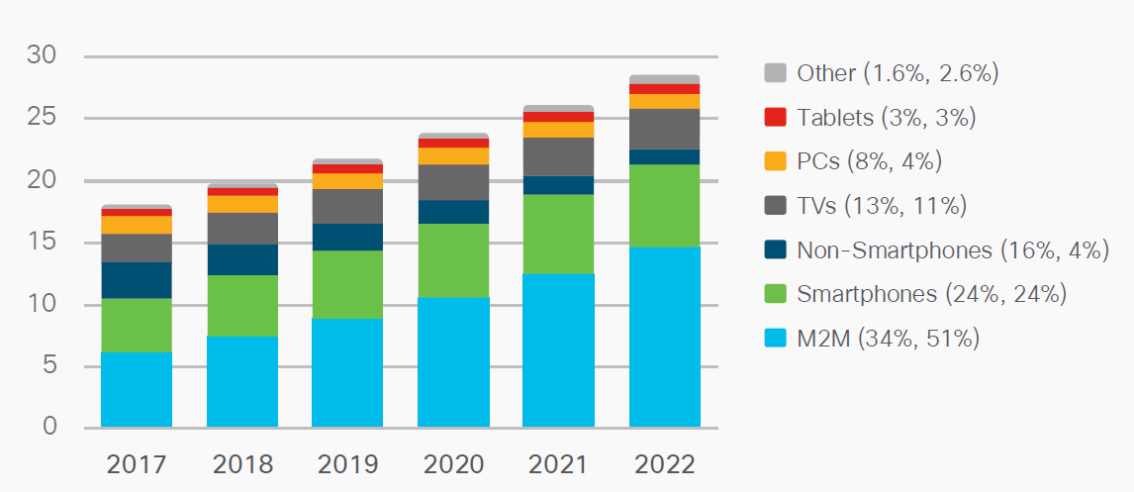


Figura 2.1. Crecimiento global de dispositivos y conexiones (en billones)

Fuente: Cisco VNI Global IP Traffic Forecast, 2017–2022

Un estudio de Cisco Systems sobre tendencias y predicción de tráfico IP [10], pronostica que para el 2022 cada persona generará un tráfico mensual de 50 GB, respecto a los 16 GB en 2017. Se espera que el número de dispositivos en red pase de unos 18 mil

millones en 2017, a unos 28.500 millones en 2022. Se predice que el tráfico de los celulares inteligentes represente el 44% respecto al 18% en 2017. El tráfico de redes inalámbricas y móviles representará el 71% del tráfico total de redes IP, mientras que el tráfico de redes alámbricas será del 29%. Respecto a los aplicativos, se espera que el tráfico de video IP represente el 82% del tráfico global. La figura 2.1 muestra el crecimiento global en billones de dispositivos y conexiones esperado en 2022.

2.1.2. Características del tráfico en una red

En redes alámbricas e inalámbricas la transmisión de la información se la realiza por medio de conmutación de paquetes [11]. El análisis de los paquetes puede ser modelado según la longitud, el tiempo de arribo entre paquetes e incluso por comportamiento de usuarios [12]. En este trabajo nos centramos en la longitud de paquete. Esta variable tiene un comportamiento estocástico [13] [14], el cual es monitoreado para el correspondiente análisis. El monitoreo del tráfico de red puede ser realizado de forma activa o pasiva [15]. El método activo consiste en inyectar tráfico en la red y analizar el comportamiento, mientras que el método pasivo consiste en capturar tráfico real de la red y analizarlo.

Una limitante del método pasivo es el tratamiento que se debe dar a la privacidad de la información que se captura, proceso que se lo realiza con un sniffer de red. Los sniffer normalmente capturan las cabeceras de los paquetes, así como los datos que contienen, comprometiendo de esa forma la privacidad [16]. Para proteger la privacidad de los datos se propone utilizar un nuevo sniffer de red que solamente captura la cabecera de cada paquete para su posterior análisis. La medición pasiva puede ser realizada a nivel de paquetes, flujos y sesiones [17] [18]. En este trabajo se utiliza la medición a nivel de paquetes.

2.1.3 Encapsulamiento y características de las aplicaciones

La red internet actual está basada en la implementación de dos protocolos de capa de red: IPv4 [19] e IPv6 [20], parte de la arquitectura TCP/IP [21]. Esta consta de 4 capas: Aplicación, Transporte, Internet y Acceso a Red. En la capa de aplicación se ejecutan los diferentes procesos que se asocian a las aplicaciones del usuario; los protocolos SSH (acceso remoto), SMTP (email), HTTP (servicios web), SSL (secure socket layer), DNS

(servicio de nombres de dominio), etc. Las aplicaciones se encapsulan en la capa de Transporte; estas capas se relacionan por medio de un conector lógico que se denomina el puerto de aplicación; cada aplicación tiene su propio identificador y se encapsula sobre uno de los dos protocolos de transporte definidos: TCP [22] o UDP [23].

Los protocolos de transporte a su vez se encapsulan en la capa de red o internet; estas dos capas se relacionan por un conector llamado protocolo; toda comunicación en este nivel se identifica además por las direcciones IP origen y destino. Finalmente, los paquetes IP (IPv4 o IPv6) se encapsulan sobre los protocolos de la capa de acceso, por ejemplo, Ethernet o MPLS (protocolo de conmutación de etiquetas). El conector lógico entre estas capas de denomina Tipo de paquete. La figura 2.2 muestra el modelo de capas de la arquitectura TCP/IP, los protocolos comunes que se asocian a cada capa, y el proceso de encapsulamiento desde la capa de aplicación hasta el acceso a red.

Para analizar el tráfico de red, se utiliza un capturador de paquetes (sniffer), que trabaja en la capa de acceso a red, por lo que un paquete capturado tiene información de todas las cabeceras de las capas, como se observa en la figura 2.2. Esto permite que el análisis pueda ser detallado, y pueda tener diferentes finalidades, por ejemplo, para la seguridad de red, la calidad de servicio, entre otros.

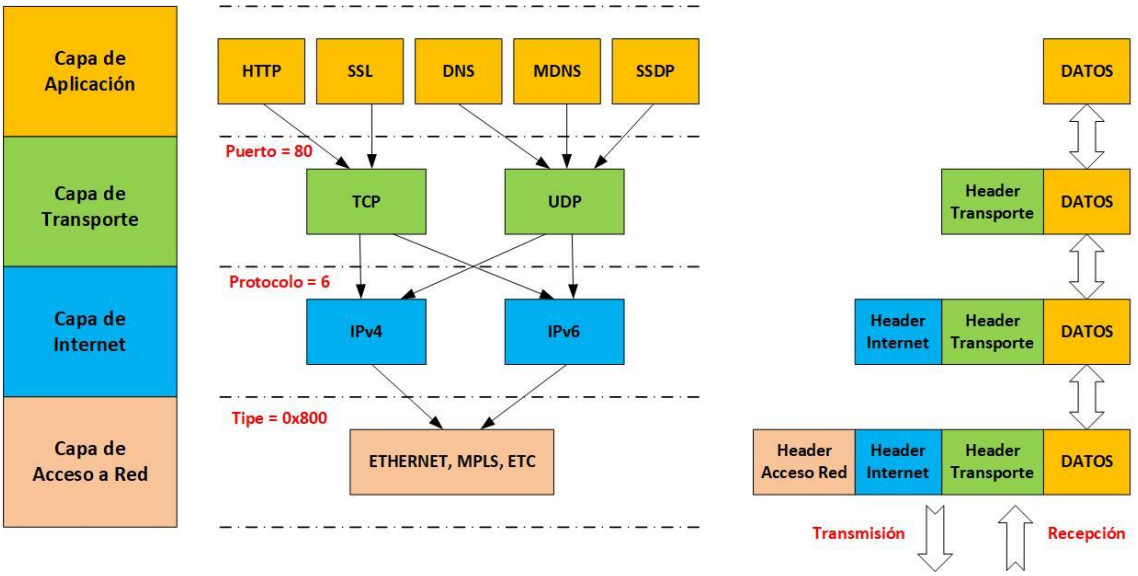


Figura 2.2. Arquitectura TCP/IP, protocolos y encapsulamiento

Tabla 2.1. Aplicativos por puerto comunes encontrados en nuestro estudio

Aplicación	Puerto	Transporte	Descripción
HTTP	80	TCP	Hyper Text Transfer Protocol es el protocolo de comunicación que permite el intercambio de información sobre la World Wide Web
SSL/HTTPS	443	TCP	HTTP Secure, Secure Socket Layer, o Tranport Layer Security son protocolos criptográficos que proporcionan comunicaciones seguras.
DNS	53	UDP	Domain Name System es un Sistema de nomenclatura jerárquico que permite resolver un nombre de dominio a una dirección IP.
BOOTP/ DHCP	67-68	UDP	El Boostrap Protocol es un protocolo de red utilizado para gestionar direcciones IP de forma automática. El servicio más conocido es el Dynamic Host Configuration Protocol o DHCP.
NETBIOS	137	UDP	El Network Basic Input Output System es una especificación de interfaz para acceso a los servicios de red, enlaza al sistema operativo con el hardware.
QUIC	443	UDP	Quick UDP Internet Connections es un protocolo de red que permite mejor rendimiento y bajo retardo en un intercambio de información.
SSDP	1900	UDP	Simple Service Discovery Protocol permite la búsqueda de dispositivos plug and play en una red.
MDNS	5353	UDP	Multicast DNS es un protocolo de descubrimiento para traducción de nombres de dominio en formato multidifusión.

Una desventaja del proceso de captura de paquetes es el manejo de la privacidad de la información del usuario de red, ya que los datos quedan expuestos en caso de que no estén cifrados. En la actualidad, se prioriza el uso de aplicativos basados en HTTPS o SSL, o incluso protocolos como TLS (Transport Layer Secure), con la finalidad de evitar comprometer la data del usuario. La tabla 2.1 describe los principales protocolos de capa de Aplicación, definidos por puerto, que se utilizan con mayor frecuencia, y observados en las redes analizadas en el presente trabajo.

Un estudio de Visual Capitalist [24] que analiza el comportamiento de aplicaciones en internet cada 60 segundos, muestra el tráfico de las Apps más comunes a nivel global, lo que se detalla en la tabla 2.2.

Tabla 2.2. Tráfico de las Apps más comunes de internet

App	Tráfico de la App en 60 segundos
YouTube	Alrededor de 700.000 horas de video son observados
Google	Se realizan más de 3.8 millones de búsqueda
Facebook	Se suben más de 240.000 fotos y cerca de 70.000 horas sobre contenido de video son observados
Twitter	Se envían más de 350.000 tweets
Instagram	Más de 65.000 fotos son subidas a la red social
WhatsApp	Se envían más de 29 millones de mensajes, 1 millón de fotos y 175.000 videos son compartidos.
Email	Se envían 156 millones de correos
Skype	Se realizan más de 2 millones de minutos de llamadas

2.1.4 Métodos de Clasificación del Tráfico

El objetivo de la clasificación del tráfico es comprender el tipo de tráfico transportado en las redes de datos, que evoluciona continuamente en alcance y complejidad. Por razones de seguridad y privacidad, han surgido muchas aplicaciones que utilizan técnicas de ofuscación, como puertos aleatorios, transmisión de datos cifrados o protocolos de comunicación patentados. Además, las aplicaciones se adaptan rápidamente ante los intentos de detectar ciertos tipos de tráfico, creando un desafío para los esquemas de clasificación de tráfico [25] [26] [27] [28]. Es importante considerar factores que influyen en el comportamiento del tráfico, tales como el despliegue de IPv6 en la red internet, el uso masivo de dispositivos como teléfonos inteligentes, y nuevas tecnologías como Wi-Fi 6 y el 5G.

Los métodos de clasificación dependen de las características de protocolos y/o aplicaciones que se van a analizar. Por ejemplo, si se lo hace por puertos, hay aplicativos que pueden usar puertos fijos, y otras que utilizan puertos aleatorios (menos comunes). En la figura 2.3 se muestran las diferentes características de evaluación.

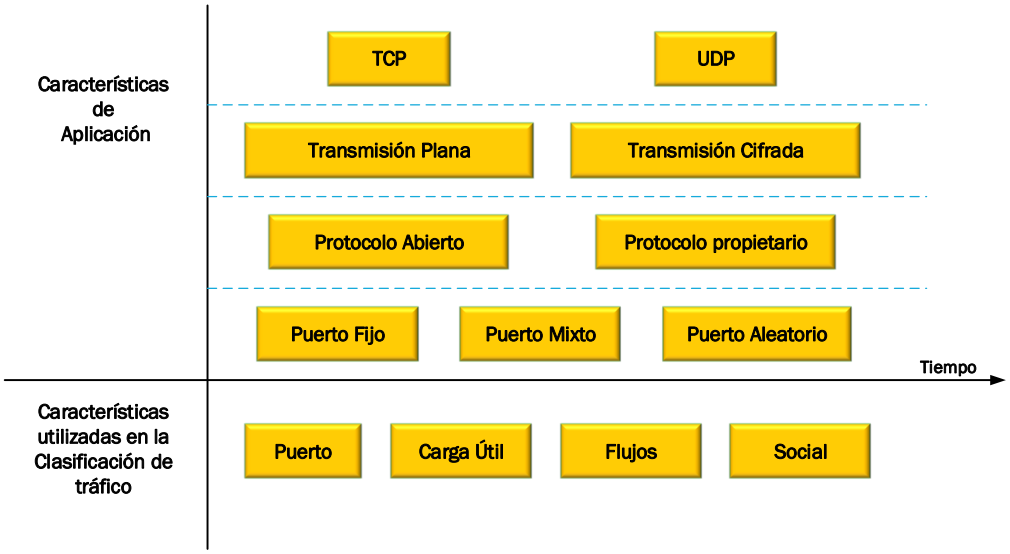


Figura 2.3. Características del tráfico

Estos métodos para clasificación pueden seguir su análisis basado en uno de los siguientes tipos de comparación:

1. Comparaciones exactas: basadas en características como números de puertos TCP o UDP, protocolos de encapsulamiento, carga útil, etc.
2. Comparación de patrones (heurísticos): basados en el análisis de patrones del tráfico de datos.
3. Métodos de aprendizaje automático: basados en el uso de análisis inteligente de datos, tales como mecanismos supervisados y no supervisados.

2.1.5 Estado del arte sobre clasificación y modelamiento de tráfico

La mayoría de los trabajos anteriores han analizado el tráfico de red utilizando la longitud de paquete, aplicando diferentes métodos. En [29], Zhang et al. presentan un estado del arte sobre clasificación del tráfico, donde se identifican métodos de coincidencia exacta, los heurísticos y los basados en características estadísticas o aprendizaje de máquina. Existe un análisis del comportamiento de la distribución de tráfico por longitud de paquete sin considerar el tipo de aplicación cuya distribución es bimodal [31] o trimodal [30].

Otros enfoques buscan la caracterización de aplicaciones en redes híbridas a partir de la longitud de paquetes de los data set analizados [31,32,33,35,36,39,40,41]. Enfoque similar, pero con data de aplicativos basados en puertos aleatorios y/o data cifrada se encuentran en [40,41]. Con respecto a la caracterización de Apps en redes móviles [42,43,44,47]. Finalmente, otros tipos de análisis en enfocan en la auto-similaridad del tráfico de red usando distribución de frecuencias y ancho de banda [34], en el número de conexiones activas [38], aplicación de algoritmos genéticos [45], y en la predicción del tráfico en una red de campus basados en el tráfico de internet [37].

Respecto al modelamiento de tráfico, se presentan modelos basados en funciones de distribución [48,50,51,54]. Otros se enfocan en modelamiento mediante aprendizaje de máquina [49,53,56]. Finalmente, hay enfoques basados en modelos matemáticos híbridos [52,55,57].

Los trabajos presentados sobre la longitud de los paquetes se centran y limitan en el análisis de protocolos de transporte (TCP y UDP) y en los puertos que sirven de interfaz lógica entre las capas de transporte y aplicación. Salvo el caso de utilizar datos de un aplicativo que está basado en cifrado, como SSL o HTTPS, en la mayoría la privacidad de la información está comprometida. En cuanto al modelamiento, se plantean estimaciones que no son tan aproximadas respecto al comportamiento estocástico de la longitud de los paquetes, nuestra variable de estudio, y del tiempo de arribo de los paquetes.

En este trabajo, en su primera fase, se propone analizar el tráfico de una red de campus híbrida (alámbrica e inalámbrica), y de una red celular con tecnología LTE, generados por diversos dispositivos como un computador de escritorio, un computador portable, y un celular, utilizando un novel sniffer que descarta la carga útil de los paquetes, y determinar la contribución de protocolos y aplicaciones más comunes entre los usuarios, y finalmente estimar modelos estadísticos que representen el comportamiento estocástico del tráfico analizado. Estos modelos se utilizan en la segunda fase para el análisis del retardo unidireccional y de encolamiento.

2.2 Análisis del retardo de Extremo a Extremo

Los paquetes en una red experimentan cierto retardo cuando son enviados por el dispositivo origen, pasando a través de una serie de dispositivos intermedios como ruteadores, hasta llegar al dispositivo destino; la mayoría de las políticas de red (QoS, filtrado de paquetes, etc.) están relacionados de alguna manera con el retardo [58]. El retardo de red de extremo a extremo es la medición desde el instante en que los datos son creados por una aplicación, entregados al Sistema Operativo, pasados a una tarjeta de red (NIC), codificados, transmitidos por un medio físico (cobre, fibra, aire), recibidos por un dispositivo intermedio (conmutador, ruteador), analizado, retransmitido a través de otro enlace, hasta llegar al destino.

Para la segunda fase de nuestro trabajo, es importante analizar y estimar el comportamiento del retardo de encolamiento, ya que es el componente del retardo unidireccional que se desea reducir mediante la aplicación de un modelo matemático de calidad de servicio.

2.2.1 Retardo Unidireccional (OWD)

En algunos puntos de la red, la demora es tan pequeña que puede ignorarse por razones prácticas, pero en otros casos la demora es significativa y es donde podemos ayudar a reducir dicho comportamiento. Este retardo de la red puede ser causado y clasificado como (ver figura 2.4):

- Serialización o retardo de transmisión (es fijo)
- Retardo de propagación (es fijo, pero usualmente pequeño)
- Retardo de procesamiento (es variable, pero usualmente pequeño)
- Retardo de encolamiento (es variable)

La medición más común utiliza una utilidad basada en ICMP llamada ping para el tiempo total de ida y vuelta o RTT [59]. Otra forma es medir el retardo unidireccional (OWD), definido en el RFC7679 (métrica de retardo unidireccional para IPPM) [60]. De acuerdo con este estándar IETF, comprender el retardo es útil por algunas razones:

- Si el retardo de extremo a extremo entre ambos hosts es relativamente grande, las aplicaciones no funcionan bien.
- Es difícil admitir aplicaciones en tiempo real en presencia de retardos significativos.
- Es más complicado para los protocolos de la capa de transporte mantener grandes anchos de banda.
- Proporciona una indicación del retardo debido solo a los componentes de propagación y transmisión.
- Una indicación de la demora que probablemente se experimentará cuando la ruta recorrida esté ligeramente saturada.

La medición del retardo unidireccional en lugar del retardo de ida y vuelta es más práctica porque la ruta desde una fuente a un destino puede ser diferente en sentido contrario ("rutas asimétricas"). Igual aplica a los tipos de redes, tales como capacidades de enlace, acceso inalámbrico versus acceso por cable, características de rendimiento, colas asimétricas. Además, el rendimiento de una aplicación puede depender principalmente del rendimiento en una dirección. Finalmente, el aprovisionamiento de calidad de servicio en una dirección puede ser radicalmente diferente al aprovisionamiento en la dirección inversa y, por lo tanto, las garantías de QoS difieren.

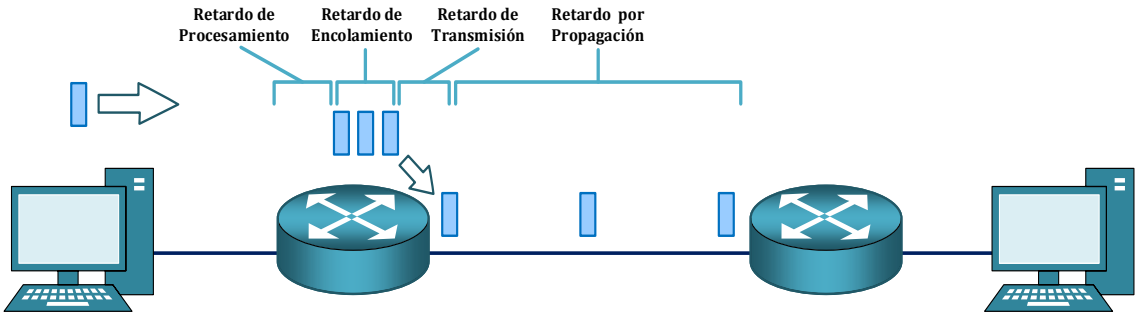


Figura 2.4. Componentes de Retardo Unidireccional (OWD) [Fuente: Cisco Systems]

2.2.2 Cálculo del retardo unidireccional

Para la ruta unidireccional desde el origen hasta el destino, podemos calcularlo mediante la fórmula 2.1 [60].

$$Nd = \sum_{i=1}^{n}(Sdi + Pdi + Fdi + Qdi) \quad (2.1)$$

Dónde:

- *Nd* es el retardo unidireccional de la red
- *n* es el número de saltos en la ruta
- Sdi es el retardo de serialización por cada salto
- Pdi es el retardo de propagación por cada salto
- Fdi es el retardo de reenvío o retardo de procesamiento por cada salto
- Qdi es el retardo de encolamiento por cada salto

La contribución de estos componentes de retardo puede variar significativamente. Por ejemplo, *Pdi* puede ser insignificante (un par de microsegundos) para un enlace que conecta dos ruteadores en el mismo campus universitario; pero pueden ser cientos de milisegundos si están interconectados por un enlace de satélite geoestacionario.

Se puede suponer que el retardo de propagación (*Pd*) y el retardo de procesamiento (*Fd*) son insignificantes para el presupuesto de retardo unidireccional [60]. Modelamos la ruta entre el origen y el destino como una serie de retardos en los enlaces (t) y las colas (q), ya que estos son los contribuyentes dominantes de OWD (en la medición activa, los hosts de origen y destino contribuyen con un retardo mínimo), como se define en el RFC 6703 [61].

$$Nd = \sum_{i=1}^{n}(ti + qi) \quad (2.2)$$

Dónde:

- *Nd* es el retardo unidireccional de la red
- *n* es el número de saltos en la ruta
- *ti* es el retardo en los enlaces
- *qi* es el retardo en las colas

2.2.3 Marcas de tiempo y mecanismos de sincronización

Para una estimación adecuada del OWD, es esencial considerar los parámetros que pueden influenciar en dicha medición. Consideramos el uso de la opción Timestamp para los paquetes IP definida en el RFC 3161 [62], lo que permite que un paquete enviado desde un dispositivo origen pueda ser etiquetado con una marca de tiempo, y luego monitoreado en el dispositivo receptor. La diferencia entre ambas mediciones nos indicaría el valor de OWD, asumiendo que los relojes de los dispositivos están perfectamente sincronizados, lo que en la práctica no puede ser garantizado. Se plantean las soluciones utilizadas para conseguir tal propósito según la precisión, como se muestra en la tabla 2.3: el protocolo de sincronización en red (NTP) [63], los sistemas de posicionamiento global (GPS) [64], o el estándar IEEE 1588 o protocolo de precisión de tiempo (PTP) [65]. La tabla 2.4 muestra una comparación de los métodos de sincronización.

Tabla 2.3. Precisión de los métodos de sincronización

Protocolo	**Características**	**Interface**	**Precisión de sincronismo**
GPS	Salida estándar de receptores GPS	Conectado directamente	1 us
NTP	Estándar para sincronismo a través de la red	Ethernet	WAN: 10 – 20 ms LAN < 1 ms
PTP	Estándar para dispositivos de instrumentación	Ethernet	< 1 us

Tabla 2.4. Comparación de los métodos de sincronización

Característica	**PTP (IEEE 1588)**	**NTP**	**GPS**
Cobertura	Pocas subredes	Área amplia	Área amplia
Comunicación en red	Ethernet	Internet	Satélite
Precisión	Sub microsegundo	milisegundos	Sub microsegundo
Implementación	Maestro / Esclavo	Peer to peer	Cliente / servidor
Recursos	Red pequeña y bajo recurso computacional	Red pequeña y moderado recurso computacional	moderado recurso computacional
Corrección de retardo	Si	Si	Si
Seguridad de protocolo	No	Si	No
Hardware requerido	Para alta precision	No	Receptor RF
Intervalo de actualización	2 segundos	Configurable	1 segundo

2.2.4 Retardo de Transmisión (Serialización)

Este retardo define los tiempos que lleva codificar los bits de un paquete en la interfaz física [66]. Depende de la velocidad de los enlaces (por ejemplo, una red Gigabit Ethernet), y de la longitud de los paquetes. Calculamos el retardo de serialización con la fórmula 2.3.

$$Sd = \frac{\#\ bits\ sent}{Link\ speed\ (bps)} \tag{2.3}$$

El componente de serialización a través de Gigabit Ethernet es insignificante, mientras que se convierte en un componente más significativo en los enlaces seriales de baja velocidad. Esto se muestra en la tabla 2.5:

Tabla 2.5. Ejemplo de Retardo por Transmisión (milisegundos)

Velocidad de enlace	**Retardo de Transmisión (64 bytes)**	**Retardo de Transmisión (1500 bytes)**
1000 Mbps	0.0005	0.012
100 Mbps	0.005	0.12
10 Mbps	0.05	1.2
1,544 Mbps	0.33	8
512 Kbps	1	24

2.2.5 Retardo de propagación

Este tipo de retardo define el tiempo que tarda un paquete en llegar de un extremo del enlace al otro, cuando se coloca una señal eléctrica u óptica en el medio de transmisión [67]. La señal no se propaga instantáneamente al otro extremo del medio, ya que se agrega un retardo debido a la resistencia presente (medios conductores como el cobre), o a la refracción y propiedad de reflexión interna total (fibra óptica). La única variable que afecta el retardo de propagación es la longitud del enlace, siguiendo la fórmula 2.4. La tabla 2.6 muestra dicha relación.

$$Pd = \frac{Length\ of\ link\ (meters)}{2.1\ x\ 10^{8}\ (meters\ /\ second)} \tag{2.4}$$

Tabla 2.6. Ejemplo de retardo de propagación

Distancia	Retardo de propagación
1000 Km	4.8 ms
10 Km	0.048 ms
1 Km	4.8 us
10 m	0,048 us
1 m	4.8 ns

2.2.6 Retardo de procesamiento

También llamado retardo de reenvío se refiere al tiempo de análisis en un dispositivo intermediario, como un ruteador, y se produce cuando un paquete se recibe por completo hasta cuando el paquete se ha colocado en una cola de salida (software o hardware) [67]. Se considera el tiempo requerido para analizar la cabecera del paquete y decidir a dónde reenviar el mismo (enrutamiento). Aunque es un retardo variable, por lo general es un componente muy pequeño que podemos ignorar en los cálculos generales del presupuesto de OWD.

En un ruteador, puede depender del número de entradas en la tabla de enrutamiento, de la implementación de las estructuras de datos, del hardware en uso, el nivel de utilización de la CPU, etc. Puede incluir también la verificación de errores, como los cálculos de suma de comprobación de encabezado IPv4 o IPv6.

2.2.7 Retardo de Encolamiento

Es un componente clave del retardo de la red. Consiste en el tiempo que un paquete permanece en las colas (software y/o hardware) del dispositivo, asociadas a las interfaces de salida de un ruteador [67]. Las colas ayudan a evitar la pérdida de paquetes, pero si el tamaño de una cola es grande puede causar un retardo considerable. Una forma de evitar el retardo de la cola es aliviando la congestión de paquetes.

El retardo en la cola de un paquete específico dependerá de la cantidad de paquetes que lleguen antes y que estén en cola y esperando la transmisión a través del enlace; el

retardo de un paquete determinado puede variar de forma significativa de un paquete a otro. El número de paquetes que esperan en la cola dependerá de la intensidad del tráfico y del tipo de tráfico.

Los algoritmos de encolamiento del ruteador intentan adaptar los retardos a preferencias específicas o imponer un retardo común en todo el tráfico. El comportamiento predeterminado para la cola es FIFO (Primero en entrar, primero en salir) donde los paquetes tienen un peso similar. Por otro lado, en lugar del modelo FIFO con una sola cola (software y hardware), otros mecanismos crean múltiples colas a nivel de software, colocan paquetes clasificados en estas colas establecidas y luego seleccionan paquetes de acuerdo con una política de despacho. Para una estimación rápida del retardo en la cola, se puede usar la fórmula 2.5.

$$Qd = Sd * Lq \tag{2.5}$$

Dónde:

- *Sd* es el retardo de serialización
- *Lq* es la longitud de la cola

¿Cuándo es considerable el retardo en la cola o cuándo es insignificante? Depende en gran medida de la velocidad a la que llega el tráfico a la cola, la velocidad de transmisión del enlace y la naturaleza del tráfico que llega (en ráfaga o sostenido) [68]. La dependencia cualitativa del retardo medio de la cola en la intensidad del tráfico se muestra en la figura 2.5.

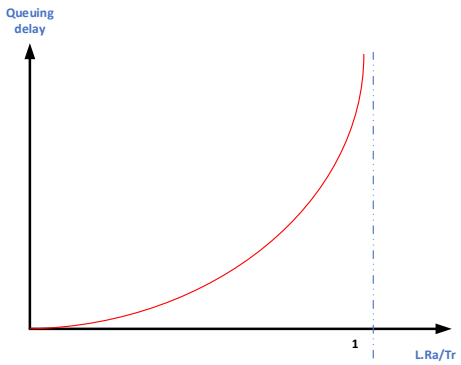


Figura 2.5. Efecto de la intensidad de tráfico sobre el retardo de encolamiento

2.2.8 Estado del arte sobre el retardo de encolamiento

La mayoría de los trabajos anteriores han analizado el RTT y el OWD, utilizando diferentes topologías y métodos, para simular el tráfico de red y estimar modelos del retardo de encolamiento, que es el componente más significativo en los cálculos analizados.

Unos trabajos se propone métodos de estimación de retardo usando o no mecanismos de sincronización de reloj para determinar la precisión y la sensibilidad del retardo en las colas [69,72,73,77]. Otros analizan los componentes variables y constantes del retardo unidireccional [70,71,74,75,76].

En los trabajos consultados, se emplean mecanismos de sincronización que buscan exactitud de la toma de muestras, pero a un alto costo; no se obtienen en base de modelos de tráfico a nivel de Apps que nos ayuden a obtener una medida adecuada del retardo de encolamiento en un escenario de prueba como el propuesto en este trabajo. Para una mejor estimación del retardo en las colas, se emplea generación de tráfico basados en las distribuciones reales obtenidas en la primera fase de la investigación; se utiliza un mecanismo de sincronismo más estandarizado como NTP (aunque con menos precisión) y de bajo costo en su implementación (presente en los sistemas operáticos de todos los equipos utilizados en las pruebas). Además, se estiman modelos predictivos del retardo de encolamiento mediante regresiones polinómicas. Finalmente, se comparan para varios mecanismos comunes de calidad de servicio.

2.3 Calidad de servicio

En las redes actuales, los usuarios esperan que el contenido esté disponible al instante. Pero si el tráfico excede el ancho de banda de los enlaces entre el origen del contenido y el usuario, ¿cómo se puede garantizar en una red de datos una experiencia de calidad? El diseño de la red puede incluir herramientas de calidad de servicio (QoS) para garantizar que ciertos tipos de tráfico, como voz y video, tengan prioridad respecto al tráfico no urgente, como el correo electrónico y la navegación web. Un dispositivo

implementa QoS solo cuando experimenta algún tipo de congestión. La QoS aplicada en las colas, afecta al retardo de encolamiento.

Los tipos de demandas de voz, video y tráfico de datos en la red son muy diferentes. El tráfico de voz tiene una necesidad de ancho de banda predecible y tiempos de llegada de paquete conocidos, pero es sensible a los retardos y a la pérdida de paquetes; el retardo no debe superar los 150 milisegundos, las fluctuaciones no deben superar los 30 ms y la pérdida de paquetes no debe ser superior al 1% [78]. El tráfico de voz requiere al menos 30 Kbps de ancho de banda. El tráfico de video tiende a ser imprevisible, inconsistente y a transmitirse por ráfagas, en comparación con el tráfico de voz. Adicionalmente es menos resistente a pérdidas y tiene un mayor volumen de los datos por paquete. El retardo no debe ser superior a 400 milisegundos, las fluctuaciones no deben ser mayores a 50 ms, y la pérdida de paquetes no debe ser superior al 1%. El tráfico de video requiere al menos 384 Kbps de ancho de banda.

