UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

La Universidad Católica de Loja

ÁREA TÉCNICA

MAGÍSTER EN CIENCIAS Y TECNOLOGÍA DE LA COMPUTACIÓN

TRABAJO DE TITULACIÓN

Propuesta de evaluación y mejora de una transferencia confiable en redes MANET

Autor: Jaramillo Campoverde, Byron Gustavo

Director: Torres Tandazo, Rommel Vicente

LOJA - ECUADOR

2021

2021

Aprobación del director del trabajo de titulación

Loja, 15 de agosto del 2021

Dr.
Rommel Vicente Torres Tandazo
Coordinador de programa de posgrados
Ciudad. -

De mi consideración:

El presente trabajo de titulación denominado: “Propuesta de evaluación y mejora de una transferencia confiable en redes MANET” realizado por Byron Gustavo Jaramillo Campoverde, ha sido orientado y revisado durante su ejecución, por cuanto se aprueba la presentación del mismo. Así mismo, doy fe que dicho trabajo de titulación ha sido revisado por la herramienta anti plagio institucional.

Particular que comunico para los fines pertinentes.

Atentamente,

Director del Trabajo de Titulación
Rommel Vicente Torres Tandazo
C.I: 1102885330

Declaración de autoría y cesión de derechos

"Yo, Byron Gustavo Jaramillo Campoverde, declaro y acepto en forma expresa lo siguiente:

- Ser autor del Trabajo de Titulación denominado: Propuesta de evaluación y mejora de una transferencia confiable en redes MANET, del Programa de posgrados Maestría en Ciencias y Tecnologías de la Computación, específicamente de los contenidos comprendidos en: Introducción, Capítulo 1. La transferencia confiable de datos, Capítulo 2. Las redes móviles sin infraestructura (MANET) y la transferencia confiable de datos, Capítulo 3. Rendimiento de los algoritmos para el control de congestión en una red móvil, Capítulo 4. Propuesta de modificación TCP-Algoritmos para el control de la congestión, Capítulo 5. Análisis de rendimiento – TCP algoritmos para el control de la congestión modificado, Conclusiones y Recomendaciones, siendo Rommel Vicente Torres Tandazo, director del presente trabajo; y, en tal virtud, eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones judiciales o administrativas, en relación a la propiedad intelectual. Además, ratifico que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo son de mi exclusiva responsabilidad.
- Que mi obra, producto de mis actividades académicas y de investigación, forma parte del patrimonio de la Universidad Técnica Particular de Loja, de conformidad con el artículo 20, literal j), de la Ley Orgánica de Educación Superior; y, artículo 91 del Estatuto Orgánico de la UTPL, que establece: "Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado que se realicen a través, o con el apoyo financiero, académico o institucional (operativo) de la Universidad".
- Autorizo a la Universidad Técnica Particular de Loja para que pueda hacer uso de mi obra con fines netamente académicos, ya sea de forma impresa, digital y/o electrónica o por cualquier medio conocido o por conocerse, sirviendo el presente instrumento como la fe

de mi completo consentimiento; y, para que sea ingresada al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública, en cumplimiento del artículo 144 de la Ley Orgánica de Educación Superior.

Firma: .....

Autor: Byron Gustavo Jaramillo Campoverde

C.I.: 1103657282

Dedicatoria

A ti mi amor.

Agradecimiento

A Dios, por la vida y conocimientos; a la Universidad Técnica Particular de Loja por el apoyo brindado como parte de la capacitación a su personal; al Dr. Rommel Torres que, en calidad de director y, sobre todo guía, permitió la culminación de este trabajo.

A mis compañeras de la maestría, con quienes compartimos gratos momentos y no apoyamos para alcanzar nuestros objetivos.

Índice de contenido

Carátula I

Aprobación del director del trabajo de titulación II

Declaración de autoría y cesión de derechos III

Dedicatoria V

Agradecimiento VI

Índice de contenido VII

Resumen 1

Abstract 2

Introducción 3

Capítulo uno 5

La transferencia confiable de datos 5

1.1 Características de una transferencia de datos confiable 5
1.2 Protocolo de transferencia confiable en el modelo TCP/IP 6
1.3 Indicadores que definen a una transferencia confiable 8

Capítulo dos 10

Las redes móviles sin infraestructura (MANET) y la transferencia confiable de datos 10

2.1 Redes móviles Adhoc (MANET) 10
2.2 Transferencia de datos confiable en una red móvil 12
2.3 Control de la congestión en TCP 13
2.4 Detección de la congestión 15
2.5 Algoritmos para el control de la congestión 16
2.6 Control de la congestión en una red inalámbrica 20

Capitulo tres 23

Rendimiento de las variantes para el control de la congestión en una red MANET 23

3.1 Definición de indicadores de evaluación y escenarios de simulación 23
3.2 Simulación de escenarios y registro de resultados 24
3.3 Análisis de resultados 35
3.4 Conclusión, rendimiento de los algoritmos para el control de la congestión en una red Manet 37

Capitulo cuatro 38

Propuesta de optimización de TCP para MANET 38

4.1 Comportamiento del algoritmo de control de congestión original 38
4.2 Optimización de New Reno 39
4.3 Análisis de rendimiento New Reno y New Reno Optimizado 42
4.4 Simulación comparativa y resultados 43
Conclusiones 47
Recomendaciones 48
Referencias 49
Apéndice 53

Índice de Tablas

Tabla 1 - Definición de indicadores de evaluación 23
Tabla 2 - Parámetros de simulación y escenarios 24
Tabla 3 - Registro de valores de los indicadores 32
Tabla 4 - *Comparativo porcentual, rendimiento de los algoritmos de control de congestión, escenario AODV.* 36
Tabla 5 Comparativo porcentual, rendimiento de los algoritmos de control de congestión, escenario OLSR 36
Tabla 6 - *Definición de escenarios de evaluación detallados.* 43
Tabla 7 - *Resultados comparativos New Reno Original vs. New Reno Modificado* 43
Tabla 8 - Comparativo de variación ACK Delayed 53

Índice de Figuras

Figura 1 - Esquema de un canal fiable 6
Figura 2 - Esquema general de control de congestión de TCP – adaptado Sharma 2016 14
Figura 3 - Detección de la congestión de TCP – adaptado Kurose 2017 15
Figura 4 - Esquema general algoritmo arranque lento – adaptado Kurose 2017 17
Figura 5 - Máquina de estados finitos para control de congestión – adaptado Jero 2018 19
Figura 6 - Control de congestión con medios de conexión mixtos (Guiados e inalámbricos). 21
Figura 7 - Esquema base de simulación 25
Figura 8 - Tráfico recibido - en bytes, nodo receptor 26
Figura 9 - Latencia Wireless LAN – en segundos (media) – Nodo receptor 26
Figura 10 - Variación de la Cwnd – Nodo emisor en bytes 27
Figura 11 - Contador de retransmisiones (media) – Nodo emisor 28
Figura 12 - Throughput Wi-Fi - en bits/seg (media) – Nodo emisor 29
Figura 13 - Tráfico recibido - en bytes (media) – Nodo receptor 29
Figura 14 - Latencia Wireless LAN – en segundos (media) – Nodo receptor 30

Figura 15 - Variación de la Cwnd – Nodo emisor en bytes 31
Figura 16 - Contador de retransmisiones (media) – Nodo emisor 31
Figura 17 - Throughput Wi-Fi - en bits/seg (media) – Nodo emisor 32
Figura 18 - Comparativa gráfica de resultados de la simulación.....33
Figura 19 - Comparativa porcentual total del rendimiento de las variantes de control de congestión.....36
Figura 20 - ACK Delay.....40
Figura 21 - Comportamiento ACK Delayed, original.....41
Figura 22 - Comportamiento de disminución del ACK Delayed 42
Figura 23 - Tráfico recibido-New Reno Original vs. New Reno Modificado–AODV.....44
Figura 24 - Latencia - New Reno Original vs. New Reno Modificado – AODV.....44
Figura 25 - Cwnd - New Reno Original vs. New Reno Modificado – AODV 45
Figura 26 - Contador transmisiones- New Reno Original vs. New Reno Modificado – AODV 45
Figura 27 - Throughput - New Reno Original vs. New Reno Modificado – AODV.....46
Figura 28 - Comparativo, variación ACK Delayed 54

Resumen

La transferencia de datos confiable tiene como base de su funcionamiento al Protocolo de Control de Transferencia (TCP). El presente trabajo, tuvo como objetivo identificar las características de una transferencia confiable de datos en redes MANET, simularlas y proponer alternativas de modificación orientadas a mejorar el rendimiento. Se evalua la ventana de congestión, throughput, tráfico recibido, retardo y cantidad de retransmisiones; se realizó la simulación del comportamiento de las variantes de congestión de TCP (Tahoe, Reno, New Reno y CUBIC) usando como herramienta a OPNET v.17.5 y, se determinó porcentualmente a New Reno como el protocolo que presenta mejores resultados (96,02% de cumplimiento de los indicadores). Sobre éste y, tomando en cuenta las definiciones del RFC 2581, se implementaron ajustes a la variable ACK Delayed. Como resultados, para el tamaño de ventana de congestión se obtuvo un incremento de 8,82%; throughput 7,45% y tráfico recibido 10,67%; para los indicadores de cantidad de retransmisiones y latencia, se tuvo disminución de 2,75% y 7,61% respectivamente, demostrando que, TCP en sus características de control de congestión puede ser modificado positivamente para entornos MANET.

*Palabras claves***:** MANET, ACK Delayed, New Reno.

Abstract

Reliable data transfer is based on the Transfer Control Protocol (TCP). The present work aimed to identify the characteristics of a reliable data transfer in MANET networks, simulate them and propose modification alternatives aimed at improving performance. The congestion window, throughput, received traffic, delay and number of retransmissions are evaluated; The behavior of TCP congestion variants (Tahoe, Reno, New Reno and CUBIC) was simulated using OPNET v.17.5 as a tool, and New Reno was determined in percentage terms as the protocol with the best results (96.02 % compliance with the indicators). On this and, taking into account the definitions of RFC 2581, adjustments to the variable ACK Delayed were implemented. As results, for the congestion window size an increase of 8.82% was obtained; throughput 7.45% and received traffic 10.67%; for the number of retransmissions and latency indicators, there was a decrease of 2.75% and 7.61% respectively, showing that TCP in its congestion control characteristics can be positively modified for MANET environments.

Keywords: MANET, ACK Delayed, New Reno.

Introducción

El transporte confiable en una comunicación de datos está bajo responsabilidad del Protocolo de Control de Transferencia (TCP, por sus siglas en inglés), que tiene como variantes de control para la congestión a Tahoe, Reno, New Reno y Cubic, entre los principales y, oficialmente implementados en los diversos sistemas operativos (RFC 2581). El proceso de percibir que existe congestión en la red utilizado por estas variantes es el registro de pérdida de paquetes o llegada de acuses de recibo duplicado. Este método es eficiente en redes estables (pérdidas menores al 1% por medio errores en capas 3, 2 y 1 – modelo OSI), pero, en redes MANET (sin infraestructura e Inalámbricas), las tasa de errores puede llegar hasta el 10%, lo que genera una interpretación errónea de congestión en la red, como respuesta, las variantes de control de congestión reducen o controlan el tamaño de la ventana de congestión, limitando, a veces innecesariamente, el uso del medio de transmisión. Con base a lo expuesto, se plantea el presente trabajo de investigación que tiene como objetivo general "Identificar las características de transporte confiable en una red MANET" y, tomando como base la teoría y trabajos similares por otros autores, se definió como características de transporte confiable a los indicadores tamaño de ventana de congestión, throughput, delay, cantidad de retransmisiones y tráfico recibido, cumpliendo así el objetivo planteado.

Respecto a los objetivos específicos se tiene:

a. Plantear una estrategia de validación de transporte confiable en redes MANET; con el desarrollo de dos escenarios que combinan las variantes de control de congestión y y, usando como herramienta de simulación a Opnet 17.5 versión educativa, se obtuvo diferencias en el comportamiento en los indicadores.

b. Proponer e implementar mejoras a la estructura y funcionamiento de una transferencia confiable en redes MANET; con base a las definiciones en el RFC 2581 respecto a la variable ACK Delayed y, tomando en cuenta el comportamiento de la estructura del envío y recepción de ACKs, se propuso ajustes a esta variable.

c. Emular la transferencia confiable bajo el nuevo esquema de mejoras; tomando en cuenta la estrategia de validación definida en el objetivo específico del literal (a), se comparó los indicadores obtenidos en las variante modificada (literal b).

Dentro de las dificultades encontradas durante el desarrollo de la investigación, se puede citar la construcción del escenario de simulación, definición de los indicadores a evaluar en el rendimiento de la simulación y, la interpretación de los resultados. Para el desarrollo de la investigación y, como metodología de análisis, se planteó la simulación de varios escenarios de transferencia de datos usando TCP mediante la herramienta Opnet 17.5 versión educativa. A través del registro de valores de los indicadores definidos como claves para la evaluación de una transferencia confiable en una red MANET, se definió porcentualmente la variante de Control de Congestión que presenta mejores resultados globales; luego, se plantea mejoras a sus características originales y se procede nuevamente con la simulación y comparación con su versión inicial, obteniendo así, los resultados finales que permitieron determinar una propuesta de mejora a la transferencia confiable en redes MANET. El trabajo tiene cuatro capítulos que, en resumen, abarcan, capítulo 1 describe desde el punto de vista teórico y base, la importancia, ventajas y desventajas de mantener los controles para garantizar la entrega de la información; en el capítulo 2 se abarca las características de una red MANET y su comportamiento frente a una transferencia confiable; en el capítulo 3, procedemos con la definición de los indicadores y escenarios de simulación, registramos y analizamos los resultados. Ya en el capítulo 4, se desarrolla la modificación que planteamos a la variante de control de la congestión que mejores resultados brindó en las simulaciones y, realizamos nuevamente la simulación de los escenarios tomando en cuenta la variante original frente a la modificada.

La investigación tiene relevancia desde la perspectiva de desarrollo de un nuevo proceso de control de congestión para redes MANET, con base en la cantidad de usuarios móviles existentes en la actualidad 5,22 billones (revista MarketingEComerce, 2021), la mejora a los procesos de comunicación en redes MANET brindará estabilidad y mejor rendimiento a las aplicaciones en este entorno.

Capítulo uno

La transferencia confiable de datos

1.1 Características de una transferencia de datos confiable

Cuando hablamos de una transferencia confiable de datos, nos enfocamos en que la trasmisión de información entre el origen y el destino, no presente variación en su estructura, para algunos investigadores la transferencia confiable se la puede representar como, la abstracción del servicio proporcionada a las entidades de la capa superior es la de un canal fiable a través del cual se pueden transferir datos. Disponiendo de un canal fiable, ninguno de los bits de datos transferidos está corrompido (cambia de 0 a 1, o viceversa) ni se pierde, y todos se entregan en el orden en que fueron enviados (Vadivel, 2016).

Desde el análisis del modelo de referencia OSI (Open System Interconection) el servicio de una conexión fiable se define en la Capa de Transporte tomando en cuenta que la capa subyacente, la Capa de Red, tiene como característica de transmisión el “mejor esfuerzo”, es decir, no garantiza la entrega de los paquetes y, ése es precisamente el propósito de la capa de transporte: ofrecer un servicio confiable en una red no confiable.

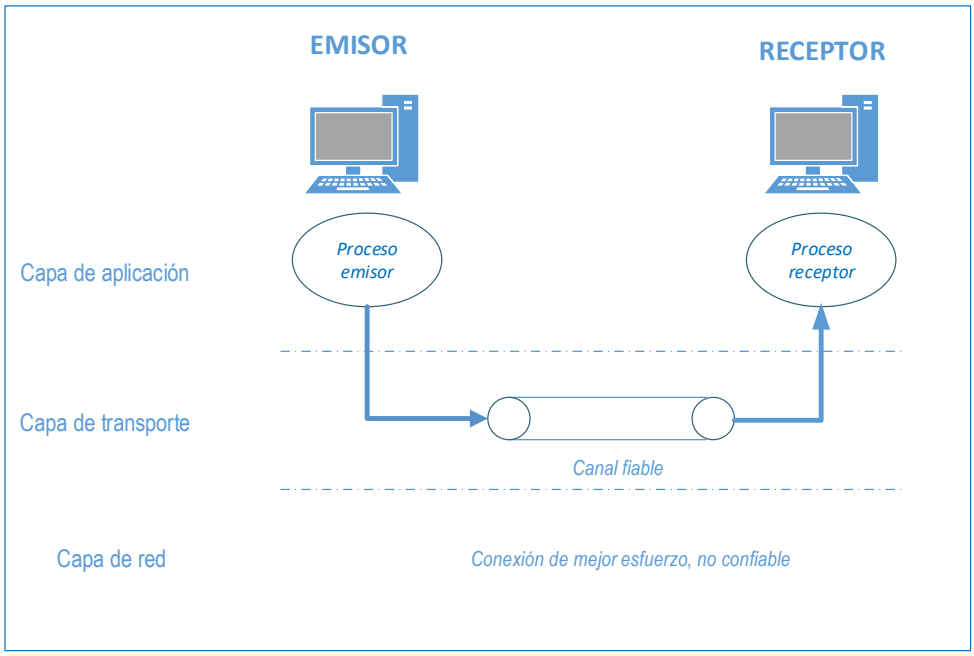
Bajo esta definición y, tomando en cuenta la pila de protocolos de Internet TCP/IP, la transferencia de datos confiable se puede implementar en diferentes niveles de la transmisión de la información, desde la capa de Aplicación, hasta la capa de Enlace o Acceso a la red. Para Kurose (2017), la transferencia de datos fiable, encabeza los 10 problemas que afectan a las redes de datos; es decir, la presencia de un protocolo o proceso que garantice la entrega fiable de datos entre dispositivos, es de vital importancia para la correcta transmisión de los datos.

En la figura 1, se puede evidenciar la importancia de la implementación de un canal fiable en la comunicación, sobre todo si tomamos en cuenta que el Protocolo de Internet (IP), por sus características propias, es un protocolo de mejor esfuerzo; es decir, no garantiza la entrega de los paquetes entre el emisor y el receptor, teniendo como respaldo o soporte, al

Protocolo de Control de Transferencia (TCP – por sus siglas en inglés) para ejecutar procesos que brinden fiabilidad a los procesos de comunicación.

Figura 1

Esquema de un canal fiable



Nota. Adaptado de *Transferencia de datos fiable* de Kurose (2017)

El objetivo fundamental de la capa de transporte es proporcionar un servicio de transmisión de datos extremo a extremo, confiable, libre de errores y eficiente, características propias de la capa de aplicación. La capa de transporte se apoya en los servicios de capas subyacentes (capa de red), estos normalmente están alojados en el núcleo del sistema operativo, en alguna librería de aplicaciones de red, en procesos de usuario o, en último de los casos, en la propia tarjeta de red; normalmente, en Internet, se implementan en el núcleo del sistema operativo y en librerías de aplicaciones de red. (Tanenbaum, 2012).

1.2 Protocolo de transferencia confiable en el modelo TCP/IP

El Protocolo de Control de Transmisión (TCP, por sus siglas en inglés), fue desarrollado a finales del año 81, con el objetivo de brindar un flujo de datos confiable de

extremo a extremo (host a host) usando una infraestructura no confiable (capas inferiores). Bajo este concepto, TCP fue diseñado para adaptarse a los cambios que pudieran surgir durante una transmisión (Tanenbaum, 2012).

(Kurose, 2017) TCP es el protocolo de transferencia fiable con el que cuenta la capa de Transporte, las características son descritas continuación:

a. Acuses de recibo (ACKs y NACKs). - TCP cuenta con un tipo de segmento que permiten al receptor informar a su emisor sobre la recepción de un segmento o, en otros casos, la recepción fallida de un segmento, es decir, con errores; esta característica permite al emisor identificar el segmento que debería retransmitir.

b. Números de secuencia. - Como los segmentos que se transmiten pueden usar diferentes rutas hasta llegar al destino, no se puede asegurar un arribo ordenado, es decir, en el mismo orden en que se enviaron, TCP usa un campo llamado número de secuencia, el cual le permite identificar el orden del segmento y reagrupar ordenadamente los segmentos previos a su paso a la capa de aplicación.

c. Establecimiento y cierre de la sesión. - TCP cuenta con procesos propios para que el emisor “contacte” al receptor previo al envío de la información (segmentos de inicio de sesión – SYN) y, una vez terminada la transferencia de los datos, TCP permite informar al receptor y emisor, que, la transferencia de datos ha finalizado (segmentos de finalización – FIN).

d. Control de flujo. - Como una conexión TCP es una comunicación lógica entre los dispositivos finales (hosts), éstos tienen recursos limitados, como CPU, memoria, entre otros; en este contexto, TCP es capaz de regular la cantidad de segmentos enviados con el objetivo de no desbordar los recursos del hardware en el receptor.

Como se ha indicado, las capas subyacentes a la capa de transporte, no garantizan la entrega de segmentos de forma correcta o, algún tipo de señalización que advierta a los nodos sobre la velocidad o capacidad con la que se debe enviar la información entre ellos;

corresponde a TCP (ejecutado en cada host terminal) sensar el estado del canal y determinar la velocidad de envío evitando la congestión o, en este tipo de situaciones, realizar las correcciones necesarias para minimizar la pérdida de información; además, TCP es responsable de la gestión (inicio y finalización) de temporizadores y retransmitir los segmentos que no hayan llegado hasta su destino (o no se haya recibido el acuse respectivo). En el caso de segmentos que llegen hasta su destino, éstos pueden arribar en un orden diferente al cual fueron enviados, TCP, usando variables propias (números de secuencia), es capaz de de reensamblarlos de forma ordenada previo al envío a capa de aplicación.

En resumen, TCP debe proporcionar un buen desempeño con la confiabilidad que la mayoría de las aplicaciones desean y que IP no proporciona (Tanenbaum, 2012).

1.3 Indicadores que definen a una transferencia confiable

Referente a las variables que brindan información sobre al rendimiento de una transferencia confiable, Sing-Borrajo (2014), menciona que las métricas más importantes con el fin de evaluar el desempeño real de una red son: el retardo, el jitter y la pérdida de paquetes, esto es reafirmado y complementado por Govindarajan (2017) en el que indica que, para objetivar la evaluación del desempeño de una red y poder establecer conclusiones cuantitativas, los indicadores medibles recomendados son:

- Paquetes perdidos, es un paquete que ha sido transmitido al núcleo de la red pero que nunca sale de la red para llegar a su destino (Sharma, 2016).
- Paquetes recibidos, es un paquete que ha sido transmitido al núcleo de la red y que llegó a su destino (Rath, 2016).
- Paquetes duplicados, es un paquete que ha sido transmitido al núcleo de la red y que llegó a su destino más de una vez (Sirajuddin, 2016).
- Latencia, es el tiempo que le toma a un paquete llegar hasta su destino (Bhatia, 2015).

- Variación del tiempo de restablecimiento, es el tiempo de recuperación del envío de información, o restitución del enrutamiento que pudo afectar de manera externa a la transmisión de datos en la capa de transporte (Torres, 2009).
- Tamaño de ventana, es la cantidad de bytes que el dispositivo de destino de una sesión TCP puede aceptar y procesar al mismo tiempo (Kurose, 2017).
- Caudal efectivo (Throughput), es el volumen de trabajo o de información neto que fluye a través de un sistema (Govindarajan, 2017).

Es importante valorar cada uno de estos indicadores, así como la relación entre ellos y determinar objetivamente la efectividad del protocolo que se desea evaluar; por ejemplo, si el tamaño de ventana calculado por el algoritmo de control de congestión es menor al Throughput del canal, tendremos una capacidad subutilizada; por el contrario, si es mayor, el contador de retransmisiones y congestión del canal se afectará (Torres, 2009).

Capítulo dos

Las redes móviles sin infraestructura (MANET) y la transferencia confiable de datos

2.1 Redes móviles Adhoc (MANET)

Las redes inalámbricas, desde el punto de vista de los elementos que la componen, pueden operar en modo infraestructura o, en modo Adhoc. Las primeras, son aquellos tipos de redes que cuentan con una infraestructura para la comunicación de los nodos, por ejemplo, un *Access Point*. Como contra parte, en las redes Adhoc, los nodos no cuentan con una infraestructura para la comunicación, como consecuencia de esta característica, los propios nodos brindan servicios como la resolución de nombres de dominio (DNS) o la asignación dinámica de direccionamiento IP (DHCP). Bajo la definición de las redes Adhoc, existe la posibilidad de que los nodos tengan movimiento y, cuando un nodo se desplaza desde la cobertura de una estación o nodo diferente al cual se encuentra asociado, cambia su punto de conexión al nuevo nodo, esto se conoce como transferencia (han off) y, es una de las características principales de las redes móviles sin infraestructura o, redes MANET (Mobile Adhoc Network).

Se puede clasificar a las redes inalámbricas, tomando en cuenta estos dos criterios base:

a) Si un paquete dentro de la red realiza un único salto o varios saltos inalámbricos.

b) Si existe infraestructura de comunicación.

Bajo estos criterios, se puede combinar las definiciones y tendremos una clasificación de las redes inalámbricas con el fin de caracterizar a las redes MANET, así:

1. *Redes basadas en infraestructura y un único salto*. - Son redes que tienen una estación base conectada a la red mayor (red cableada o Internet) y, los nodos inalámbricos realizan un solo salto inalámbrico para alcanzar la conectividad a equipos remotos, por ejemplo 802.11.

2. *Redes sin infraestructura y un único salto*. - En este tipo de redes no existe la estación base, pero, existe un único nodo que coordina las transmisiones con el resto de nodos, por ejemplo, redes Bluetooth.

3. *Redes basadas en infraestructura y múltiples saltos*. - Existe una estación base conectada a la red mayor, sin embargo, los nodos pueden tener que realizar la transferencia de datos mediante otros nodos y no directamente, por ejemplo, las redes de sensores.

4. *Redes sin infraestructura y múltiples saltos*. - En este tipo de redes, no existe una estación base y, los nodos deben retransmitir los datos de otros nodos para alcanzar cierto destino; este tipo de redes son las conocidas como MANET (Mobile Ad Hoc network).

Bajo estas premisas, se puede definir a una red MANET como la colección de nodos inalámbricos que, para conectarse o intercambiar información entre ellos, no requieren de infraestructura de comunicación; además, estos nodos son móviles, es decir, pueden cambiar su ubicación de manera aleatoria o controlada, inclusive, pueden salir o volver a ingresar a la topología de comunicación sin control alguno. Cada uno de los nodos puede ser un transmisor o receptor, e inclusive, puede ser simplemente un rúter de paso de información (Sing-Borrajo, 2014).

Otra de las características importantes en los nodos de una red MANET, es su acceso a la energía; (Rath 2016) los nodos de las redes móviles ad hoc están equipados con batería limitada que disminuye gradualmente a medida que se ejecutan tareas de procesamiento, por ejemplo, enviando y recibiendo mensajes de saludo, control información, paquetes de datos, reenvío de paquetes y mensajes, procesamiento de la lógica de enrutamiento, entre otros; muchos protocolos de enrutamiento y otros procesos propios de la transferencia de datos, son propuesto por muchos investigadores eminentes basados en diferentes parámetros de evaluación como consumo de energía, retraso, técnicas de gestión, etc., teniendo así, un vasto alcance de análisis que va, desde el comportamiento físico de los componentes del nodo, hasta el procesamiento lógico de los procesos internos de éste. En el presente trabajo,

nos centraremos en el análisis lógico de los procesos de comunicación en Capa de Transporte, quedando temas físicos como, consumo de energía del nodo, procesamiento, uso de memoria, entre otros, fuera del alcance y objetivos planteados.

2.2 Transferencia de datos confiable en una red móvil

Las redes móviles es un área de las telecomunicaciones que está presentando un auge muy importante; según datos de la revista MarketingECommerce, hasta enero 2021, existen 5,22 billones de usuarios móviles en el Internet, lo que representa en 66% de la población mundial, teniendo un incremento del 1,8% respecto a enero del 2020. Bajo esta realidad, se puede concluir que existe una gran cantidad de dispositivos móviles conectados a Internet y, que, por sus características, estos cuentan con los protocolos para generar una red MANET.

Recordemos que, para tener una transferencia de datos confiable, el modelo de referencia OSI y TCP/IP, definen esta característica en la capa de Transporte usando el protocolo TCP.

TCP fue desarrollado en los años 80s y posteriormente optimizado, pero siempre fue enfocado en redes con medios de transmisión guiados; en las redes fijas actuales las pérdidas de paquetes son debidas, fundamentalmente, a situaciones de congestión (ya que las tasas de error son muy bajas), las implementaciones modernas del TCP asocian cualquier error a este problema e incorporan toda una serie de mecanismos destinados a combatirlo. Ahora bien, en los entornos móviles las pérdidas de paquetes son muy frecuentes debido a las altas tasas de error y a las desconexiones temporales asociadas al traspaso de las llamadas; por lo tanto, los algoritmos de control de la congestión que TCP activa al detectar los errores, pueden ocasionar un comportamiento muy poco eficiente, cuyos efectos deben ser evaluados (Rodriguez, 2016).

El supuesto al que se asocia la pérdida de segmentos, es que la conexión TCP atraviesa una red congestionada, es aceptable en los enlaces cableados donde la tasa de error es mínima (del orden de $1x10^{-12}$ en el caso de enlaces de fibra óptica y $1x10^{-9}$ en el

caso de UTP Cat.5), pero no se puede extender al caso de enlaces con tasas de errores mayores, del orden de $1x10^{-6}$ o peores (por ejemplo inalámbricos), donde la probabilidad de pérdida de un segmento debido a problemas en el medio físico deja de ser despreciable.

2.3 Control de la congestión en TCP

TCP al ser un protocolo de transporte que implementa control de congestión, debe ser independiente de la red subyacente y de las tecnologías de capa de enlace. Teóricamente válido, pero en la práctica las redes inalámbricas pueden presentar problemas. El tema puntual es que la pérdida de paquetes se usa con frecuencia como señal de congestión. Las redes inalámbricas pierden paquetes todo el tiempo debido a errores de transmisión (Tanenbaum 2012).

Cuando se analiza una situación de congestión en la red, hay que entender el origen de esta información, según Leemaroselin (2015), puede originarse de dos maneras:

- Control de congestión de terminal a terminal, en este método, el control de congestión debe ser inferido por los sistemas terminales basándose únicamente en el comportamiento observado de la red (pérdida de paquetes y retardos).
- Control de congestión asistido por la red; en este método, los enrutadores ofrecen información al terminal emisor/receptor informando el estado de congestión en la red.

Las variantes de control para la congestión predeterminadas de TCP, adoptan el control de la congestión de terminal a terminal, sin embargo, TCP puede implementar de manera opcional el control de congestión asistido por la red.

A continuación, se presenta un esquema de control de la congestión con TCP, adaptado de Sharma, 2016.

Figura 2

Esquema general de control de congestión de TCP – adaptado Sharma 2016

Como se indica en la figura 2, el emisor tiene como característica el poder identificar/percibir la congestión en la red, así mismo, puede identificar/percibir que ésta no existe y, de esta manera, usar los medios de transmisión al máximo; los métodos propuestos para el control de congestión en una red se pueden agrupar en dos categorías según Rodriguez (2015).

a. Control de la Congestión (mecanismo reactivo) que busca restablecer la normalidad de uso del canal cuando la congestión ha sido detectada, y,

b. Evitación de la Congestión (mecanismo proactivo) que buscan evitar la congestión en el canal.

Para realizar estas tareas; es decir, variar la velocidad de la transmisión (incremento/decremento), llegamos a identificar algunas variables de las cuales TCP hace uso para cumplir su objetivo; entre las principales:

- Ventana de congestión (cwnd); impone una restricción sobre la capacidad que el emisor puede enviar tráfico a la red.
- Retrasmision timer (RTT); con esta variable (timer) se busca asegurar la entrega de datos frente al hecho de no tener ninguna realimentación desde el receptor de los datos

- Retrasmision timeout (RTO); es la duración de esta variable (timer). Un segmento no debe ser retransmitido hasta RTO después de su transmisión previa.
- ACKs duplicados
- Temporizadores

Estas variables, sufren cambios durante la ejecución de una transmisión y, son las responsables de controlar la congestión en la red.

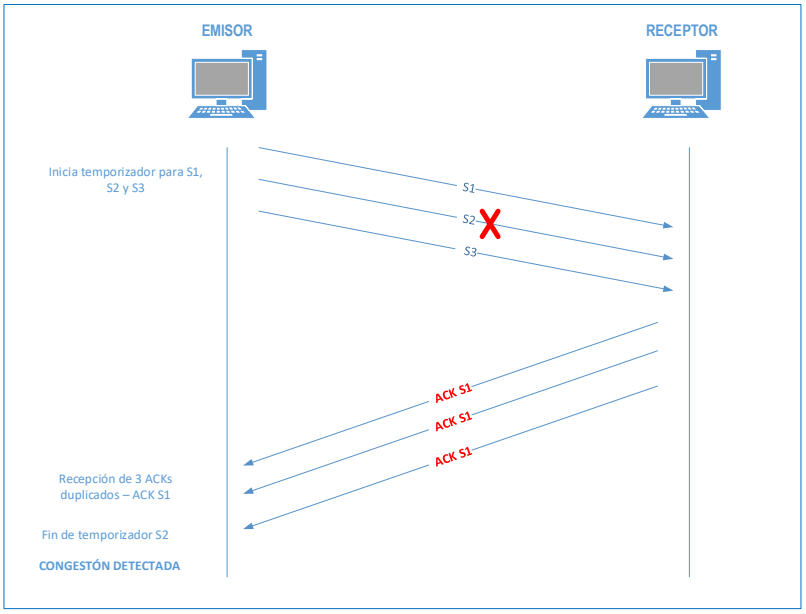
2.4 Detección de la congestión

Según Kurose (2017), TCP no ejecuta un control de la congestión asistido por la red, al contrario, el proceso de control de congestión que TCP realiza, es terminal a terminal, la razón, la capa de red no suministra información detallada que pueda ser utilizada por los nodos terminales como apoyo para determinar la existencia o no de congestión en el canal, por lo tanto, el proceso utilizado se describe a continuación:

1. El emisor percibe un “suceso de pérdida” validado por dos variables, fin de temporizador o la recepción de tres ACKs duplicados procedentes del receptor.
2. El emisor pasa al estado “congestión detectada”.
3. El emisor dispara un proceso de control de congestión.