El tráfico de datos no es en tiempo real, y tiene una necesidad impredecible de ancho de banda. La mayoría de las aplicaciones de datos utilizan TCP, y en menor grado UDP. Por ejemplo, el tráfico de YouTube y Facebook representa el mayor volumen de tráfico en internet, y se basan más en video y streaming, y trabajan sobre TCP. Se debe considerar si el tráfico de la App es interactivo o de misión crítica.

En la tercera fase de la investigación se propone un mecanismo de QoS que analice los patrones de tráfico de la red en función de las longitudes de paquetes, y mediante una política de despacho ajustada, permita obtener menores tiempos de retardo de encolamiento, comparada con otras técnicas comunes de QoS.

2.3.1 La congestión como métrica de calidad de servicio

La congestión se produce cuando se agregan varias líneas de comunicación en un mismo dispositivo, y luego muchos de esos datos se colocan en menos interfaces salientes o en una interfaz más lenta [79]. La congestión también puede producirse cuando los paquetes grandes impiden que paquetes más pequeños se transmitan a tiempo. Cuando el volumen de tráfico es mayor de lo que se puede transportar en la red, los dispositivos

colocan los paquetes en cola en la memoria hasta que haya recursos disponibles para transmitirlos. Los paquetes en cola causan retardos, dado que los nuevos paquetes no se pueden transmitir hasta que no se hayan procesado los anteriores. Si sigue aumentando la cantidad de paquetes que se pondrán en cola, la memoria del dispositivo se llenará y los paquetes se descartarán. Se puede mejorar aplicando clasificación de los paquetes en varias colas, como se muestra en la figura 2.6.

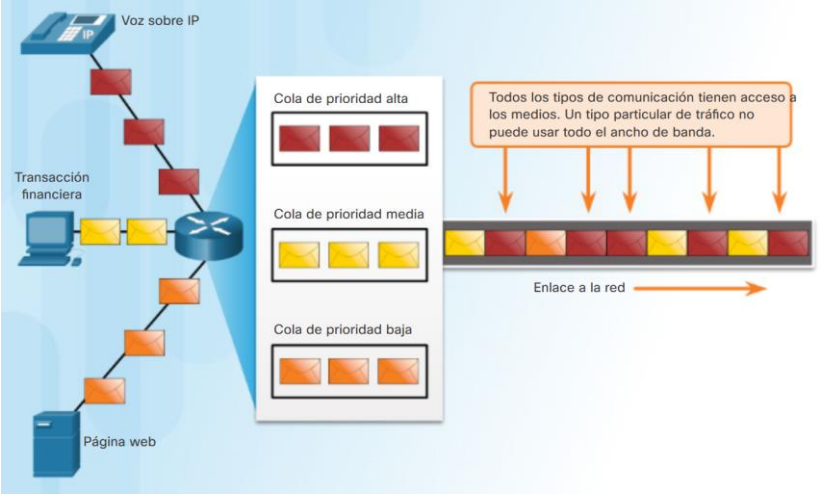


Figura 2.6. Calidad de Servicio [Fuente: Cisco Systems]

La congestión de red también puede generar jitter, que es la variación en el retardo de los paquetes recibidos. En el extremo emisor, los paquetes se envían de manera constante con un espacio de separación uniforme entre paquetes. Debido a la congestión de red, la demora entre cada paquete al arribar al destino puede variar en lugar de permanecer constante [80].

2.3.2 Modelos de arquitectura de QoS

La calidad de servicio puede implementarse en base a los tres modelos conocidos [81] [82], descritos en la tabla 2.7, que son:

- Modelo de mejor esfuerzo (Best Effort)
- Servicios integrados (IntServ)
- Servicios diferenciados (DiffServ)

El modelo de mejor esfuerzo en realidad no se trata de ninguna implementación; se aplica por defecto cuando no se requiere calidad de servicio. El diseño básico de Internet ofrece la entrega de paquetes mediante este modelo y no brinda ninguna garantía [83]. El consumo de ancho de banda, los retardos y la fluctuación, son impredecibles en este modelo.

Tabla 2.7. Ventajas y Desventajas de las arquitecturas de QoS

	Ventajas	**Desventajas**
Best Effort	Es el modelo más escalable	No hay garantías de entrega
	Su limitante es el ancho de banda, lo que aplica por igual a todo el tráfico.	En caso de llegar, los paquetes pueden llegar en cualquier orden.
	No se requieren mecanismos de QoS	Ningún paquete tiene trato preferencial
	Es el modelo más fácil y rápido de implementar.	Los datos de misión crítica se manejan igual que cualquier App informal.
IntServ	Control explícito de admisión de recursos de extremo a extremo.	Uso intensivo de recursos debido a la señalización continua de arquitectura activa.
	Control de admisión de políticas por solicitud	Enfoque basado en flujos no escalable para despliegues grandes como en la red internet.
Diffserv	Gran escalabilidad	Sin garantía de extremo a extremo
	Proporciona diferentes niveles de calidad	Requiere conjunto de mecanismos para la operación en la red

Las necesidades de aplicaciones en tiempo real, como video remoto, conferencias multimedia, entre otros, motivaron el desarrollo del modelo de arquitectura IntServ [84] [85] [86]. IntServ es un modelo de servicio múltiple que puede acomodar múltiples requisitos de Calidad de Servicio. Utiliza reservación de recursos y mecanismos de control de admisión para establecer y mantener QoS.

El modelo DiffServ especifica un mecanismo simple y escalable para clasificar y administrar el tráfico de red y proporcionar las garantías de la QoS en redes IP modernas. El diseño de DiffServ supera las limitaciones de los modelos de Best Effort e IntServ. Este modelo se describe en [87] [88] [89]. DiffServ no puede otorgar garantías de extremo a extremo. Su operación se representa en la figura 2.7.

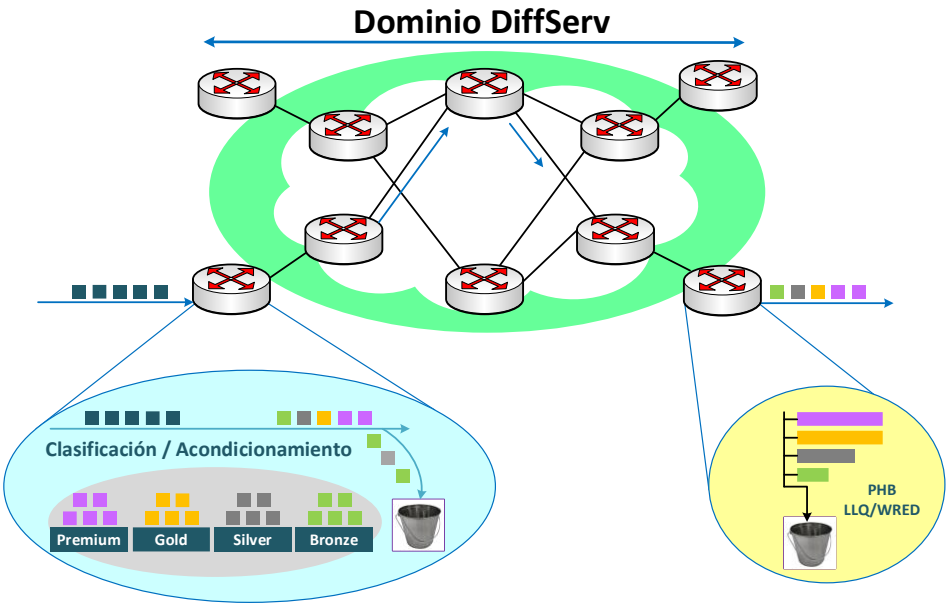


Figura 2.7. Modelo de QoS DiffServ [Fuente: Cisco Systems]

2.3.3 Componentes de QoS: Marcación y Clasificación

Hay tres procesos donde se aplican diferentes herramientas de la QoS [82], como se puede visualizar en la figura 2.8. Se clasifican los paquetes de ingreso (cuadros grises) y se marca su encabezado IP respectivo (cuadros de color). Para evitar la congestión, luego se asignan recursos a los paquetes en base a las políticas definidas. Los paquetes son luego puestos en la cola y reenviados a la interfaz de egreso según la política definida de modelado y regulación de tráfico de la QoS.

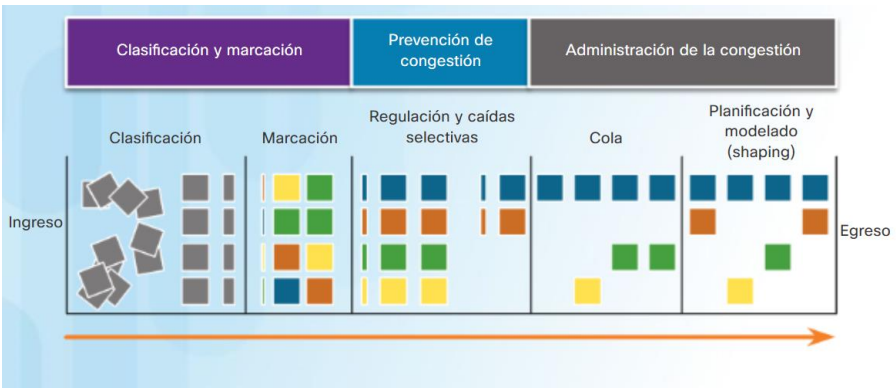


Figura 2.8. Componentes de QoS [Fuente: Cisco Systems]

Antes de que a un paquete se le pueda aplicar una política de QoS, el mismo tiene que ser clasificado. La clasificación y la marcación nos permiten identificar o "marcar"

los tipos de paquetes. La clasificación determina la clase de tráfico al cual los paquetes o tramas pertenecen. Solo pueden aplicarse las políticas al tráfico después del marcado. IPv4 e IPv6 especifican un campo de marcación de paquetes. Los dispositivos receptores remiten al campo para aplicar la política de QoS correspondiente.

El estándar original de IP (RFC 791) especificaba el campo de preferencia, el mismo que no proporcionaba suficiente granularidad para implementar QoS. El RFC 2474 [90] redefine el campo de ToS, como campo de punto de código de servicios diferenciados (DSCP).

2.3.4 Congestión y políticas de despacho de las Colas

La política de QoS de la red se activa cuando se produce una congestión en el enlace. La colocación de los paquetes en cola es una herramienta administrativa para la congestión que puede almacenar en búfer, priorizar, y, si corresponde, reordenar los paquetes antes de que estos se transmitan al destino. De allí que este análisis y despacho, introduzca un retardo adicional al retardo de extremo a extremo. Sin el encolamiento, los paquetes que exceden las capacidades del dispositivo intermedio o de las colas, serán descartados. Se encuentran disponibles varios algoritmos de encolamiento [91] [92], entre los que mencionamos:

- Primero en entrar, primero en salir (FIFO)
- Mecanismo de cola equitativo ponderado (WFQ)
- CBWFQ (mecanismo de cola de espera equitativo y ponderado basado en clases)

El encolamiento FIFO, implica el almacenamiento en búfer y el reenvío de paquetes en el orden de llegada. FIFO no tiene concepto de prioridad ni clases de tráfico, por lo que no toma decisiones sobre la prioridad de los paquetes. Hay una sola cola, y todos los paquetes se tratan por igual. Su operación se representa en la figura 2.9.

La WFQ es un método de programación automatizada que proporciona la asignación del ancho de banda justo a todo el tráfico de red. WQF aplica prioridades o ponderaciones para identificar el tráfico y clasificarlo en conversaciones o flujos, como

se muestra en la figura 2.10. El WFQ permite dar prioridad al tráfico de bajo volumen e interactivo sobre el tráfico de gran volumen.

Figura 2.9. Encolamiento FIFO [Fuente: Cisco Systems]

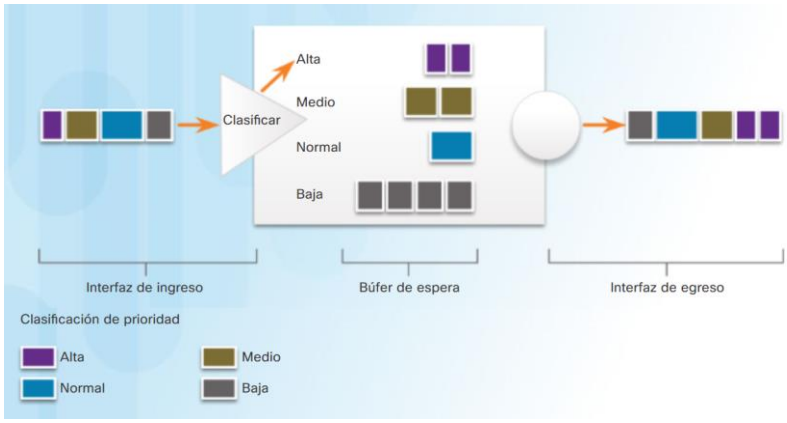


Figura 2.10. Encolamiento WFQ [Fuente: Cisco Systems]

El CBWFQ extiende la funcionalidad estándar de WFQ para admitir las clases de tráfico definidas en función del tráfico de la red. Estas clases. Los paquetes que cumplen los criterios de coincidencia para una clase constituyen el tráfico para esa clase. Se reserva una cola FIFO para cada clase y el tráfico de cada clase se dirige a su respectiva cola, como se muestra en la figura 2.11.

Figura 2.11. Encolamiento CBWFQ [Fuente: Cisco Systems]

2.3.5 Estado del arte de QoS

Los conceptos de QoS son aplicables a una serie de tecnologías de red y servicios para aplicaciones, por lo que existe un amplio estudio para mejorar la experiencia del usuario en aplicativos que son susceptibles a retardo como la Voz y el Video sobre IP, o el streaming [94,95,101], con el fin de mejorar el rendimiento. Otros trabajos en enfocan en el análisis de garantizar QoS sobre tecnologías como Ethernet, MPLS, LTE, CDMA, redes inalámbricas, entre otras [96,97,98,99,100,101,102], con la finalidad de garantizar confiabilidad y el retardo.

Los trabajos consultados, no basan sus modelos en función de patrones de longitudes de paquetes. Y en lo general, los modelos obedecen a aspectos diferentes como topologías reales, y las tecnologías aplicadas. Este trabajo en su fase # 3, presenta un modelo matemático aplicado a calidad de servicio, asociando colas en función de longitudes de paquetes, clasificando el tráfico de acuerdo con patrones estimados de protocolos y aplicaciones, y estableciendo una política de despacho de colas que permita reducir el retardo de encolamiento de los paquetes.

CAPÍTULO III

MODELAMIENTO DEL TRÁFICO DE RED Y DE APPS

3.1. Metodologías aplicadas

En la primera fase de la investigación, se propone analizar de forma pasiva el tráfico de red que generan dispositivos conectados a una red de campus alámbrica e inalámbrica, y el tráfico generado por dispositivos móviles que se conectan a una red LTE o una red inalámbrica. En el primer caso, el análisis permite modelar el tráfico a nivel de protocolos y puertos de aplicación; en el segundo caso, modelar el tráfico de las principales aplicaciones que utilizan los usuarios. Para la captura de la información se utiliza un sniffer de red; para el modelamiento y estimación de los patrones de red se utilizan distribuciones de Poisson, que reflejan el comportamiento estocástico del tráfico. Los escenarios de captura de tráfico, resultados obtenidos y modelos estimados se presentan en las siguientes secciones.

En el proceso de captura de los paquetes de red, un aspecto importante es el manejo de la privacidad de la información que contienen. Normalmente los sniffer de red capturan tanto las cabeceras como la data o carga útil de los paquetes. Por esta razón, se propone el desarrollo y uso de un sniffer que evita el almacenamiento de la data de aplicación del usuario, garantizando la privacidad de la carga útil de los paquetes. Además, con el despliegue de IPv6, el sniffer debe diferenciar un entorno de red que utiliza doble pila con IPv4. Finalmente, debe consumir pocos recursos de hardware, lo que permite una captura más eficiente de los paquetes a analizar.

El sniffer propuesto le llamamos TinySniff; está escrito en lenguaje C y se ejecuta bajo el sistema operativo Linux (ver anexo A). Es un software portátil y liviano que consume una pequeña cantidad de recursos (memoria y CPU). Puede capturar tráfico en escenarios LAN y WLAN, y almacenar las cabeceras capturadas en archivos planos, en formato texto. TinySniff está diseñado para capturar los siguientes campos en el encabezado de un paquete para su posterior análisis: longitud total (IPv4) o longitud de

la carga útil (IPv6), dirección de origen, dirección de destino, protocolo (IPv4) o siguiente cabecera (IPv6), puerto de origen, y puerto de destino, como se muestra en la figura 3.1.

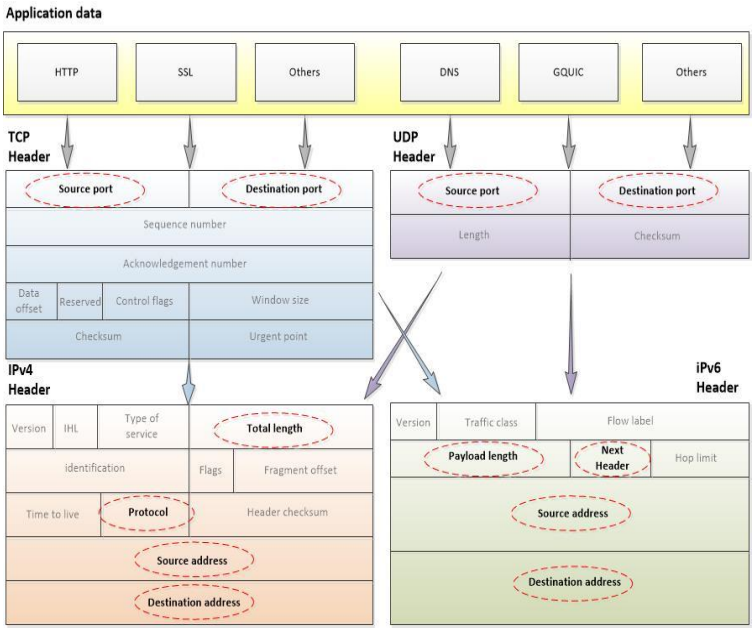


Figura 3.1. Campos analizados por TinySniff

3.2. Escenarios de estudio

Para la captura de tráfico y su posterior análisis, se utilizan dos escenarios, el primero de una red híbrida de un campus universitaria, y el segundo de una conexión de datos heterogénea LTE/Wi-Fi utilizada por dispositivos móviles.

El primer escenario se muestra en la figura 3.2, una red híbrida de un campus universitario, en la que se procede a capturar tráfico de la red virtual (VLAN) de datos más significativos, y de la VLAN inalámbrica de campus. Para lograr este objetivo, se configura en modo espejo (Port Mirror) un puerto de un switch de capa de distribución, en el cual se replican los paquetes de la red en el puerto asignado. Se procede a la captura de paquetes por medio de TinySniff, y se almacenan las cabeceras en un archivo plano. Finalmente, se procede al análisis de la información. El switch de distribución indicado es de marca CISCO, y la capacidad de port mirror en este fabricante se denomina SPAM (Switch Port Analyzer Monitor). Esta característica debe ser soportada por el sistema operativo conocido como IOS.

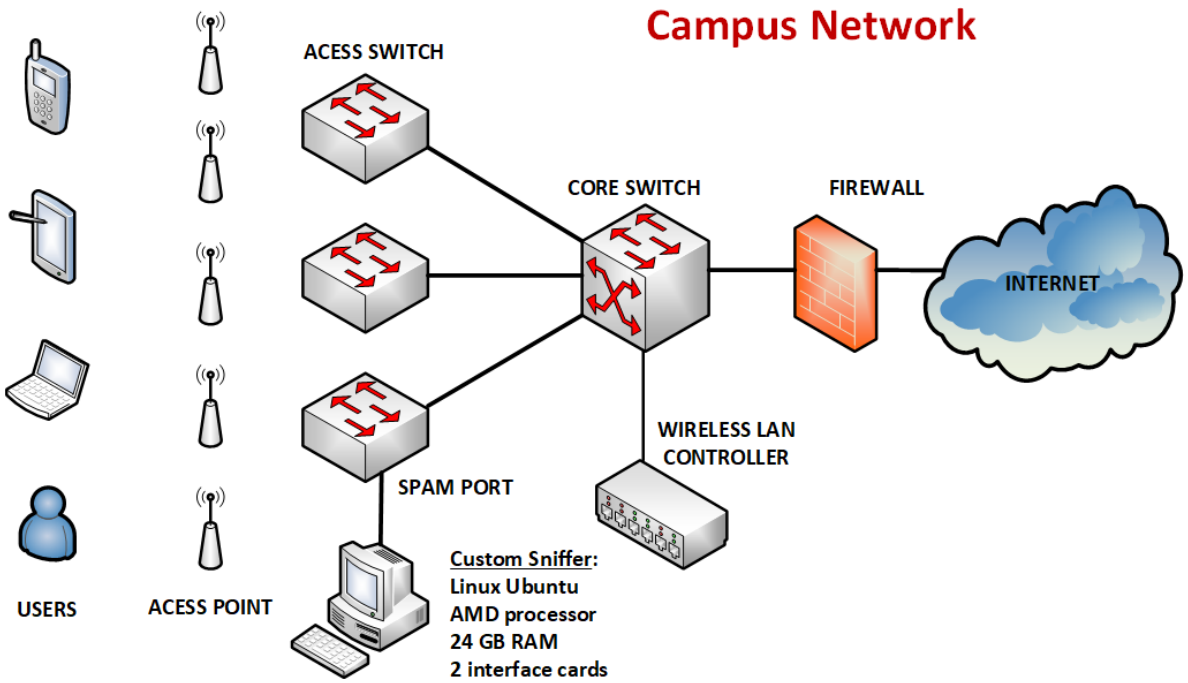


Figura 3.2. Escenario #1 de captura de paquetes en red híbrida de campus

La red híbrida de campus corresponde a una universidad en Ecuador. Este campus ocupa una extensión de 610 hectáreas, tiene 8 facultades, y tiene alrededor de 10.000 estudiantes en sus 27 carreras de pregrado. Además, cuenta con una planta docente de 602 profesores e investigadores, y 713 empleados administrativos. La tabla 3.1 describe la red tanto alámbrica como inalámbrica.

Tabla 3.1. Descripción de la red de campus

Característica	Red alámbrica	Red inalámbrica
Capacidad y enlaces	Backbone 10 Giga Ethernet, enlaces de fibra óptica hacia cada facultad a 10 Gbps. Se distribuye por medio de cableado horizontal en cada edificio con tasas de 1 Gbps	Compuesta por cerca de 300 puntos de acceso, capacidad hasta 300 Mbps de acceso, administrados por controladores de red inalámbricos (WLC)
Ancho de banda de acceso a internet	2.200 Mbps Pico de tráfico 60%	300 Mbps Pico de tráfico 70%
Número de dispositivos de usuario final	3.000	6.000

En el segundo escenario se recopila tráfico real de aplicaciones móviles de un smartphone conectado en primera instancia a una red inalámbrica del campus universitario que se muestra en la figura 3.2, y luego conectamos el dispositivo móvil a la red LTE que se muestra en la Fig. 3.3. El teléfono inteligente es un Samsung Galaxy S8 +, Se puede conectar a redes Wi-Fi utilizando 802.11 a/b/g/n/ac 2.4G + 5GHz, VHT80 MU-MIMO, 1024-QAM. También se puede conectar a otras redes celulares basadas en 3G TD-SCDMA, 4G LTE FDD, 4G LTE TDD.

En ambos casos, se recopila paquetes de datos de las 8 aplicaciones más comunes en teléfonos inteligentes, que incluyen: Google Drive, Facebook, búsqueda de Google, correo electrónico, Twitter, YouTube, WhatsApp e Instagram. Para cada aplicación, ejecutamos las tareas típicas de acuerdo con el comportamiento del usuario y recopilamos los paquetes para su análisis. En la prueba de Drive cargando y descargando archivos; en Facebook, subiendo fotos, videos, navegando por los perfiles de usuario, etc. En Google, buscar información y visualizar enlaces, imágenes, videos, etc. Y similar para otras aplicaciones para el estudio exploratorio de la longitud de los paquetes.

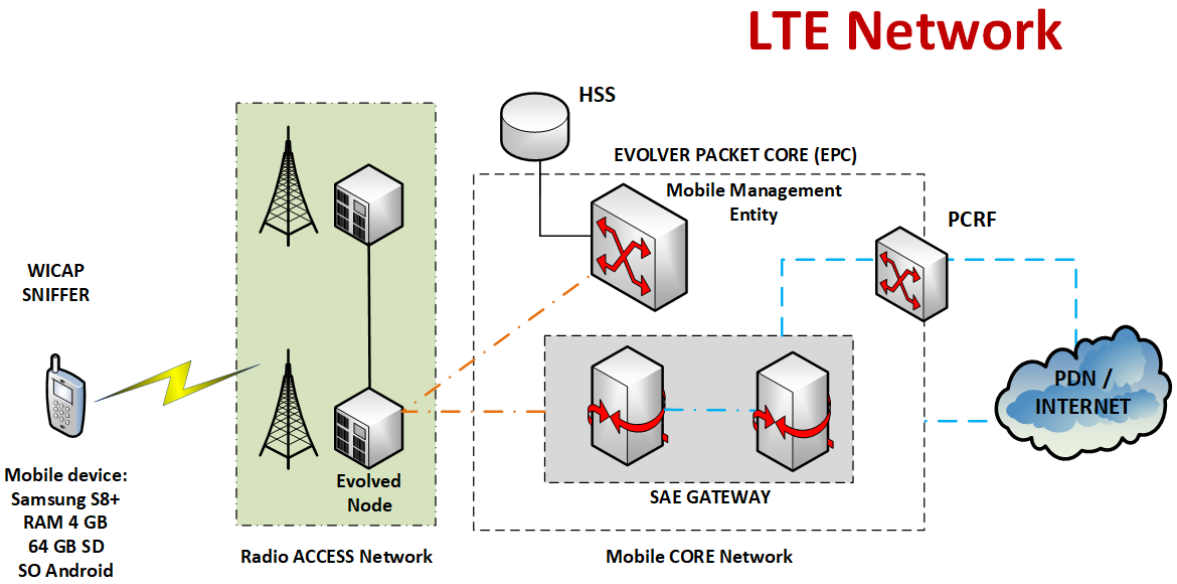


Figura 3.3. Escenario #2 de red LTE para captura de tráfico de Apps

Para la captura de paquetes en el segundo escenario se propone utilizar un sniffer gratuito para redes Wi-Fi llamado Wireshark que recolecta y procesa los paquetes, y un sniffer comercial llamado WICAP que se ejecuta en teléfonos inteligentes con sistemas

operativos Android y funciona en redes celulares LTE. Además, WICAP guarda los datos de captura en formato PCAP, lo que permite abrir y analizar los datos con otros sniffers más avanzados como Wireshark. Se instala Wireshark en una computadora de escritorio con Windows 10. Sus especificaciones técnicas son: procesador AMD FX-8300 de ocho núcleos, 24 GB de RAM y Ethernet de dos tarjetas de interfaz de red (NIC). Una NIC está dedicada a la administración de PC y la segunda para capturar el tráfico. Se conecta la NIC de captura de paquetes en un puerto gigabit de conmutador Cisco de capa de acceso, y configuramos este puerto en modo espejo (SPAM) para reflejar el tráfico inalámbrico de la VLAN correspondiente.

La red LTE pertenece a un operador celular en nuestro país. Esta utiliza IP RAN como capa de acceso, LTE para acceso móvil, anillos de fibra óptica para redundancia y alta disponibilidad, ruteadores ALCATEL, uno por cada radio de acceso, y protocolo IS-IS (sistema intermedio a sistema intermedio) como protocolo IGP (protocolo de puerta de enlace de interior).

3.3. Resultados

En el caso del primer escenario, las capturas de tráfico de red se realizan durante los picos de mayor tráfico de la red alámbrica e inalámbrica, de acuerdo con las gráficas de uso de ancho de banda proporcionadas por el departamento técnico de Tecnologías de la Información de la red híbrida de campus analizada. El departamento técnico maneja aplicativos para medición de uso de ancho de banda basados en herramientas de software libre MRTG y NETFLOW. En la VLAN de datos, se capturaron cerca de 10 millones de paquetes, a una tasa promedio de 1.899 pps (paquetes por segundo) y tamaño promedio de 709 bytes; esta captura se realizó en octubre 25 de 2018 entre las 08:54:33 y las 10:21:12. Mientras que en el escenario de la VLAN inalámbrica de Campus se capturaron cerca de 12 millones de paquetes, a una tasa promedio de 21.562 PPS y tamaño promedio de 742 bytes; dicha captura se realizó en enero 9 de 2019 entre las 15:18:48 y las 15:28:03. Las tablas 3.2 y 3.3 muestran en resumen los datos obtenidos, clasificados por protocolo y aplicación.

Tabla 3.2. Tráfico de red clasificado de la VLAN alámbrica

	Protocolo	IPv4		IPv6	
		Frecuencia	Porcentaje	Frecuencia	Porcentaje
TCP	SSL	4.294.926	55,64%	26.592	87,64%
	HTTP	2.556.705	33,12%	3.749	12,36%
	Otros	867.062	11,24%	0	0,00%
	Total	7.718.693	100,00%	30.341	100,00%
UDP	GQUIC	314.816	44,27%	9.494	12,09%
	MDNS	123.093	17,31%	37.293	47,47%
	SSDP	83.116	11,69%	4.154	5,29%
	BOOTSTRAP	47.422	6,67%	3.310	4,21%
	NETBIOS	36.124	5,08%	0	0,00%
	DNS	32.232	4,53%	1.594	2,03%
	Otros	74.261	10,44%	22.708	28,91%
	Total	711.064	100,00%	78.553	100,00%

Tabla 3.3. Tráfico de red clasificado de la VLAN inalámbrica

	Protocolo	IPv4		IPv6	
		Frecuencia	Porcentaje	Frecuencia	Porcentaje
TCP	SSL	10.273.141	92,91%	0	0,00%
	HTTP	691.052	6,25%	0	0,00%
	Otros	93.489	0,85%	0	0,00%
	Total	11.057.682	100,00%	0	0,00%
UDP	MDNS	192.704	41,63%	98.118	81,07%
	SSDP	96.851	20,92%	4.425	3,66%
	DNS	56.321	12,17%	0	0,00%
	BOOTSTRAP	24.938	5,39%	0	0,00%
	GQUIC	3.498	0,76%	0	0,00%
	NETBIOS	0	0,00%	0	0,00%
	Otros	117.044	19,14%	18.479	15,27%
	Total	462.920	100,00%	121.022	100,00%

Con respecto al tráfico de la red alámbrica (tabla 3.2), podemos observar que IPv4 representa el 97% del total del tráfico, comparado con IPv6 que apenas es del 3%. De igual manera, el tráfico de las aplicaciones sobre TCP (HTTP y SSL) equivale al 91.42% versus el 8.42% de aplicativos que utilizan UDP (GQUIC y MDNS). De la tabla 3.3 podemos deducir que, en la red inalámbrica de campus, el tráfico IPv4 es muy superior al de IPv6 (98.26% versus 1.74%). Sobre IPv4, el tráfico de aplicativos TCP representa un 95.93% versus 4.02% de los aplicativos UDP. En este ámbito, HTTP y SSL representan el 99.15% del total del tráfico TCP. Tráfico de otros aplicativos no es

significativo. En términos generales el tráfico en ambos escenarios tiene un comportamiento similar, predominan protocolos como IPv4, TCP, SSL y HTTP. Se muestran datos obtenidos para los diferentes protocolos de aplicación descritos en la tabla 2.1.

Para el análisis del tráfico en función de la longitud de paquete, hemos clasificado a los paquetes en intervalos de 10 bytes (ejemplo: 0-10, 11-20, 11-30, etc.). Esta variable normalmente toma valores entre 40 y 1.500 bytes. La figura 3.4 nos muestra la probabilidad del tráfico alámbrico e inalámbrico en estos intervalos. Podemos observar que tanto el tráfico alámbrico como el inalámbrico tienen un comportamiento bimodal. El tráfico alámbrico sigue una frecuencia acumulada del 48.32% de paquetes alrededor de 60 bytes, y 38.42% alrededor de los 1.500 bytes. Con respecto al tráfico inalámbrico de campus, el 37.78% de paquetes se encuentra alrededor de 60 bytes, y un 50.08% alrededor de los 1.300 bytes. Existe una diferencia en cuanto a la longitud máxima de un paquete entre ambos escenarios, lo que nos lleva a definir modelos de tráfico diferentes para estos patrones de paquetes de red alámbrica e inalámbrica. Esto puede apreciarse mediante los diagramas de Pareto en las figuras 3.5 y 3.6 para ambos escenarios, con una diferencia marcada para la distribución de los paquetes grandes.