Figura 3

Detección de la congestión de TCP – adaptado Kurose 2017



Una vez que TCP ha detectado que existe congestión en la red, se procede a modificar la variable “ventana de congestión”, el cual puede reducir al mínimo o a la mitad, dependiendo de la variante de control de congestión que esté utilizando. Es importante mencionar que, en el caso de que TCP no detecte congestión en la red; es decir, los reconocimientos de los segmentos y temporalizadores fluyen de manera correcta, la variable “ventana de congestión” se incrementa, esto con el fin de optimizar el uso de la capacidad del canal. Similar que, en el caso del control de la congestión, TCP usará el algoritmo definido para realizar estos incrementos.

Para Kong, 2018, TCP tiene como objetivo de no saturar el canal, cuando exista algún tipo de problema en la transmisión o, no subutilizar la capacidad total, cuando los segmentos fluyen de manera correcta, para lograr esto se basa en los siguientes principios:

a. Un segmento perdido implica congestión y, por tanto, la velocidad del emisor TCP debe reducirse cuando se pierde un segmento.

b. Un segmento que ha sido reconocido indica que la red está entregando los segmentos del emisor al receptor y, por tanto, la velocidad de transmisión del emisor puede incrementarse cuando llega un paquete ACK correspondiente a un segmento que todavía no había sido reconocido.

c. Tanteo del ancho de banda; validando un ACK fallido, reduce la velocidad, pero intenta nuevamente incrementar hasta encontrar una nueva falla.

2.5 Algoritmos para el control de la congestión

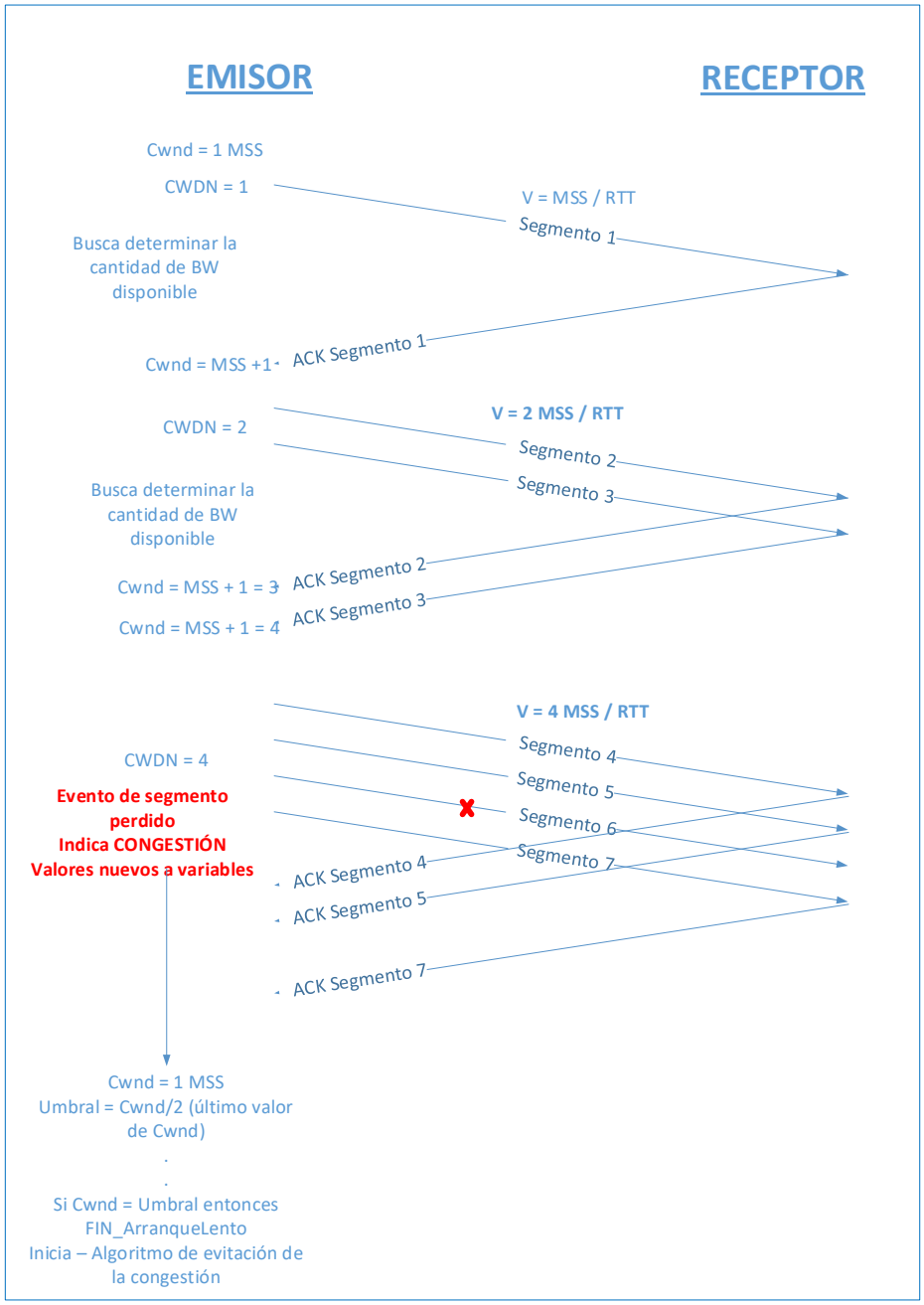
Con el objetivo de llegar a evaluar la congestión en la red, TCP utiliza tres algoritmos que interactúan entre sí, estos son arranque lento (slow start), evitación de la congestión (congestion avoidance) y recuperación rápida (fast recovery). A continuación, una explicación del funcionamiento de cada uno.

a. Arranque lento (slow start). - El algoritmo de arranque lento realiza su tarea con tres variables principales, ventana de la congestión (*Cwnd*), temporizadores

(contador de ACKs duplicados, RTT, etc.) y umbral (*Cwnd*/2); gráficamente, la ejecución global de este proceso se describe en la figura 4.

Figura 4

Esquema general algoritmo arranque lento – adaptado Kurose 2017



En resumen, el emisor inicia la transmisión con un valor de *Cwnd* = 1 MSS (Tamaño Máximo de Segmento, por sus siglas en inglés). MSS es un valor relacionado con la Unidad Máxima de Transferencia (MTU) y, por defecto en TCP tiene un valor de 1460 bytes (MSS = MTU – Cabecera IP (20 bytes) – Cabecera

TCP (20 bytes), éste valor se va incrementando en 1 MSS por cada reconocimiento positivo recibido; este proceso se mantiene hasta que, el emisor detecta la pérdida de un segmento (fin de temporizador), esto le indica que existe congestión en la red; a continuación, el emisor regresa el valor de *Cwnd* a 1 MSS e inicia nuevamente el proceso de arranque lento. Adicional, incluye una nueva variable, umbral, que tiene el valor de *Cwnd/*2 (valor de *Cwnd* cuando se detectó la congestión); continúa la transmisión de segmentos hasta que cwnd iguala a umbral, momento en el cual, inicia el proceso de evitación de la congestión.

b. Evitación de la congestión (congestión avoidance).- El objetivo de este proceso es, una vez que *Cwnd* es igual al *umbral*, realizar incrementos discretos que no presenten de forma inmediata una nueva congestión (y volver al arranque lento); en este caso, una vez que la ventana de congestión iguala el valor del umbral (valor de la ventana de congestión durante la última pérdida de un segmento), ésta se calcula de acuerdo a la ecuación (1) por cada confirmación de segmento:

$$Cwnd_{nuevo} = Cwnd + \frac{1MSS}{Cwnd} \qquad (1)$$

Este proceso se ejecuta hasta detectar un nuevo proceso de congestión; por ejemplo, si éste se da por un nuevo segmento perdido, se ejecuta el proceso de arranque lento definiendo nuevos valores a las variables de *Cwnd* y *umbral*; pero, si el nuevo proceso de congestión se da por la recepción de tres confirmaciones duplicadas (ACKs duplicados), TCP define que el nuevo valor para la ventana de congestión de acuerdo a (2) y (3):

$$Umbral = \frac{Cwnd}{2} \qquad (2)$$

$$Cwnd = \frac{Cwnd}{2} + 3MSS \qquad (3)$$

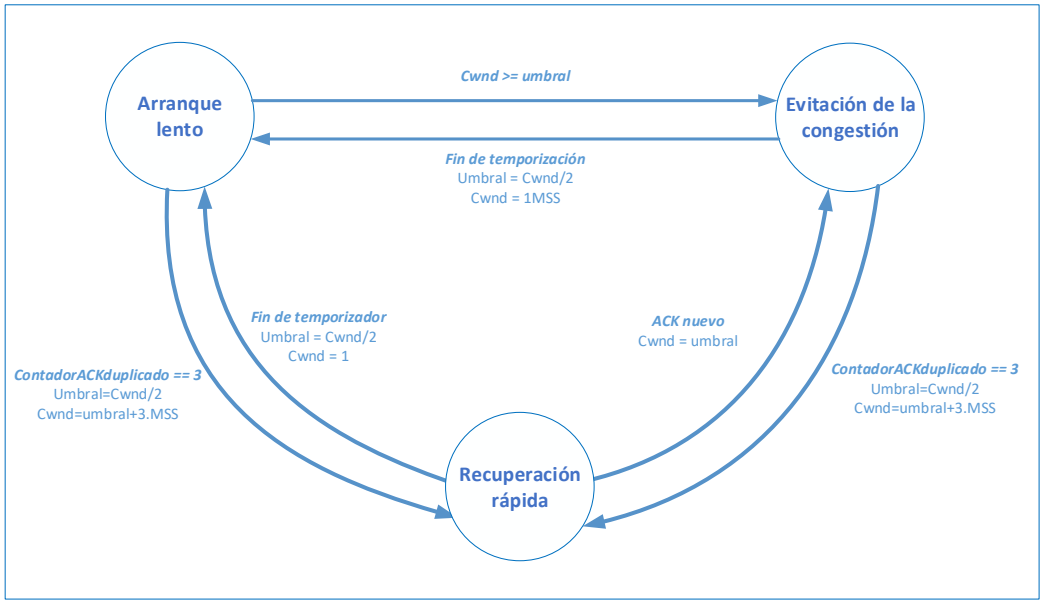
Luego de esta definición, TCP entra en el proceso de recuperación rápida.

c. Recuperación rápida (fast recovery). - Este proceso se ejecuta cuando se detecta la congestión mediante la recepción de tres confirmaciones duplicadas (ACKs duplicados), ya sea cuando se está ejecutando el algoritmo de arranque lento o de evitación de la congestión.

La máquina de estados finitos – FSM por sus siglas en inglés (Jero, 2018) describe los estados, variables e interacción que se da entre ellos para la ejecución de un proceso de control de congestión; en la figura 5 se muestra una adaptación de la FSM desarrollada por Jero, 2018; ahí se evidencia la interacción de cada uno de los algoritmos (arranque lento, evitación de la congestión y recuperación rápida)**.**

Figura 5

Máquina de estados finitos para control de congestión – adaptado Jero 2018



Según Rodriguez (2016), dentro de las variantes de control de congestión más populares, tenemos Reno, New reno, Tahoe y Cubic; a continuación, una introducción al comportamiento y características de cada uno.

TCP Tahoe, ejecuta los algoritmos de arranque lento y evitación de la congestion; luego, se incluye el algoritmo de retransmisión rápida. Una de las desventajas de su funcionamiento es que, ante la pérdida de un segmento, regresa a ejecutar el proceso de

arranque lento, lo que termina afectando el valor de la ventana de congestión por la demora hasta alcanzar los valores óptimos.

TCP Reno; en el caso de Tahoe, al registrar congestión en la red, la venta de congestión toma el valor de un segmento (1 MSS), dando inicio al algoritmo de arranque lento; en cambio, TCP Reno, en una situación similar, ejecuta al algortimo de Recuperación Rápida, reenviando el paquete perdido y reduciendo el *umbral* a la mitad, evitando así, la ejecución del algoritmo de arranque lento. Pasa de Recuperación rápida a evitación de la congestión, cuando recibe un nuevo reconocimiento (*ACK*). Con estas defincciones, Reno se enfoca en mantener a la Cwnd lo más cerca al valor que tenía la Cwnd cuando se registró la congestión.

TCP New Reno, esta variante para el control de la congestión tiene como objetivo reducir la pérdida aislada e individual de segmentos, sin embargo, a un escenario de pérdida de segmentos sucesivos, no presenta alternativas. El contraste con su predecesor (TCP Reno) se da en la fase de recuperación rápida, la llegada de un nuevo reconocimiento no hace que salga de esta etapa, hasta que lleguen las confirmaciones de todos los segmentos pendientes; así, un ACK parcial, indica que el segmento siguiente no llegó a su destino y debe ser retransmitido.

TCP Cubic, surge con la idea de tomar ventaja del hecho de que actualmente los enlaces de comunicaciones disponen de niveles de ancho de banda cada vez mayores, en una red compuesta por enlaces de amplios anchos de banda un algoritmo de control de congestión que lentamente incrementa el porcentaje de transmisión puede terminar por desperdiciar la capacidad los enlaces. La intención es disponer de un algoritmo que trabaje con ventanas de congestión cuyos procesos de incremento sean más agresivos, pero que se restrinjan de sobrecargar la red.

2.6 Control de la congestión en una red inalámbrica

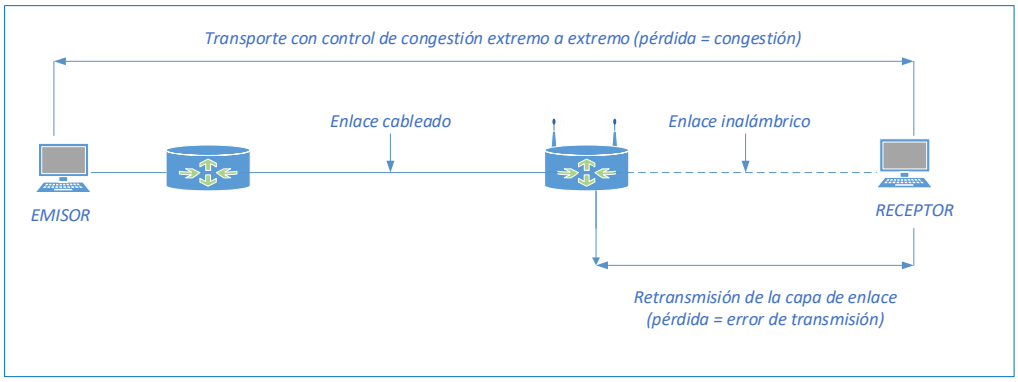
Para Tanenbaum 2012, en la práctica, la tasa de pérdidas para las conexiones TCP es muy pequeña: 1% es una tasa de pérdidas moderada y, para cuando la tasa de pérdidas

llega a 10% o superior, la conexión ha dejado de trabajar por completo; sin embargo, para las redes inalámbricas (802.11, bluetooth, WiMAX , entre otras), son comunes las tasas de pérdidas de tramas de por lo menos un 10%. Este comportamiento significa que, sin medidas de protección, los esquemas de control de congestión que utilizan la pérdida de paquetes como señal, regulan de manera innecesaria las conexiones que pasen a través de enlaces inalámbricos para generar tasas muy bajas de envío de segmentos. Para funcionar bien, las únicas pérdidas de paquetes que debe observar el algoritmo de control de congestión son las pérdidas debido a un ancho de banda insuficiente, no las pérdidas debido a errores de transmisión. Una solución a este problema es enmascarar las pérdidas inalámbricas mediante el uso de retransmisiones a través del enlace inalámbrico. Por ejemplo, 802.11 usa un protocolo de parada y espera para entregar cada trama, y reintenta las transmisiones varias veces si es necesario antes de reportar la pérdida de un paquete a la capa superior. En el caso normal, cada paquete se entrega a pesar de los errores de transmisión transitorios que no son visibles para las capas superiores.

En la figura 6, se visualiza un esquema general de una conexión, que muestra una trayectoria de enlace cableado e inalámbrico; aquí se puede apreciar cómo una pérdida de paquetes en el tramo inalámbrico, puede interpretarse como congestión para TCP.

Figura 6

Control de congestión con medios de conexión mixtos (Guiados e inalámbricos).



Nota. Tanenbaum (2012).

Hay que tener en cuenta dos aspectos, en primer lugar, el emisor no necesariamente sabe que la trayectoria incluye un enlace inalámbrico, ya que todo lo que ve es la tecnología directamente conectada, en este caso el enlace cableado. Las trayectorias son heterogéneas, debido a que no existe un método general para que el emisor sepa qué tipo de enlaces conforman la trayectoria. Esto complica el problema de control de congestión, ya que no hay una manera fácil de usar un protocolo para enlaces inalámbricos y otro para enlaces cableados.

Como segundo punto, se tiene dos mecanismos basados en la pérdida de paquetes: las retransmisiones de la capa de enlace de datos y, el control de congestión en la capa de transporte. La clave es, que ambos mecanismos deben coexistir sin confundirse, identificando que una pérdida de paquete/retransmisión hace referencia a un problema en la transmisión y no de congestión; esta diferenciación la realizan tomando en cuenta que, estos mecanismos trabajan en escalas de tiempo distintas, por ejemplo, el control de la transmisión en la capa de enlace, se da en el rango de microsegundos a milisegundos, por el contrario, los controles en la capa de transporte se activan en milisegundos a segundos; esta diferencia, permite a los enlaces inalámbricos detectar las pérdidas de tramas y retransmitirlas para reparar los errores de transmisión mucho antes de que la entidad de transporte deduzca la pérdida de paquetes y asuma un problema de congestión desencadenando sus mecanismos de control. Después de todo, una pérdida sólo debe provocar que un mecanismo actúe, ya que puede ser un error de transmisión o una señal de congestión. No puede ser ambos. Si los dos mecanismos actúan (al retransmitir la trama y reducir la tasa de envío), entonces regresamos al problema original de los transportes que operan con demasiada lentitud a través de los enlaces inalámbricos.

Capitulo tres

Rendimiento de las variantes para el control de la congestión en una red MANET

Una vez analizado el comportamiento de TCP frente a un problema de congestión, en el presente capítulo se presenta la simulación, registro y análisis de resultados del comportamiento de las variantes de control de congestión en un ambiente MANET, mediante el software de simulación OPNET Modeler 17.5; adicional a la evaluación del control de la congestión, se complementa con el análisis del comportamiento de los protocolos de enrutamiento en redes MANET (proactivos y reactivos) como variable externa a los objetivos de este trabajo. Mediante una combinación de estas variables externas y, las diferentes variantes de control de congestión de TCP, se define los escenarios para el estudio.

3.1 Definición de indicadores de evaluación y escenarios de simulación

Respecto al comportamiento de control de congestión en las variantes de TCP, Tahoe, Reno, New Reno y CUBIC, el presente trabajo busca registrar, comparar y analizar las variables definidas en el punto 1.3 de este documento (Indicadores que definen a una transferencia confiable), a continuación se muestra en la tabla 1 de equivalencia entre el indicador y, la variable usada en el simulador y, en la tabla 2, la descripción de los parámetros de simulación y escenarios utilizados para el análisis.

Tabla 1

Definición de indicadores de evaluación

Indicador	Proceso de evaluación – simulador
Paquetes recibidos (Sharma,, 2016).	**Tráfico recibido** en el nodo receptor durante la simulación.
Latencia (Bhatia,, 2015).	***Delay*** de la conexión en el nodo receptor.
Tamaño de ventana (Kurose, 2017).	Variación del **tamaño de la ventana de congestión** durante la transmisión para el nodo transmisor y el **tamaño de ventana anunciado por el receptor.**

Cantidad de retransmisiones	***Retransmition Count**;* indica el número de retransmisiones TCP por conexión.
Caudal efectivo (Throughput), Govindarajan (2017).	**Throughput total de la conexión Wi-Fi** del dispositivo emisor.

Escenarios de simulación

Tabla 2

Parámetros de simulación y escenarios

Parámetro	**Valor**	
Cantidad de nodos	3	
Aplicación	FTP	
Tiempo de simulación	3 minutos	
Área de simulación	1000 m^2 (100x100)	
Data rate	11 Mbps	
Movimiento del nodo	Desplazamiento 1, 2, 3 4 y 5 = duración 30 seg. (c/u)	
Variación de parámetros	**Protocolo de enrutamiento**	**Algoritmo de control de congestión**
Escenario 1.1	AODV	Tahoe, reno, new reno y CUBIC
Escenario 1.2	OLSR	Tahoe, reno, new reno y CUBIC

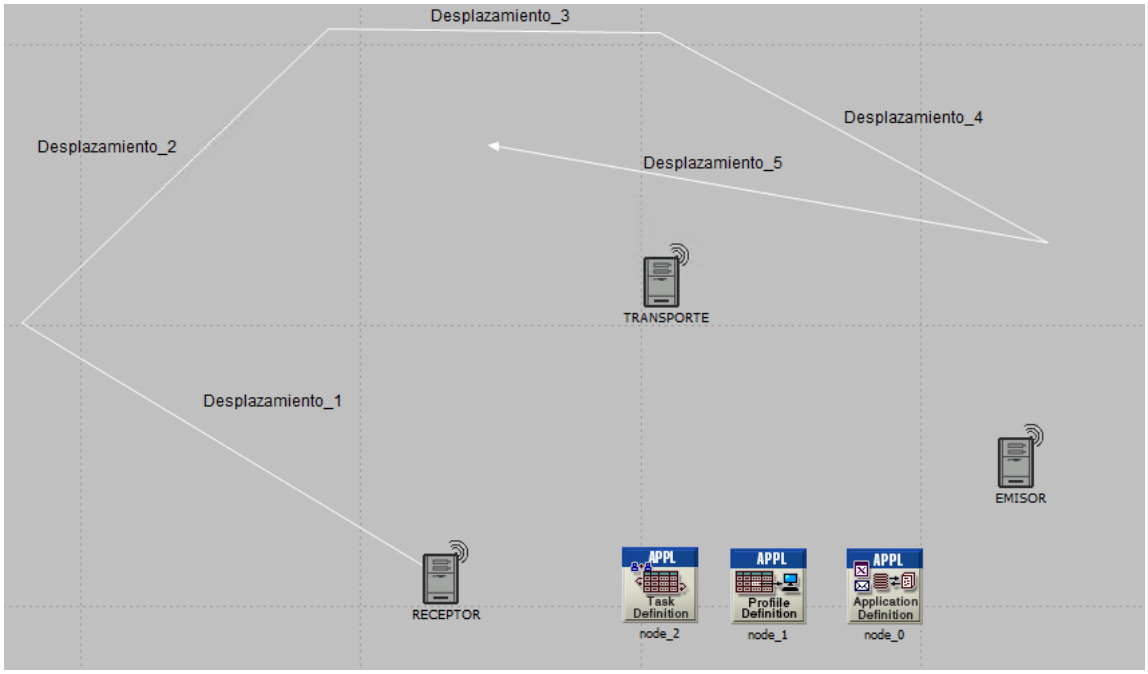
Nota. Los parámetros de simulación fueron seleccionados con base en trabajos de investigación similares como los ejecutados por Rath, 2016; Casoni, 2015, entre otros; además, con el parámetro de movimiento del nodo, lo que buscamos es generar pérdida de paquetes (por la pérdida de conexión) y, validar el comportamiento de TCP y sus características de control de congestión.

3.2 Simulación de escenarios y registro de resultados

Conforme a la definición del escenario base y, tomando en cuenta los escenarios de evaluación detallados, se configura el proyecto de simulación de una red MANET usando el simulador OPNET. A continuación, la figura 7 representa el esquema simulado.

Figura 7

Esquema base de simulación



Como podemos apreciar en la figura 7, existen tres tipos de nodos; el nodo “emisor”, el cual es el encargado de generar el tráfico direccionado al destino, nodo “receptor”. El nodo central del esquema le hemos llamado “nodo transporte”, el cual es el encargado de generar comunicación en los momentos de desplazamiento del “nodo receptor”. Adicional, el movimiento configurado para el “nodo receptor”, está definido en cinco desplazamientos, cada uno, con una duración de 30 segundos.

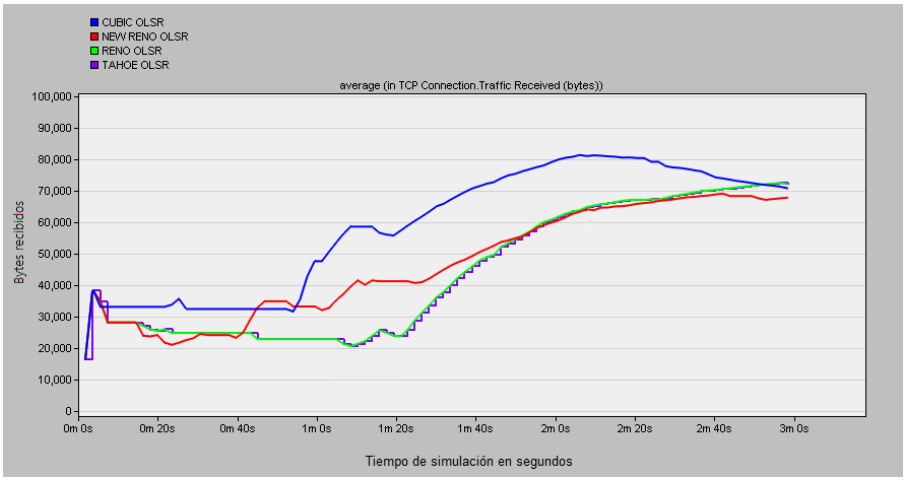
Gráficas de resultados

Escenario 1.1.- Tahoe, Reno, New Reno y CUBIC con **OLSR**

a. **Paquetes recibidos**, la figura 8 muestra la cantidad de bytes recibidos en el nodo receptor durante el tiempo de la simulación; cada línea marcada, representa el valor obtenido por cada proceso de control de congestión.

Figura 8

Tráfico recibido - en bytes, nodo receptor

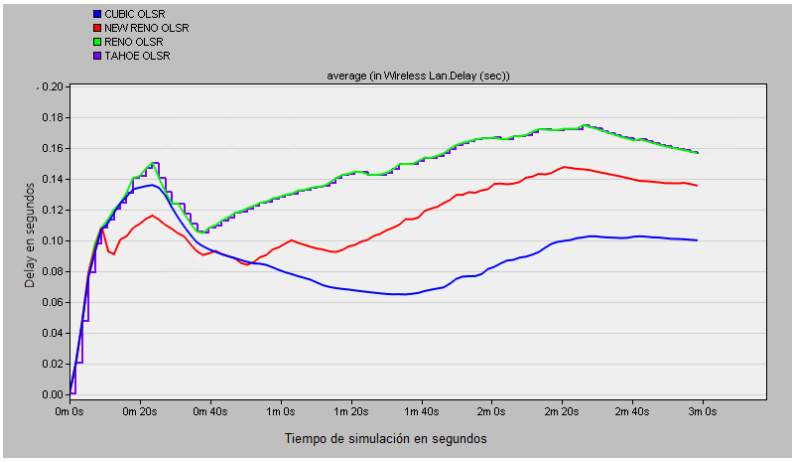


Este indicador refleja la cantidad de información efectiva que llega hasta el "nodo receptor"; es decir, la información validada y, que pasará a la capa de aplicación; mientras mayor sea el valor registrado, evidencia un mejor comportamiento en los procesos de envío/recepción de la información; como podemos apreciar, en la figura 8, el comportamiento es diferente por los procesos de control de congestión que cada variante ejecuta, como se indicó en la sección 2.5 de este documento.

b. Latencia, la figura 9 indica la media de la latencia registrada en el nodo receptor durante la simulación

Figura 9

Latencia Wireless LAN – en segundos (media) – Nodo receptor

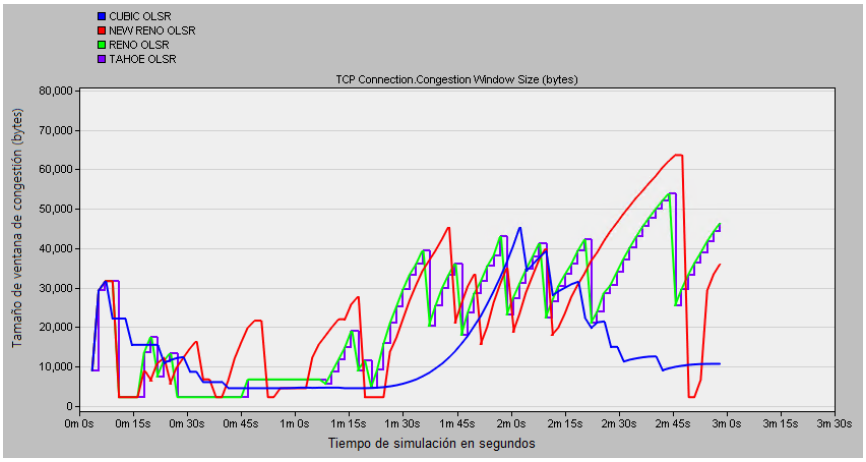


La latencia es el tiempo requerido por un segmento para llegar desde el emisor hasta el receptor, mientras menor sea el valor registrado, significa que le tomó menos tiempo a la información ser entregada de forma correcta en el receptor; la figura 9 indica la media de la latencia registrada por cada variante de control de congestión y, gráficamente se puede apreciar a CUBIC con valores menores, una vez que la topología es covergente (aproximadamente a partir de los 60 segundos, el tráfico fluye a través del nodo de transporte).

c. **Tamaño de ventana de congestión (Cwnd),** la figura 10, muestra el comportamiento de la ventana de congestión (nodo emisor) según avanza el tiempo de la simulación.

Figura 10

Variación de la Cwnd – Nodo emisor en bytes

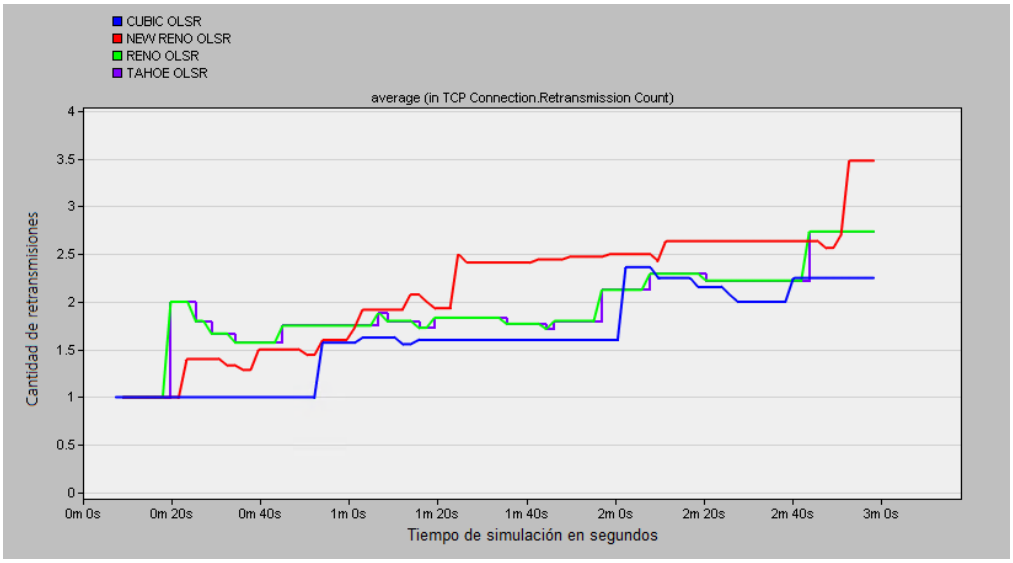


El valor de la ventana de congestión (*Cwnd*) es la capacidad máxima censada por el emisor para el envío de los segmentos, en la figura 10 se puede apreciar a New Reno como la variante que llega a experimentar un valor de la *Cwnd* más alto que el resto en varios momentos de la simulación.

d. **Contador de retransmisiones,** la figura 11 representa el valor medio de la cantidad de retransmisiones registradas por el nodo emisor durante la simulación.

Figura 11

Contador de retransmisiones (media) – Nodo emisor

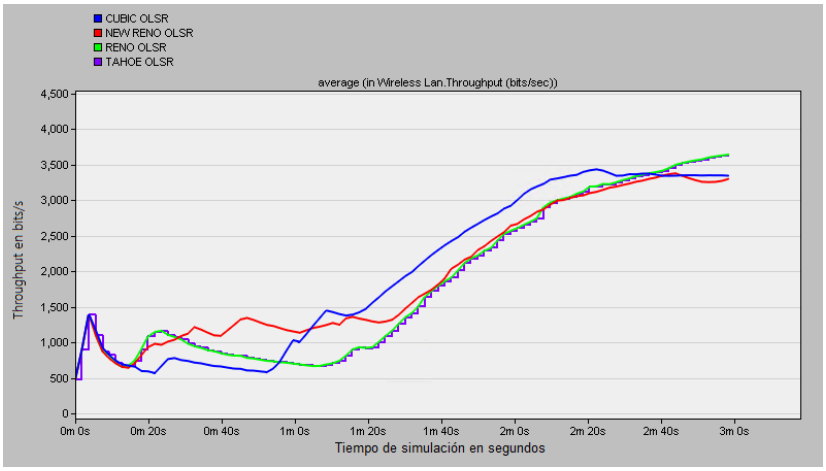


Cuando un segmento es transmitido sin haber recibido un acuse de recibo, éste se registra como una retransmisión e indica, que el segmento ha sido perdido; estos valores pueden estar relacionados a varios orígenes, como, por ejemplo, el comportamiento del protocolo de enrutamiento. La figura 11 registra los valores medios de retransmisiones para cada variante de control de congestión, teniendo a CUBIC como el que menor cantidad de retransmisiones registra.

e. **Caudal efectivo (Throughput),** la figura 12, representa el valor medio del Throughput registrado en la interfaz inalámbrica en el nodo emisor durante la simulación.