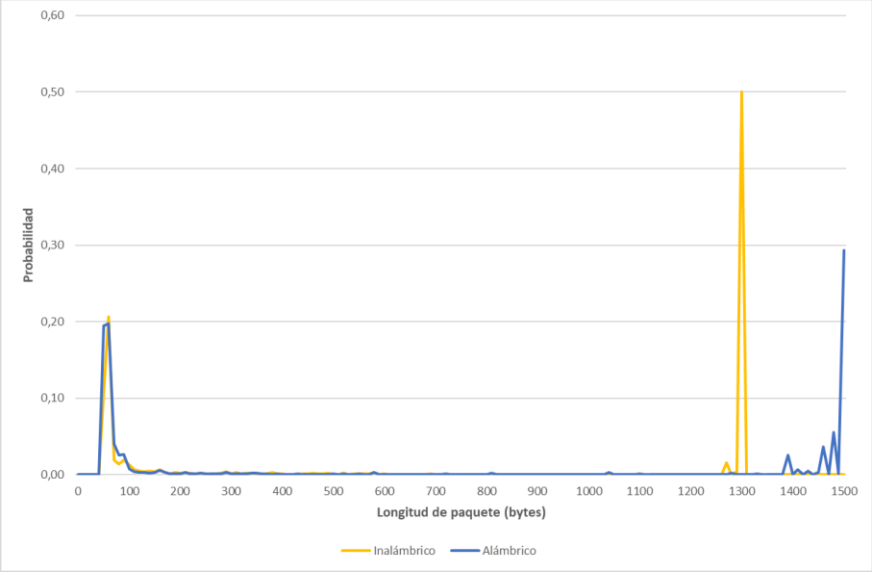


Figura 3.4. Tráfico comparativo red híbrida por longitud de paquete

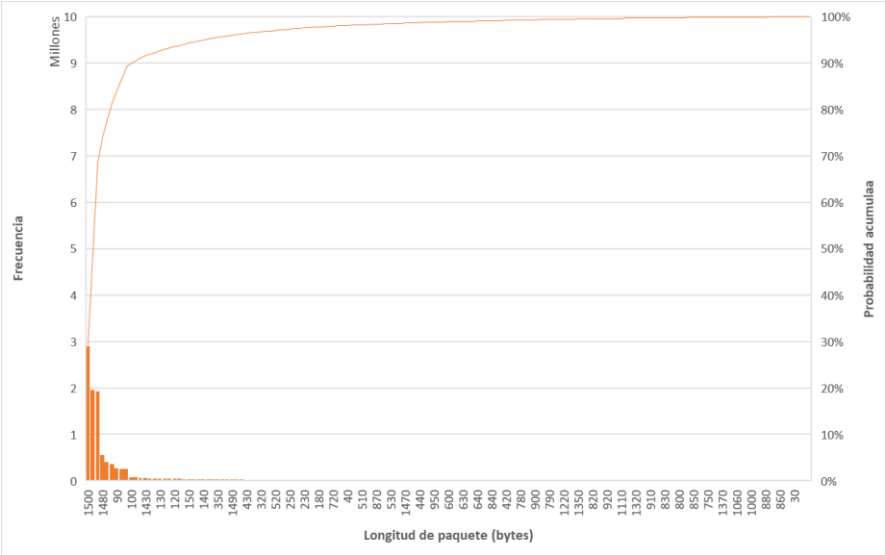


Figura 3.5. Diagrama de Pareto del tráfico alámbrico

Se puede observar que el tráfico IPv4 es más relevante que IPv6; que el protocolo TCP es más significativo que UDP; y además que las aplicaciones HTTP y SSL son las más representativas en ambos escenarios. Si modelamos estas tendencias, con toda seguridad obtendremos una estimación muy cercana al tráfico total según cada escenario. Las figuras 3.7 a 3.10 reflejan el comportamiento bimodal del protocolo IPv4 sobre la red alámbrica e inalámbrica, así como del protocolo TCP sobre el que trabajan aplicativos basados en SSL y HTTP, para el tráfico de la red alámbrica e inalámbrica, respectivamente.

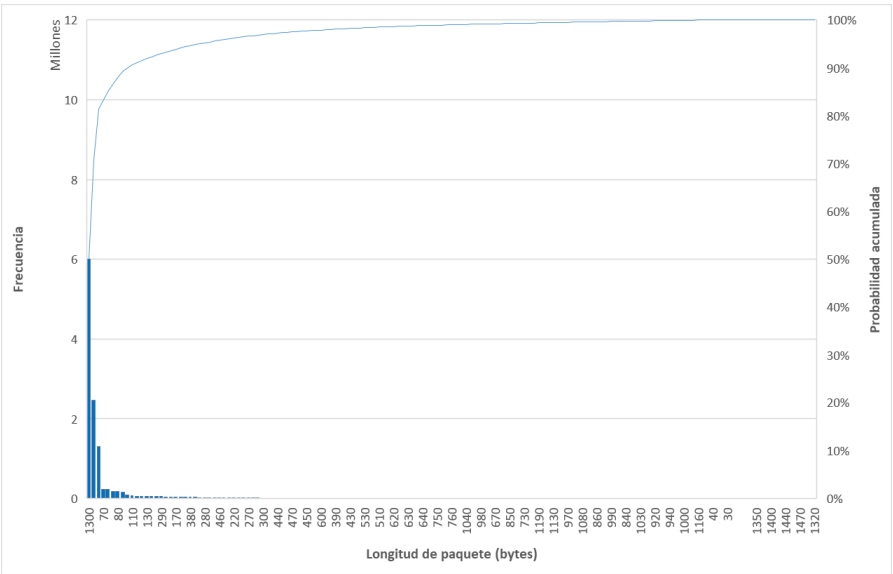


Figura 3.6. Diagrama de Pareto del tráfico inalámbrico

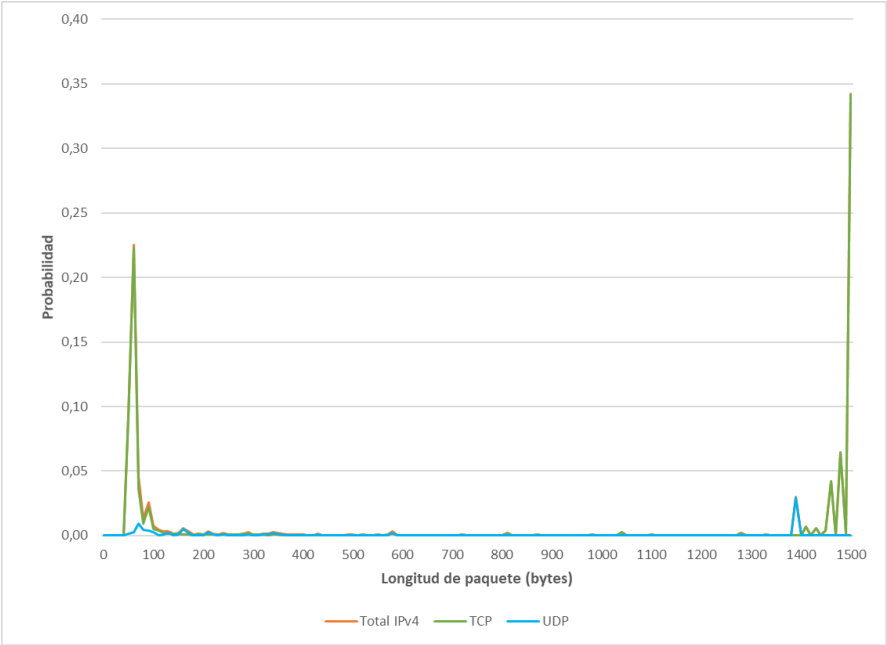


Figura 3.7. Distribución de tráfico IPv4 red alámbrica

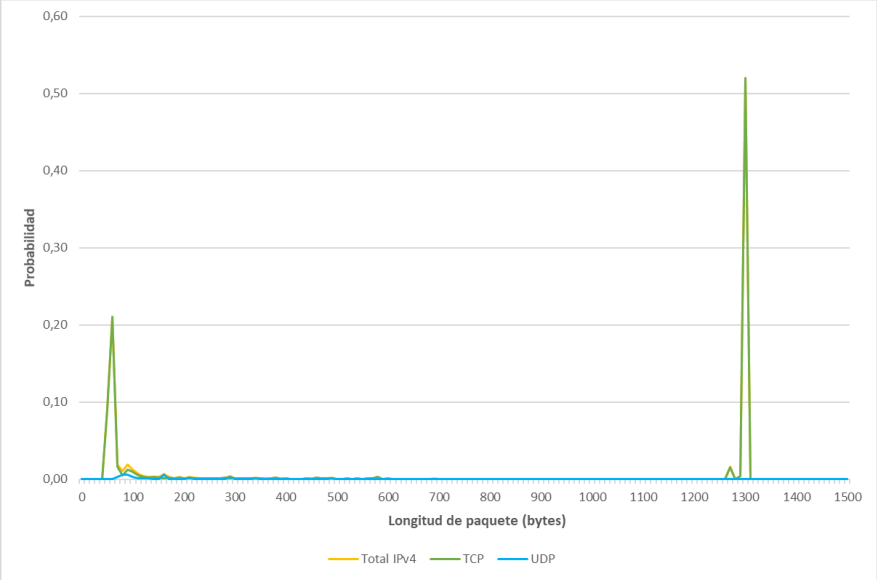


Figura 3.8. Distribución de tráfico IPv4 red inalámbrica

De acuerdo con nuestro segundo escenario, y según las estadísticas de statcounter sobre el sistema operativo móvil más utilizado en Ecuador (julio de 2019), el 86.96% de los dispositivos móviles usan Android, el 11.72% usa iOS y el 1.32% usa otros sistemas operativos, como Windows, Nokia, Blackberry, etc. En el uso de aplicaciones, mostramos estadísticas en el operador de red LTE más grande de nuestro país (ver figura 3.11). para el entorno de la red de campus, aplicamos una encuesta en línea a una muestra de 380 usuarios (tamaño de la población de 6,000 usuarios, con un nivel de confianza del 95% y

un margen de error del 5%) y encontramos que las aplicaciones más comunes son: en las redes sociales, WhatsApp, Facebook, Instagram y Twitter. Para streaming, YouTube. Y para la productividad: Drive, Gmail y Google Search. Otras aplicaciones utilizadas son: Messenger, Snapchat, Spotify, Netflix, Tinder.

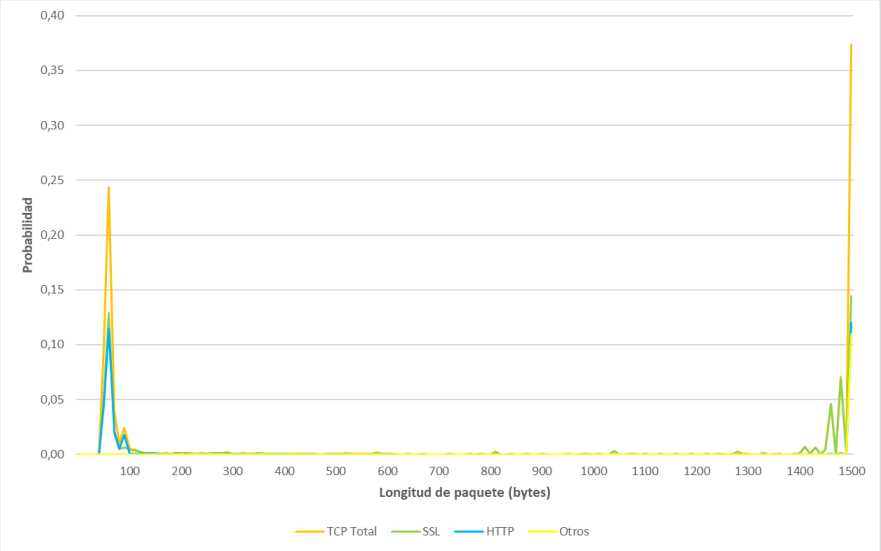


Figura 3.9. Distribución TCP por puertos sobre red alámbrica

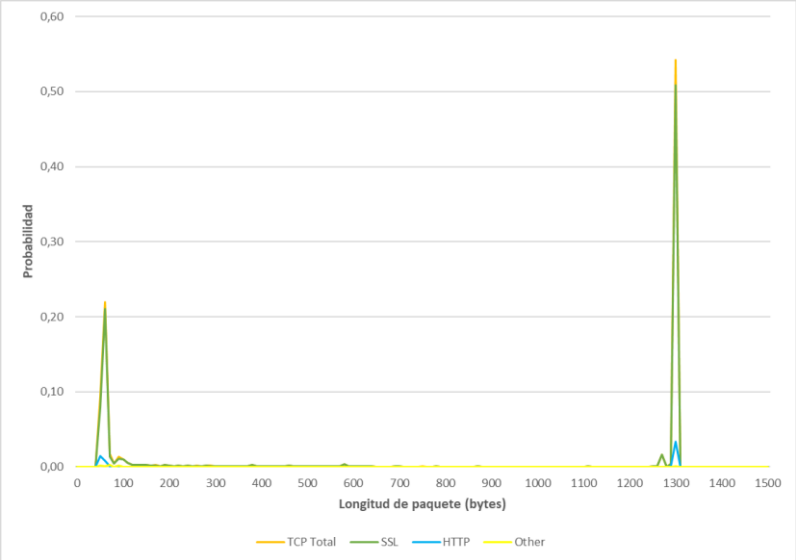


Figura 3.10. Distribución TCP por puertos sobre red inalámbrica

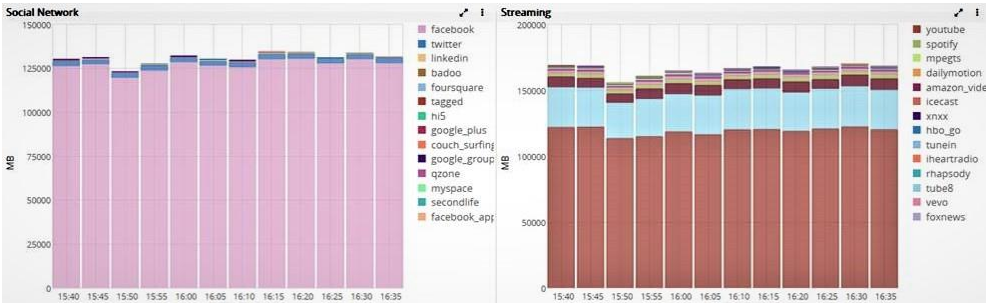


Figura 3.11. Aplicaciones más comunes en un operador LTE en Ecuador

Se analiza la longitud variable de los paquetes generados por las aplicaciones típicas en dispositivos móviles. Las Figuras 3.12 – 3.19 muestran la probabilidad de la longitud de paquete por aplicación móvil analizada en este escenario.

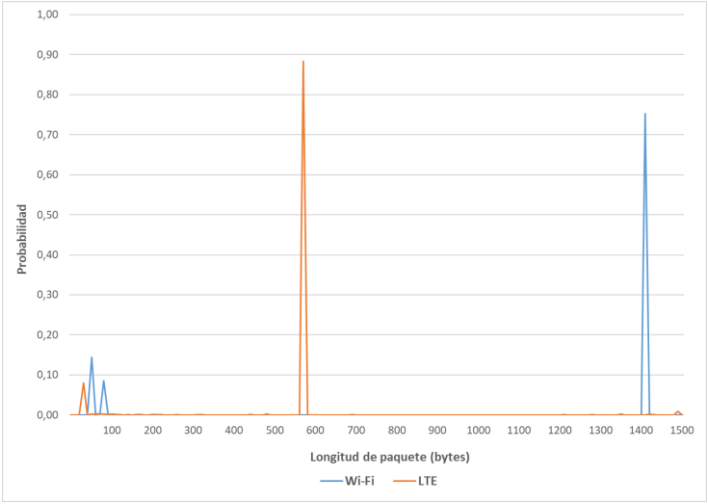


Figura 3.12. Patrón de paquetes de la App Google Drive

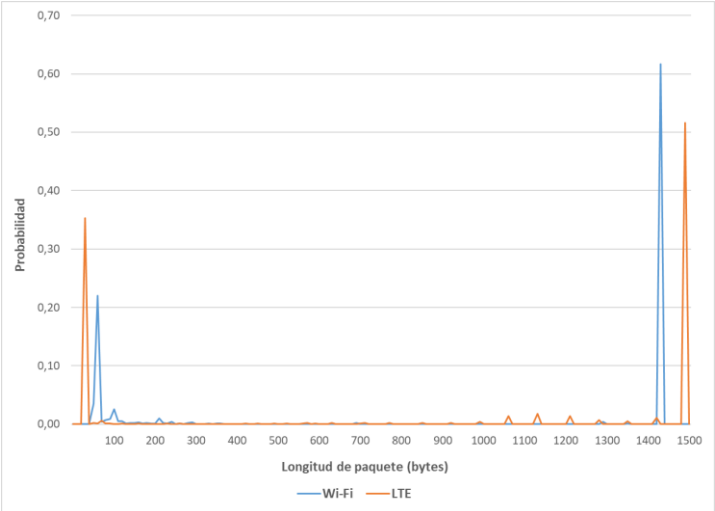


Figura 3.13. Patrón de paquetes de la App Facebook

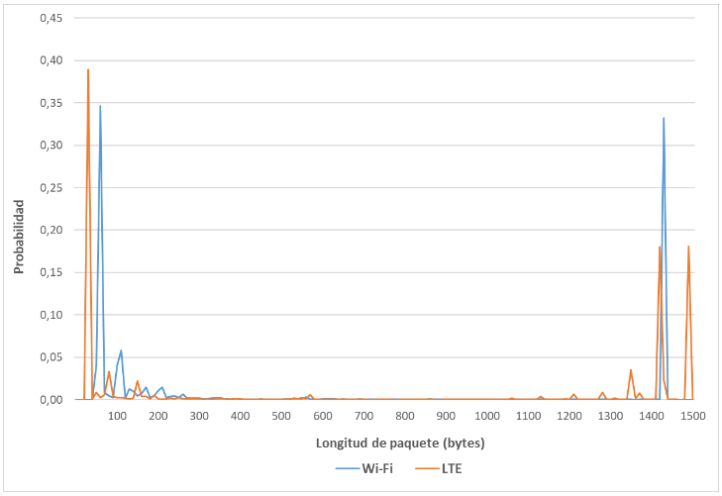


Figura 3.14. Patrón de paquetes de la App Google search

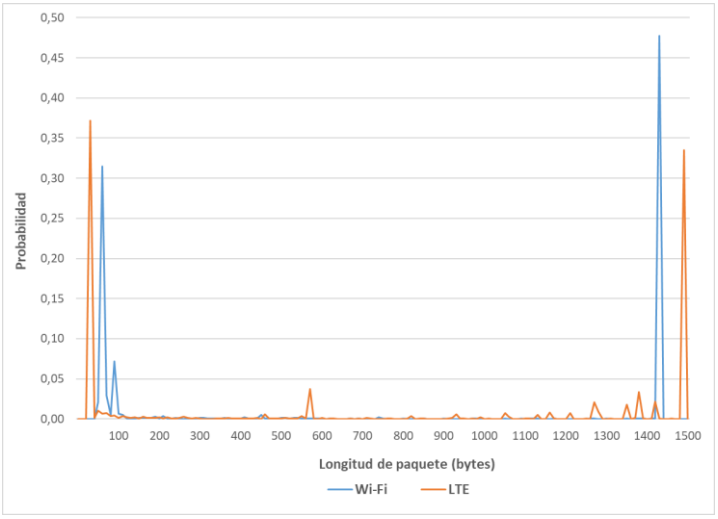


Figura 3.15. Patrón de paquetes de la App Google mail

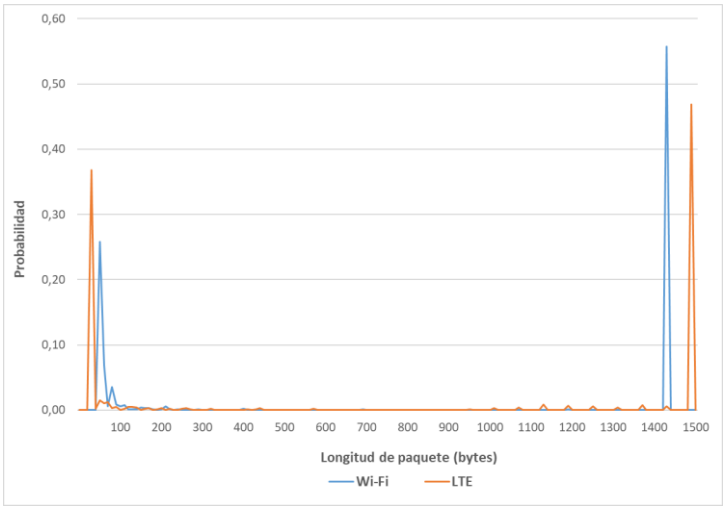


Figura 3.16. Patrón de paquetes de la App Twitter

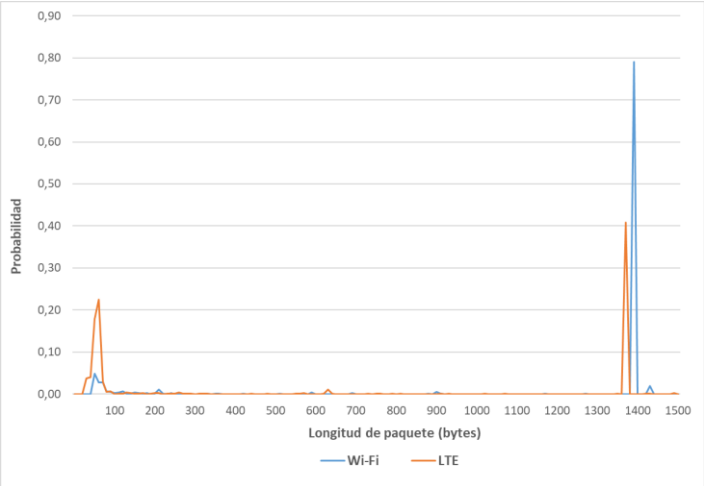


Figura 3.17. Patrón de paquetes de la App YouTube

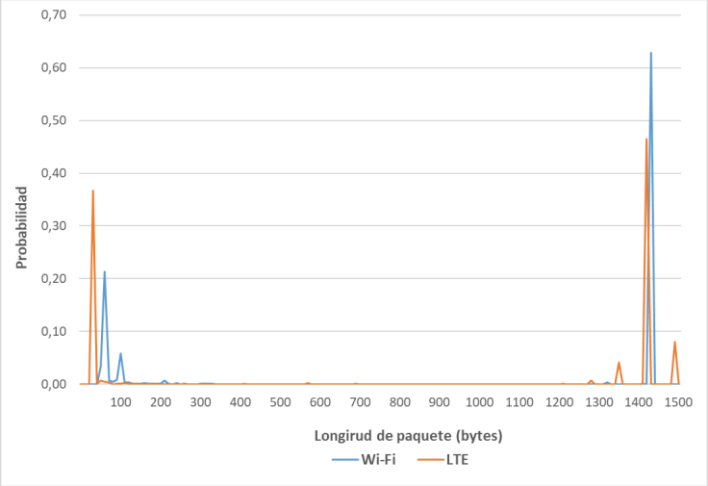


Figura 3.18. Patrón de paquetes de la App WhatsApp

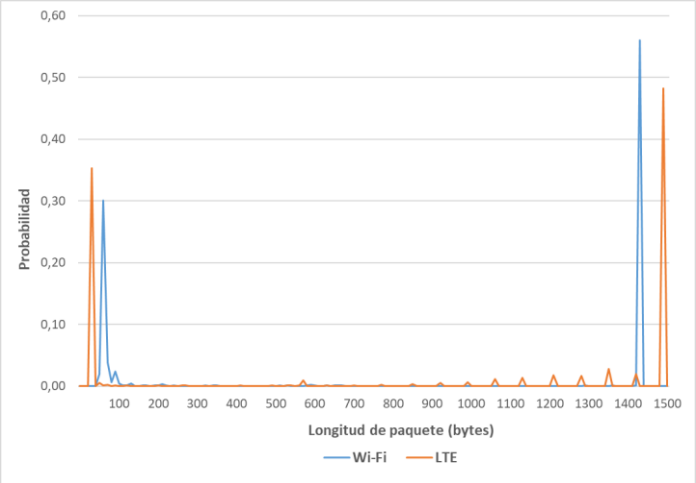


Figura 3.19. Patrón de paquetes de la App Instagram

En la Figura 3.12, se puede observar que hay una distribución de tráfico bimodal para Google Drive App en ambos escenarios, similar en paquetes pequeños pero

diferentes en paquetes grandes. Mediante la red Wi-Fi, esta aplicación tiene un 23% de paquetes con una longitud alrededor de 60 bytes y un 75% con alrededor de 1410 bytes. Para la red LTE, se puede observar que los paquetes pequeños representan 8% alrededor de 30 bytes, mientras que los paquetes grandes contribuyen 88% alrededor de 570 bytes. Esta es una diferencia importante en el comportamiento de esta aplicación y en los modelos para estimar este tráfico real en ambos escenarios.

En la Figura 3.13, podemos ver que hay una distribución de tráfico bimodal para la aplicación de Facebook en ambos escenarios, similar en paquetes pequeños y grandes. A través de la red Wi-Fi, esta aplicación tiene un 31% de paquetes con una longitud alrededor de 60 bytes y un 62% con alrededor de 1430 bytes. Para la red LTE, puede observar que los paquetes pequeños representan el 35% alrededor de 30 bytes, mientras que los paquetes grandes contribuyen con el 52% alrededor de 1490 bytes. El comportamiento de esta aplicación es similar para otras aplicaciones como Google search, Twitter, YouTube, WhatsApp e Instagram, y sobre los modelos que estiman este tráfico real en ambos escenarios.

Finalmente, en la Figura 3.15, podemos observar que existe una distribución de tráfico trimodal para la aplicación de correo electrónico a través de la red LTE, con un 37% de paquetes (pequeños) alrededor de 30 bytes, un 4% alrededor de 570 bytes (medios) y un 34% para paquetes grandes alrededor 1490 bytes. Esta aplicación a través de la red Wi-Fi tiene una distribución bimodal.

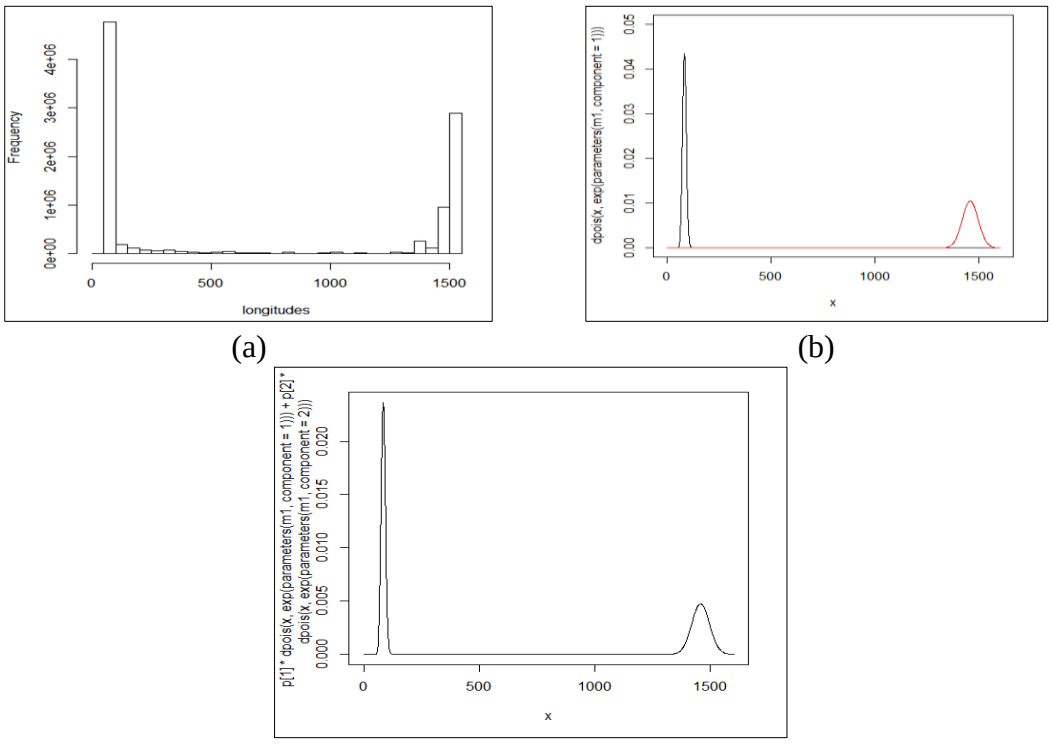
3.4. Estimación de modelos de tráfico

De acuerdo con los datos analizados en los dos escenarios estimamos varios modelos de tráfico utilizando la función de distribución de probabilidades de Poisson. Para el primer escenario de red híbrida, asociado al tráfico total, y a los protocolos con mayor significancia como son IPv4 y aplicativos sobre TCP. Los modelos obtenidos se calculan tanto en la red alámbrica como en la red inalámbrica de campus, debido a que la tendencia observada es similar.