Figura 12

Throughput Wi-Fi - en bits/seg (media) – Nodo emisor



El throughput (o rendimiento) es el valor efectivo de transmisión, si este indicador crece, significa que el nodo está enviando mayor cantidad de información (bits/seg). En la figura 12 se aprecia comportamientos similares entre las variantes de control de congestión, con ligeras diferencias entre ellos.

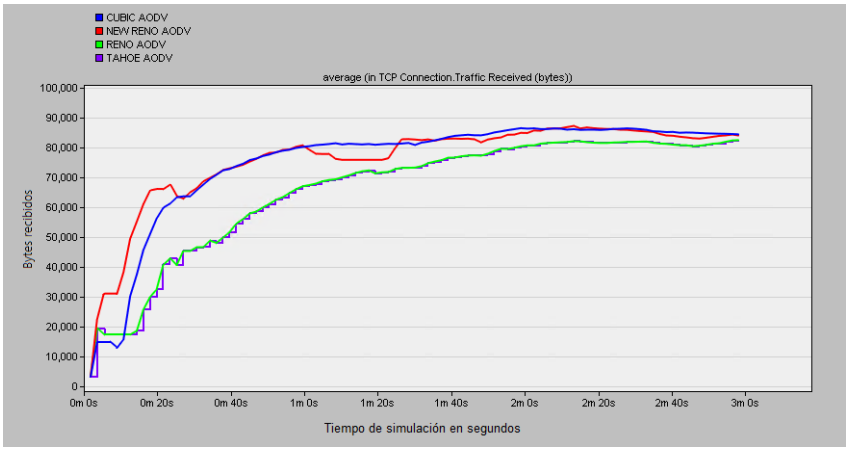
Gráficas de resultados

Escenario 1.2.- Tahoe, Reno, New Reno y CUBIC con **AODV**

a. **Paquetes recibidos**, la figura 13 muestra la cantidad media de bytes recibidos en el nodo receptor durante la simulación; cada línea marcada, representa el valor obtenido con las variantes de control de congestión.

Figura 13

Tráfico recibido - en bytes (media) – Nodo receptor

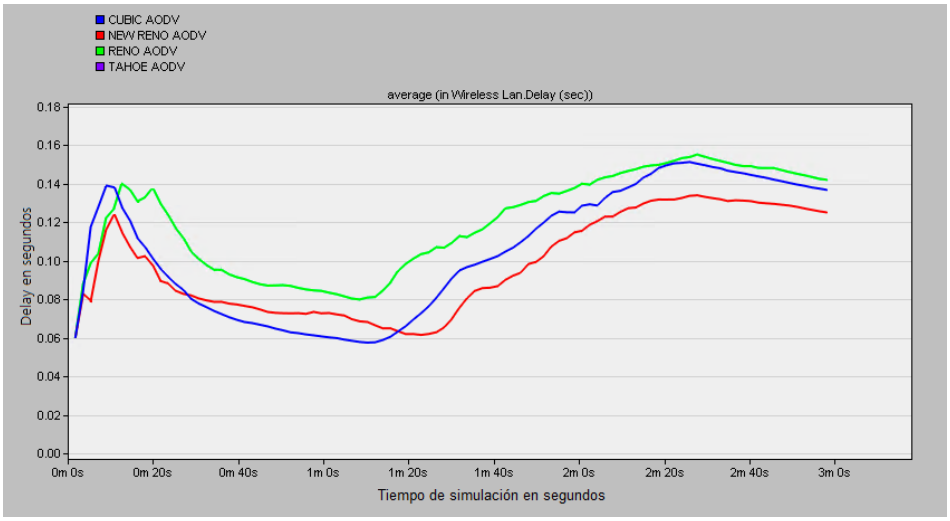


Similar al gráfico 8, este indicador refleja la cantidad de información efectiva que llega hasta el “nodo receptor”; en la figura 13, se puede apreciar una tendencia similar de capacidad de entrega de información conforme avanza el tiempo de simulación.

b. **Latencia**, la figura 14 indica la media de la latencia registrada en el nodo receptor durante la simulación.

Figura 14

Latencia Wireless LAN – en segundos (media) – Nodo receptor

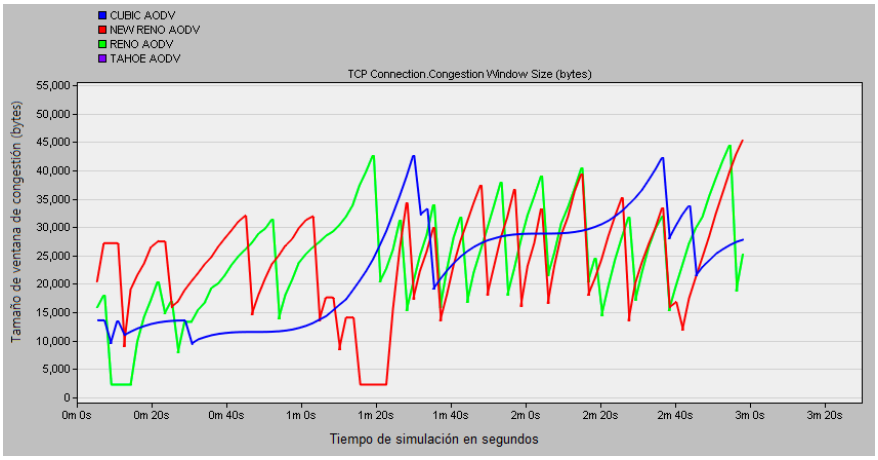


En la figura 14 se puede apreciar a New Reno y CUBIC como las variantes que presentan mejores resultados.

c. **Tamaño de ventana de congestión (Cwnd)**, la figura 15, muestra el comportamiento de la ventana de congestión (nodo emisor) según avanza el tempo de la simulación.

Figura 15

Variación de la Cwnd – Nodo emisor en bytes

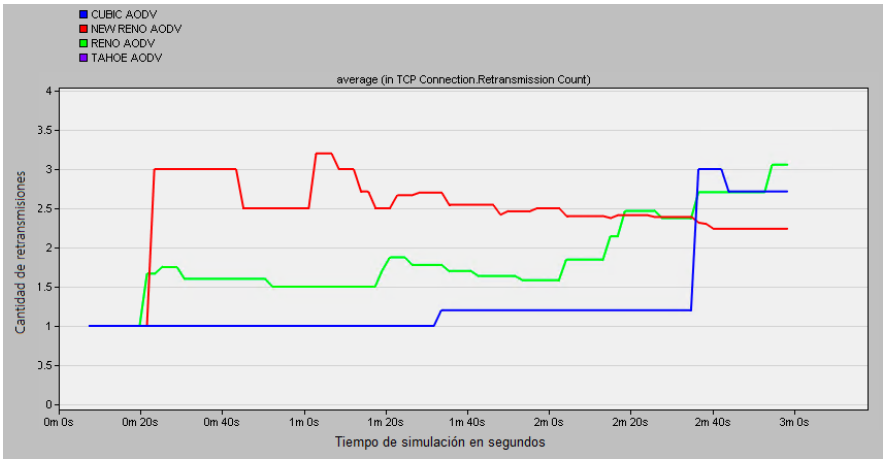


En la figura 15 se puede apreciar comportamientos diferentes de cada variante de control de congestión, esto se debe a que existen diferencias en el manejo de la congestión por cada uno.

d. Contador de retransmisiones, la figura 16 representa el valor medio de la cantidad de retransmisiones registradas por el nodo emisor durante la simulación.

Figura 16

Contador de retransmisiones (media) – Nodo emisor

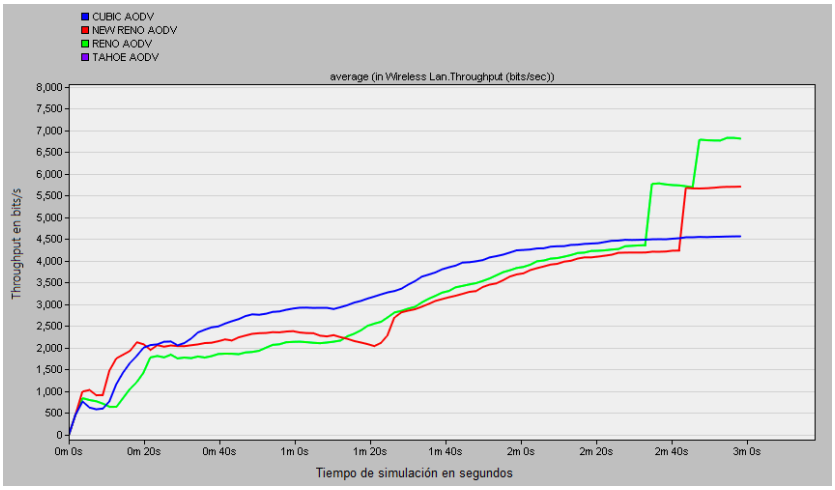


La figura 16 registra los valores medios de retransmisiones para cada variante de control de congestión, teniendo a CUBIC como el que menor cantidad de retransmisiones registra durante aproximadamente el 75% de la simulación.

e. Caudal efectivo (Throughput), la figura 17, representa el valor medio del Throughput registrado en la interfaz inalámbrico el nodo emisor durante la simulación.

Figura 17

Throughput Wi-Fi - en bits/seg (media) – Nodo emisor



En la figura 17 se aprecia comportamientos similares entre las variantes de control de congestión, con ligeras diferencias entre ellos.

A continuación, la tabla 3 describe los valores medios registrados en la herramienta de simulación, para cada uno de los indicadores definidos y, la figura 18, representa en forma gráfica y comparativa los resultados obtenidos.

Tabla 3

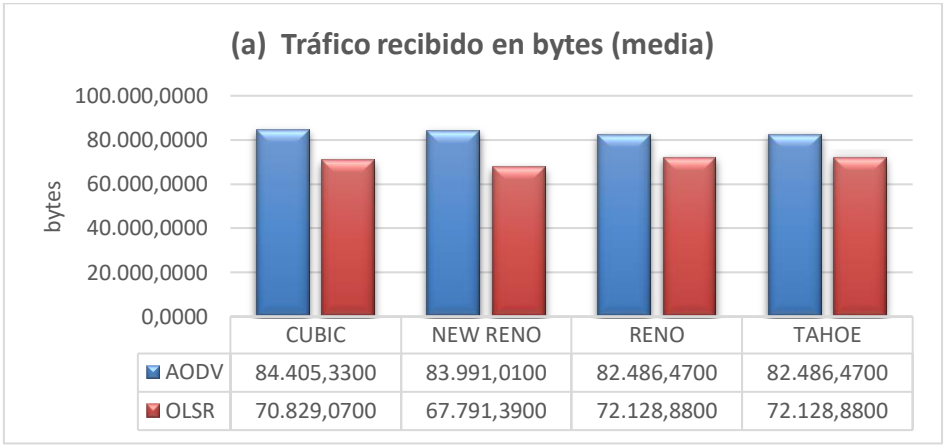
Registro de valores de los indicadores

AODV	Nodo Transmisor			Nodo Receptor	
	Tamaño de ventana, emisor (bytes)	Throughput Wi-Fi (bits/seg)	Contador de retransmisiones	Tráfico recibido (bytes)	Latencia LAN Wi-Fi (seg)
	Media	Media	Media	Media	Media
CUBIC	22.590,0900	4.566,2200	2,7143	84.405,3300	0,1368
NEW RENO	23.775,8500	5.709,9600	2,2381	83.991,0100	0,1251
RENO	24.560,1500	6.814,8400	3,0556	82.486,4700	0,1420
TAHOE	24.560,1500	6.814,8400	3,0556	82.486,4700	0,1420
OLSR	Nodo Transmisor			Nodo Receptor	

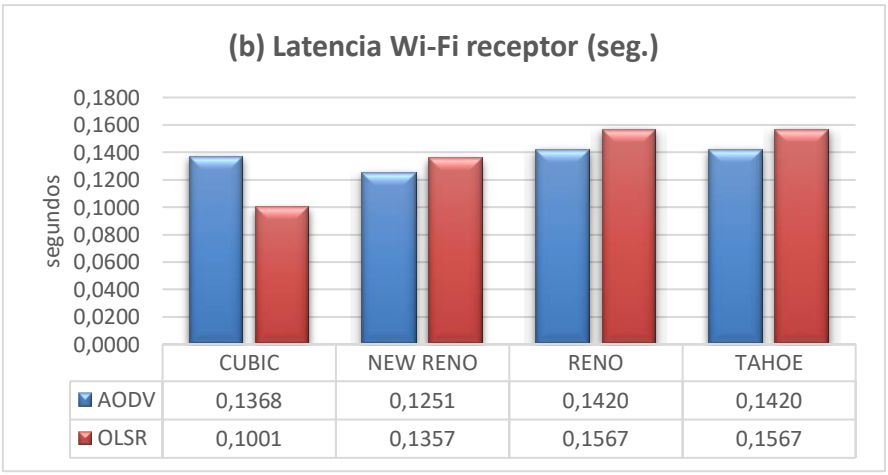
	Tamaño de ventana, emisor (bytes)	Throughput Wi-Fi (bits/seg)	Contador de retransmisiones	Tráfico recibido (bytes)	Latencia LAN Wi-Fi (seg)
	Media	Media	Media	Media	Media
CUBIC	14.121,9600	3.345,3300	2,2500	70.829,0700	0,1001
NEW RENO	23.470,1900	3.298,6700	3,4800	67.791,3900	0,1357
RENO	22.166,3300	3.642,4800	2,7368	72.128,8800	0,1567
TAHOE	22.166,3300	3.642,4800	2,7368	72.128,8800	0,1567

Figura 18

Comparativa gráfica de resultados de la simulación

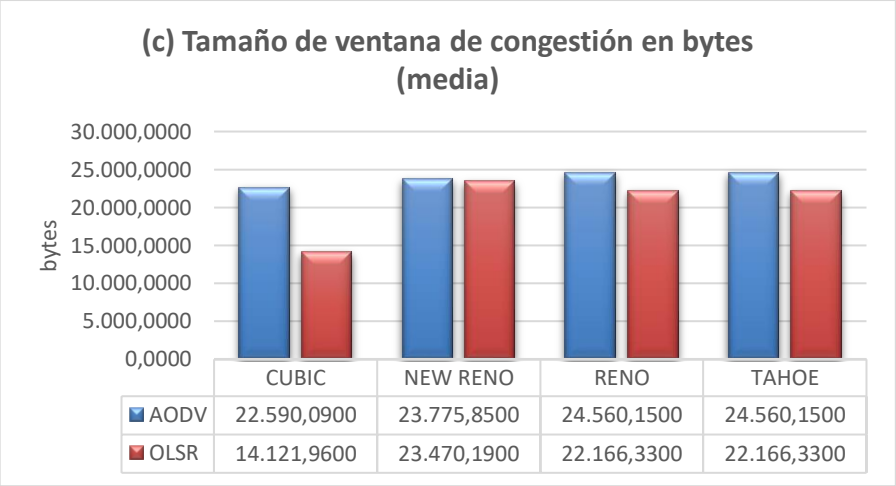


En la figura 18 (a), podemos evidenciar un mejor comportamiento de las variantes de control de congestión en combinación con AODV, respecto al tráfico recibido y, puntualmente con Reno o Tahoe, obtenemos los mejores resultados.