El modelo ajustado para el tráfico total de la red alámbrica es una mezcla de dos distribuciones Poisson con parámetros $\lambda_1 = 84.38$ y $\lambda_2 = 1457.11$. La probabilidad de

que la longitud de un paquete pertenezca a la primera distribución es 0.545, mientras que para la segunda distribución la probabilidad de que un paquete siga esa distribución es de 0.455. Al final la ecuación que representa este modelo resulta de la suma de dos distribuciones poisson como se indica en (3.1). La figura 3.20 nos muestra el modelo obtenido a partir del histograma de los datos clasificados por longitud (a), y la estimación del modelo (b) y (c). La primera distribución se representa con color negro, mientras que la segunda distribución se representa con color rojo. El modelo estimado es la suma de ambas distribuciones.

$$P(X = x) = 0.545 * \frac{e^{-84.38}84.38^{x}}{x!} + 0.455 * \frac{e^{1457.11}1457.11^{x}}{x!} \qquad (3.1)$$



(a) (b)

(c)

Figura 3.20. Modelo de Poisson para tráfico total de la red alámbrica

Para el tráfico IPv4 de la red alámbrica el modelo ajustado consta igualmente de dos distribuciones poisson con parámetros $\lambda_1 = 90.61$ y $\lambda_2 = 1458.72$. La probabilidad de que la longitud de un paquete pertenezca a la primera distribución es 0.469, mientras que para la segunda distribución la probabilidad de que un paquete siga esa distribución es de 0.531. El modelo es el resultado de la suma de dos distribuciones poisson como se indica en (3.2), y su estimación en la figura 3.21.

$$P(X = x) = 0.469 * \frac{e^{-90.61}90.61^{x}}{x!} + 0.531 * \frac{e^{1458.72}1458.72^{x}}{x!} \qquad (3.2)$$

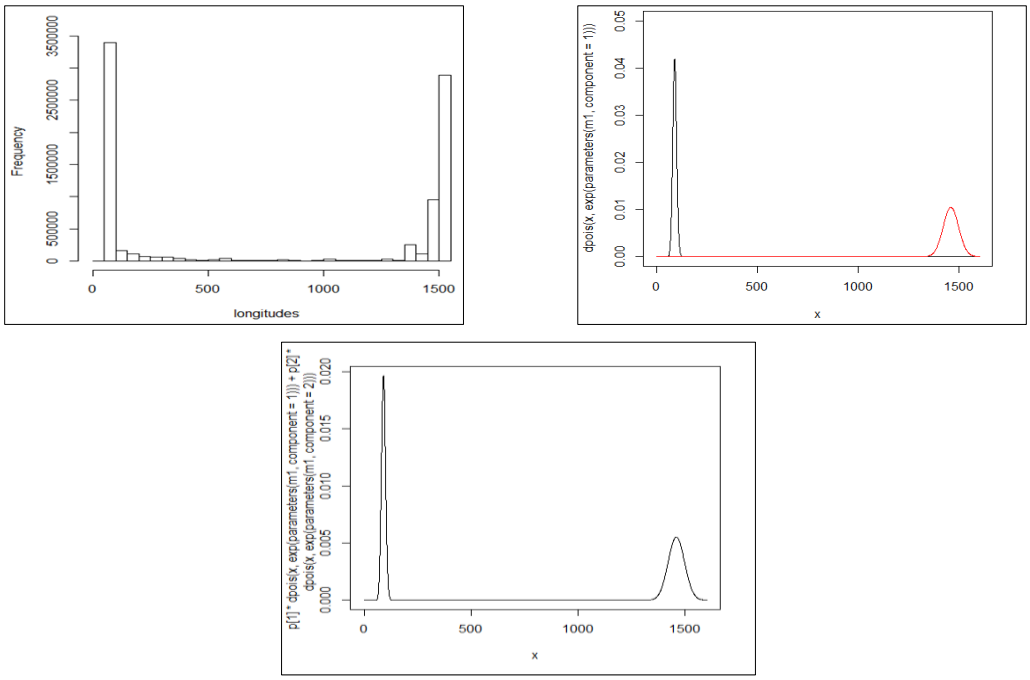


Figura 3.21. modelo de Poisson para tráfico IPv4 de la red alámbrica

Para el tráfico de IPv6 en la red alámbrica, el modelo tiene parámetros $\lambda_1 = 1083.92$ y $\lambda_2 = 103.86$. La probabilidad de que la longitud de un paquete pertenezca a la primera distribución es 0.0505, mientras que para la segunda distribución la probabilidad es 0.9495, como se muestra en (3.3).

$$P(X = x) = 0.0505 * \frac{e^{-1083.92}1083.92^{x}}{x!} + 0.9495 * \frac{e^{103.86}103.86^{x}}{x!} \qquad (3.3)$$

Adicionalmente, se presentan modelos para el tráfico de protocolos TCP and UDP, sobre IPv4 and IPv6 en la red alámbrica. La tabla 3.4 resume los parámetros de estos modelos, donde λ_1 representa la ocurrencia promedio en el intervalo 1, λ_2 representa la ocurrencia promedio en el intervalo 2, P_1 es la probabilidad de que un paquete siga la primera distribución y P_2 la probabilidad de que un paquete siga la segunda distribución. Para IPv6 solo es necesaria una distribución para ajustarse a los datos. Las figuras 3.22 y 3.23 muestran la simulación de los modelos obtenidos, y sus ecuaciones en (3.4) (3.5) (3.6) y (3.7). La tabla 3.5 presenta los parámetros para las aplicaciones, clasificadas por puerto, que contribuyen principalmente al tráfico total de la red.

$$P(X = x) = 0.544 * \frac{e^{-1467.92}1467.92^{x}}{x!} + 0.456 * \frac{e^{81.21}81.21^{x}}{x!} \quad (3.4)$$

$$P(X = x) = 0.611 * \frac{e^{-169.84}169.84^{x}}{x!} + 0.389 * \frac{e^{1327.37}1327.37^{x}}{x!} \quad (3.5)$$

$$P(X = x) = \frac{e^{-94.99}94.99^{x}}{x!} \quad (3.6)$$

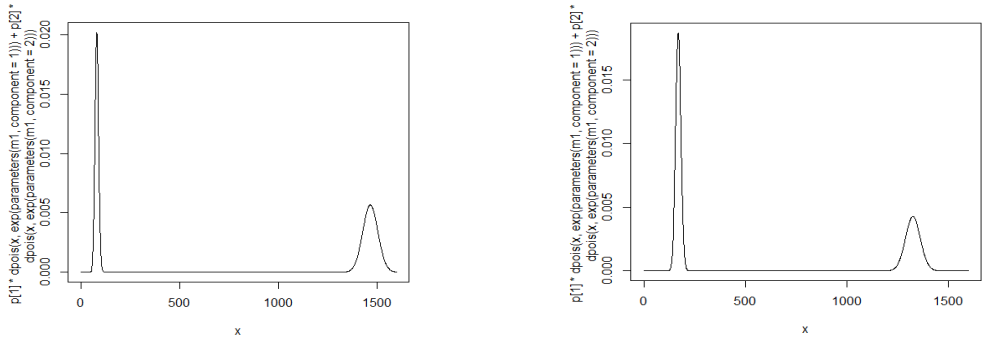
$$P(X = x) = \frac{e^{-305.12}305.12^{x}}{x!} \quad (3.7)$$

Tabla 3.4. Parámetros de Poisson para tráfico por protocolos red alámbrica

Protocol		λ_1	λ_2	P_1	P_2
IPv4	TCP	1467.92	81.21	0.544	0.456
	UDP	169.84	1327.37	0.611	0.389
IPv6	TCP	94.99	-	1	-
	UDP	305.12	-	1	-

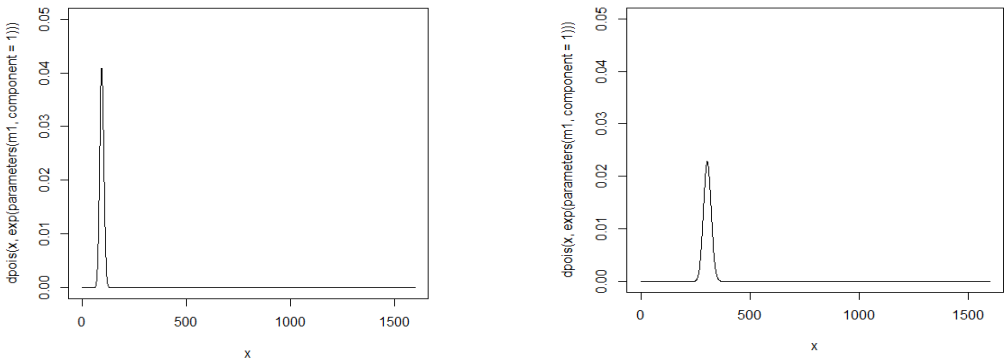
Tabla 3.5. Parámetros de Poisson para tráfico por puertos de aplicación red alámbrica

Protocol			λ_1	λ_2	P_1	P_2
IPv4	TCP	HTTP	1496.52	-	1	-
		SSL	87.93	1437.97	0.444	0.556
	UDP	GQUIC	1383.01	79.65	0.807	0.193
		MDNS	129.25	422.95	0.61	0.31
IPv6	TCP	SSL	94.36	-	1	-
	UDP	MDNS	212.75	-	1	-



(a) TCP (b) UDP

Figura 3.22. Modelo de Poisson para tráfico IPv4 por protocolos de la red alámbrica



(a) TCP (b) UDP

Figura 3.23. Modelo de Poisson para tráfico IPv6 por protocolos de la red alámbrica

Adicionalmente, respecto al primer escenario, se muestran los modelos para el tráfico total y de IPv4 de la red inalámbrica de campus en (3.8) y (3.9). La estimación del tráfico de la red inalámbrica de campus se muestra en la figura 3.24, y el modelo estimado para el tráfico SSL sobre TCP de este dataset en la figura 3.25.

$$P(X = x) = 0.448 * \frac{e^{-93.22}93.22^{x}}{x!} + 0.552 * \frac{e^{1270.11}1270.11^{x}}{x!} \quad (3.8)$$

$$P(X = x) = 0.409 * \frac{e^{-93.42}93.42^{x}}{x!} + 0.591 * \frac{e^{1267.86}1267.86^{x}}{x!} \quad (3.9)$$

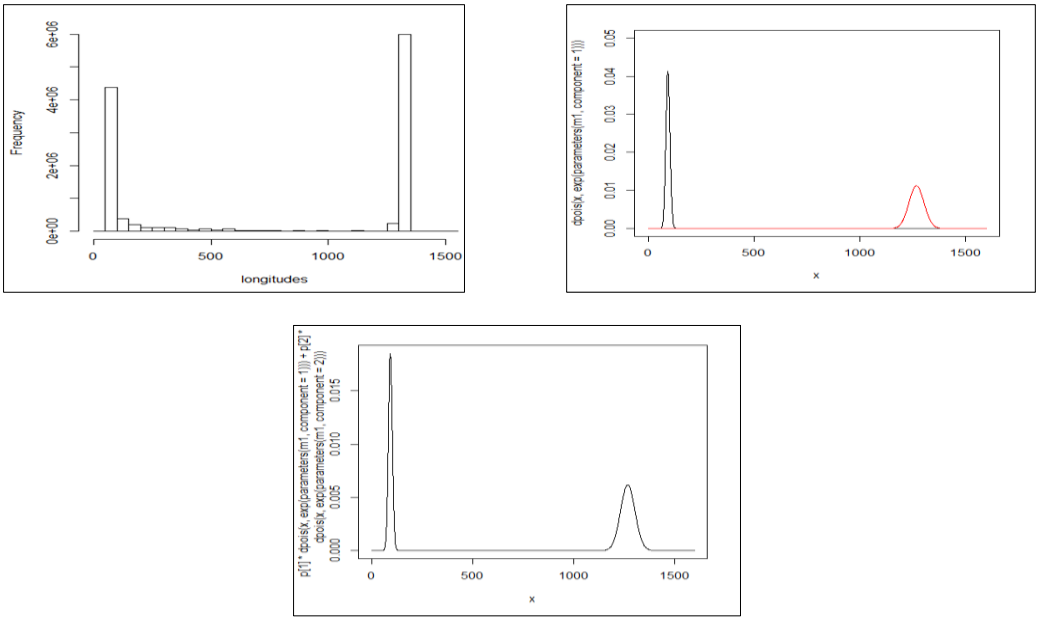


Figura 3.24. Modelo de Poisson para tráfico total de la red inalámbrica

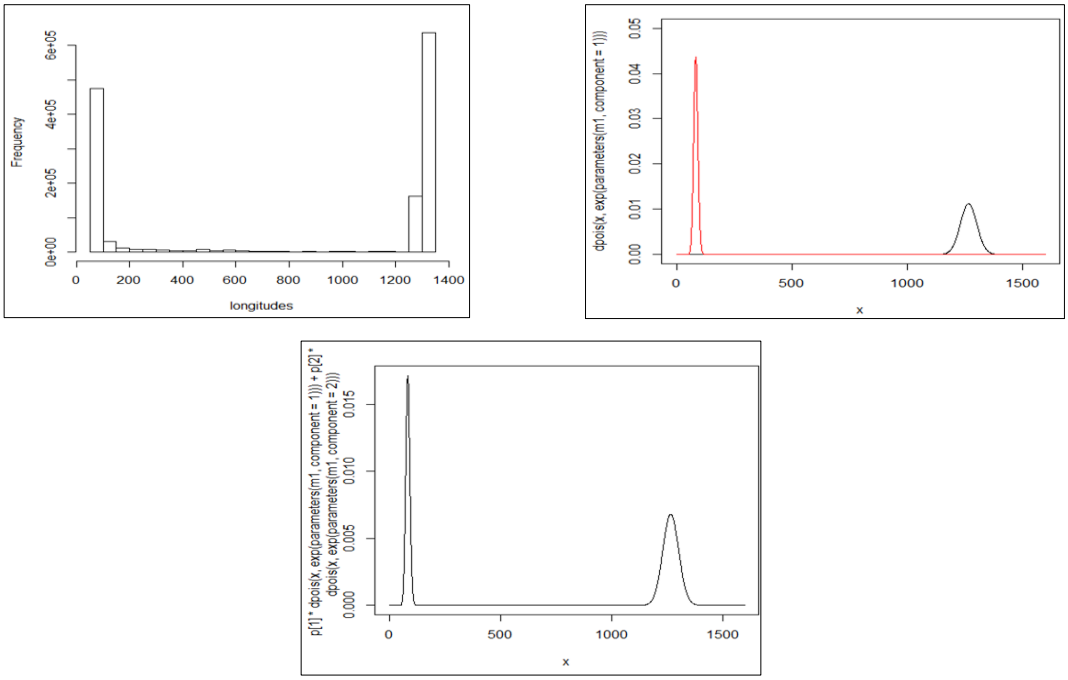


Figura 3.25. Modelo de Poisson para tráfico SSL sobre TCP de la red inalámbrica

Para el tráfico inalámbrico del primer escenario, se presentan modelos para protocolos TCP and UDP, sobre IPv4 e IPv6. La tabla 3.6 resume los parámetros de los modelos por protocolos. La tabla 3.7 muestra los parámetros para las aplicaciones que contribuyen de forma significativa al tráfico total de la red. Como se describe en la table 3.2, aplicativos sobre SSL y HTTP representan un 99.15% del tráfico sobre TCP, y este un 95.93% sobre el tráfico total de IPv4. El 98.26% del tráfico total de la red inalámbrica de campus es IPv4.

Tabla 3.6. Parámetros de Poisson para tráfico por protocolos red inalámbrica

Protocol		λ_1	λ_2	P_1	P_2
IPv4	TCP	1270.48	85.67	0.612	0.388
	UDP	418.09	119.06	0.212	0.788
IPv6	TCP	-	-	-	-
	UDP	405.15	143.58	0.482	0.518

Finalmente, para nuestro primer escenario, se demuestra que los datos se ajustan en efecto a una distribución de Poisson. Para ello se establece como hipótesis nula H_0: los datos siguen una distribución de Poisson; y se define como hipótesis alterna H_1: Los datos no siguen una distribución de Poisson. Para el análisis, se seleccionan 31 categorías de las longitudes de paquete y sus correspondientes frecuencias.

Tabla 3.7. Parámetros de Poisson para tráfico por puertos de aplicación red inalámbrica

Protocol			λ_1	λ_2	P_1	P_2
IPv4	TCP	HTTP	1296.14	85.50	0.685	0.315
		SSL	1267.80	83.61	0.607	0.393
	UDP	DNS	81.94	177.39	0.935	0.065
		MDNS	107.33	353.83	0.670	0.330
		SSDP	362.64	178.61	0.067	0.933
IPv6	UDP	MDNS	405.15	143.58	0.482	0.518
		SSDP	398.54	214.47	0.420	0.580

Se calcula la probabilidad de Poisson (p_i) para cada categoría, asumiendo que siguen dicha distribución, usando un $\lambda = 1$. Se define las frecuencias observadas (n_i) para cada categoría y a partir del producto $n_i \; x \; p_i$ se obtuvieron las frecuencias esperadas. Se toman en cuenta aquellas aportaciones al estadístico que son mayores o iguales a 5, de acuerdo con la fórmula indicada en (3.10), para medir la cantidad de divergencia entre la distribución de los datos de la muestra y la distribución de Poisson esperada. El valor obtenido del estadístico fue de 1.8353, por lo que a partir de la distribución de Pearson con 29 grados de libertad se buscó un p-valor tal que su probabilidad sea mayor o igual a 1.8353. Se obtuvo un p-valor aproximado de 0.175. Este valor se compara con un nivel de significancia típico de 0.05, observando que $p - valor \geq \alpha$, entonces se acepta la hipótesis nula y consideramos que los datos siguen una distribución de Poisson.

$$D = \sum_{i=1}^{31} \frac{(ni - npi)^2}{npi} \tag{3.10}$$

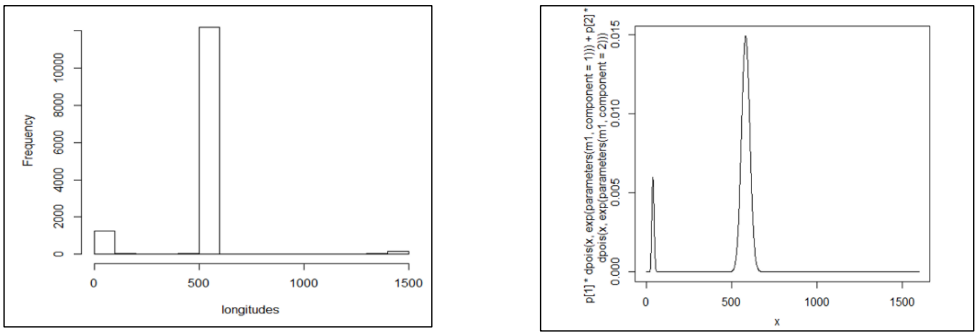


Figura 3.26. Distribución de Poisson para patrón de paquetes de la App Google Drive

Para el segundo escenario, se toma en cuenta el análisis del tráfico de red para estimar los modelos que utilizan la función de distribución de probabilidad de Poisson, basada en aplicaciones para dispositivos móviles.

Para el patrón de paquetes de la aplicación Drive que se presenta en la fig. 3.12, da como resultado un modelo ajustado como una mezcla de dos distribuciones de Poisson con parámetros $\lambda_1 = 39.82$ *y* $\lambda_2 = 583.79$. La probabilidad de que la longitud de un paquete pertenezca a la primera distribución es 0.095, mientras que para la segunda distribución la probabilidad de que un paquete siga esa distribución es 0.905. Finalmente, el modelo es el resultado de la suma de dos distribuciones de Poisson como en (3.11). En la figura 3.26 se muestra la simulación del modelo para el patrón de paquetes de la App de Google Drive.

$$P(X = x) = 0.095 * \frac{e^{-39.82}39.82^{x}}{x!} + 0.905 * \frac{e^{-583.79}583.79^{x}}{x!} \qquad (3.11)$$

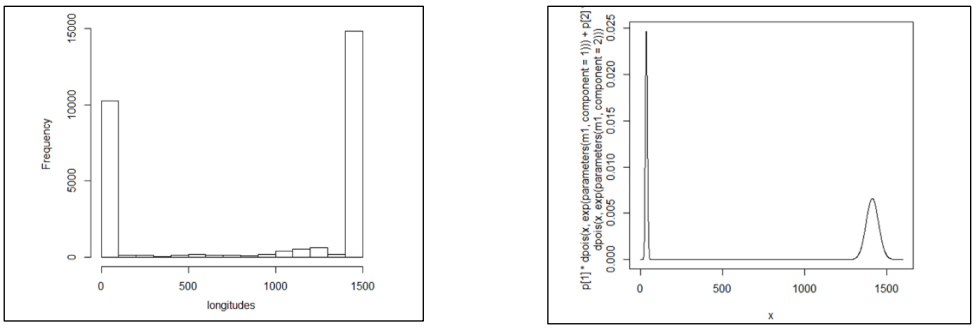


Figura 3.27. Distribución de Poisson para patrón de paquetes de la App Facebook

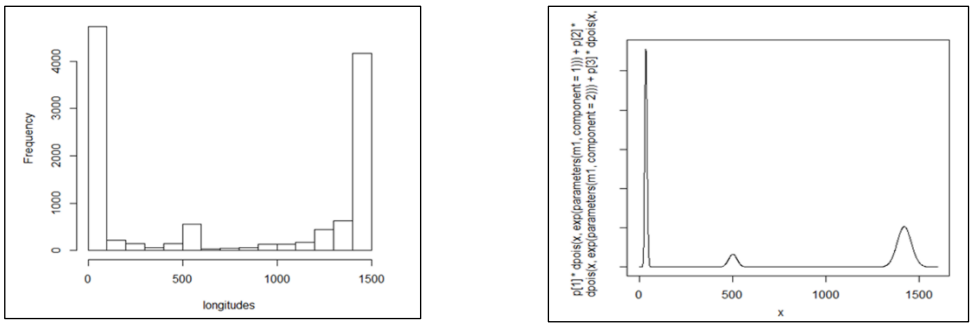


Figura 3.28. Distribución de Poisson para patrón de paquetes de la App Gmail

Para el patrón de paquetes de la App de Facebook, los parámetros son $\lambda_1 = 1414.78$ *y* $\lambda_2 = 36.93$. La probabilidad de que la longitud de un paquete pertenezca a la primera

distribución es 0.623, mientras que para la segunda distribución la probabilidad de que un paquete siga esa distribución es 0.377. El modelo se muestra en (3.12) y la simulación en la figura 3.27. Para Gmail, el modelo es el resultado de la suma de tres distribuciones de Poisson como se indica en (3.13), y la simulación en la figura 3.28. La tabla 3.8 muestra los parámetros para otras aplicaciones analizadas, donde λ_1 representa la ocurrencia promedio en el intervalo 1, λ_2 representa la ocurrencia promedio en el intervalo 2, P_1 es la probabilidad de que un paquete siga la primera distribución y P_2 es la probabilidad de que un paquete siga la segunda distribución. Y sus modelos en (3.14) (3.15) (3.16) (3.17) y (3.18). Adicionalmente la tabla 3.9 muestra los parámetros de las distribuciones de Poisson para los patrones de las Apps analizadas sobre la red Wi-Fi.

Tabla 3.8. Parámetros de distribución de Poisson para Apps sobre red LTE

App	λ1	λ2	P1	P2
Google	1383.51	57.75	0.485	0.515
Twitter	1434.44	49.59	0.540	0.460
YouTube	1324.33	62.31	0.444	0.556
WhatsApp	35.78	1414.33	0.398	0.602
Instagram	34.96	1405.98	0.370	0.630

Tabla 3.9. Parámetros de distribución de Poisson para Apps sobre red Wi-Fi

App	λ1	λ2	P1	P2
Drive	1406.74	65.69	0757	0.243
Facebook	87.86	1412.47	0.360	0.640
Google	1365.83	98.28	0.371	0.629
Email	84.61	1394.65	0.493	0.507
Twitter	1412.38	71.23	0.575	0.425
YouTube	104.41	1374.74	0.167	0.833
WhatsApp	79.78	1422.31	0.359	0.641
Instagram	70.82	1404.54	0.414	0.586

$$P(X = x) = 0.623 * \frac{e^{-1414.78}1414.78^{x}}{x!} + 0.377 * \frac{e^{-36.93}36.93^{x}}{x!} \qquad (3.12)$$

$$P(X = x) = 0.487 * \frac{e^{-1422.32}1422.32^{x}}{x!} + 0.091 * \frac{e^{-502.88}502.88^{x}}{x!} + 0.421 * \frac{e^{-36.56}36.56^{x}}{x!} \quad (3.13)$$

$$P(X = x) = 0.485 * \frac{e^{-1383.51}1383.51^{x}}{x!} + 0.515 * \frac{e^{-57.75}57.75^{x}}{x!} \quad (3.14)$$

$$P(X = x) = 0.540 * \frac{e^{-1434.44}1434.44^{x}}{x!} + 0.460 * \frac{e^{-49.59}49.59^{x}}{x!} \quad (3.15)$$

$$P(X = x) = 0.444 * \frac{e^{-1324.33}1324.33^{x}}{x!} + 0.556 * \frac{e^{-62.31}62.31^{x}}{x!} \quad (3.16)$$

$$P(X = x) = 0.398 * \frac{e^{-35.78}35.78^{x}}{x!} + 0.602 * \frac{e^{-1414.33}1414.33^{x}}{x!} \quad (3.17)$$

$$P(X = x) = 0.370 * \frac{e^{-34.96}34.96^{x}}{x!} + 0.630 * \frac{e^{-1405.98}1405.98^{x}}{x!} \quad (3.18)$$

Finalmente, proponemos una prueba de hipótesis para comparar el promedio de los paquetes en la red LTE versus la red Wi-Fi. En cada caso se propusieron dos contrastes de hipótesis, H_0: el promedio de los paquetes es igual; y en caso de que la hipótesis nula no sea cierta, se propone una hipótesis alternativa H_1: el promedio de los paquetes de la red Wi-Fi es mayor que los paquetes de la red LTE. La única aplicación que muestra que el promedio de los paquetes en ambos escenarios es el mismo, es la aplicación de correo electrónico. Para aplicaciones como Drive, Facebook, Twitter, YouTube y WhatsApp, el resultado fue que el paquete promedio de la red Wi-Fi es mayor que el de la red LTE. Para aplicaciones como Instagram y búsqueda de Google, el paquete promedio de la red Wi-Fi es más pequeño que la red LTE. A continuación, mostramos en la tabla 3.10 esta comparación para la aplicación Drive:

Tabla 3.10. Contraste de Hipótesis para la App Google Drive

Estimaciones	t	df	p-value	Evaluación de Hipótesis
Media de x = 1081.4132	129.54	28717	2.2e-16	H_0 es rechazada, el promedio de longitud de paquetes no es igual
Media de y = 532.3175 Intervalo de confianza: 95%	129.54	28717	1	H_1 no es rechazada, el promedio de la longitud de paquetes sobre la red Wi-Fi es mayor que sobre la red LTE

CAPÍTULO IV
MODELAMIENTO DEL RETARDO UNIDIRECCIONAL Y DE ENCOLAMIENTO

4.1. Metodología aplicada

En la segunda fase de la investigación, se propone analizar el comportamiento del retardo unidireccional, con énfasis en el retardo de encolamiento, sobre un escenario de red TCP/IP con topología punto a punto, y estimar su comportamiento mediante modelos predictivos basados en regresión polinómica,

Para el estudio del retardo, se utilizan los modelos de estimación de tráfico obtenidos en la primera fase de investigaciones basadas en distribuciones de Poisson [104-107], en función de la longitud de paquete, con la finalidad de generar tráfico sobre la topología propuesta, que simule la carga de enlace. De igual manera se generará tráfico de interés para el estudio del retardo, en cuyos paquetes se agregan marcas de tiempo (timestamp), sincronizadas por medio de NTP, y procesadas en los puntos de medición.

Se obtienen los valores aproximados de retardo de transmisión, de propagación, de procesamiento y encolamiento a lo largo de la topología planteada, y se procede a modelar el retardo unidireccional (OWD) y el retardo de encolamiento, para obtener valores de referencia o de línea base, que serán usados en la fase 3 de la investigación con el objetivo de comprobar si es posible validar la hipótesis de este trabajo.

4.2. Escenario de estudio

Para la estimación de los retardos, se utiliza el escenario de la figura 4.1, en el que se configuran 4 segmentos lógicos o subredes con la finalidad de simular una red empresarial, y que se utilizan para generar la carga del enlace serial punto a punto que representa una conexión hacia un proveedor de servicios de internet, y sobre el cual se toman las medidas de las marcas de tiempo para su posterior análisis.

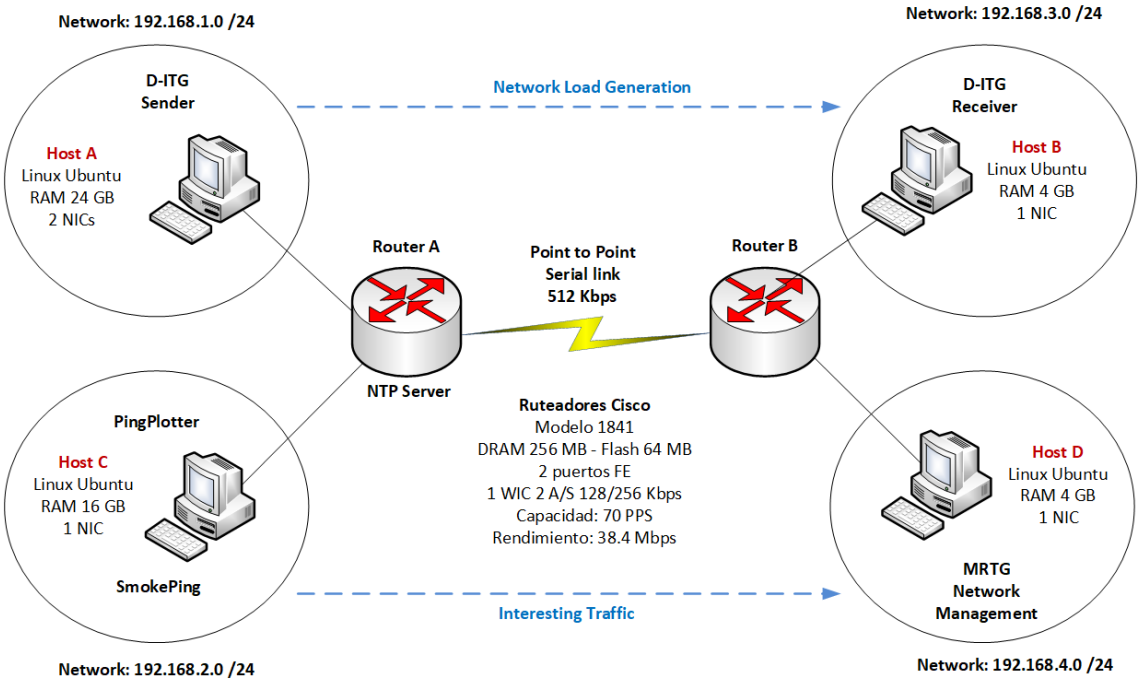


Figura 4.1. Escenario propuesto para estudio del retardo

Entre las redes 192.168.1.0 /24 y 192.168.3.0 /24 se genera el tráfico que permite generar la carga sobre el enlace punto a punto, y que sigue un patrón real en base a protocolos y/o aplicaciones, cuyos modelos estimados se presentan en el capítulo 3. Para esta tarea se utiliza una herramienta de código abierto y gratuita llamada D-ITG desarrollada por el grupo de investigación Traffic, del departamento de Ingeniería Eléctrica y de Tecnologías de la Información de la Universidad de Nápoles Federico II [103]. D-ITG permite generar tráfico siguiendo el comportamiento estocástico de dos variables aleatorias como lo son el tiempo de partida entre paquetes (IDT) y la longitud de paquete. Soporta generación de tráfico IPv4 e IPv6.

Para las mediciones de la utilización o carga de los enlaces se utiliza el software MRTG, un monitor de tráfico basado en el protocolo de administración de redes (SNMP), de código abierto y libre distribución.

En los ruteadores del escenario se configuran diferentes políticas de QoS con la finalidad de determinar su comportamiento en cuanto al manejo del retardo de encolamiento: WFQ (por defecto), FIFO, PQ (encolamiento por prioridad) y CBWFQ. Para el mismo escenario se configura cada política de QoS y se toman los datos para su posterior análisis. Esto se lo realiza para diferentes niveles de carga de enlace: 0 – 32 –

64 – 128 – 256 – 512 Kbps. Adicionalmente, uno de ellos se configura como servidor de NTP, para sincronizar a todos los equipos presentes en el escenario, y que permiten que el manejo de las marcas de tiempo en los paquetes de tráfico interesante sea oportuno.

Se instala el sniffer de red Wireshark para la captura de los paquetes a la entrada y salida del ruteador. Se mide OWD entre los dos equipos, y se validan dichas mediciones por medio de SmokePing y Ping Plotter, herramientas que permiten medir OWD y RTT, además de fluctuación de retardo, y paquetes perdidos. Los paquetes son generados con diferentes longitudes: 50, 100, 200, 300, etc., hasta 1500 bytes, y se lo hace para cada carga de enlace (0-32-64-128-256-512 Kbps) y para cada política de QoS considerada para el análisis de este trabajo (WFQ-FIFO-PQ-CBWFQ).

4.3. Resultados

La recolección de datos en el escenario de estudio se la realiza para varias políticas de QoS (4). Por cada una de ellas, se establecen diferentes niveles de carga (6). Por cada nivel se configuran 16 tamaños de paquete, entre 50 y 1500 bytes. Y para cada longitud se toman 11 muestras diferentes. La tabla 4.1 refleja las muestras para la carga de 256Kbps usando WFQ como política de QoS.

Tabla 4.1. Muestras de OWD para WFQ con carga de 256 Kbps (en milisegundos)

Longitud de paquete	Muestras										
	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11
50	12,85	13,44	15,01	15,95	16,38	17,00	17,05	17,19	17,42	17,68	17,95
100	18,20	18,90	19,56	20,12	20,81	21,30	21,91	22,80	24,10	25,24	28,10
200	26,50	27,65	29,51	30,30	31,05	31,70	32,65	33,10	35,02	37,48	42,40
300	37,20	37,78	39,40	40,88	41,38	42,20	43,80	44,17	47,32	49,21	53,50
400	48,35	49,44	49,98	51,23	52,65	53,35	53,92	54,46	55,34	56,78	59,90
500	58,55	49,90	52,30	57,43	58,67	60,21	61,34	62,05	63,54	66,40	71,21
600	67,00	68,20	69,78	70,65	71,95	73,00	74,10	76,34	77,54	80,21	93,25
700	76,95	78,30	79,80	81,05	81,80	83,85	84,80	86,75	88,42	90,12	109,60
800	87,75	89,44	89,98	90,45	91,26	91,95	92,67	93,01	93,98	95,34	97,25
900	94,50	98,66	102,54	104,33	108,01	116,65	119,23	124,29	132,45	135,66	152,75
1000	105,15	107,30	109,01	109,34	109,90	111,60	112,40	114,21	115,98	117,34	120,95
1100	113,75	120,39	132,34	135,90	137,00	138,05	139,10	144,64	147,51	153,88	182,85
1200	125,80	128,37	131,10	135,98	136,90	137,60	139,65	141,33	148,77	155,00	192,75
1300	132,95	134,89	137,44	138,01	138,74	140,45	144,50	147,12	151,56	157,11	180,95
1400	143,55	146,64	148,33	149,10	151,05	153,75	155,10	158,30	162,86	169,23	191,05
1500	154,30	158,92	162,78	164,23	166,37	168,60	170,20	173,21	177,67	180,10	218,65

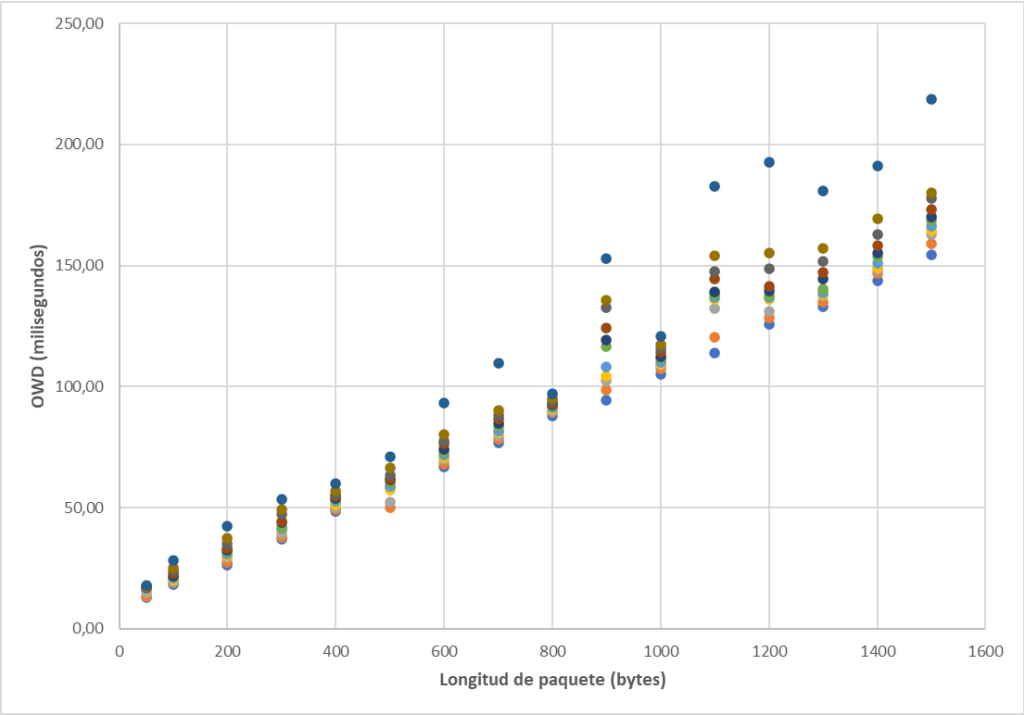


Figura 4.2. Dispersión de las muestras de OWD para WFQ con carga de 256 Kbps

En las tablas de datos, se obtienen el valor mínimo, máximo y la mediana de los valores. La figura 4.2 muestra la dispersión de los datos de la tabla 4.1. Para nuestro posterior análisis de regresión se utilizarán los valores de las medianas. La tabla 4.2 muestra los valores detallados de los componentes del retardo OWD considerando los valores de las medianas de OWD para WFQ con carga de 256 Kbps.