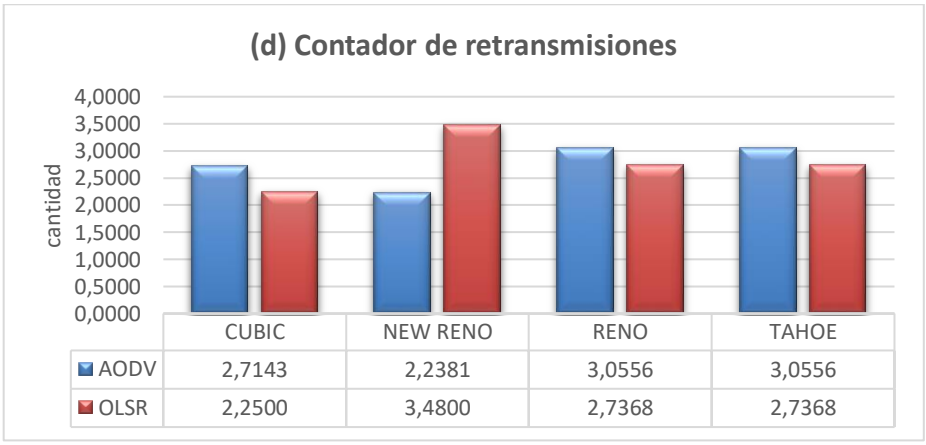


Respecto a la latencia, la figura 18 (b) muestra a CUBIC en combinación con OLSR con los mejores resultados; este es el único punto en el que OLSR supera a AODV ya que,

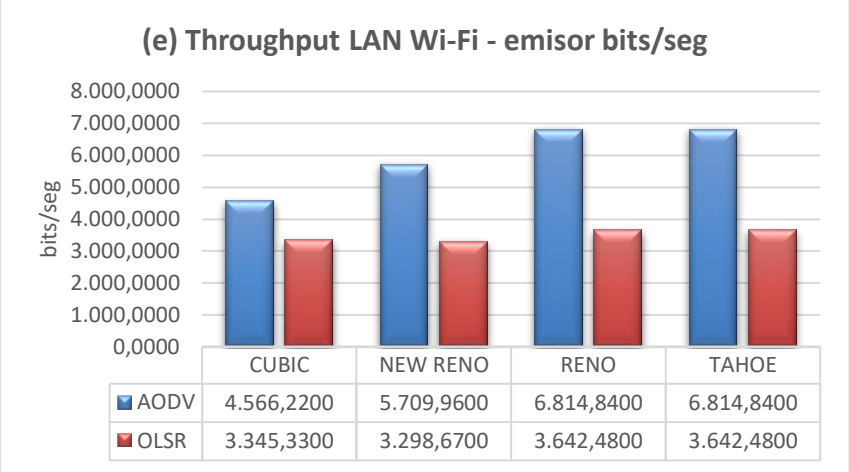
como se aprecia, en el resto de variantes, la tendencia es AODV para obtener valores más bajos de latencia.



Analizando el Tamaño de ventana de congestión (*Cwnd*), la figura 18 (c) registra AODV con valores más altos respecto a OLSR, independiente de la variante de control de congestión, pero, con Reno y Tahoe, registran la *Cwnd* más alta.



En la figura 18 (d) el contador de retransmisiones marca una diferencia entre Reno y New Reno, siendo el segundo el que cuenta con el valor más bajo y, por lo tanto, presenta ventajas en la comunicación; tomando en cuenta que AODV la combinación para cumplir lo indicado.



La figura 18 (e) muestra una marcada diferencia positiva en los resultados al usar AODV como protocolo de enrutamiento, respecto a las variantes de control de congestión, Reno y Tahoe tienen los mejores resultados respecto al throughput en el emisor.

3.3 Análisis de resultados

En el presente apartado, se definirá, según la evaluación de cada indicador y, tomando en cuenta los resultados comparativos de cada variante de control de congestión, cual de ellas, brinda el mejor comportamiento global según nuestra simulación.

Para la valoración objetiva del comportamiento de cada indicador según la variante de control de congestión, se utilizó la regla de tres con proporciones (Obando 2018) :

1. Se identifica la variante de control de congestión que presenta el mejor comportamiento y, se le asigna una proporción de 100%.
2. Se identifica en orden descendente o ascendente, según el indicadorla siguiente variante y, con el valor obtenido, calculamos la proporcionalidad que representa este valor, respecto al 100% de la variante con mejor comportamiento.

Bajo lo expuesto, realizamos los cálculos correspondientes y obtenemos la tabla 4 y 5, que representan los resultados proporcionales de cada indicador y totales de cada escenario, tanto para los escenarios con AODV y OLSR, respectivamente.

Tabla 4

Comparativo porcentual, rendimiento de los algoritmos de control de congestión, escenario AODV.

Control de Congestión	*Cwnd*	*Throughput*	*Cant. Retransmisiones*	*Tráfico recibido*	*Delay Wi-Fi*	*TOTAL*
NEW RENO	96,81%	83,79%	100,00%	99,51%	100,00%	**96,02%**
RENO	100,00%	100,00%	63,47%	97,73%	86,46%	**89,53%**
TAHOE	100,00%	100,00%	63,47%	97,73%	86,46%	**89,53%**
CUBIC	91,98%	67,00%	78,72%	100,00%	90,66%	**85,67%**

Tabla 5

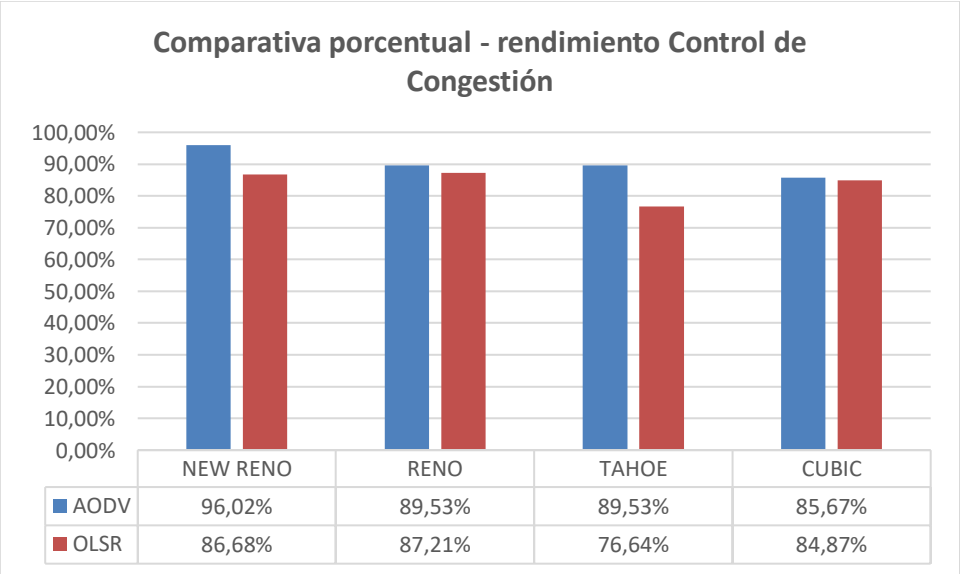
Comparativo porcentual, rendimiento de los algoritmos de control de congestión, escenario OLSR.

Control de Congestión	*Cwnd*	*Throughput*	*Cant. Retransmisiones*	*Tráfico recibido*	*Delay Wi-Fi*	*TOTAL*
NEW RENO	100,00%	90,56%	78,36%	100,00%	64,46%	**86,68%**
RENO	94,44%	100,00%	100,00%	98,20%	43,43%	**87,21%**
TAHOE	94,44%	100,00%	45,33%	100,00%	43,43%	**76,64%**
CUBIC	60,17%	91,84%	78,36%	93,99%	100%	**84,87%**

La figura 19 indica en un gráfico comparativo, el valor total obtenido por cada proceso de control de congestión, en los escenarios planteados.

Figura 19

Comparativa porcentual total del rendimiento de las variantes de control de congestión



Como se aprecia en la figura 19, para los escenarios propuestos (AODV y OLSR), la variante de control de congestión New Reno, presenta un resultado general mayor al de las otras variantes (96,02% con AODV); con OLSR, Reno se ubica en primer lugar con 87,21%. Si bien, no presentan un comportamiento óptimo para cada uno de los indicadores evaluados (100%), en el total presentan el mejor resultado respecto a las otras variantes.

3.4 Conclusión, rendimiento de los algoritmos para el control de la congestión en una red Manet

Con base al análisis realizado en el apartado 3.3 del presente capítulo y, tomando en cuenta los resultados obtenidos en la tabla 5, concluimos:

a. Los escenarios propuestos permiten identificar diferencias en el comportamiento de los indicadores evaluados, por lo tanto, se pudo realizar una comparativa porcentual de los resultados y determinar la variante de control de congestión que presenta mejores resultados.

b. En la valoración final, la variante New Reno en complemento con AODV (como protocolo de enrutamiento) obtiene la valoración más alta de 96,02%; por lo tanto, se define a New Reno - AODV como el que mejores prestaciones presenta las simulaciones realizadas.

c. El algoritmo utilizado por CUBIC se ubica en la última posición de la valoración total, lo que indica que, en un entorno MANET no presentará las mejores características y resultados.

Capitulo cuatro

Propuesta de optimización de TCP para MANET

Con base en los resultados comparativos entre las diferentes variantes de control de congestión de TCP en un escenario MANET, el proceso propuesto por New Reno en combinación con AODV (como protocolo de enrutamiento Adhoc) presenta el resultado óptimo en comparación con las otras variantes analizadas; por lo tanto, se evalúa el detalle y características de TCP – New Reno para luego proponer, implementar y setear configuraciones adaptables a redes MANET.

4.1 Comportamiento del algoritmo de control de congestión original

Según Cadin (2021), existe una variación en TCP New Reno (respecto a Reno) que está especificada en RFC3782; puntualmente en el algoritmo de Recuperación Rápida optimizando los resultados en una situación de pérdida múltiples. Algo en común de estos algoritmos en que, en la situación de múltiples segmentos duplicados, ambos pasan a retransmisión rápida, pero, New Reno, no sale de esta fase mientras no se reciba la confirmación completa de los segmentos transmitidos en la ventana de congestión; para llegar a este nivel de control, TCP New Reno incrementa una variable en la que registra el número de secuencia más alto antes de entrar a la fase de Recuperación Rápida.

Cuando se advierte la llegada de una nueva confirmación de segmentos, implica que todos los segmentos transmitidos, previo al error detectado, se entregaron sin error y, cualquier nueva pérdida detectada, se hará referencia a un nuevo proceso de congestión en el canal, por lo tanto, New Reno sale de Recuperación Rápida y define el valor umbral para los nuevos cálculos de la ventana de congestión, mientras tanto continúa en fase de evitación de la congestión similar a Tahoe.

En el caso de que una confirmación hace referencia a un *ACK* parcial, New Reno, enviará la confirmación de sólo el primer error e indica que existe más pérdidas en la ventana

original de segmentos y, únicamente sale de la fase de recuperación rápida cuando todos los segmentos de la ventana han sido reconocidos.

Un punto importante que destacar en TCP New Reno, es que, si bien resuelve el problema de bajo desempeño por eventos de múltiples pérdidas en la ventana de congestión, como resultado de esto, requiere tiempo excesivo en la fase de recuperación rápida, pues necesita todas las confirmaciones de los segmentos perdidos en la *Cwnd*.

Según Intelligent Automation And Soft Computing (2020), la ventana de congestión (*Cwnd*) se mide en bytes y, es asignada por el emisor según su percepción de uso y estado del canal de comunicación (basado en paquetes perdidos, *ACKs* duplicados y RTO (tiempo de espera de retransmisión); en el apartado 2.1 se indicó que, en resumen, el emisor inicia la transmisión con un valor de *Cwnd* = 1 MSS, éste valor se va incrementando en 1 MSS por cada reconocimiento positivo recibido (*ACK*); Según el RFC2581 – Control de congestión de TCP, la variable Delayed *ACK*, permite a los receptores TCP pueden retrasar brevemente sus respuestas *ACK* a nuevos datos. El retardo máximo de *ACK* es de 500 ms, aunque 200 ms es más común. Para el tráfico masivo, los *ACK* retrasados simplemente significan que se reduce el volumen de tráfico de *ACK.* Debido a que los *ACK* son acumulativos, un *ACK* del receptor puede, en principio, reconocer múltiples paquetes de datos del remitente. Desafortunadamente, reconocer demasiados paquetes de datos con un *ACK* puede interferir con el aspecto de reloj automático de las ventanas deslizantes; la llegada de ese *ACK* activará una ráfaga de paquetes de datos adicionales, que de otro modo se habrían transmitido a intervalos regulares.

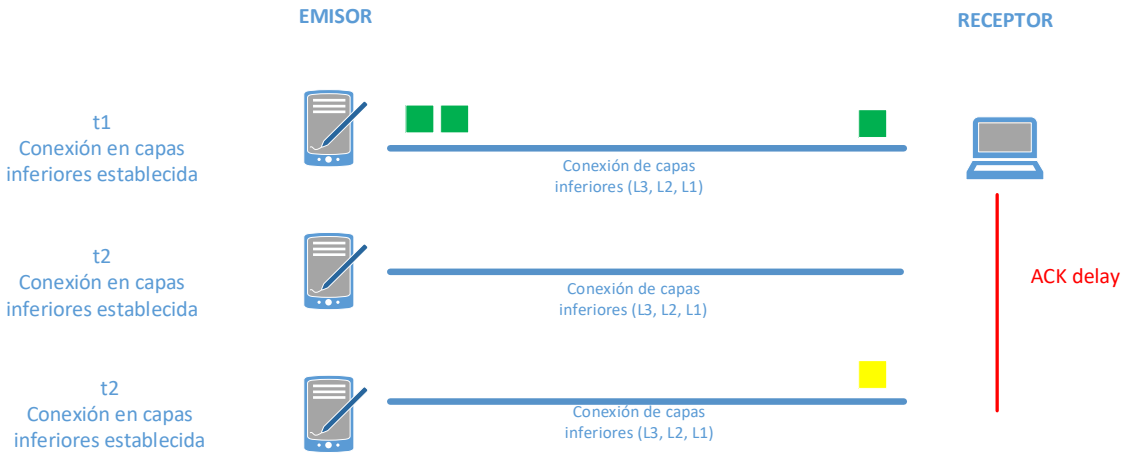
4.2 Optimización de New Reno

Bajo las premisas descritas en el RFC2581, la variable ACK Delayed especifica el tiempo máximo que TCP espera luego de recibir un segmento, para enviar el *ACK* correspondiente.

En la figura 20 se puede apreciar que, existe los tiempos *t1, t2* y *t3*; el segmento llega en el *t1* al receptor y, pasamos al *t3* cuando el receptor envía el *ACK* correspondiente; el tiempo transcurrido entre la llegada del segmento y el envío del *ACK*, es el *ACK* Delayed.

Figura 20

ACK Delay



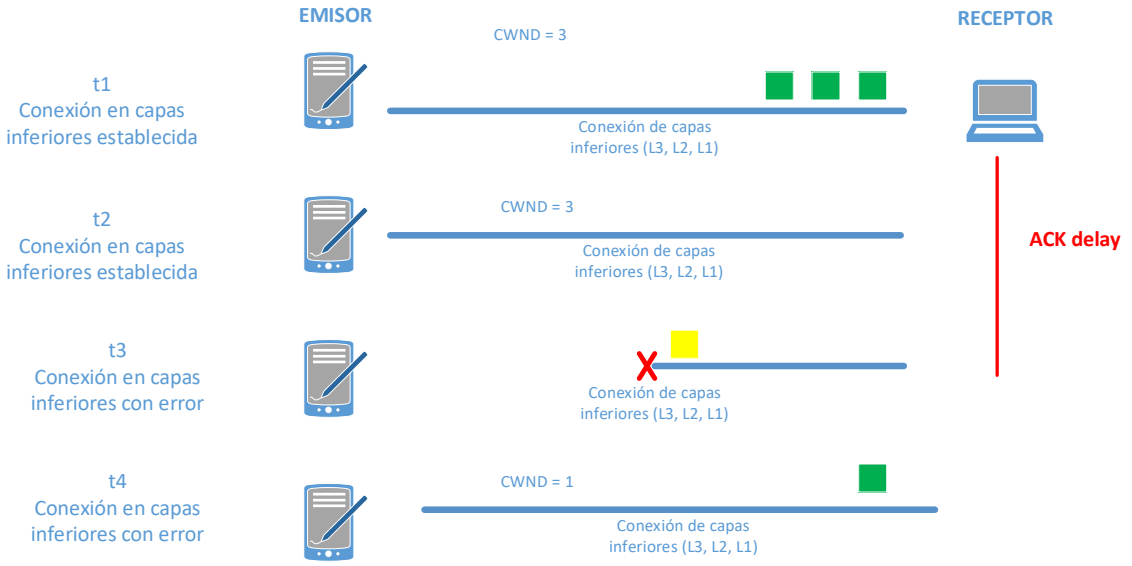
La ventana de congestión calculada por el emisor, está directamente relacionada con la llegada de los *ACKs;* si éstos no llegan (finaliza el timer *RTT*), el emisor interpreta una congestión y, en el caso de New Reno, reduce el tamaño de la ventana de congestión. Tomando en cuenta que, en una red MANET, la tasa de errores en el medio de transmisión bordea el 10% (Tanenbaum, 2012), el comportamiento del *ACK* Delayed afecta al cálculo del tamaño de ventana, subutilizando el canal.