El valor de retardo por transmisión mostrado en la tabla 4.2 incluye los 3 retardos de transmisión del escenario (host origen, ruteador A y ruteador B). el valor de retardo por propagación incluye los 3 retardos de propagación del escenario (2 cables de red UTP y un cable serial para enlace punto a punto). El valor de retardo por procesamiento incluye los 3 valores del escenario (switch y los 2 ruteadores). Y el valor de retardo de encolamiento incluye los 2 retardos que agregan los 2 ruteadores. Como se indica en [61], los valores de retardo por propagación y procesamiento son muy pequeños, prácticamente despreciables para el cálculo de OWD. El valor de retardo por transmisión es constante para cada valor de longitud de paquete, y sigue un comportamiento lineal como se muestra en la figura 4.3. A partir de este punto, para cada muestreo de datos, se indicarán los valores de OWD y del retardo de encolamiento para cada política de QoS analizada.

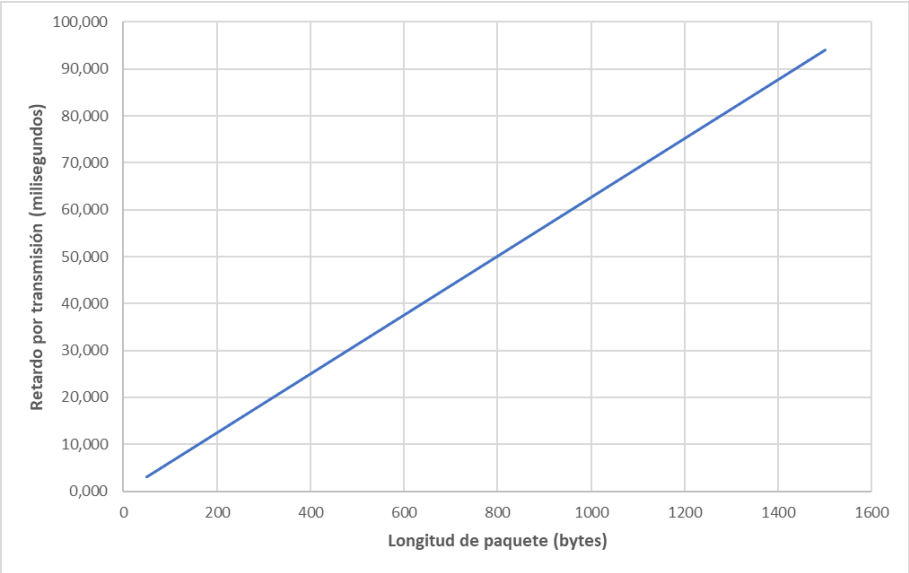


Figura 4.3. Retardo de transmisión

Tabla 4.2. Componentes de OWD para WFQ con carga de 256 Kbps (en milisegundos)

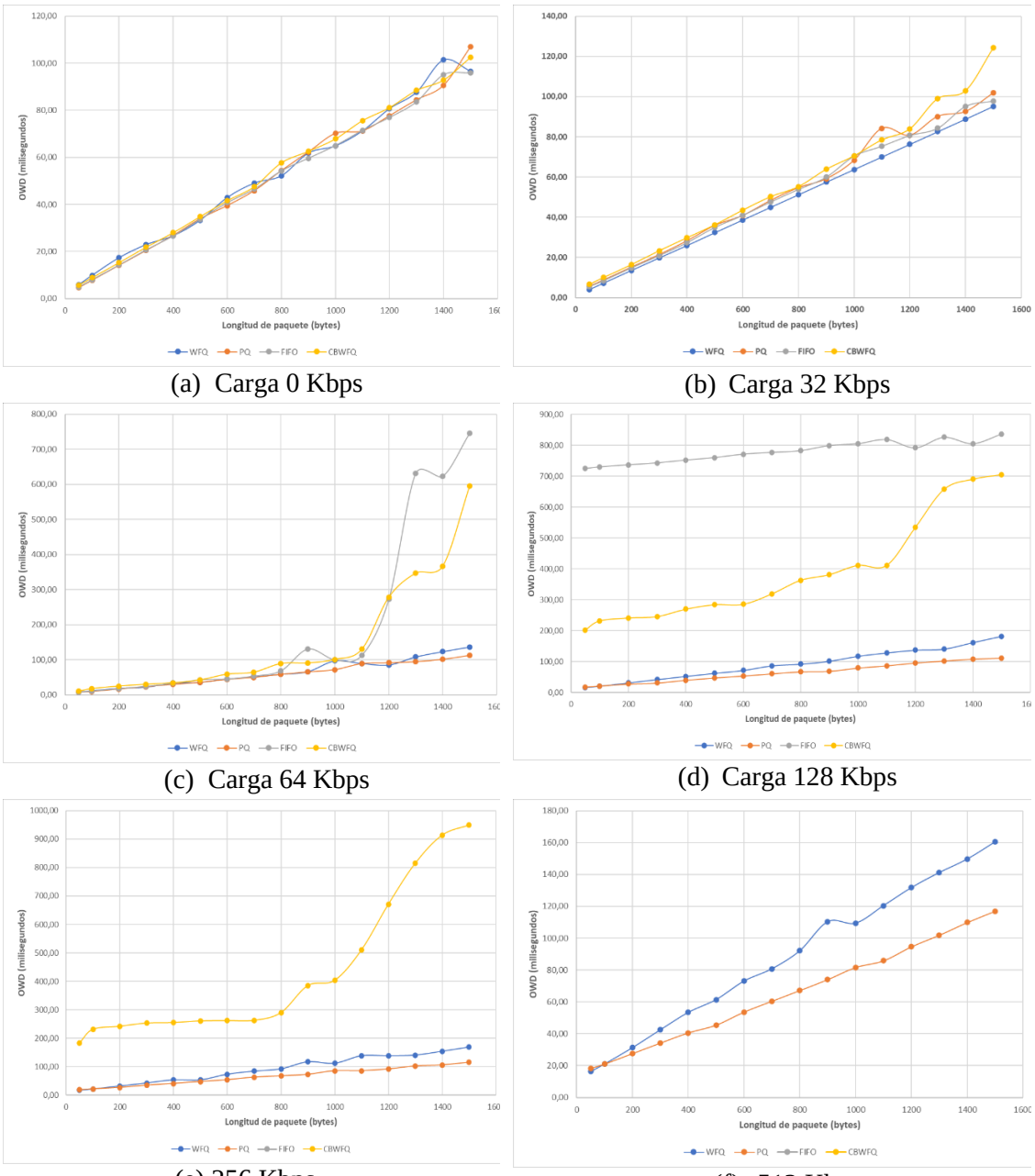
Longitud de paquete	Retardo por Transmisión	Retardo por Propagación	Retardo por Procesamiento	Retardo por Encolamiento	OWD
50	3,13	0,0001	0,048	13,8189	17,00
100	6,27	0,0001	0,048	14,9859	21,30
200	12,53	0,0001	0,048	19,1199	31,70
300	18,80	0,0001	0,048	23,3539	42,20
400	25,06	0,0001	0,048	28,2379	53,35
500	31,33	0,0001	0,048	21,9719	53,35
600	37,60	0,0001	0,048	35,3559	73,00
700	43,86	0,0001	0,048	39,9399	83,85
800	50,13	0,0001	0,048	41,7739	91,95
900	56,39	0,0001	0,048	60,2079	116,65
1000	62,66	0,0001	0,048	48,8919	111,60
1100	68,93	0,0001	0,048	69,0759	138,05
1200	75,19	0,0001	0,048	62,3599	137,60
1300	81,46	0,0001	0,048	58,9439	140,45
1400	87,72	0,0001	0,048	65,9779	153,75
1500	93,99	0,0001	0,048	74,5619	168,60

Tabla 4.3. OWD para CBWFQ por carga de enlace y longitud de paquete

Retardo Unidireccional (milisegundos)					
Longitud de paquete	**Carga de enlace**				
	0K	**32K**	**64K**	**128K**	**256K**
50	5,75	6,78	10,20	202,60	182,55
100	8,97	10,06	17,70	232,35	231,30
200	15,40	16,49	24,90	241,80	241,90
300	21,93	23,37	30,00	246,30	253,20
400	28,08	29,86	34,40	270,10	255,00
500	34,92	36,27	42,30	284,65	260,80
600	41,65	43,56	58,80	285,90	261,90
700	47,62	50,23	64,55	319,65	262,85
800	57,74	55,26	89,25	363,70	289,70
900	62,64	63,88	90,90	381,90	385,05
1000	67,90	70,56	100,95	412,10	403,50
1100	75,65	78,52	130,00	412,10	510,70
1200	81,18	83,96	278,40	534,85	671,20
1300	88,49	98,97	347,00	658,20	815,40
1400	92,88	102,92	367,10	690,90	914,20
1500	102,55	124,25	595,60	705,55	950,00

Tabla 4.4. Retardo de encolamiento para CBWFQ por carga de enlace y longitud de paquete

Retardo de encolamiento (milisegundos)					
Longitud de paquete	**Carga de enlace**				
	0K	**32K**	**64K**	**128K**	**256K**
50	1,286	1,801	3,509	99,709	89,684
100	1,328	1,872	5,693	113,018	112,493
200	1,410	1,953	6,160	114,610	114,660
300	1,541	2,261	5,577	113,727	117,177
400	1,483	2,376	4,644	122,494	114,944
500	1,769	2,448	5,461	126,636	114,711
600	2,002	2,956	10,578	124,128	112,128
700	1,856	3,162	10,320	137,870	109,470
800	3,181	2,539	19,537	156,762	119,762
900	3,097	3,716	17,229	162,729	164,304
1000	2,897	3,924	19,121	174,696	170,396
1100	3,338	4,773	30,513	171,563	220,863
1200	2,971	4,360	101,580	229,805	297,980
1300	3,493	5,600	132,747	288,347	366,947
1400	3,256	10,708	139,664	301,564	413,214
1500	4,256	15,105	250,781	305,756	427,981



(a) Carga 0 Kbps

(b) Carga 32 Kbps

(c) Carga 64 Kbps

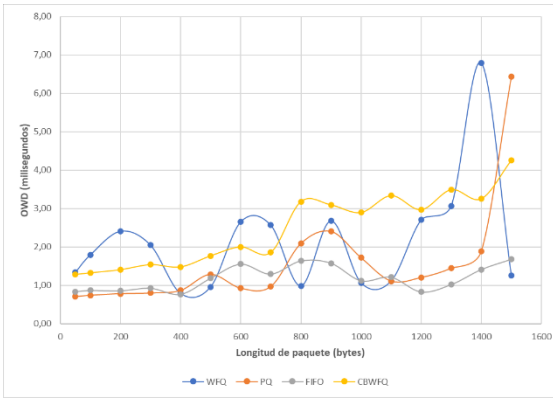
(d) Carga 128 Kbps

(e) 256 Kbps

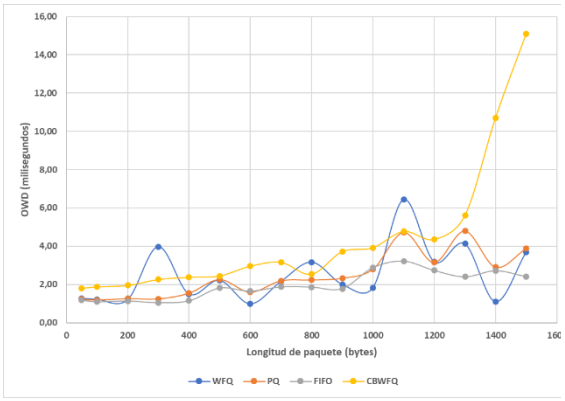
(f) 512 Kbps

Figura 4.4. Análisis de OWD por carga y técnica de QoS

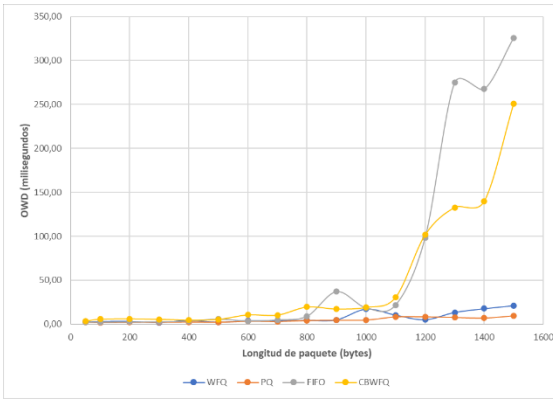
Para cada técnica de QoS, se aplica una configuración en la que se da cierto peso o prioridad a los paquetes de tráfico interesante, de tal forma que tengan preferencia de envío por la función de despacho de cada política. Eso no es aplicable en FIFO, donde la función de despacho solo se aplica en la medida que los paquetes llegan a la interfaz de salida. Las tablas 4.3 y 4.4 muestran los resultados obtenidos para la técnica CBWFQ.



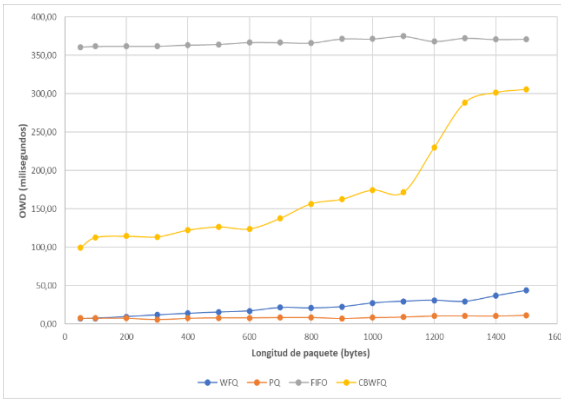
(a) Carga 0 Kbps



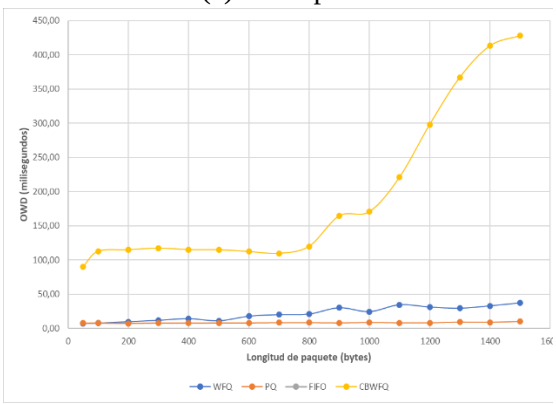
(b) 32 Kbps



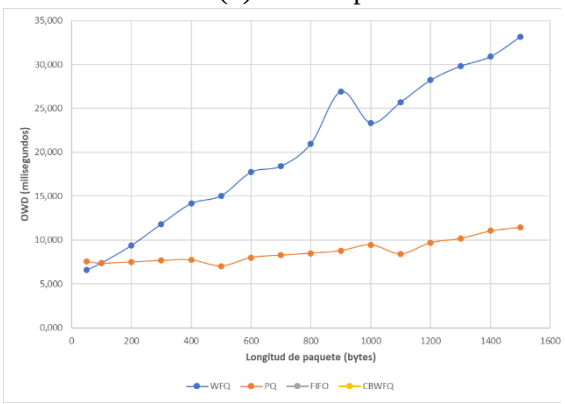
(c) 64 Kbps



(d) 128 Kbps

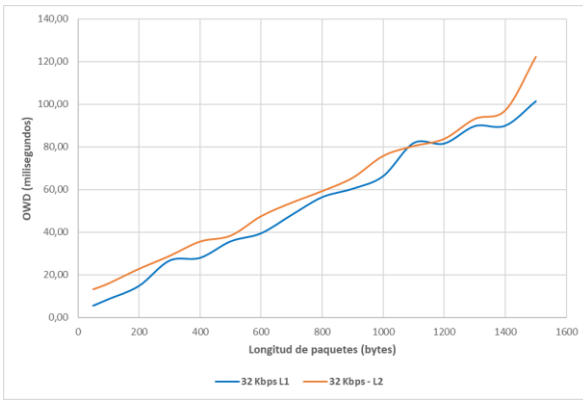


e) 256 Kbps

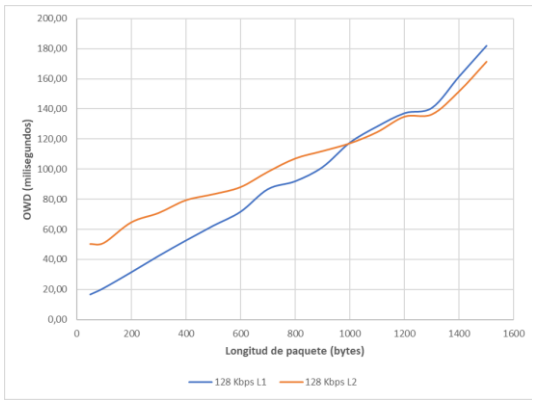


f) 512 Kbps

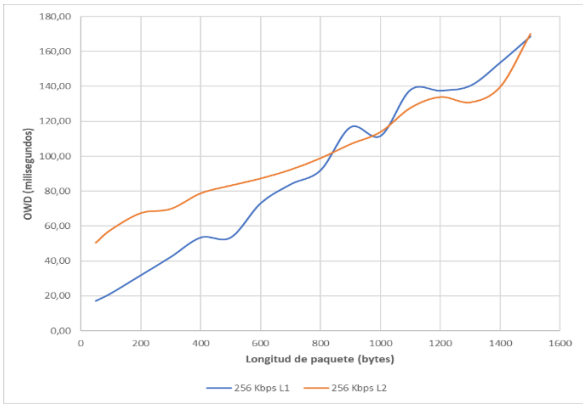
Figura 4.5. Análisis de retardo de encolamiento por carga y técnica de QoS



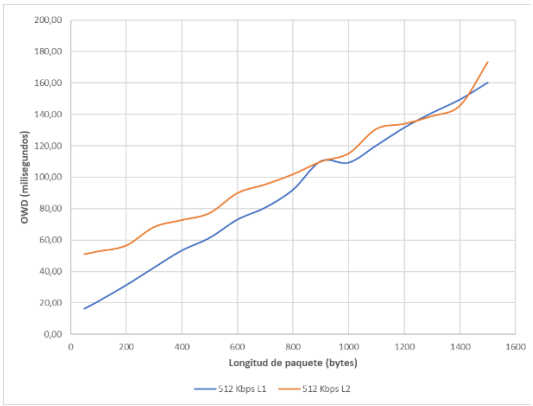
(a) 32 Kbps



(b) 64 Kbps



c) 256 Kbps



d) 512 Kbps

Figura 4.6. Comportamiento de OWD con WFQ con diferentes modelos de carga

Aunque se utiliza un hardware de tarjeta serial que soporta 128/256 Kbps para tráfico sincrónico, las técnicas de WFQ y PQ mostraron un mejor comportamiento de WFQ y del retardo de encolamiento, y del uso del ancho de banda establecido. FIFO reporta retardos excesivos de paquetes a partir de 256 Kbps, y CBWFQ a partir de 512 Kbps. Para una carga superior todas las técnicas muestran retardos muy grandes. Las figuras 4.4 y 4.5 muestran el comportamiento de los retardos medidos para las diferentes cargas de tráfico y técnicas de QoS. La carga de tráfico utilizada tiene una distribución alta de paquetes pequeños y baja de paquetes de tamaño grande, como se indica en la ecuación 3.3 (L1). Para verificar el comportamiento de OWD con otro tipo de carga, se utiliza un modelo con alta probabilidad de paquetes grandes y baja para paquetes pequeños, como se indica en la ecuación 3.12 (L2). La gráfica 4.6 muestra los resultados.

Se puede observar en la figura 4.6 que entre las dos distribuciones de tráfico existen diferencias no despreciables para OWD, para paquetes de interés de longitud pequeña, pero en la medida que la longitud de estos paquetes se incrementa, la diferencia de OWD es menos significativa. Este comportamiento se presenta para cargas de 64 – 256 – 512 Kbps, y se debe principalmente a que en la medida que el tráfico se incrementa, y la longitud de paquete crece, se tiene una mejor utilización del sistema de encolamiento. Para la carga de 32 Kbps la utilización del sistema no es adecuada por la presencia de un número bajo de paquetes pequeños o grandes.

4.4. Estimación de modelos de OWD y del retardo de encolamiento

De acuerdo con la matriz de datos de la tabla 4.1, se procede a realizar estimaciones para obtener un modelo predictivo del retardo OWD basado en regresión polinómica, tomando en cuenta todos los datos, con y sin los outliers, versus un modelo que solo tome en cuenta los valores de las medianas, y establecer cual representa una mejor estimación en función de la relación entre las variables independiente y dependiente de esta fase de la investigación.

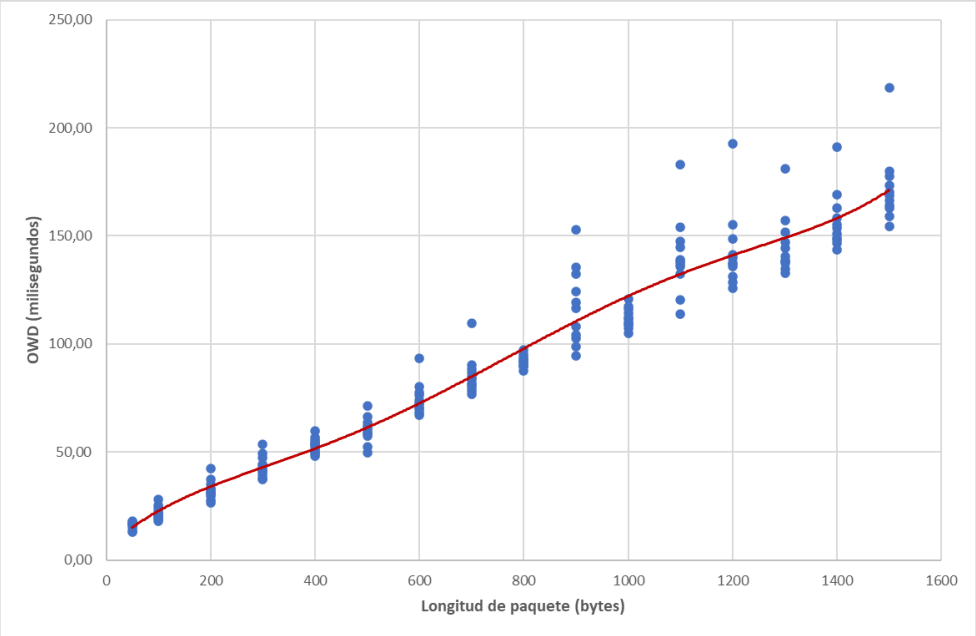


Figura 4.7. Modelo de regresión polinómica para OWD a partir de matriz de datos

Tabla 4.5. Valores de R^2 para modelo de matriz de datos con outliers

Orden	Polinomio	R^2
2	$f(x)=-2E-06x^2 + 0,1113x + 9,7125$	0,9484
3	$f(x)=-2E-08x^3 + 4E-05x^2 + 0,0844x + 12,986$	0,9491
4	$f(x)=2E-12x^4 - 2E-08x^3 + 5E-05x^2 + 0,0828x + 13,11$	0,9491
5	$f(x)=2E-13x^5 - 7E-10x^4 + 1E-06x^3 - 0,0005x^2 + 0,2183x + 5,4599$	0,9503
6	$f(x)=3E-16x^6 - 1E-12x^5 + 2E-09x^4 - 1E-06x^3 + 0,0004x^2 + 0,0563x + 12,79$	0,9508

Tabla 4.6. Valores de R^2 para modelo de matriz de datos sin outliers

Orden	Polinomio	R^2
2	$f(x)=-3E-06x^2 + 0,1089x + 10,234$	0,9753
3	$f(x)=-2E-08x^3 + 3E-05x^2 + 0,0874x + 12,831$	0,9758
4	$f(x)=-2E-13x^4 - 1E-08x^3 + 3E-05x^2 + 0,0876x + 12,811$	0,9758
5	$f(x)=2E-13x^5 - 7E-10x^4 + 1E-06x^3 - 0,0005x^2 + 0,2192x + 5,41$	0,9771

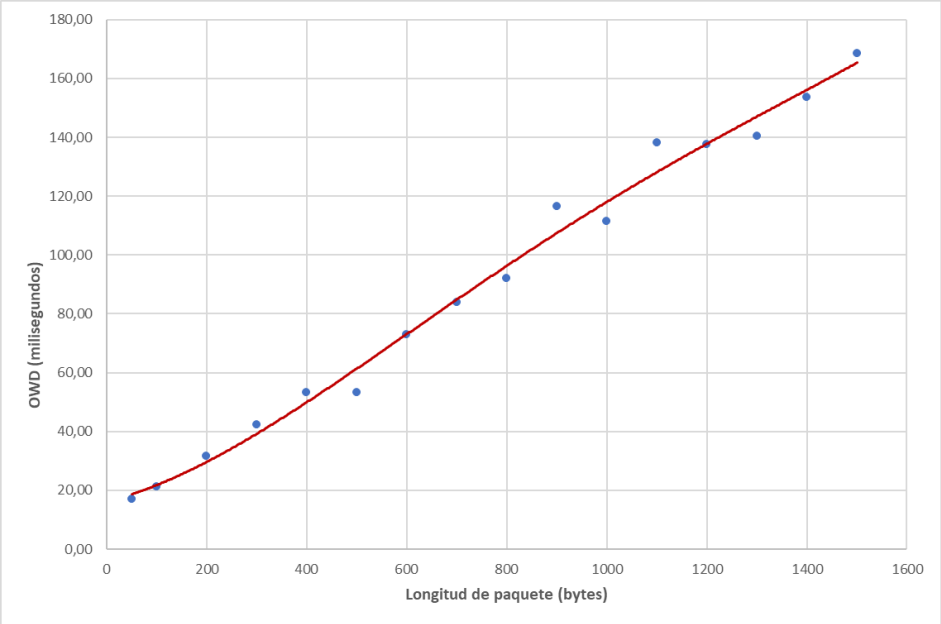


Figura 4.8. Modelo de regresión polinómica para OWD a partir de medianas

Tabla 4.7. Valores de R^2 para modelo basado en medianas

Orden	Polinomio	R^2
2	$f(x)=-3E-06x^2 + 0,1094x + 9,3976$	0,9879
3	$f(x) = -3E-08x^3 + 6E-05x^2 + 0,0741x + 13,698$	0,9891
4	$f(x) = 3E-11x^4 - 1E-07x^3 + 0,0001x^2 + 0,0429x + 16,1$	0,9894

Para el caso de considerar todas las muestras, para un R^2 aceptable, se requiere un polinomio de orden 5. Eliminando los valores de outliers, un polinomio de orden 2 es más que suficiente, similar al modelo en el que solo se consideran las medianas. Los valores obtenidos de R^2 son los siguientes:

El cálculo de R^2 con matriz de datos incluyendo los valores de outliers:

Number of observations: 176, Error degrees of freedom: 173

Root Mean Squared Error: 11.6

R-Squared: 0.9490, Adjusted R-Squared 0.949

F-statistic vs. constant model: 1.59e+03, p-value = 4.43e-112

Tabla 4.8. Pruebas de validación del modelo con outliers

Coeficiente	Estimación	Error Estándar	Estadístico t	p-valor
(Intercept)	9.7125	2.4514	3.9621	0.00010848
X	0.11127	0.0075698	14.7	2.8877e-32
X^2	-2.2976e-06	4.8169e-06	-0.47699	0.63397

El cálculo de R^2 con matriz de datos eliminando los valores de outliers:

Number of observations: 169, Error degrees of freedom: 166

Root Mean Squared Error: 7.64

R-Squared: 0.9761, Adjusted R-Squared 0.975

F-statistic vs. constant model: 3.24e+03, p-value = 9.14e-134

Tabla 4.9. Pruebas de validación del modelo sin outliers

Coeficiente	Estimación	Error Estándar	Estadístico t	p-valor
(Intercept)	10.299	1.6269	6.3302	2.199e-09
X	0.1094	0.0050796	21.538	6.1231e-50
X^2	-3.5115e-06	3.2513e-06	-1.0801	0.28169

Tabla 4.10. Pruebas de validación del modelo usando medianas

Coeficiente	Estimación	Error Estándar	Estadístico t	p-valor
(Intercept)	9.397633	4.1494	2.264817	0.04126212
X	0.1093901	0.0128133	8.537227	1.089986e-06
X^2	-2.750439e-06	8.153508e-06	-0.3373319	0.7412511

Se puede observar que el valor de R^2 del modelo de regresión que elimina los outliers, mejora la relación de las variables independiente y dependiente del modelo, pero no es mejor que el modelo que considera solo medianas. A partir de este punto se puede justificar el uso solo de las medianas para el cálculo de los modelos predictivos mediante la regresión polinomial.

Para el manejo de outliers se establece un intervalo de confianza y aquellos valores que no caen en el mismo, se los deja fuera el modelo. Se utiliza estadísticamente los valores residuales desde cada punto hacia la línea del modelo de regresión. Y se valida mediante la distancia de Cook´s si los outliers afectan negativamente al modelo de regresión. Las gráficas 4.7 - 4.11 muestran el análisis realizado.

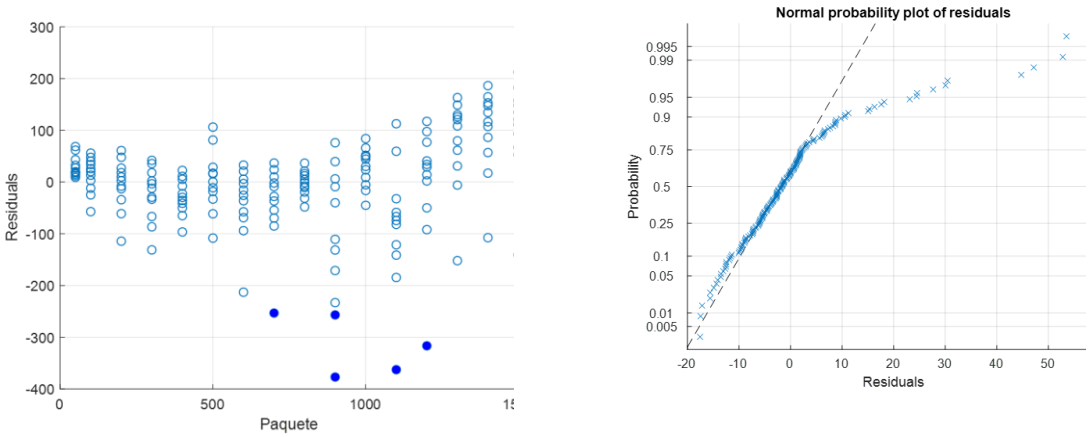
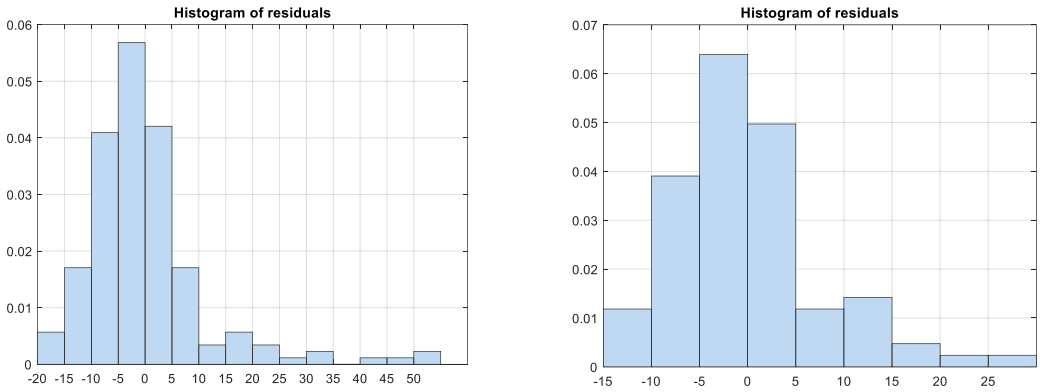


Figura 4.9. Análisis de valores residuales para modelo OWD



(a) con outliers **(b) sin outliers**

Figura 4.10. Histogramas de valores residuales para modelo OWD

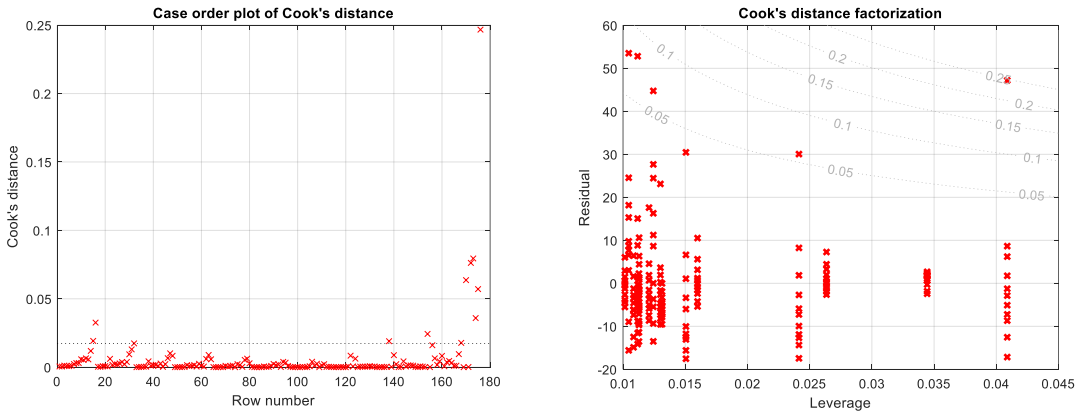
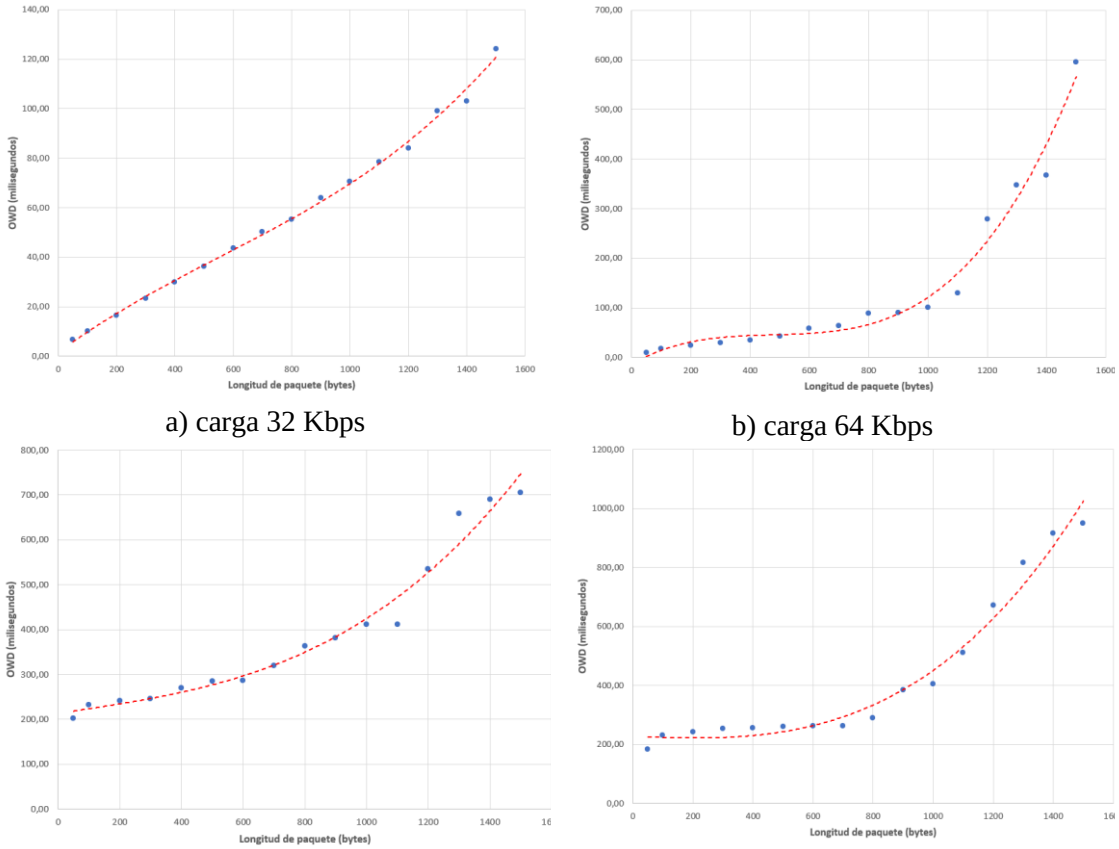


Figura 4.11. Análisis de incidencia de los outliers en el modelo

Los modelos predictivos basados en regresión polinómica para el retardo unidireccional para diferentes niveles de carga de enlace y diferentes longitudes de paquete, aplicando CBWFQ, se muestran en las ecuaciones 4.1 (carga 32K), 4.2 (carga 64K), 4.3 (carga 128K) y 4.4 (carga 256K), y en la gráfica 4.12 (a – d). Para el retardo de encolamiento. los modelos corresponden a las ecuaciones 4.5 (carga 32K), 4.6 (carga 64K), 4.7 (carga 128K) y 4.8 (carga 256K), y se reflejan en la gráfica 4.13. Estos modelos sirven como línea de referencia en la tercera fase de la investigación, para un análisis comparativo con los retardos a obtener de la simulación del modelo de QoS propuesto en este trabajo.



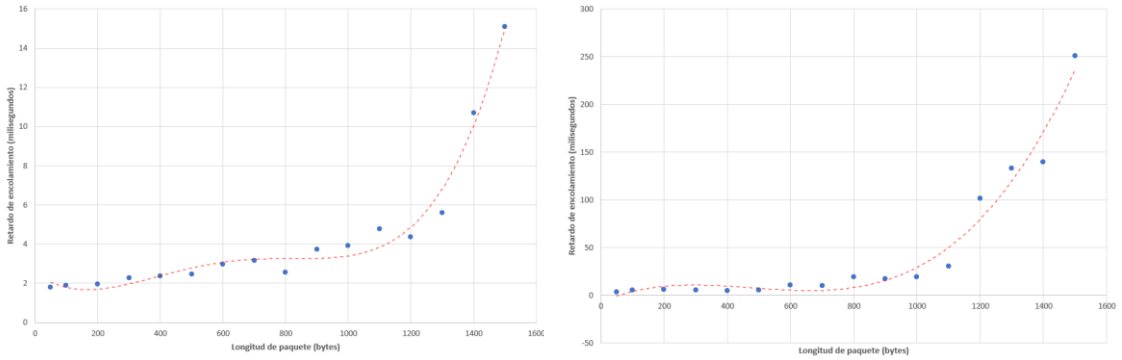
a) carga 32 Kbps

b) carga 64 Kbps

c) carga 128 Kbps

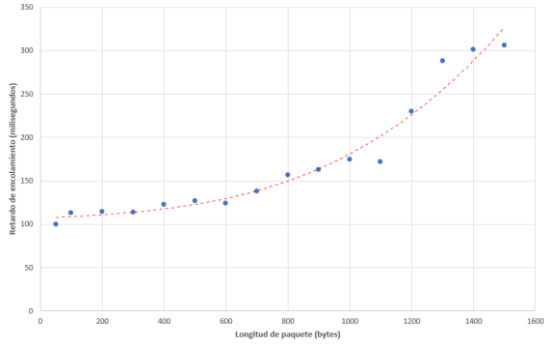
d) carga 256 Kbps

Figura 4.12. Modelos de regresión polinómica de OWD para técnica CBWFQ

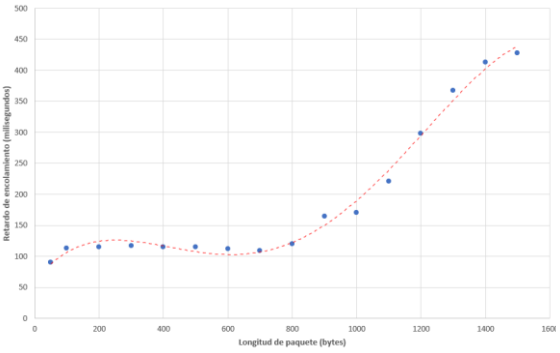


a) carga 32 Kbps

b) carga 64 Kbps



c) carga 128 Kbps



d) carga 256 Kbps

Figura 4.13. Modelos de regresión polinómica del retardo de encolamiento para técnica CBWFQ

$$f(x) = 3x10^{-8}\,X^3 - 5x10^{-5}\,X^2 + 0{,}087\,X + 1{,}5517 \quad (4.1)$$

$$f(x) = 5x10^{-7}\,X^3 - 7x10^{-4}\,X^2 + 0{,}3388\,X - 13{,}041 \quad (4.2)$$

$$f(x) = 1x10^{-7}\,X^3 - 1x10^{-5}\,X^2 + 0{,}1038\,X + 213{,}52 \quad (4.3)$$

$$f(x) = 2x10^{-7}\,X^3 + 4x10^{-5}\,X^2 - 0{,}0481\,X + 227{,}84 \quad (4.4)$$

$$f(x) = 2x10^{-11}\,X^4 - 6x10^{-8}\,X^3 - 5x10^{-5}\,X^2 - 0{,}0115\,X + 2{,}517 \quad (4.5)$$

$$f(x) = 2x10^{-7}\,X^3 - 3x10^{-4}\,X^2 + 0{,}1381\,X - 6{,}5444 \quad (4.6)$$

$$f(x) = 6x10^{-8}\,X^3 - 7x10^{-6}\,X^2 + 0{,}0205\,X + 106{,}74 \quad (4.7)$$

$$f(x) = -7x10^{-10}\,X^4 + 2x10^{-6}\,X^3 - 0.002\,X^2 + 0{,}6351\,X + 60{,}683 \quad (4.8)$$

CAPÍTULO V

MODELO PROPUESTO DE CALIDAD DE SERVICIO

El objetivo principal de este trabajo es proponer un modelo de calidad de servicio, que utilizando el comportamiento predictivo de la variable longitud de paquete (primera etapa de la investigación) y de la variable retardo de encolamiento de otros esquemas de QoS (segunda etapa), permita reducir el retardo en un dispositivo intermediario (tercera etapa). El modelo propuesto resulta una variante de la arquitectura DiffServ, pero no utiliza marcación ni clasificación a partir del campo DSCP de los paquetes IPv4 e IPv6. La descripción de este modelo, así como de los resultados de la simulación de este, se presentan a continuación.

5.1. Principios del modelo propuesto

El primer principio de un modelo de QoS es tener claro que, para ofrecer tiempos de servicios adecuados a las aplicaciones que garanticen una calidad de experiencia de los usuarios, se necesita que los enlaces de comunicación no estén saturados durante largos periodos de tiempo. En la medida que la carga se incrementa, los retardos aumentan, y si sobrepasamos los límites establecidos por los enlaces de comunicación, se producirán pérdidas de paquetes y retransmisiones, lo que causará problemas con el rendimiento de las aplicaciones y servicios de una red de datos. De allí que, en la segunda etapa de investigación, que modela los retardos unidireccionales y de encolamiento, se utilizan interfaces con anchos de banda limitados donde se pueda llegar a la congestión del enlace y obtener datos adecuados para este estudio. Esto es aplicado de igual manera en la simulación del modelo propuesto.

De acuerdo con las dos arquitecturas existentes para la implementación de calidad se servicio, DiffServ es la que representa una implementación ágil y factible para los ruteadores de acceso, como apreciamos en la figura 5.1, pero invierte tiempo en el proceso de marcación de paquetes (campo DSCP) para luego clasificarlos en función de las colas que se implementen en un modelo de QoS. Una forma de evitarlo es que los paquetes

sean marcados antes de llegar al ruteador, pero no se lo puede garantizar de forma estándar.

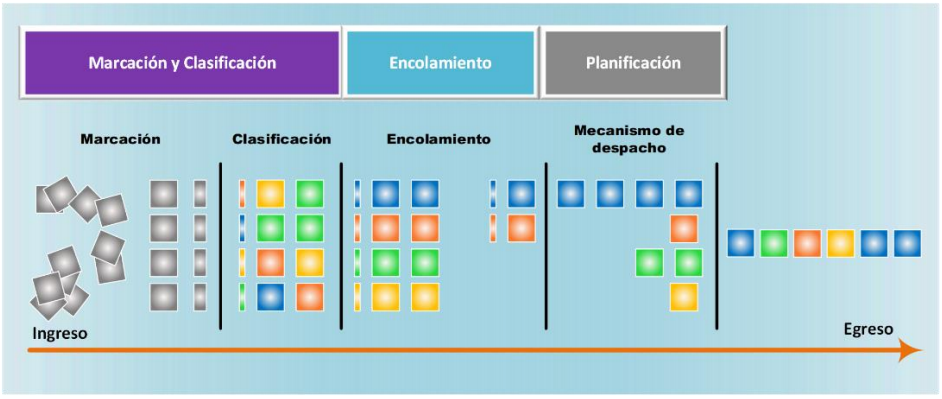


Figura 5.1. Modelo de QoS basado en DiffServ

El modelo propuesto sigue la estructura de DiffServ, pero a diferencia de este, se elimina el proceso de marcación y clasificación que se centran en el uso del campo DSCP y otros campos que identifican a los paquetes y flujos a través de un dispositivo intermediario (ver figura 5.2).

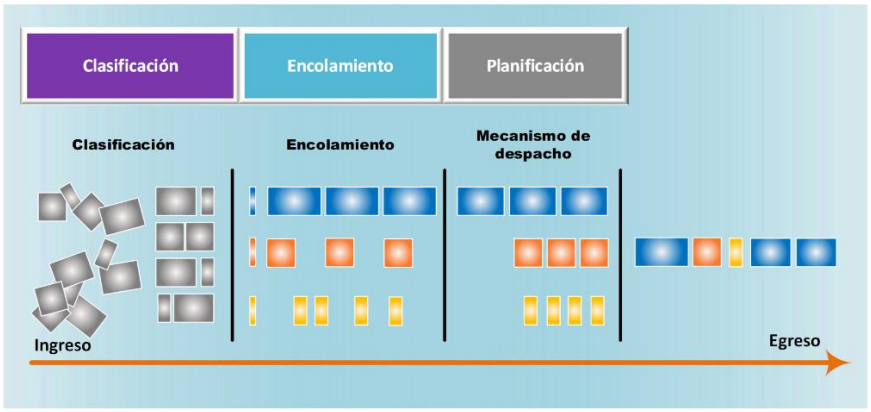


Figura 5.2. Modelo general de QoS propuesto sin fase de marcación

Como se muestra en la figura 1.4, el modelo propuesto utiliza el comportamiento predictivo de las longitudes de paquetes para reemplazar y/o eliminar el mecanismo de marcación, pues utiliza un campo que ya viene establecido en los paquetes (longitud total). Esto cambia el proceso de clasificación, que ya no utiliza varias propiedades de los paquetes que identifican a las aplicaciones, sino se centra de forma exclusiva en la variable longitud de paquete, que se analiza a lo largo de este trabajo, tanto para modelar

el tráfico de red [104-107], así como los retardos unidireccionales y de encolamiento. La figura 5.3 nos muestra un esquema del modelo propuesto.

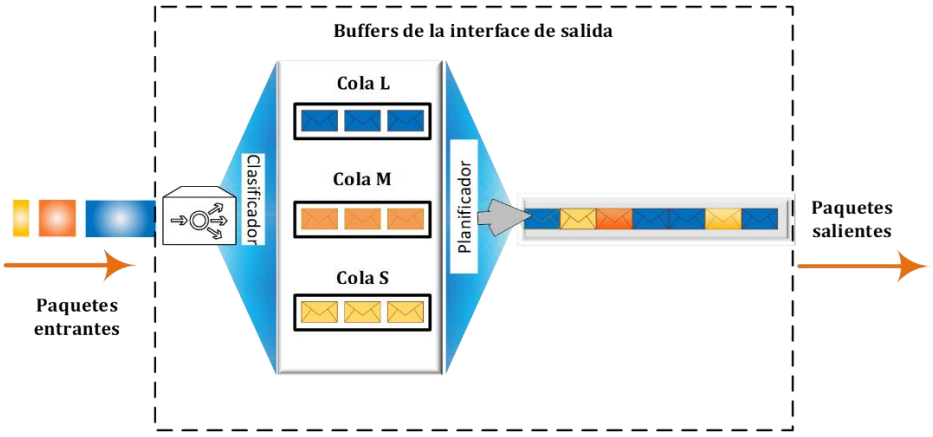


Figura 5.3. Diagrama esquemático del modelo propuesto

El modelo propuesto utiliza 3 colas, en función de las longitudes de paquetes. Se utiliza una cola de paquetes de longitud pequeña (entre 0 y 200 bytes), una de longitud grande (entre 1300 y 1500 bytes), y una de paquetes de longitud mediana (entre 200 y 1300). El clasificador asigna los paquetes a estas 3 colas según el campo longitud total. Se utilizan los modelos de tráfico estimados en la primera fase (capítulo 3), que nos establecen probabilidades de pertenencia a los rangos de longitudes indicados, así como las distribuciones que siguen dichos paquetes, clasificados por protocolos, puertos de aplicaciones y aplicaciones más utilizadas. Se asume un modelo con manejo de longitud de colas de tamaño infinito, ya que no se estudia la pérdida de paquetes; sin embargo, en función de los retardos de encolamiento que se estudian y presentan en la segunda fase (capítulo 4), se obtienen longitudes de colas sugeridas para evitar saturación en función del arribo esperado de paquetes para el ancho de banda asociado al modelo de tráfico.

Con respecto al planificador, que es el proceso encargado de tomar paquetes de las colas y asignarlos a la interfaz de salida de acuerdo con el ancho de banda establecida, se utiliza un modelo existente conocido como WRR (Round Robin por prioridad/peso). Entre las ventajas de este método están su facilidad de implementación que lo hace adecuado para redes de alta velocidad, y que en cada ciclo atiende a todas las colas, evitando el agotamiento de paquetes en colas no atendidas. Adicionalmente, y en función de la prioridad, se despachan los paquetes en mayor frecuencia de las colas de mayor peso/prioridad, con respecto a las otras colas. Una desventaja importante de WRR es que

solo asigna el ancho de banda adecuado a cada cola asumiendo que los tamaños de paquete de todas las colas son iguales. Esto es mejorado por el modelo propuesto debido a que los paquetes de las colas están asignados en función de su longitud, estableciendo una mejor utilización del sistema de colas y del ancho de banda.

5.2. Diseño y formulación

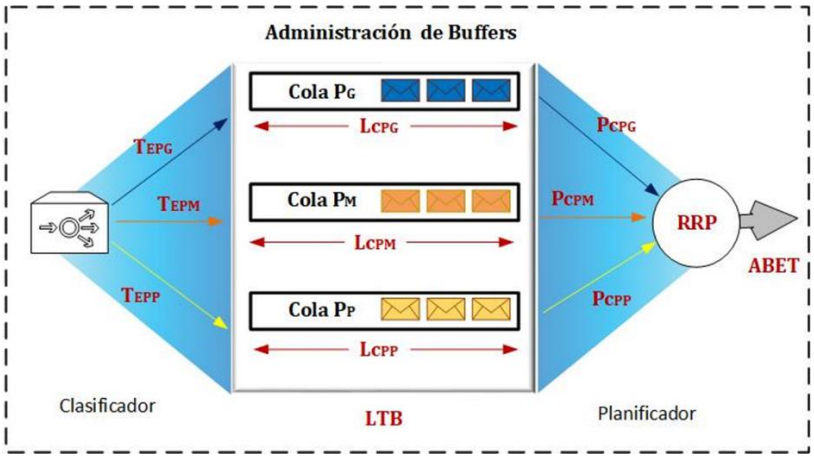


Figura 5.4. Formulación del modelo propuesto

El modelo propuesto [108-111] trabaja con un conjunto de parámetros de entrada obtenidos de los modelos predictivos de longitudes de paquete y de retardos de encolamiento; se asume que los paquetes arriban de forma constante al sistema de colas. Se definen los siguientes parámetros:

LPPG: Longitud promedio de un paquete grande
LPPM: Longitud promedio de un paquete mediano
LPPP: Longitud promedio de un paquete pequeño
PrPG: probabilidad de arribo de un paquete de longitud grande
PrPM: probabilidad de arribo de un paquete de longitud mediana
PrPP: probabilidad de arribo de un paquete de longitud pequeña
REMPG: Retardo de encolamiento medio de un paquete de longitud grande
REMPG: Retardo de encolamiento medio de un paquete de longitud mediana
REMPP: Retardo de encolamiento medio de un paquete de longitud pequeña
ABET: Ancho de banda de enlace total

Cada modelo estimado de tráfico que se muestra en el capítulo 3, permite obtener las longitudes promedio de paquetes y sus probabilidades de arribo. Se toma como referencia el modelo de tráfico de IPv4 – TCP debido a que representa el mayor volumen de tráfico, y los aplicativos obtenidos y analizados trabajan sobre estos protocolos. Respecto a los retardos de encolamiento, se utilizará como referencia los obtenidos para la técnica de QoS conocida como CBWFQ, definiendo un ancho de banda similar al utilizado en el hardware del escenario del capítulo 4, esto es 256 Kbps, para obtener un modelo comparativo y que nos permita validar o no la hipótesis de este trabajo.

Las tasas esperadas de paquetes entrantes de definen en las ecuaciones 5.1 hasta 5.4. La asignación de tamaños de las colas en bytes se define en las ecuaciones 5.5 hasta 5.7; se utiliza la siguiente nomenclatura para los parámetros y variables:

TEPG: Tasa estimada de paquetes grandes que ingresan al sistema de colas
TEPM: Tasa estimada de paquetes medianos que ingresan al sistema de colas
TEPP: Tasa estimada de paquetes pequeños que ingresan al sistema de colas
TAE: Tasa de arribos esperada total al sistema de colas
LCPG: Longitud de la cola de paquetes grandes
LCPM: Longitud de la cola de paquetes medianos
LCPP: Longitud de la cola de paquetes pequeños

$$\mathbf{T_{AE}} = \frac{\text{ABET}}{8 * \text{LPPM}} \tag{5.1}$$

$$\mathbf{T_{EPG}} = T_{AE} * Pr_{PG} \tag{5.2}$$

$$\mathbf{T_{EPM}} = T_{AE} * Pr_{PM} \tag{5.3}$$

$$\mathbf{T_{EPP}} = T_{AE} * Pr_{PP} \tag{5.4}$$

$$\mathbf{L_{CPG}} = \frac{L_{PPG} * T_{EPG} * REMPG}{T_{AE}} \tag{5.5}$$

$$\mathbf{L_{CPP}} = \frac{L_{PPP} * T_{EPP} * REMPP}{T_{AE}} \tag{5.6}$$

$$\mathbf{L_{CPM}} = \frac{L_{PPM} * T_{EPM} * REMPM}{T_{AE}} \tag{5.7}$$

De acuerdo con el planificador de paquetes WRR, las colas son servidas en función del nivel de preferencia. En cada intervalo de medición se asegura que todas las colas sean atendidas sin causar agotamiento de los paquetes en espera. Asumimos que

mientras no se alcance el nivel de saturación (utilización del sistema de encolamiento < 1) la tasa de despacho de paquetes será igual a la tasa esperada de arribos. Cuando es mayor, se producirá una saturación de las colas hasta llegar a la pérdida de paquetes, lo que aumentará el retardo, y cualquier modelo de QoS no será útil. La tasa de despacho es el equivalente al peso del algoritmo, por lo que en función de la probabilidad de arribo de cada tamaño de paquete se define cuántos paquetes grandes, medianos y pequeños serán atendidos en cada ciclo o ronda de WRR.

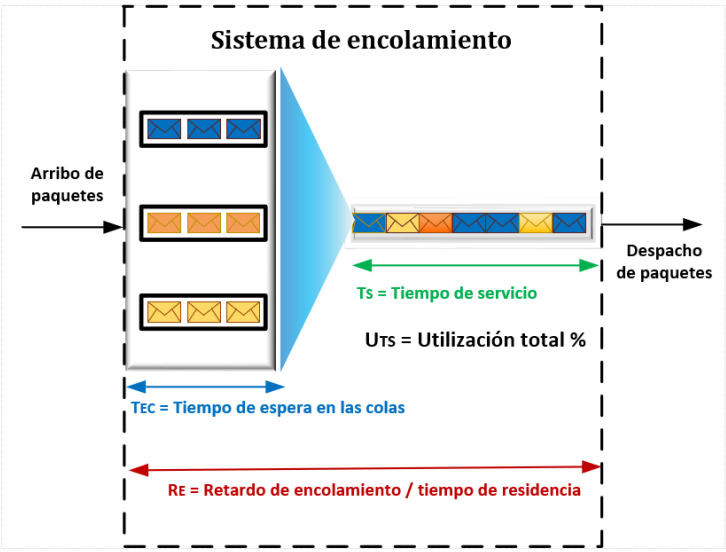


Figura 5.5. Tiempos de servicio y residencia

El retardo de encolamiento de un esquema de QoS, es conocido también como tiempo de residencia de un paquete en el sistema de encolamiento, y es equivalente al tiempo de espera en cola (Tec) más el tiempo de servicio (Ts), como se observa en la figura 5.5. Los retardos de servicio son definidos por las ecuaciones 5.8 – 5.10; en 5.11 se define el retardo promedio de servicio. En cada ciclo WRR se obtiene la utilización del sistema de colas, como se define en las ecuaciones 5.12 – 5.15. Los tiempos de residencia se definen en las ecuaciones 5.16 – 5.19. La nomenclatura asignada a los parámetros y variables es la siguiente:

RSPG: Retardo de servicio para un paquete grande

RSPM: Retardo de servicio para un paquete mediano

RSPP: Retardo de servicio para un paquete pequeño

RS: Retardo de servicio promedio

UcPG: Utilización de la cola de paquetes grandes

U_{CPM}: Utilización de la cola de paquetes medianos

U_{CPP}: Utilización de la cola de paquetes pequeños

R_{RPG}: Retardo de residencia/encolamiento para un paquete de longitud grande

R_{RPM}: Retardo de residencia para un paquete de longitud mediana

R_{RPP}: Retardo de residencia para un paquete de longitud pequeña

R_R: Retardo de residencia promedio

$$R_{SPG} = \frac{L_{PPG} * 8}{A_{BET}} \tag{5.8}$$

$$R_{SPM} = \frac{L_{PPM} * 8}{A_{BET}} \tag{5.9}$$

$$R_{SPP} = \frac{L_{PPP} * 8}{A_{BET}} \tag{5.10}$$

$$R_S = \frac{T_{EPG}}{T_{AE}} * R_{SPG} + \frac{T_{EPM}}{T_{AE}} * R_{SPM} + \frac{T_{EPP}}{T_{AE}} * R_{SPP} \tag{5.11}$$

$$\mathbf{U_{CPG}} = T_{EPG} * R_{SPG} \tag{5.12}$$

$$\mathbf{U_{CPM}} = T_{EPM} * R_{SPM} \tag{5.13}$$

$$\mathbf{U_{CPP}} = T_{EPP} * R_{SPP} \tag{5.14}$$

$$\mathbf{U_{TS}} = U_{CPG} + U_{CPM} + U_{CPP} \tag{5.15}$$

$$R_{RPM} = R_{SPM} + \frac{(U_{CPG} * R_{SPG} + U_{CPM} * R_{SPM} + U_{CPP} * R_{SPP})}{(1 - U_{CPM})} \tag{5.16}$$

$$R_{RPP} = R_{SPP} + \frac{(U_{CPG} * R_{SPG} + U_{CPM} * R_{SPM} + U_{CPP} * R_{SPP})}{(1 - U_{CPP})} \tag{5.17}$$

$$R_{RPG} = R_{SPG} + \frac{(U_{CPG} * R_{SPG} + U_{CPM} * R_{SPM} + U_{CPP} * R_{SPP})}{(1 - U_{CPP} - U_{CPM})} \tag{5.18}$$

$$R_R = \frac{T_{EPG}}{T_{AE}} * R_{RPG} + \frac{T_{EPM}}{T_{AE}} * R_{RPM} + \frac{T_{EPP}}{T_{AE}} * R_{RPP} \tag{5.19}$$

5.3. Simulación y Resultados

El modelo propuesto se lo ha simulado mediante MATLAB Simulink, implementando el diagrama de bloques que se muestra en la figura 5.6. El valor de ABET definido es de 256 Kbps, que permitirá comparar los valores a obtener del modelo con los obtenidos en el estudio del capítulo 4, usando el modelo CBWFQ con el mismo ancho de banda. Además, este valor es útil para poder llegar a un nivel de saturación donde el modelo de QoS propuesto pueda ofrecer valores de retardo adecuados para las aplicaciones.

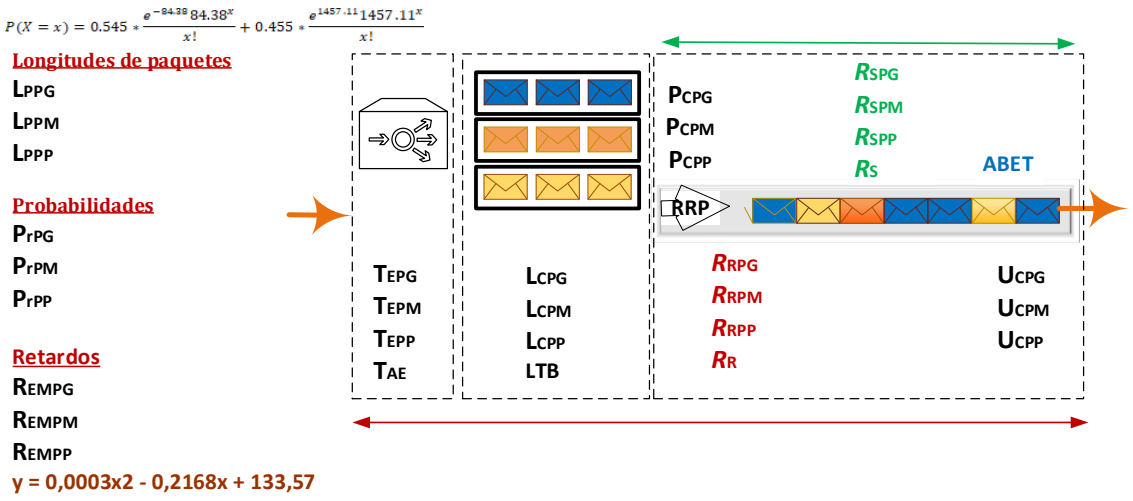


Figura 5.6. Diagrama de bloques del modelo propuesto

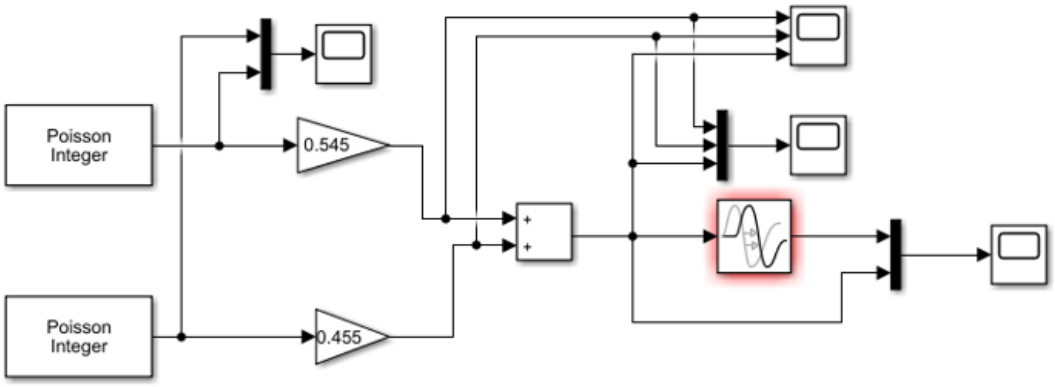


Figura 5.7. Diagrama del modelo propuesto en MATLAB

Los valores de retardo utilizados del modelo predictivo definido en capítulo 4 se utilizan para dimensionar las longitudes de las colas. Las longitudes de paquete y sus probabilidades se obtienen de las estimaciones presentadas en capítulo 3. Podemos simular el comportamiento del modelo propuesto modificando las probabilidades de ocurrencia de los paquetes, para lo cual utilizamos modelos predictivos de tráfico donde se estiman más paquetes de longitud grande que pequeños, o viceversa. Adicionalmente, se puede analizar el comportamiento del modelo propuesto para diferentes escenarios de longitudes mínimas y máximas de los paquetes. Se simula de manera similar al modelo experimental de capítulo 4, para paquetes de longitud creciente hasta el máximo tamaño de paquete permitido.