Como se puede apreciar en la figura 21; en el *t1*, el emisor envía el segmento con éxito hasta el receptor, en el *t2*, la conexión de capas inferiores está disponible, pero, el *ACK* Delayed le permite al receptor esperar un tiempo adicional (máx. 500 *ms*) hasta enviar el *ACK* correspondiente; para la llegada del *t3*, cuando el receptor envía el *ACK*, la conexión de capas inferiores ha sufrido una afectación, por lo tanto, el *ACK* no llegará al emisor, lo que será interpretado como congestión en la red, pasando a realizar los procesos de disminución de la ventana de congestión. En el *t4*; la conexión de capas inferiores está nuevamente disponible y, el emisor (por la pérdida anterior *t3*) está trabajando con una Cwnd de 1, cuando

el canal esta disponible, puede soportar una *Cwnd* de mayor tamaño, como en el *t1*; esto genera una subutilización de las capacidades del canal

Figura 21

Comportamiento ACK Delayed, original



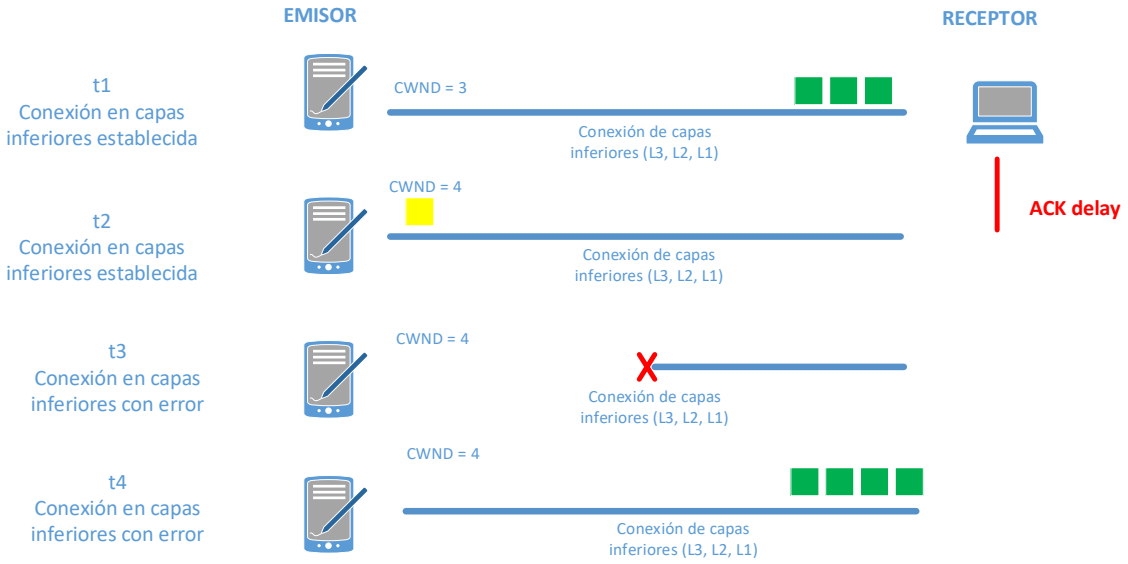
Bajo este contexto se plantea que, la variable *ACK Delayed* sea reducida al orden de 100 ms (normalmente implementada en 200 ms RFC2581); para llegar al valor de 100 milisegundos, se hizo un estudio con diferentes valores de ACK Delayed, el mismo que se puede evidenciar en el Apéndice 1, Comparativo variación de *ACK Delayed*, donde se pudo evidenciar que, al disminuir o incrementar el ACK Delayed sobre los 100 ms, los indicadores se ven afectados en su desempeño; con esta variación se busca:

- Envío de *ACKs* en el momento que el receptor atiende el segmento; usando así, “el mismo estado del canal” que usó el segmento para entregarse.
- Con la llegada temprana de los ACKs, el emisor tendrá reconocimientos que permitirán incrementar el tamaño de la ventana de congestión; esto ayudará a que la *Cwnd* crezca de forma rápida una vez que el canal esté disponible (en caso de pérdida del medio de transmisión).
- Al incrementar la *Cwnd*, crece la probabilidad de envío y recepción de los datos.

Por lo expuesto, la figura 22 indica una representación gráfica del comportamiento esperado al modifica la variable ACK Delayed.

Figura 22

Comportamiento de disminución del ACK Delayed



Bajo el esquema propuesto, el ACK del *t2*, llega hasta el emisor, incrementando en 1 MSS a la *Cwnd*; en el *t3* se produce un problema de conexión de capas inferiores (L3, L2 o L1), evitando que los segmentos sean transmitidos, para el *t4*, el canal ha sido restablecido y, el emisor envía los segmentos usando un tamaño de ventana de 4, mejorando la transmisión.

4.3 Análisis de rendimiento New Reno y New Reno Optimizado

Con el análisis expuesto en el punto anterior, en el presente capítulo realizaremos la simulación del comportamiento de la variante de control de congestión de TCP – New Reno. Así mismo, recopilaremos los resultados según los indicadores y escenarios definidos en el capítulo 3, tomando en cuenta que, únicamente se realiza el registro de resultados para la variante de TCP New Reno con AODV, por presentar mejores resultados según el análisis del aparado 3.3.

Tabla 6

Definición de escenarios de evaluación detallados.

Parámetro	Valor	
Cantidad de nodos	3	
Aplicación	FTP	
Tiempo de simulación	3 minutos	
Área de simulación	1000 m2 (100x100)	
Data rate	11 Mbps	
Movimiento del nodo	Desplazamiento 1, 2, 3 4 y 5 = duración 30 segundos (c/u)	
Max ACK Delayed	0,100 segundos	
Variación de parámetros	**Protocolo de enrutamiento**	**Algoritmo de control de congestión**
Escenario 1.1	AODV	New Reno original y New Reno modificado

4.4 Simulación comparativa y resultados

Para efectos comparativos, la tabla 6 y figuras desde la 23 a la 32 indican el comportamiento de las variantes TCP New Reno vs. TCP New Reno MODIFICADO para cada uno de los indicadores definidos anteriormente.

Tabla 7

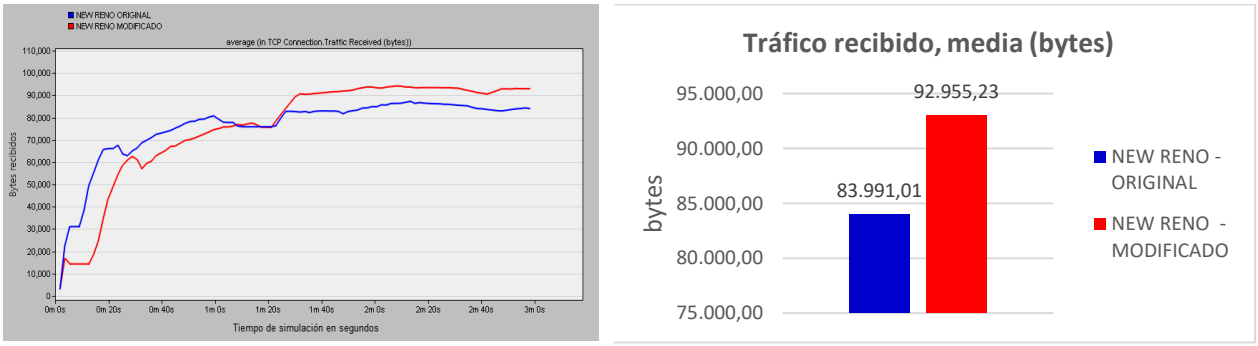
Resultados comparativos New Reno Original vs. New Reno Modificado

AODV	***Nodo Transmisor***			***Nodo receptor***	
	Tamaño de ventana, emisor (bytes)	**Throughput Wi-Fi (bits/seg)**	**Cantidad de retransmisiones**	***Tráfico recibido (bytes)***	**Latencia LAN Wi-Fi (seg)**
	Media	**Media**	**Media**	**Media**	**Media**
NEW RENO - ORIGINAL	23.775,85	5.709,96	2,2381	83.991,01	0,1251
NEW RENO - MODIFICADO	26.075,06	6.135,56	2,1765	92.955,23	0,1156

a. Tráfico recibido, en el nodo receptor durante la simulación, a continuación, la figura 23 indica el comparativo de la simulación y media de valores registrados.

Figura 23

Tráfico recibido-New Reno Original vs. New Reno Modificado–AODV

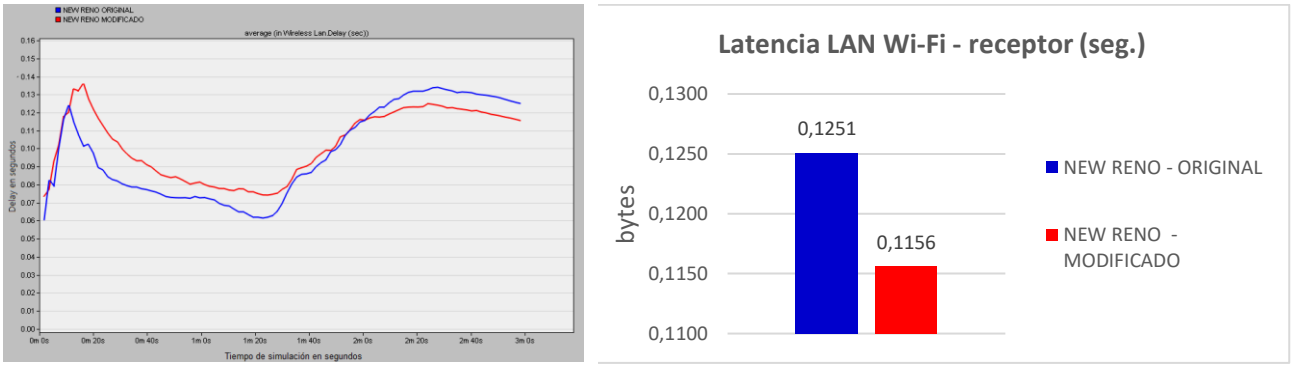


La figura 23 muestra la media de tráfico recibido en bytes durante la simulación; como se puede apreciar, existe un incremento de 23.775,85 a 26.075,06 bytes, esto representa el 10,67% de incremento en el tráfico recibido usando la variante New Reno Modificado, generando una diferencia positiva a la propuesta planteada.

b. **Latencia**, en el nodo receptor durante la simulación, a continuación, la figura 24, comparativo de la simulación y media de valores.

Figura 24

Latencia - New Reno Original vs. New Reno Modificado – AODV



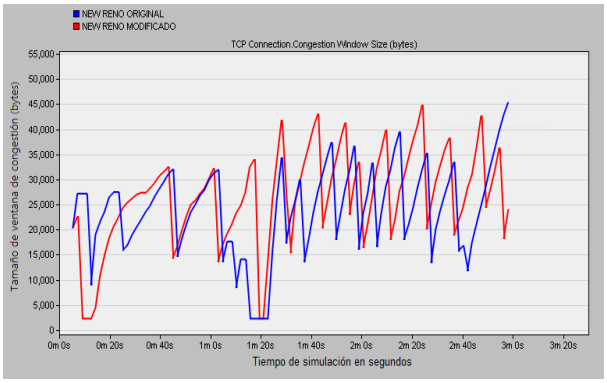
El indicador de latencia presenta una reducción total de 7,61% usando la variante New Reno Modificado; esto significa que, el tiempo de llegada de los

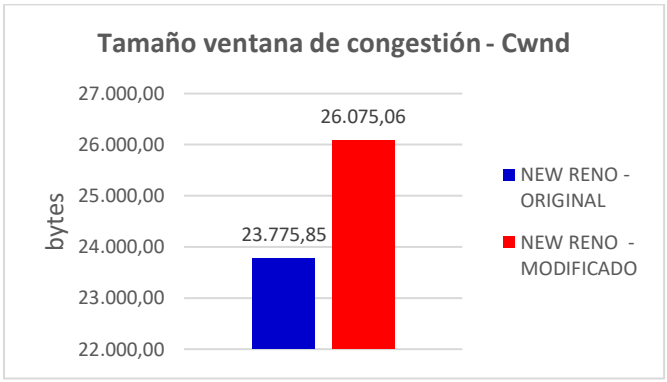
segmentos se reduce en el porcentaje indicado con la modificación realizada a New Reno.

c. Tamaño de ventana de congestión (Cwnd), en el nodo emisor durante la simulación, a continuación, la figura 25, comparativo de la simulación y media de valores.

Figura 25

Cwnd - New Reno Original vs. New Reno Modificado – AODV



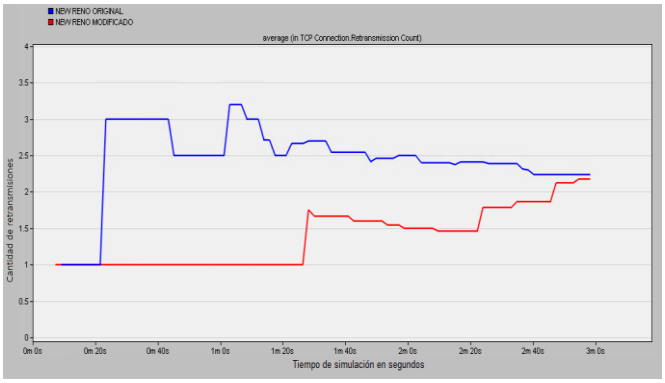


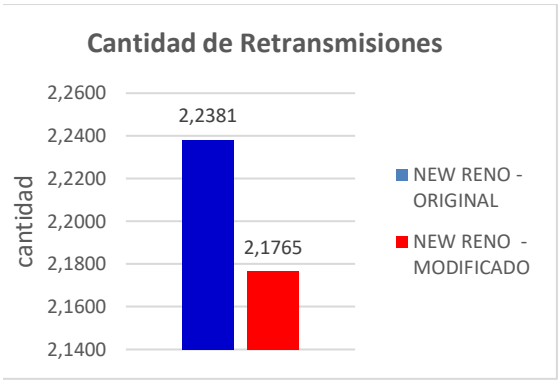
El indicador Ventana de Congestión (*Cwnd*) presenta diferencia de 8,82% a favor de la propuesta ajustada de New Reno en ambos escenarios; como se puede apreciar, el tamaño de la *Cwnd* tiende a incrementar como se proyectaba en el análisis teórico como respuesta a la disminución del *ACK Delayed*.

d. Contador de retransmisiones, en el nodo emisor durante la simulación; a continuación, la figura 26, comparativo de la simulación y media de valores.

Figura 26

Contador transmisiones- New Reno Original vs. New Reno Modificado – AODV



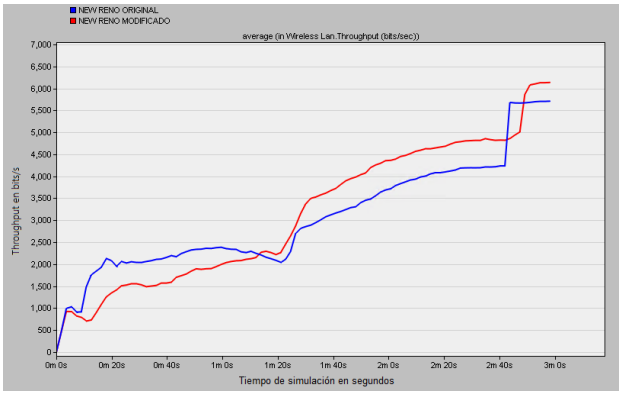


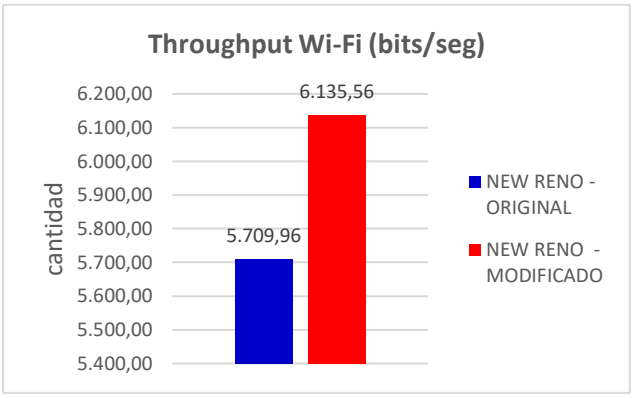
Los resultados presentan al New Reno Modificado con un registro menor en la cantidad de retransmisiones, 2,75% menor que New Reno Original.

e. Caudal efectivo (Throughput), en la interfaz Wireless el nodo emisor durante la simulación; la figura 27 indica el comparativo de la simulación y media de valores.

Figura 27

Throughput - New Reno Original vs. New Reno Modificado – AODV





La simulación y datos recolectados, indican un incremento en el throughput de 7,45% usando New Reno Modificado frente al New Reno original, lo que significa un mejor rendimiento del proceso de conectividad entre los nodos.

Conclusiones

Las características de transporte confiable en una red MANET responden a varios indicadores definidos por algunos autores, entre los principales, el tamaño de la ventana de congestión (Cwnd), la cantidad de retransmisiones, el tráfico recibido por el receptor, el retardo que percibe el receptor y throughput de la conexión; por lo tanto, estos han sido la base de la evaluación y análisis en cada uno de los escenarios planteados.

El método utilizado para evaluar el comportamiento de una tranferencia confiable (usa TCP como protocolo de capa de transporte), permitió indentificar los valores obtenidos para cada una de los indicadores de una transferencia confiable.

De las mediciones realizadas en los escenarios planteados, se logró identificar a New Reno (variante de control de congestión) en combinación con AODV (protocolo de enrutamiento), como la mejor opción en los resultados globales de medición de los indicadores planteados.

La propuesta de modificación a la variante New Reno, presenta porcentajes de mejora para cada indicador utilizado en la medición; es así que, para el tamaño de la ventana de congestión tiene un incremento de 8,82%; en throughput 7,45% y en tráfico recibido 10,67%. Para los indicadores de cantidad de retransmisiones y latencia, se tiene disminución de 2,75% y 7,61% respectivamente; por lo tanto, se concluye que se puede generar ajustes positivos a las variantes actuales de control de congestión y enforcarlos en un mejor rendimiento para redes MANET.