La figura 5.8 nos muestra el tráfico entrante que genera MATLAB para un primer escenario / modelo de tráfico, donde los paquetes de longitud pequeña se encuentran alrededor de 80 bytes, y los de longitud grande se encuentran alrededor de 1450 bytes, y que representa la estimación del tráfico IPv4 sobre una red alámbrica, un escenario experimental que se muestra en el capítulo 3.

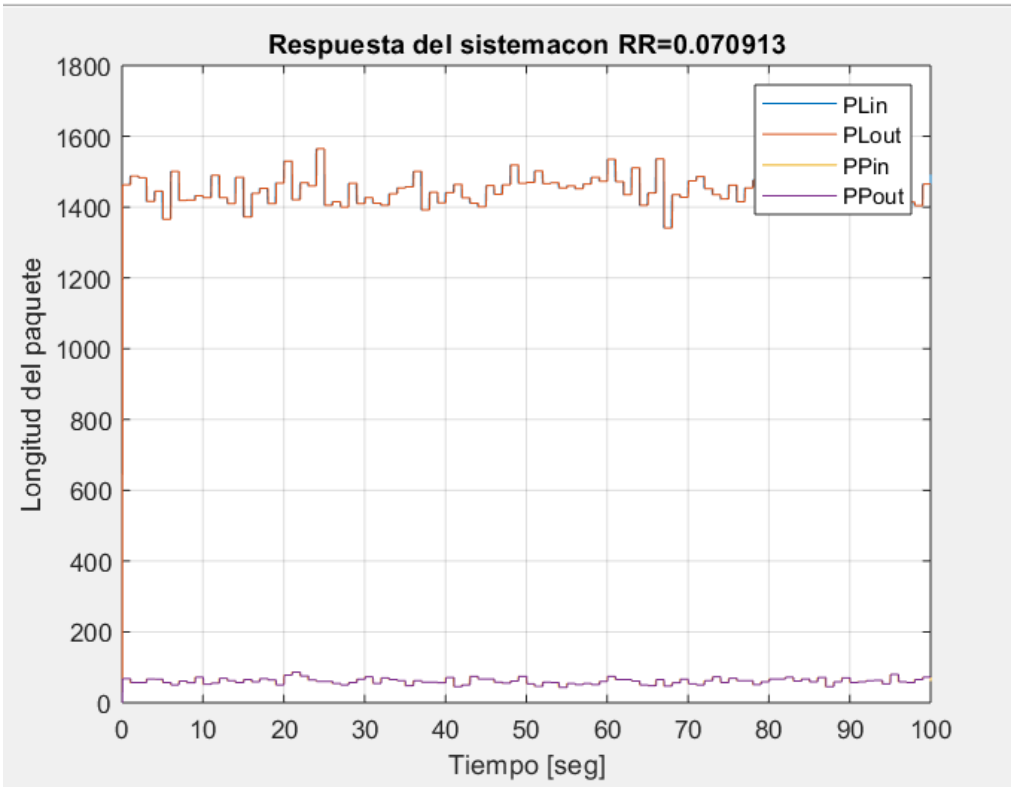


Figura 5.8. Tráfico simulado por MATLAB

Las probabilidades de ocurrencia de los paquetes generados en el primer escenario son las siguientes:

$$\mathrm{PrPG} = 0{,}5005 \quad \mathrm{PrPM} = 0{,}0686 \quad \mathrm{PrPP} = 0{,}4309$$

La figura 5.9 muestra el retardo de encolamiento promedio estimado por el modelo propuesto para paquetes de longitud pequeña y grande. La longitud de paquete pequeño se mantiene en 80 bytes. La figura 5.10 nos muestra el comportamiento de la variable dependiente retardo de encolamiento para el primer escenario de pruebas del modelo propuesto.

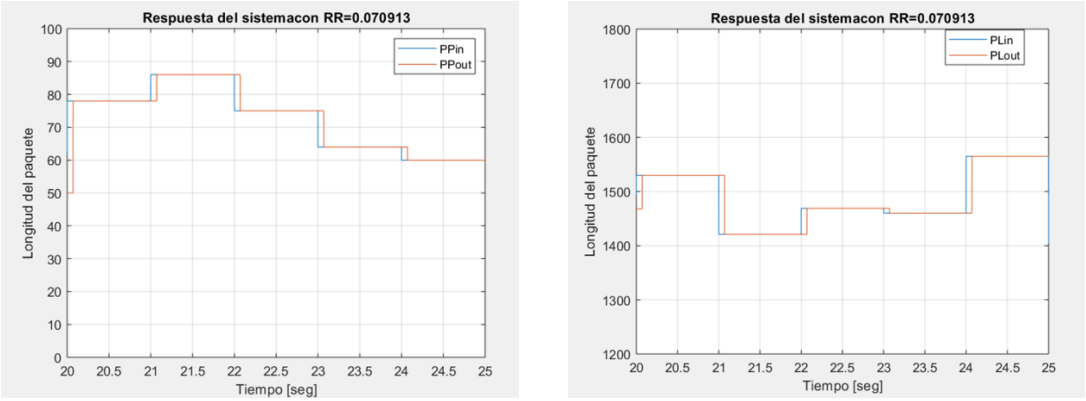


Figura 5.9. Retardo de encolamiento para paquetes de longitud pequeña y grande

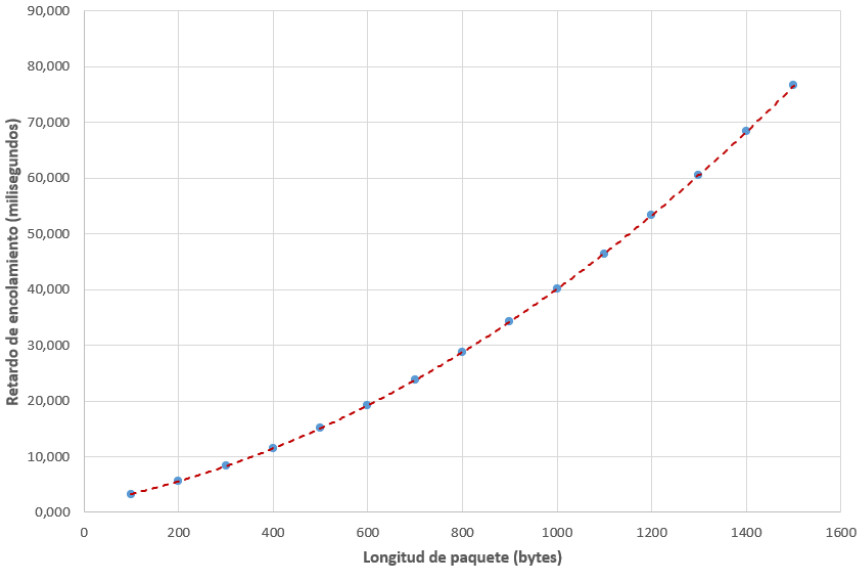


Figura 5.10. Retardo de encolamiento para escenario # 1 con LPPP = 80 bytes

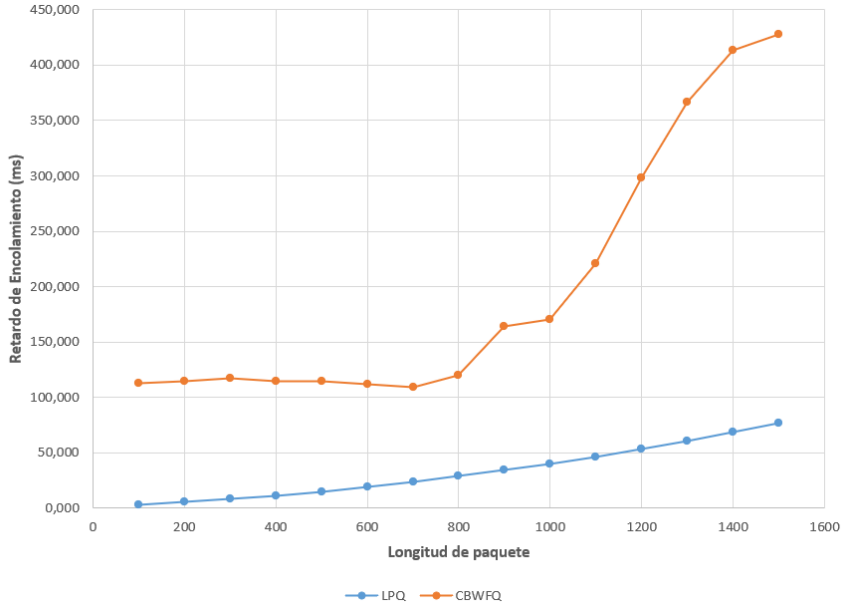
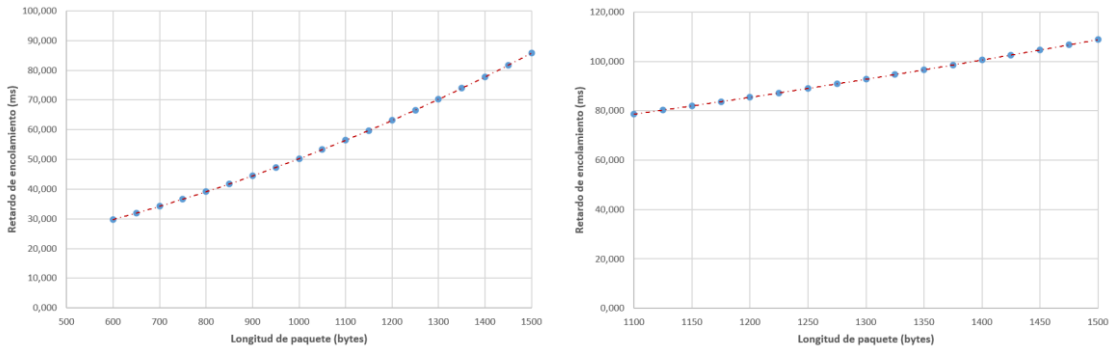


Figura 5.11. Comparativo del Retardo de encolamiento LPQ vs CBWFQ

Se procede a comparar los valores de retardo de encolamiento obtenidos a partir de la simulación del modelo propuesto, con los obtenidos en el modelo experimental implementado en el capítulo 4, usando la política de QoS CBWFQ y en similares condiciones de ancho de banda y carga de tráfico. Podemos observar en la figura 5.11 que el modelo propuesto, denominado ahora LPQ (encolamiento por longitud de paquete) nos ofrece mejores tiempos de residencia o encolamiento con respecto a CBWFQ. Los valores de utilización del sistema de colas van desde 0.10 hasta 1.07, denotando que en el caso de congestión moderada el modelo LPQ refleja un mejor comportamiento del retardo.

Utilizando los mismos valores de prioridades del escenario # 1, se procede a modificar la longitud mínima de un paquete asumiendo una distribución intermedia, con paquetes entre 500 bytes (pequeños) y 1500 bytes (grandes). La figura 5.12 (a) nos muestra los valores obtenidos. Los niveles de utilización del sistema de colas van desde 0.55 hasta 1.03 para congestión moderada. En la figura 5.12 (b) se muestra el comportamiento para paquetes entre 1000 bytes (pequeños) y 1500 bytes (grandes), donde la utilización del sistema de colas se encuentra entre 0.86 y 1.03 (congestión moderada).



a) LPPP = 500 bytes b) LPPP = 1000 bytes

Figura 5.12. Retardo de encolamiento para escenario #1 con diferentes valores de LPPP

Para un segundo escenario, se procede a modificar los valores de prioridades de los paquetes de longitud pequeña, mediana y grande. Se asocia un modelo donde sea mayor la probabilidad de un paquete grande ($PrPG = 0{,}70$) con respecto a un paquete pequeño ($PrPP = 0{,}23$). Las longitudes de paquetes pequeños y medianos se mantienen según los datos obtenidos del modelo experimental. Los resultados de esta simulación se muestran en la figura 5.13. La utilización del sistema de colas se mueve entre 0.09 y 1.09 para congestión moderada.

Finalmente, se simula un tercer escenario donde sea mayor la probabilidad de un paquete pequeño (PrPP = 0,70) con respecto a un paquete grande (PrPG = 0,23). Las longitudes de paquetes pequeños y medianos se mantienen según los datos obtenidos del modelo experimental. Los resultados de esta simulación se muestran en la figura 5.14. La utilización del sistema de colas se mueve entre 0.17 y 1.02 para congestión moderada. La figura 5.15 muestra un comparativo entre los 3 escenarios utilizando diferentes valores de probabilidad de ocurrencia de los paquetes de tamaño grande y pequeño.

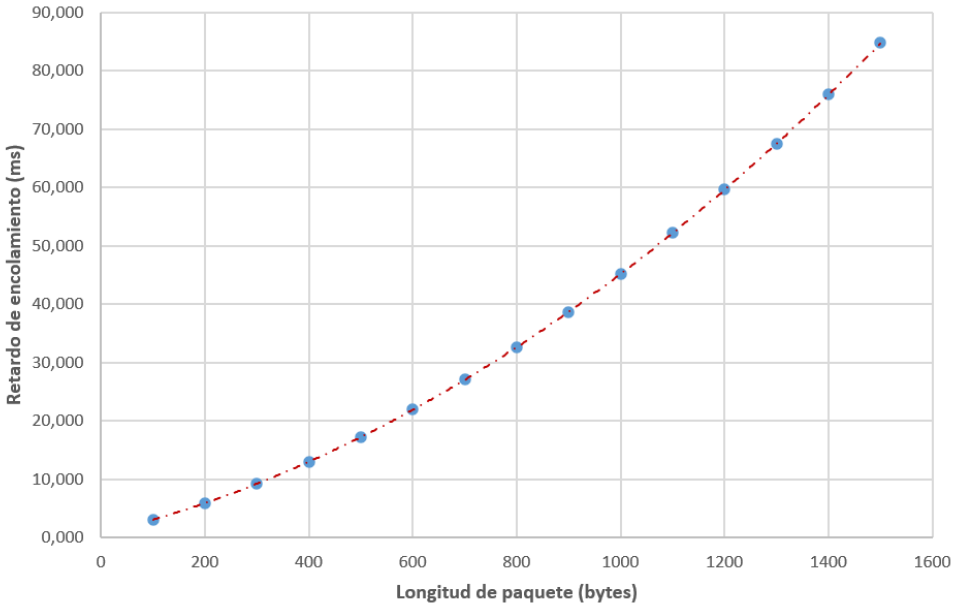


Figura 5.13. Retardo de encolamiento con Probabilidades PrPG = 0,70 y PrPP = 0,23

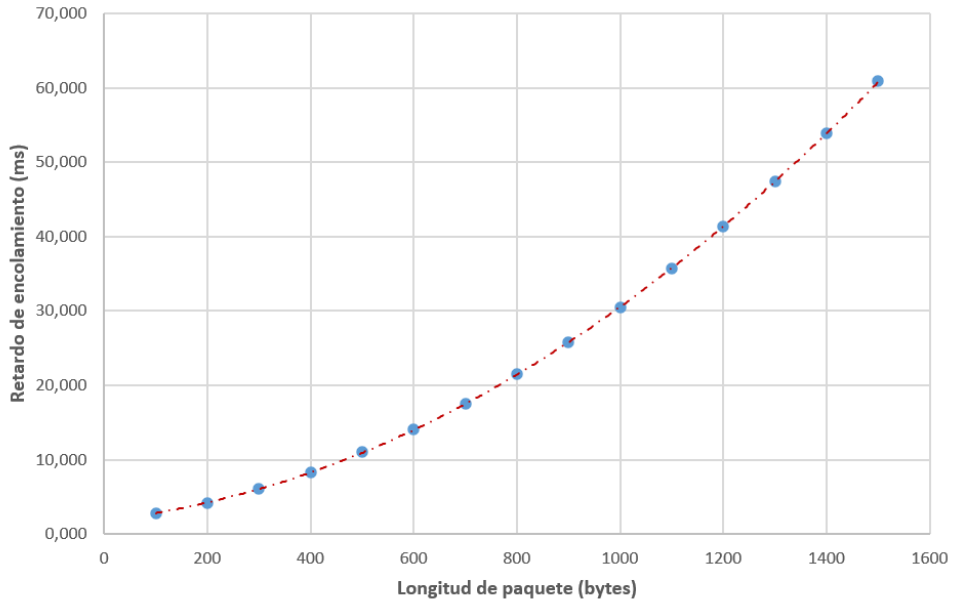


Figura 5.14. Retardo de encolamiento con Probabilidades PrPG = 0,23 y PrPP = 0,70

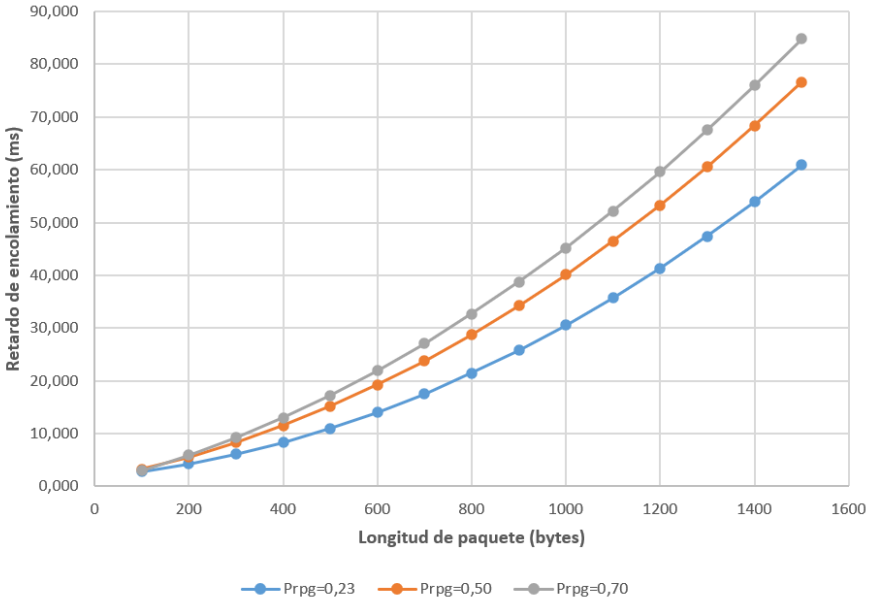


Figura 5.15. Retardo de encolamiento con diferentes valores de Probabilidades

5.4. Estimación del retardo de encolamiento del modelo propuesto

De acuerdo con los datos obtenidos, se procede a realizar estimaciones para obtener un modelo predictivo del retardo de encolamiento basado en regresión polinómica.

Para los datos del escenario presentado en la gráfica 5.10, el modelo estimado se muestra en la fórmula 5.1. El análisis estadístico de esta estimación se indica en la tabla 5.1. El error estándar calculado nos indica una alta precisión de la predicción del modelo, lo que es corroborado por el valor p que demuestra la confiabilidad del resultado obtenido. La gráfica 5.16 muestra el análisis de residuales del modelo con regresión polinómica.

$$f(x) = 2.27x10^{-5}\,X^2 + 0{,}0159\,X + 1{,}5090 \qquad (5.1)$$

Tabla 5.1. Análisis estadístico del modelo aplicado a escenario # 1 con LPPP = 80 bytes

Model	Estimate	SE	pValue
(Intercept)	1,5090	0,0432	1,9137E-13
x1	0,0159	0,0001	3,5431E-20
x1^2	2,2762E-05	7,5446E-08	1,1837E-24

Number of observations: 15, Error degrees of freedom: 12

Root Mean Squared Error: 0.0485
R-squared: 1, Adjusted R-Squared 1
F-statistic vs. constant model: 1.68e+06, p-value = 2.11e-33

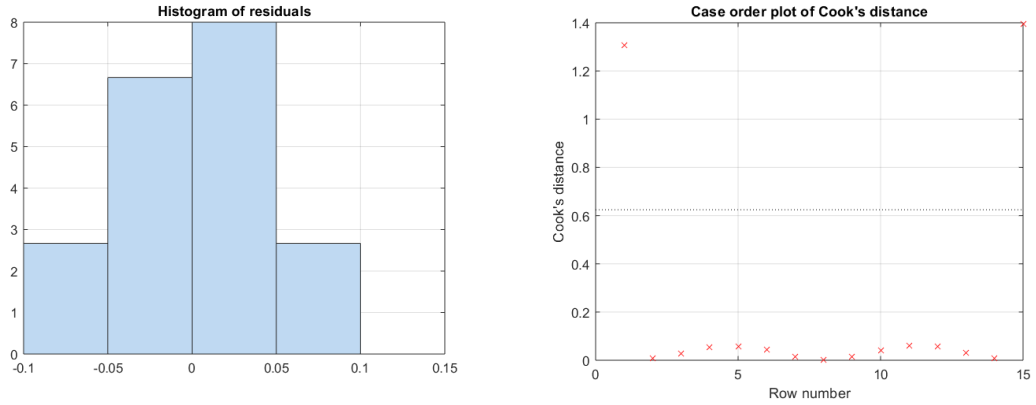


Figura 5.16. Análisis de residuales del modelo - escenario # 1 con LPPP = 80 bytes

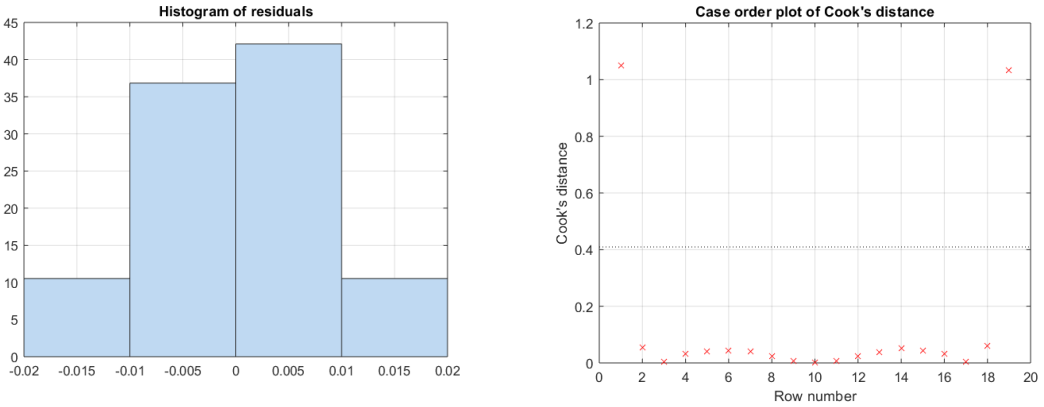


Figura 5.17. Análisis de residuales del modelo - escenario # 1 con LPPP = 500 bytes

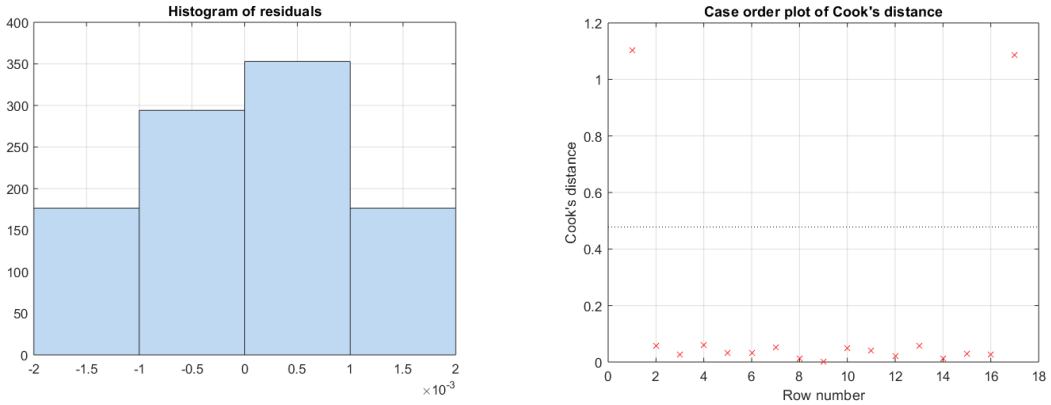


Figura 5.18. Análisis de residuales del modelo - escenario # 1 con LPPP = 1000 bytes

Para el mismo escenario #1 pero modificando la longitud promedio de un paquete pequeño a 500 bytes, la estimación se muestra en la fórmula 5.2. El análisis estadístico

de esta estimación se indica en la tabla 5.2, y en la gráfica 5.17 se muestra el análisis de residuales del modelo con regresión polinómica.

$$f(x) = 2.21x10^{-5}\,X^2 + 0{,}0157\,X + 12{,}3808 \quad (5.2)$$

Tabla 5.2. Análisis estadístico del modelo aplicado a escenario # 1 con LPPP=500 bytes

Model	Estimate	SE	pValue
(Intercept)	12,3808	0,0349	1,3150E-32
x1	0,0157	6,95E-05	1,8646E-29
x1^2	2,2182E-05	3,2855E-08	4,5223E-37

Number of observations: 19, Error degrees of freedom: 16
Root Mean Squared Error: 0.00957
R-squared: 1, Adjusted R-Squared 1
F-statistic vs. constant model: 3.04e+07, p-value = 2.3e-53

Finalmente, para el escenario #1 se modifica la longitud promedio de un paquete pequeño a 1000 bytes, la estimación se muestra en la fórmula 5.3. El análisis estadístico de esta estimación se indica en la tabla 5.3, y el análisis de residuales del modelo con regresión polinómica en la gráfica 5.18.

$$f(x) = 2.30x10^{-5}\,X^2 + 0{,}0156\,X + 33{,}5426 \quad (5.3)$$

Tabla 5.3. Análisis estadístico del modelo aplicado a escenario # 1 con LPPP = 1000 bytes

Model	Estimate	SE	pValue
(Intercept)	33,5426	0,0338	2,4697E-35
x1	0,0156	5,23E-05	4,8903E-28
x1^2	2,3028E-05	2,0114E-08	3,3207E-36

Number of observations: 17, Error degrees of freedom: 14

Root Mean Squared Error: 0.00111

R-squared: 1, Adjusted R-Squared 1

F-statistic vs. constant model: 5.94e+08, p-value = 3.15e-56

Del análisis del escenario #1 modificando en el modelo propuesto la longitud promedio de un paquete pequeño, se puede observar que la variable dependiente retardo

de encolamiento se incrementa respecto a la variable independiente longitud de paquete, corroborando los resultados experimentales presentados en el capítulo 4.

En el escenario # 2 se procede a modificar la probabilidad de que los paquetes entrantes sigan una distribución diferente, con una mayor frecuencia de paquetes grandes versus los paquetes pequeños, que representan de acuerdo con los modelos de tráfico presentados en el capítulo 3, las longitudes de paquete más comunes. En este caso se modifica la probabilidad de un paquete grande a 0.70 y la de un paquete pequeño a 0.23. Las longitudes de paquete se mantienen de acuerdo con el primer análisis del escenario #1. Los datos obtenidos a partir del modelo se presentan en la gráfica 5.13, y la estimación del modelo se muestra en la fórmula 5.4. El análisis estadístico de esta estimación se indica en la tabla 5.4. El error estándar calculado nos indica una alta precisión de la predicción del modelo, lo que es corroborado por el valor p que demuestra la confiabilidad del resultado obtenido. La gráfica 5.19 muestra el análisis de residuales del modelo con regresión polinómica.

$$f(x) = 2.29x10^{-5}\,X^2 + 0{,}0216\,X + 0{,}6799 \qquad (5.4)$$

Tabla 5.4. Análisis estadístico del modelo aplicado a escenario # 2 con PrPG = 0.7

Model	Estimate	SE	pValue
(Intercept)	0,6799	0,0361	2,7727E-10
x1	0,0216	1,04E-04	1,0035E-22
x1^2	2,2962E-05	6,3033E-08	1,2325E-25

Number of observations: 15, Error degrees of freedom: 12
Root Mean Squared Error: 0.0405
R-squared: 1, Adjusted R-Squared 1
F-statistic vs. constant model: 2.97e+06, p-value = 6.73e-35

Finalmente, se plantea un escenario # 3 donde se procede a modificar la probabilidad de que los paquetes entrantes sigan una distribución con una mayor frecuencia de paquetes pequeños (0.70) versus los paquetes grandes (0.23). Las longitudes de paquete se mantienen de acuerdo con el primer análisis del escenario #1. Los datos obtenidos a partir del modelo se presentan en la gráfica 5.14, y la estimación del modelo se muestra en la fórmula 5.5. El análisis estadístico de esta estimación se

indica en la tabla 5.5. La gráfica 5.20 muestra el análisis de residuales del modelo con regresión polinómica.

$$f(x) = 2.11x10^{-5}\,X^2 + 0{,}0075\,X + 1{,}8806 \qquad (5.5)$$

Tabla 5.5. Análisis estadístico del modelo aplicado a escenario # 3 con PrPP = 0.7

Model	Estimate	SE	pValue
(Intercept)	1,8806	0,0416	8,8258E-15
x1	0,0075	1,20E-04	1,7121E-16
x1^2	2,1165E-05	7,2629E-08	1,7948E-24

Number of observations: 15, Error degrees of freedom: 12

Root Mean Squared Error: 0.0466

R-squared: 1, Adjusted R-Squared 1

F-statistic vs. constant model: 1.14e+06, p-value = 2.08e-32

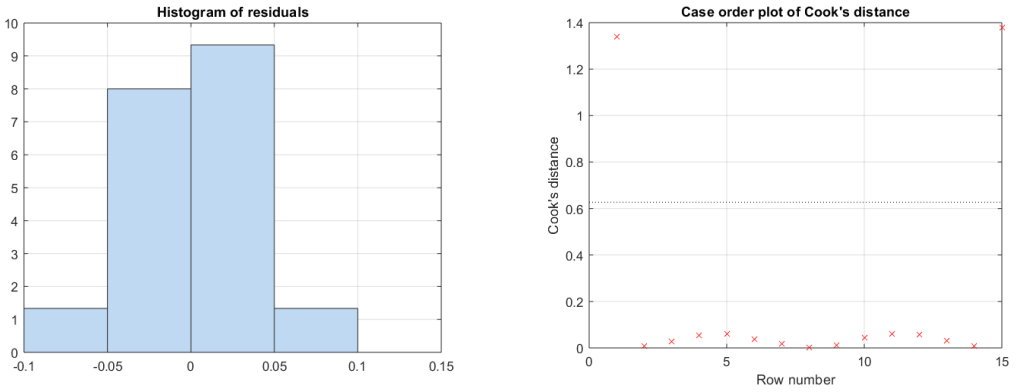


Figura 5.19. Análisis de residuales del modelo - escenario # 2 con PrPG = 0.70

Al comparar el modelo propuesto en los 3 escenarios basados en las probabilidades de ocurrencia de diferentes longitudes de paquetes, podemos observar que en la medida que la ocurrencia de paquetes grandes es mayor que la de paquetes pequeños, la utilización del sistema de colas crece de forma progresiva, causando un incremento en el retardo de encolamiento.

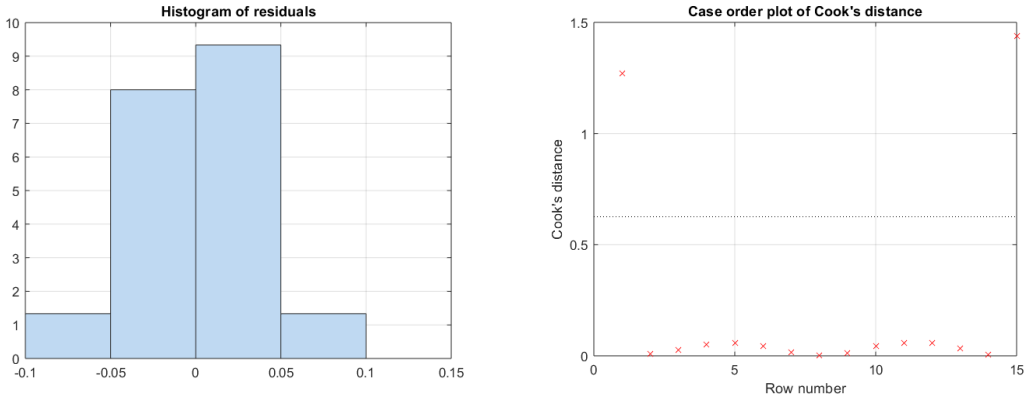


Figura 5.20. Análisis de residuales del modelo - escenario # 3 con PrPP = 0.70

Finalmente, realizamos un contraste de hipótesis del escenario #1 con respecto a la media de LPQ (μ_1) y la media de CBWFQ (μ_2): $H_0: \mu_1 = \mu_2$ $vs.$ $H_0: \mu_1 \neq \mu_2$ (ver figura 5.11). Previo a esto, se tomaron datos adicionales para completar 30 muestras para un mejor análisis, con longitudes de paquete en saltos de 50, entre el rango de 50 a 1500 bytes. En la tabla 5.6 (test de ANOVA) y 5.7 (muestras emparejadas) se muestran los datos estadísticos obtenidos. Podemos observar que el p-valor de la prueba es menor a 0.05, lo que rechaza la hipótesis nula, y se concluye que existe evidencia estadística de que las medias entre LPQ y CBWFQ son diferentes. El modelo LPQ demuestra obtener menores valores para el retardo de encolamiento.

Tabla 5.6. Test de ANOVA entre LPQ y CBWFQ

	Suma de cuadrados	gl	Media cuadrática	F	Valor p
Entre grupos	388772,658	1	388772,658	58,955	0,000
Dentro de grupos	382477,255	58	6594,435		
Total	771249,913	59			

Tabla 5.7. Estadística de muestras emparejadas LPQ vs CBWFQ

		Media	N	Desv. Desviación	Desv. Error promedio
Media	**LPQ**	31,7320	30	22,84171	4,17031
	CBWFQ	192,7232	30	112,54833	20,54842

De igual manera se realiza un contraste de hipótesis entre los 3 escenarios de LPQ, que utilizan diferentes probabilidades entre paquetes grandes y pequeños (ver figura

5.15), para verificar si existen diferencias entre sus medias respectivas: $H_0: \mu_1 = \mu_2 = \mu_3 \ \ vs. H_0: al\ menos\ un\ par\ \mu_i \neq \mu_j$. Esto se documenta en las tablas 5.8 y 5.9. A través del análisis de varianza, el valor p de la prueba es mayor a 0.05 con lo que no se rechaza la hipótesis nula y se concluye que existe evidencia estadística de que las medias son iguales. El modelo LPQ es válido para los 3 escenarios.

Tabla 5.8. Test de ANOVA entre 3 escenarios de LPQ

	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Entre grupos	1923,412	2	961,706	1,925	0,152
Dentro de grupos	43471,402	87	499,671		
Total	45394,813	89			

Tabla 5.9. Estadística de muestras emparejadas entre escenarios de LPQ

			Diferencia de medias (I-J)	Desv. Error	Sig.	Intervalo de confianza al 95%	
						Límite inferior	Límite superior
HSD Tukey	Prpg=0,23	Prpg=0,50	-7,34920	5,77160	0,414	-21,1115	6,4131
		Prpg=0,70	-11,13533	5,77160	0,137	-24,8976	2,6269
	Prpg=0,50	Prpg=0,23	7,34920	5,77160	0,414	-6,4131	21,1115
		Prpg=0,70	-3,78613	5,77160	0,789	-17,5484	9,9761
	Prpg=0,70	Prpg=0,23	11,13533	5,77160	0,137	-2,6269	24,8976
		Prpg=0,50	3,78613	5,77160	0,789	-9,9761	17,5484
DMS	Prpg=0,23	Prpg=0,50	-7,34920	5,77160	0,206	-18,8209	4,1225
		Prpg=0,70	-11,13533	5,77160	0,057	-22,6070	0,3364
	Prpg=0,50	Prpg=0,23	7,34920	5,77160	0,206	-4,1225	18,8209
		Prpg=0,70	-3,78613	5,77160	0,514	-15,2578	7,6856
	Prpg=0,70	Prpg=0,23	11,13533	5,77160	0,057	-0,3364	22,6070
		Prpg=0,50	3,78613	5,77160	0,514	-7,6856	15,2578

CAPÍTULO VI

CONCLUSIONES Y TRABAJO FUTURO

6.1. Conclusiones

Basado en los objetivos de este trabajo y sus fases de investigación, concluimos lo siguiente:

El tráfico de la red internet sigue siendo considerablemente mayor sobre IPv4 que sobre IPv6, actualmente un 70/30 aproximadamente. De igual manera, las aplicaciones utilizan más el transporte TCP antes que UDP (90/10), con una notable variación debido al periodo de cuarentena, teletrabajo, y educación virtual, consecuencia de la pandemia actual del covid-19, con el uso de herramientas sobre todo de streaming. Las longitudes de paquetes más comunes están alrededor de 60 y 1480 bytes. Los paquetes de menor tamaño representan un 50% versus un 43% de los paquetes de mayor longitud. Sin embargo, la carga y consumo de ancho de banda de los paquetes de longitud grande es significativamente mayor. Los patrones de tráfico analizados en la han sido modelados mediante distribuciones de Poisson y se han estimado sus parámetros respectivos. Se ha demostrado que cada patrón a nivel de protocolo y por aplicaciones, puede representarse como la suma de dos distribuciones de Poisson.

En este trabajo se ha establecido un esquema para medir el retardo unidireccional en un enlace punto a punto, utilizando NTP para minimizar la incertidumbre del sincronismo, y la política de QoS CBWFQ con la finalidad de analizar el retardo en función de la variable longitud de paquete. Adicionalmente se hace especial énfasis en el análisis del retardo de encolamiento que agregan los dispositivos intermedios. Se estima el comportamiento predictivo de OWD y del retardo de encolamiento mediante regresiones polinómicas.

Finalmente, se presentó un modelo matemático que representa a un modelo predictivo de QoS, que, utilizando los modelos de tráfico y de retardos obtenidos

previamente, ayuda a reducir el retardo de encolamiento en dispositivos intermedios en un escenario punto a punto. Este modelo definido como LPQ, ha sido simulado mediante MATLAB, y los resultados numéricos obtenidos demuestran que genera un menor retardo de encolamiento comparado con el retardo obtenido de forma experimental usando la política de QoS CBWFQ, aplicada comúnmente por fabricantes de dispositivos intermedios. Estos resultados fueron obtenidos para diferentes niveles de carga, con diferentes tamaños de paquete, y diferentes valores de probabilidad de ocurrencia de longitudes de paquete. El modelo predictivo LPQ no utiliza una fase de marcación como la arquitectura DiffServ, ni esquemas complejos de clasificación de paquetes en función de campos de control, sino que clasifica a los paquetes basados en un campo que ya está establecido como es la longitud, lo que lo hace más sencillo de implementar en dispositivos de borde, y hace un mejor uso del ancho de banda al emplear WRR como política de despacho de paquetes.

6.2. Recomendaciones

Entre las limitantes del desarrollo y modelamiento de LPQ está el hecho de que fue validado mediante una simulación y comparado sus resultados numéricos con un escenario experimental. De allí que se sugiera implementarlo en un dispositivo intermedio usando plataformas de software libre como un ambiente Linux, y obtener una mejor comparación de las políticas de QoS.

Adicionalmente se asumió en el proyecto que la longitud de las colas de paquetes tuviera un tamaño infinito. En la práctica los fabricantes sugieren el equivalente a 500 milisegundos de almacenamiento en los buffers de memoria asignados a las colas de paquetes en espera. Sería recomendable en una implementación verificar el comportamiento del tamaño de los buffers.

Finalmente, se manejaron valores reales de utilización del sistema de colas, asumiendo lapsos de congestión temporal para analizar el comportamiento del retardo de encolamiento. Si la congestión se produjera en lapsos más largos, y para evitar un desbordamiento de los buffers, sería necesario aliviar la congestión mediante el descarte selectivo o aleatorio de los paquetes.

6.3. Aporte realizado

El presente trabajo presenta un modelo predictivo de calidad de servicio que se diferencia de los esquemas tradicionales de DiffServ, evitando las fases de marcación y clasificación, haciéndolo más liviano y eficiente, con la finalidad de reducir el retardo en dispositivos intermedios de enrutamiento. Utiliza para ello modelos predictivos de tráfico real, del retardo unidireccional y del retardo de encolamiento, de los que se obtienen los parámetros de entrada del modelo. Se han presentado la formulación de los modelos indicados para su uso en diferentes áreas de investigación asociadas a redes de datos.

Los fabricantes de dispositivos intermedios para la conectividad de redes de acceso pueden utilizar este estudio e implementar LPQ como una política adicional y de bajo costo. Los proveedores de internet pueden utilizar los modelos de tráfico, retardos y QoS para la implementación de políticas de ingeniería de tráfico en sus redes.

6.4. Trabajo futuro

Frente al contenido y resultados obtenidos en el presente trabajo de investigación, queda abierta la posibilidad de analizar y estudiar las sugerencias presentadas en la sección de recomendaciones de esta tesis, como la implementación en un software de ruteador del modelo LPQ, el manejo ajustable del tamaño de los buffers de memoria, y la utilización del sistema de colas en casos de congestión persistente.

De igual manera, los modelos predictivos de tráfico, del retardo unidireccional y del retardo de encolamiento, basados en el análisis de la longitud de paquetes, pueden aportar a otras áreas de investigación tales como la seguridad de la información y la ingeniería de tráfico.

Los ataques de red pueden manejar patrones que se identifiquen por la longitud de paquetes. Un análisis de estos podría ayudar a mitigar ataques de seguridad que afectan a la disponibilidad de servicios y redes en general, en capas de bajo nivel permitiendo una detección más rápida.

El estudio de los patrones de tráfico y del retardo, puede aportar en el diseño, implementación y gestión de redes, para una operación en condiciones óptimas de las mismas, mejorando parámetros de medición.

BIBLIOGRAFÍA

[1] O. J. S. Parra, A. P. Rios, and G. Lopez Rubio, "Quality of Service over IPV6 and IPV4," 2011, pp. 1–4.

[2] K. Cheshmi, J. Trajkovic, M. Soltaniyeh, and S. Mohammadi, "Quota setting router architecture for quality of service in GALS NoC," in *Proceedings of the 2013 International Symposium on Rapid System Prototyping: Shortening the Path from Specification to Prototype, RSP 2013*, 2013, pp. 44–50.

[3] M. Jutila, "An Adaptive Edge Router Enabling Internet of Things," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1061–1069, Dec. 2016.

[4] H. A. Mohammed, A. H. Ali, and H. J. Mohammed, "The Affects of Different Queuing Algorithms within the Router on QoS VoIP application Using OPNET," *Int. J. Comput. Networks Commun.*, vol. 5, no. 1, pp. 117–124, Feb. 2013.

[5] B. Hartpence, "Packet Guide to Voice over IP: A system administrator's guide to VoIP technologies," 2013. [Online].

[6] J. Evans and C. Filsfils, "Deploying IP and MPLS QoS for multiservice networks [electronic book]: theory and practice / John Evans, Clarence Filsfils.," 2007. [Online].

[7] P. Rukmani and R. Ganesan, "Enhanced low latency queuing algorithm for real time applications in wireless networks," *Int. J. Technol.*, vol. 7, no. 4, pp. 663–672, 2016.

[8] O. J. Salcedo Parra, A. P. Rios, and G. L. Rubio, "IPV6 and IPV4 QoS mechanisms," in *ACM International Conference Proceeding Series*, 2011, pp. 463–466.

[9] T. Szigeti; C. Hattingh; R. Barton; and K. Briley, "End-to-End QoS Network Design: Quality of Service for Rich-Media & Cloud Networks, Second Edition, Video Enhanced Edition," pp. 1–6, 2013.

[10] H. Min, F. Zhou, S. Jui, T. Wang, and X. Chen, "Cisco Visual Networking Index: Forecast and Trends, 2017–2022," *Distribution*, 2018. [Online]. Available: https://es.slideshare.net/itsitio/cisco-visual-networking-index-forecast-and-trends-20172022.

[11] D. Arrowsmith and R. Mondragón, "Modelling Network Data Traffic.", 2005

[12] I. W. C. Lee and A. O. Fapojuwo, "Stochastic processes for computer network traffic modeling," *Comput. Commun.*, vol. 29, no. 1, pp. 1–23, 2005.

[13] A. Dainotti, A. Pescapé, and G. Ventre, “A packet-level characterization of network traffic,” 2006.

[14] G. Mansfield, T. K. Roy, and N. Shiratori, “Self-similar and fractal nature of Internet traffic data,” *Int. Conf. Inf. Netw.*, pp. 227–231, 2001.

[15] R. Pries, F. Warmer, D. Staehle, K. Heck, and P. Tran-Gia, “Traffic measurement and analysis of a broadband wireless internet access,” *IEEE Veh. Technol. Conf.*, 2009.

[16] C. Gandhi, G. Suri, R. P. Golyan, P. Saxena, and B. K. Saxena, “Packet Sniffer-A Comparative Study,” *Int. J. Comput. Networks Commun. Secur.*, vol. 2, no. 5, pp. 179–187, 2014.

[17] S. Maheshwari, K. Vasu, C. Kumar, and S. Mahapatra, “Measurement and Comparative Analysis of UDP Traffic over Wireless Networks,” *Int. Conf. Wirel. Networks*, 2011.

[18] A. Callado *et al.*, “A survey on internet traffic identification,” *IEEE Commun. Surv. Tutorials*, vol. 11, no. 3, pp. 37–52, 2009.

[19] J. Postel, “RFC 791: Internet Protocol,” *Ietf Rfc 791*, 1981. [Online]. Available: http://www.ietf.org/rfc/rfc791.txt.

[20] S. Deering and R. Hinden, “RFC 2460 - Internet Protocol, Version 6 (IPv6) Specification,” *Internet Requests for Comments*, 1998. [Online]. Available: https://tools.ietf.org/pdf/rfc2460.pdf.

[21] C. J. Kale and T. J. Socolofsky, “RFC 1180 - TCP/IP tutorial.” [Online]. Available: https://tools.ietf.org/html/rfc1180.

[22] J. Postel, “RFC 793: Transmission control protocol, September 1981,” *Status: Standard*, 1981. [Online]. Available: https://tools.ietf.org/html/rfc793.

[23] J. Postel, “RFC 768: User Datagram Protocol,” *Network Information Center*, 1980. [Online]. Available: https://tools.ietf.org/html/rfc768.

[24] C. Callahan and L. Lewis, “Infographic: What Happens in an Internet Minute in 2019?” *VisualCapitalists.com*, 2019. [Online]. Available: https://www.visualcapitalist.com/internet-minute-2019/.

[25] Y. D. Goli and R. Ambika, “Network Traffic Classification Techniques-A Review,” *Proc. Int. Conf. Comput. Tech. Electron. Mech. Syst. CTEMS 2018*, pp. 219–222, 2018.

[26] Z. A. G. H. Shaikh, “An Overview of Network Traffic Classification Methods,” *Int. J. Recent Innov. Trends Comput. Commun. IJRITCC*, vol. 3, no. 2, pp. 482–488, 2015.

[27] N. Al Khater and R. E. Overill, “Network traffic classification techniques and challenges,” in *the 10th International Conference on Digital Information Management, ICDIM 2015*, 2016, pp. 43–48.

[28] S. Valenti, D. Rossi, A. Dainotti, A. Pescapè, A. Finamore, and M. Mellia, “Reviewing Traffic Classification,” *Springer*, pp. 123–147, 2013.

[29] M. Zhang, W. John, K. C. Claffy, N. Brownlee, and U. C. S. Diego, “State of the Art in Traffic Classification: A Research Review,” 2009.

[30] W. John and S. Tafvelln, “Analysis of internet backbone traffic and header anomalies observed,” in *Proceedings of the ACM SIGCOMM Internet Measurement Conference, IMC*, 2007, pp. 111–116.

[31] R. Sinha, C. Papadopoulos, and J. Heidemann, “Internet packet size distributions: Some observations,” *USC/Information Sci. Inst. [...*, pp. 1–7, 2007.

[32] W. M. Li, X. N. Liu, C. Miao, L. Y. Chen, and Z. M. Lei, “Packet size distribution of typical internet applications,” *Dianzi Keji Daxue Xuebao/Journal Univ. Electron. Sci. Technol. China*, vol. 43, no. 2, 2014.

[33] A. Hajjar, J. Khalife, and J. Díaz-Verdejo, “Network traffic application identification based on message size analysis,” *J. Netw. Comput. Appl.*, vol. 58, pp. 130–143, 2015.

[34] S. Lee, Y. Won, and D. J. Shin, “On the multi-scale behavior of packet size distribution in internet backbone network,” *NOMS 2008 - IEEE/IFIP Netw. Oper. Manag. Symp. Pervasive Manag. Ubiquitous Networks Serv.*, pp. 799–802, 2008.

[35] H. Kim, K. C. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Y. Lee, “Internet traffic classification demystified: Myths, caveats, and the best practices,” in *Proceedings of 2008 ACM CoNEXT Conference - 4th International Conference on Emerging Networking EXperiments and Technologies, CoNEXT ’08*, 2008.

[36] M. Zhang, M. Dusi, W. John, and C. Chen, “Analysis of UDP traffic usage on internet backbone links,” *Proc. - 2009 9th Annu. Int. Symp. Appl. Internet, SAINT 2009*, pp. 280–281, 2009.

[37] O. J. Adeyemi, S. I. Popoola, A. A. Atayero, D. G. Afolayan, M. Ariyo, and E. Adetiba, “Exploration of daily Internet data traffic generated in a smart university campus,” *Data Br.*, vol. 20, pp. 30–52, 2018.

[38] J. Cao, W. S. Cleveland, D. Lin, and D. X. Sun, “Internet Traffic Tends Toward Poisson and Independent as the Load Increases,” 2003, pp. 83–109.

[39] D. J. Parish, L. Bo, J. M. Sandford, and P. J. Sandford, “using TcP Packet Size Distributions for Application Detection’’ Implementing the Orwell ATM protocol over an optical fibre ring View project Processing of Network Performance Monitoring Data View project Using TCP Packet Size Distributions for Application Detection,” 2006.

[40] F. Liu, Z. Li, and J. Yu, “P2P applications identification based on the statistics analysis of packet length,” *Proc. - 2009 Int. Symp. Inf. Eng. Electron. Commer. IEEC 2009*, pp. 160–163, 2009.

[41] W. Zhang, “Peer-to-peer traffic anti-identification based on packet size,” *Proc. 2011 Int. Conf. Comput. Sci. Netw. Technol. ICCSNT 2011*, vol. 4, pp. 2277–2280, 2011.

[42] V. F. Taylor, R. Spolaor, M. Conti, and I. Martinovic, “AppScanner: Automatic Fingerprinting of Smartphone Apps from Encrypted Network Traffic,” *ieeexplore.ieee.org*, 2016.

[43] Z. Liu, R. Wang, and D. Tang, “Extending labeled mobile network traffic data by three levels traffic identification fusion,” *Futur. Gener. Comput. Syst.*, vol. 88, pp. 453–466, 2018.

[44] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, “Multi-classification approaches for classifying mobile app traffic,” *J. Netw. Comput. Appl.*, vol. 103, pp. 131–145, 2018.

[45] F. Ertam and E. Avcı, “A new approach for internet traffic classification: GA-WK-ELM,” *Meas. J. Int. Meas. Confed.*, vol. 95, pp. 135–142, 2017.

[46] S. Maheshwari, K. Vasu, S. Mahapatra, and C. S. Kumar, “Measurement and Analysis of UDP Traffic over Wi-Fi and GPRS,” 2017.

[47] H. F. Alan and J. Kaur, “Can android applications be identified using only TCP/IP headers of their launch time traffic?,” in *WiSec 2016 - Proceedings of the 9th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2016, pp. 61–66.

[48] N. Vicari, “Modeling of Internet Traffic: Internet Access Influence, User Interference, and TCP Behavior” 2003.

[49] S. Maheshwari, S. Mahapatra, and K. Cheruvu, “Measurement and Forecasting of Next Generation Wireless Internet Traffic,” 2018.

[50] I. W. C. Lee and A. O. Fapojuwo, “Analysis and modeling of a campus wireless network TCP/IP traffic,” *Comput. Networks*, vol. 53, no. 15, pp. 2674–2687, 2009.

[51] C. M. Mueller, “On the importance of realistic traffic models for wireless network evaluations,” 2010.

[52] S. A. Mushtaq and A. A. Rizvi, “Statistical analysis and mathematical modeling of network (segment) traffic,” *Proc. - IEEE 2005 Int. Conf. Emerg. Technol. ICET 2005*, vol. 2005, pp. 246–251, 2005.

[53] A. Dainotti, A. Pescapé, and H. C. Kim, “Traffic classification through joint distributions of packet-level statistics,” 2011.

[54] E. R. S. Castro, M. S. Alencar, and I. E. Fonseca, “Probability Density Functions of the Packet Length for Computer Networks with Bimodal Traffic,” *Int. J. Comput. Networks Commun.*, vol. 5, no. 3, pp. 17–31, 2013.

[55] F. Al-Turjman, A. Radwan, S. Mumtaz, and J. Rodriguez, “Mobile traffic modelling for wireless multimedia sensor networks in IoT,” *Comput. Commun.*, vol. 112, pp. 109–115, 2017.

[56] H. Shi, H. Li, D. Zhang, C. Cheng, and X. Cao, “An efficient feature generation approach based on deep learning and feature selection techniques for traffic classification,” *Comput. Networks*, vol. 132, pp. 81–98, 2018.

[57] J. Cao, Z. Fang, G. Qu, H. Sun, and D. Zhang, “An accurate traffic c lassification model based on support vector machines,” *Int. J. Netw. Manag.*, vol. 27, no. 1, Jan. 2017.

[58] S. Floyd, “RFC 5166: Metrics for the Evaluation of Congestion Control Mechanisms” IETF 2008. [Online]. Available: https://www.ietf.org/rfc/rfc5166.txt.

[59] P. Martinsen, T. Reddy, D. Wing, and V. Singh, “Measurement of Round-Trip Time and Fractional Loss Using Session Traversal Utilities for NAT (STUN),” 2016. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7982.txt.

[60] G. Almes, S. Kalidindi, M. Zekauskas, and A. Morton, “RFC 7679: A One-Way Delay Metric for IP Performance Metrics (IPPM),” 2016. [Online]. Available: https://tools.ietf.org/html/rfc7679.

[61] R. Ip, N. Performance, and D. Points, “RFC 6703 Reporting IP Network Performance Metrics: Different Points of View,” 2017. [Online]. Available: https://tools.ietf.org/html/rfc6703.

[62] C. Adams, P. Cain, D. Pinkas, R. Zuccherato, “RFC 3161 Internet X.509 Public Key Infrastructure Time-Stamp Protocol,” 2001. Available: https://www.ietf.org/rfc/rfc3161.txt

[63] D. L. Mills, “RFC1305: Network Time Protocol (Version 3) specification, implementation and analysis,” *RFC 1305, March* 1992. [Online]. Available: https://tools.ietf.org/html/rfc1305.

[64] *Global Positioning System: Theory and Applications, Volume I*. American Institute of Aeronautics and Astronautics, 1996.

[65] IEEE Standards Association, “1588-2008 - IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems,” 2017. [Online]. Available: https://standards.ieee.org/findstds/standard/1588-2008.html.

[66] Cisco, “Design Best Practices for Latency Optimization,” 2007.

[67] M. Z. Stanikzai, “Types of Delay | CCIE #40010,” 2017. [Online]. Available: http://zahid-stanikzai.com/types-of-delay/.

[68] A. Abdelkefi and Y. Jiang, “A structural analysis of network delay,” in *Proceedings - 2011 9th Annual Communication Networks and Services Research Conference, CNSR 2011*, 2011, pp. 41–48.

[69] J. Liu, “A novel method for estimating the variable and constant components of one-way delays without using the synchronized clocks,” *2014 Int. Conf. Comput. Netw. Commun. ICNC 2014*, pp. 1028–1033, 2014.

[70] A. Abdelkefi and Y. Jiang, “A structural analysis of network delay,” in *Proceedings - 2011 9th Annual Communication Networks and Services Research Conference, CNSR 2011*, 2011, pp. 41–48.

[71] M. Ulbricht and J. Wagner, “Accelerated processing delay optimization in hierarchical networks using low cost hardware,” *2016 10th Int. Symp. Commun. Syst. Networks Digit. Signal Process. CSNDSP 2016*, 2016.

[72] K. Salehin, R. Rojas-Cessa, and K. W. Kwon, “COMPRESS: A Self-Sufficient Scheme for Measuring Queueing Delay on the Internet Routers,” in *2019 International Conference on Computing, Networking and Communications, ICNC 2019*, 2019, pp. 624–629.

[73] B. Ferencz and T. Kovacshazy, “One-way delay measurement system for local area network delay and jitter characterization,” *Proc. 2014 15th Int. Carpathian Control Conf. ICCC 2014*, pp. 14–18, 2014.

[74] A. Csoma, L. Toka, and A. Gulyas, “On lower estimating internet queuing delay,” in *2015 38th International Conference on Telecommunications and Signal Processing, TSP 2015*, 2015, pp. 299–303.

[75] Y. Wang, M. C. Vuran, and S. Goddard, “Cross-layer analysis of the end-to-end delay distribution in wireless sensor networks,” *IEEE/ACM Trans. Netw.*, vol. 20, no. 1, pp. 305–318, Feb. 2012.

[76] A. M. Sukhov, M. A. Astrakhantseva, A. K. Pervitsky, S. S. Boldyrev, and A. A. Bukatov, “Generating a function for network delay,” *J. High Speed Networks*, vol. 22, no. 4, pp. 321–333, Oct. 2016.

[77] R. R. Kompella, K. Levchenko, A. C. Snoeren, and G. Varghese, “Router support for fine-grained latency measurements,” *IEEE/ACM Trans. Netw.*, vol. 20, no. 3, pp. 811–824, 2012.

[78] P. K. Verma and L. Wang, *Voice over IP Networks*, vol. 71. 2011.

[79] C. Vaughan, “Network quality of service,” *Network Quality of Service*, 2012. [Online]. Available: https://ozrobotics.com/shop/network-quality-of-service/.

[80] T. Braun, M. Diaz, J. Enríquez-Gabeiras, and T. Staub, *End-to-End quality of service over heterogeneous networks*. Springer Berlin Heidelberg, 2008.

[81] Y. Ramberg, “RFC 3644 (Policy Quality of Service (QoS) Information Model,” *Request for comments*, 2003. [Online]. Available: https://tools.ietf.org/html/rfc3644.

[82] “RFC 4594 QoS Recommendation > QoS Design Principles and Best Practices | Cisco Press.” [Online]. Available: https://www.ciscopress.com/articles/article.asp?p=2756478&seqNum=7.

[83] S. Floyd and M. Allman, “Comments on the Usefulness of Simple Best-Effort Traffic,” *Internet Engineering Task Force*, 2007. [Online]. Available: http://www.ietf.org/.

[84] IETF Network Working Group, “{Integrated Services in the Internet Architecture: An Overview (RFC 1633)},” 1997. [Online]. Available: https://tools.ietf.org/html/rfc1633.

[85] J. Wroclawski, “RFC 2211: Specification of the Controlled-Load Network Element Service,” 1999. [Online]. Available: https://tools.ietf.org/html/rfc2211

[86] S. Shenker, C. Partridge, and R. Guerin, “RFC 2212: Specification of Guaranteed Quality of Service,” 1999. [Online]. Available: https://tools.ietf.org/html/rfc2212.

[87] J. Rosenberg and P. Kyzivat, “Guidelines for Usage of the Session Initiation Protocol (SIP) Caller Preferences Extension,” 2006. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4596.txt%0Ahttp://www.ietf.org/rfc/rfc4596.txt.

[88] V. Firoiu, W. Courtney, B. Davie, and A. Charny, “RFC 3246 - An Expedited Forwarding PHB (Per-Hop Behavior),” *Internet Engineering Task Force (IETF)*, 2002. [Online]. Available: https://tools.ietf.org/html/rfc3246

[89] J. Heinanen, T. Finland, F. Baker, C. System, and W. Weiss, “RFC 2597 - Assured Forwarding PHB Group,” *IETF - Network Working Group*, 1999. [Online]. Available: https://www.rfc-editor.org/rfc/pdfrfc/rfc2597.txt.pdf.

[90] H. Nichols, B. S., B. F., and B. D., “‘Definition of the Differentiated Services Filed (DiffServ Field) in the IPv4 and IPv6 Headers’, RFC 2474,” 1998. [Online]. Available: https://tools.ietf.org/html/rfc2474.

[91] B. M. Al-shawi and A. Laurent, “QoS Design Principles and Best Practices,” 2019. [Online]. Available: https://www.ciscopress.com/articles/article.asp?p=2756478.

[92] T. Szigeti, C. Hattingh, R. Barton, K. Briley, Jr., “End-to-End QoS Network Design: Quality of Service for Rich-Media & Cloud Networks, Second Edition, Video Enhanced Edition,” 2013. [Online]. Available: https://www.ciscopress.com/store/end-to-end-qos-network-design-quality-of-service-for-9780133116106

[93] S. Varma, “Internet Congestion Control,” *Internet Congestion Control*, 2015. [Online]. Available: https://www.sciencedirect.com/book/9780128035832/internet-congestion-control.

[94] M. Fiedler, T. Hossfeld, and P. Tran-Gia, “A generic quantitative relationship between quality of experience and quality of service,” 2010.

[95] W. M. Kempa, “A direct approach to transient Queue-Size distribution in a finite-buffer Queue with AQM,” *Appl. Math. Inf. Sci.*, vol. 7, no. 3, pp. 909–905, 2013.

[96] G. Gómez, Q. Pérez, J. Lorca, and R. García, “Quality of service drivers in LTE and LTE-A networks,” *Wirel. Pers. Commun.*, vol. 75, no. 2, pp. 1079–1097, 2014.

[97] A. Malik, J. Qadir, B. Ahmad, K. L. Alvin Yau, and U. Ullah, “QoS in IEEE 802.11-based wireless networks: A contemporary review,” 2015.

[98] C. Ghazel and L. Saïdane, “Satisfying QoS requirements in NGN networks using a dynamic adaptive queuing delay control method,” *Procedia Comput. Sci.*, vol. 56, no. 1, pp. 225–232, 2015.

[99] Y. Wenbin, C. Yin, Z. Ming, and W. Dongbin, “QoS-oriented packet scheduling scheme for opportunistic networks,” *J. China Univ. Posts Telecommun.*, vol. 24, no. 3, pp. 51–57, 2017.

[100] V. G. Vassilakis, I. D. Moscholios, and M. D. Logothetis, “Quality of service differentiation in heterogeneous CDMA networks: a mathematical modelling approach,” 2018.

[101] F. Yihunie and E. Abdelfattah, “Simulation and Analysis of Quality of Service (QoS) of Voice over IP (VoIP) through Local Area Networks,” 2018.

[102] M. Gheisari, J. Alzubi, X. Zhang, U. Kose, and J. A. M. Saucedo, “A new algorithm for optimization of quality of service in peer to peer wireless mesh networks,” *Wirel. Networks*, 2019.

[103] A. Botta, A. Dainotti, A. Pescapè, "A tool for the generation of realistic network workload for emerging networking scenarios", Computer Networks (Elsevier), 2012, Volume 56, Issue 15, pp 3531-3547

[104] A. Espinal, R. Estrada, and C. Monsalve, “ Modelling TCP/IP Traffic of a Convergent Campus Wireless Network,” *Int. J. CIRCUITS, Syst. SIGNAL Process.*, vol. 13, pp. 611–616, 2019.

[105] A. Espinal, R. Estrada, and C. Monsalve, “Análisis comparativo y modelamiento del tráfico ip en una red de campus heterogénea,” Revista de Investigación Operacional, vol. 41, pp. 499-504. 2020.

[106] A. Espinal, R. Estrada, and C. Monsalve, “Traffic model using a novel sniffer that ensures the user data privacy,” *MATEC Web Conf.*, vol. 292, p. 03002, 2019.

[107] A. Espinal, R. Estrada, and C. Monsalve, “Traffic analysis of internet applications on mobile devices over lte and wireless networks,” *RISTI - Rev. Iber. Sist. e Tecnol. Inf.*, vol. 2019, no. E22, pp. 81–94, 2019.

[108] J. F. Shortle, J. M. Thompson, D. Gross, and C. M. Harris, Fundamentals of Queueing Theory: Fifth Edition. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2017.

[109] J. Resing, I Adan, Queueing systems, First Edition, vol. 1. Netherlands: Eindhoven University of Technology, 2015.

[110] G. R. Ash, Traffic Engineering and QoS Optimization of Integrated Voice & Data Networks. Elsevier Inc., 2007.

[111] P. Lundqvist, M. Barreiros, QOS-Enabled Networks: Tools and Foundations, 2nd Edition. Wiley series in communications networking & distributed systems, 2016.

ANEXOS