Las variantes de control de congestión se ejecutan en el nodo emisor, teniendo como alternativa variar sus características, pero, siempre se va a depender de la calidad y estado de la conexión de capas inferiores; por lo tanto, podemos incrementar valores de ventana de congestión, por ejemplo, pero, la entrega de la información, siempre va a depender del estado de las conexiones físicas y lógicas bajo la capa de transporte.

Recomendaciones

Continuar con la investigación, explorar la ejecución de los ajustes planteados en otros escenarios y combinar con nuevos protocolos de enrutamiento, por ejemplo, protocolos híbridos.

Usar nuevas herramientas de simulación generando escenarios similares o nuevos a los propuestos en el presente trabajo e, incluir nuevas variables o indicadores de medición, por ejemplo, temas físicos, como rendimiento energético y uso de recursos de los nodos.

Aplicar los ajustes propuestos a otros entornos inalámbricos y validar su comportamiento, por ejemplo, enlaces satelitales, redes vehiculares Ad Hoc (VANET), entre otros.

Implementar en un entorno físico real un escenario de evaluación que permita validar el comportamiento de los indicadores definidos (o nuevos).

Referencias

Agung Rahmat Adi Yuwono, Istikmal and L. V. Yovita, (2013). Performance analysis of Cubic and Yeah TCP implementation using BATMAND over MANET. International Conference on Robotics, Biomimetics, Intelligent Computational Systems, 40-45, https://doi: 10.1109/ROBIONETICS.2013.6743575.

Barreto, M., Belino, M., & San Martín, M. TCP SACK.

Bhatia, B. (2015, February). Performance analysis of AODV based congestion control protocols in MANET. In 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE) (pp. 453-458). IEEE.

Casoni, M., Grazia, C. A., Klapez, M., & Patriciello, N. (2015, May). Implementation and validation of TCP options and congestion control algorithms for ns-3. In Proceedings of the 2015 Workshop on Ns-3 (pp. 112-119).

Chen, X., Zhai, H., Wang, J., & Fang, Y. (2004). TCP performance over mobile ad hoc networks. Canadian Journal of Electrical and Computer Engineering, 29(1/2), 129-134.

Gañán de Andres, R. (2021). Caracterización experimental del comportamiento de TCP sobre enlaces 802.11 n de 40 MHz.

Ghanem, T. F., Elkilani, W. S., & Hadhood, M. M. (2020). Improving TCP Performance over Mobile Ad Hoc Networks with an Adaptive Backoff and Response approach (Dept. E). MEJ. Mansoura Engineering Journal, 33(1), 19-27.

Gopinath, S., & Nagarajan, N. (2015). Energy based reliable multicast routing protocol for packet forwarding in MANET. Journal of applied research and technology, 13(3), 374-381.

Govindarajan J., Vibhurani N., Kousalya G. (2018) Enhanced TCP NCE: A Modified Non-Congestion Events Detection, Differentiation and Reaction to Improve the End-to-End Performance Over MANET. In: Sa P., Sahoo M., Murugappan M., Wu Y., Majhi B. (eds) Progress in Intelligent Computing Techniques: Theory, Practice, and

Applications. Advances in Intelligent Systems and Computing, 719. Springer, Singapore. https://doi.org/10.1007/978-981-10-3376-6_48

Haldorai, A. Modified congestion control algorithm in TCP westwood for mobile ad-hoc network environment.

Hamrioui, S., Lloret, J., Lorenz, P., & Lalam, M. (2013). TCP performance in mobile ad hoc networks. Network Protocols and Algorithms, 5(4), 117-142.

Hanin, M. H., Amnai, M., & Fakhri, Y. (2021). New adaptation method based on cross layer and TCP over protocols to improve QoS in mobile ad hoc network. International Journal of Electrical & Computer Engineering (2088-8708), 11(3).

Holland, G., & Vaidya, N. (2002). Analysis of TCP performance over mobile ad hoc networks. Wireless Networks, 8(2), 275-288.

Jero, S., Hoque, M. E., Choffnes, D. R., Mislove, A., & Nita-Rotaru, C. (2018, July). Automated Attack Discovery in TCP Congestion Control Using a Model-guided Approach. In NDSS.

Jing Wu, Yu Shi, Peng Zhang, Shiduan Cheng and Jian Ma (1999). "ACK delay control for improving TCP throughput over satellite links. IEEE International Conference on Networks. ICON '99 Proceedings (Cat. No.PR00243), 303-312, https://doi: 10.1109/ICON.1999.796192.

Kong, Y., Zang, H., & Ma, X. (2018, August). Improving TCP congestion control with machine intelligence. In Proceedings of the 2018 Workshop on Network Meets AI & ML (pp. 60-66).

Kurose, J. F., Mañoso, H. C., Pérez, . M. P. Á., & Ross, K. W. (2017). Redes de computadoras: Un enfoque descendente. Madrid: Pearson Educación.

Lee, S. B., Ahn, G. S., & Campbell, A. T. (2001). Improving UDP and TCP performance in mobile ad hoc networks with INSIGNIA. IEEE Communications Magazine, 39(6), 156-165.

Leemaroselin, S. (2015). A Review on Congestion Control Algorithms in MANET. International Journal of Computer Science & Engineering Technology, 6(04), 198-204.

Mahamune, A. A., & Chandane, M. M. (2021). TCP/IP Layerwise Taxonomy of Attacks and Defence Mechanisms in Mobile Ad Hoc Networks. Journal of The Institution of Engineers (India): Series B, 1-19.

Obando Zapata, G. (2018). Regla de tres simple directa: avatares de un algoritmo.

Pérez Palacio, A. (2016). Análisis del comportamiento de TCP Cubic sobre la plataforma ns-3.

Ramírez Aragón, L. (2014.). Simulación y evaluación del desempeño de dos variantes de TCP-Por defecto y TCP-Linux en NS-2. Universidad del Valle.

Rath, M., Pattanayak, B. K., & Pati, B. (2016). Energy efficient MANET protocol using cross layer design for military applications. Defence Science Journal, 66(2), 146-150.

Rodríguez Herlein, D. R., & Talay, C. A. (2016, May). Explorando posibles mejoras de protocolo TCP en redes móviles. In XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina).

Rodríguez Herlein, D. R., Talay, C. A., González, C. N., Trinidad, F. A., & Marrone, L. A. (2017). Consideraciones sobre el comportamiento del protocolo TCP en sus variantes Vegas, Reno, Cubic y Westwood ante errores en ráfaga en una topología híbrida. In XXIII Congreso Argentino de Ciencias de la Computación (La Plata, 2017)..

Sharma, N., Gupta, A., Rajput, S. S., & Yadav, V. K. (2016, February). Congestion control techniques in MANET: a survey. In 2016 Second international conference on computational intelligence & communication technology (CICT) (pp. 280-282). IEEE.

Sirajuddin, M. D., Rupa, C., & Prasad, A. (2016). Advanced congestion control techniques for MANET. In Information Systems Design and Intelligent Applications (pp. 271-279). Springer, New Delhi.

Sing-Borrajo, P. (2014). Evaluación de desempeño de VoIP en redes MANET. ITECKNE: Innovación e Investigación en Ingeniería, 11(1), 7-16.

Tanenbaum, A, (2012), Redes de Computadoras, México, Pearson Educación.

Torres, R., & Tomás, J. G. EVALUACIÓN EFECTIVA DE TCP EN REDES INALÁMBRICAS AD HOC.

Vadivel, R., & Bhaskaran, V. M. (2017). Adaptive reliable and congestion control routing protocol for MANET. Wireless Networks, 23(3), 819-829.Wang, F., & Zhang, Y. (2002, June). Improving TCP performance over mobile ad-hoc networks with out-of-order detection and response. In Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing (pp. 217-225).

Apéndice

Apéndice 1: Comparativo de variable *Ack Delayed*

Con el objetivo de evidenciar el comportamiento de la variable *ACK Delayed* en un ambiente MANET, realizamos variaciones positivas (0,250 y 0,300 segundos) y negativas (0,025, 0,050, 0,100, 0,150 segundos) a su valor original (*ACK Delayed* = 200 ms, RFC 2581) y, tomando en cuenta las conclusiones emitidas en el apartado 3.4, realizamos la simulación usando a New Reno como variante de congestión y a AODV como protocolo de enrutamiento MANET. Los resultados se detallas en la tabla 8.

Tabla 8

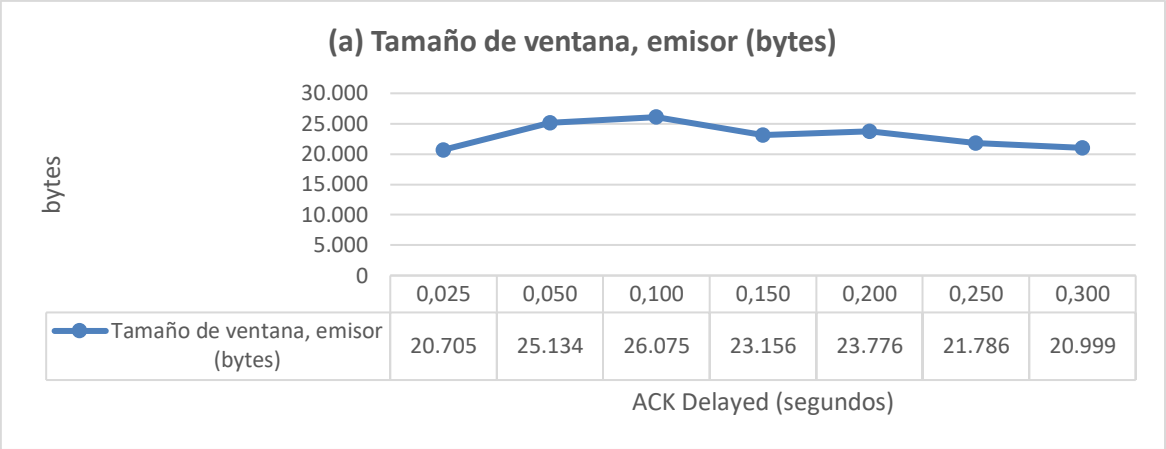
Comparativo de variación ACK Delayed

ACK DELAYED (s)	***Nodo Transmisor***			***Nodo receptor***	
	Tamaño de ventana, emisor (bytes)	**Throughput Wifi (bits/seg)**	**Cantidad de retransmisiones**	**Tráfico recibido (bytes)**	**Latencia LAN Wifi (seg)**
	Media	**Media**	**Media**	**Media**	**Media**
0,025	20.704,90	5.484,36	3,2917	74.865,22	0,0977
0,050	25.134,03	5.573,69	3,8333	72.554,43	0,1071
0,100	26.075,06	6.135,56	2,1765	92.955,23	0,1156
0,150	23.156,43	4.167,56	2,6190	78.244,98	0,1236
0,200	23.775,85	5.709,96	2,2381	83.991,01	0,1251
0,250	21.785,68	5.698,40	2,1765	76.776,73	0,1432
0,300	20.998,94	3.806,76	2,2857	80.155,76	0,1340

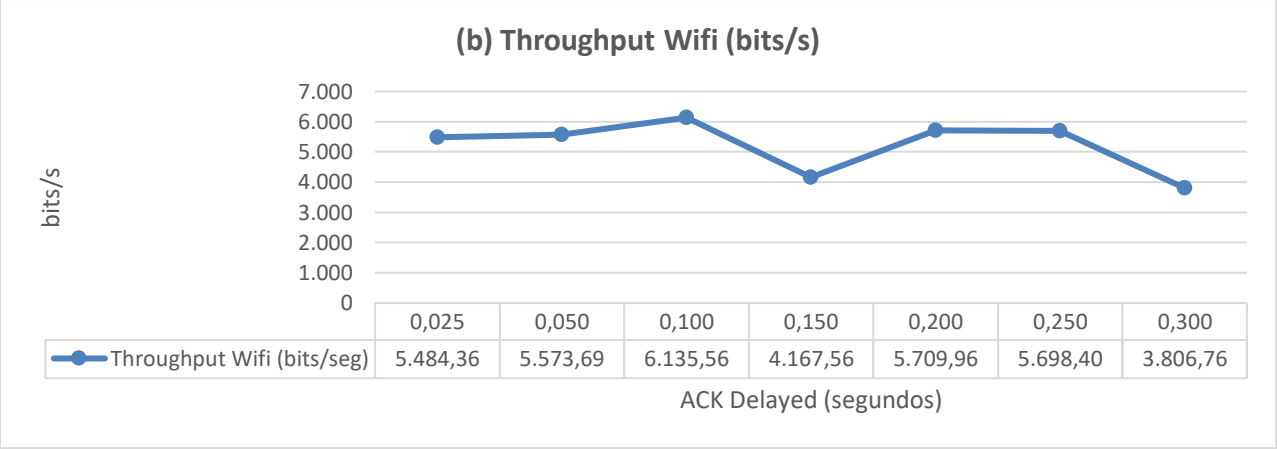
La figura 28 describe el comportamiento de los indicadores respecto a la variación del ACK Dealey

Figura 28

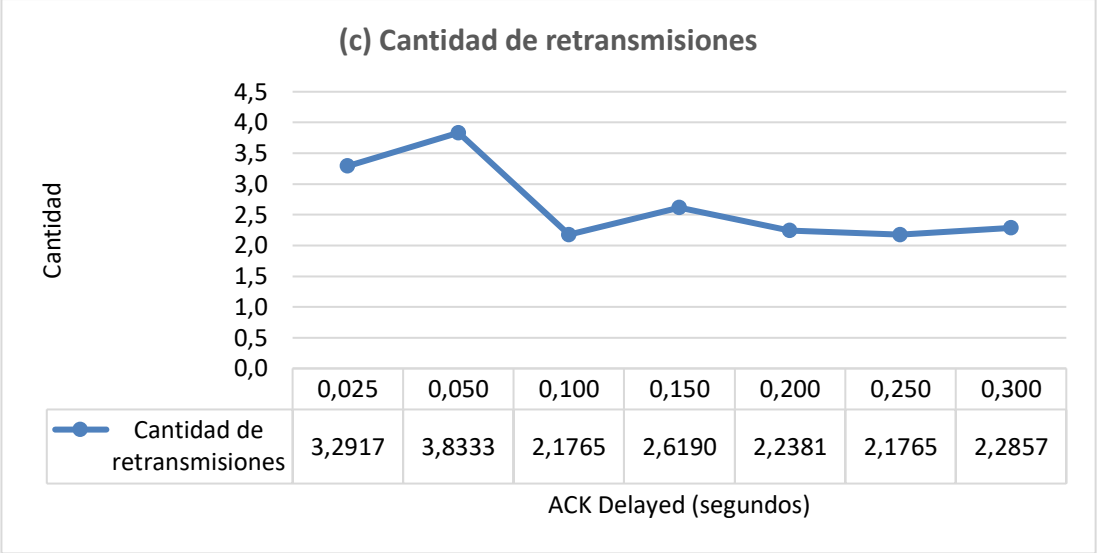
Comparativo, variación ACK Delayed



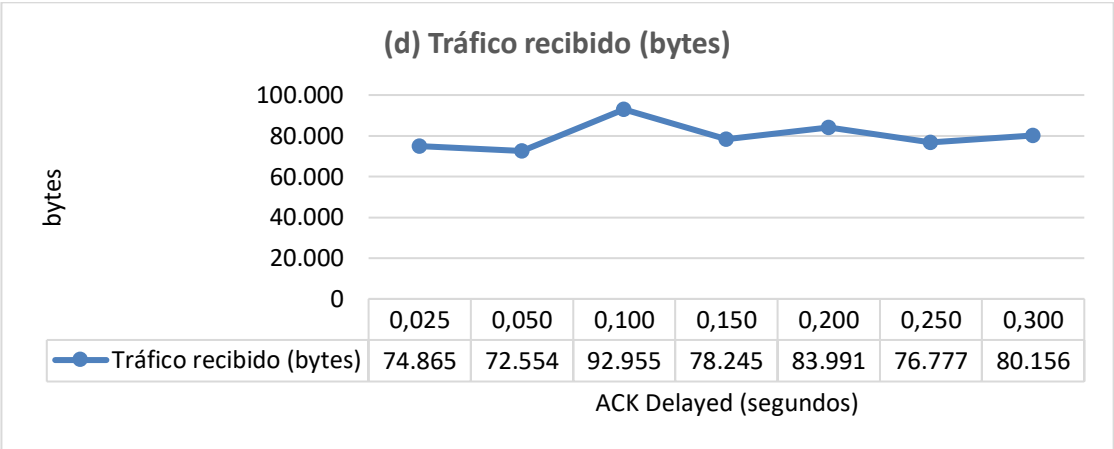
La figura 28 (a), indica que la Cwnd tiende a disminuir cuando la variable ACK Delayed se configura con valores menores o mayores 0,100 segundos; el valor mas alto registrado en la simulación para el tamaño de la ventana de congestión es con el ACK Delayed configurado a 0,100 segundos.



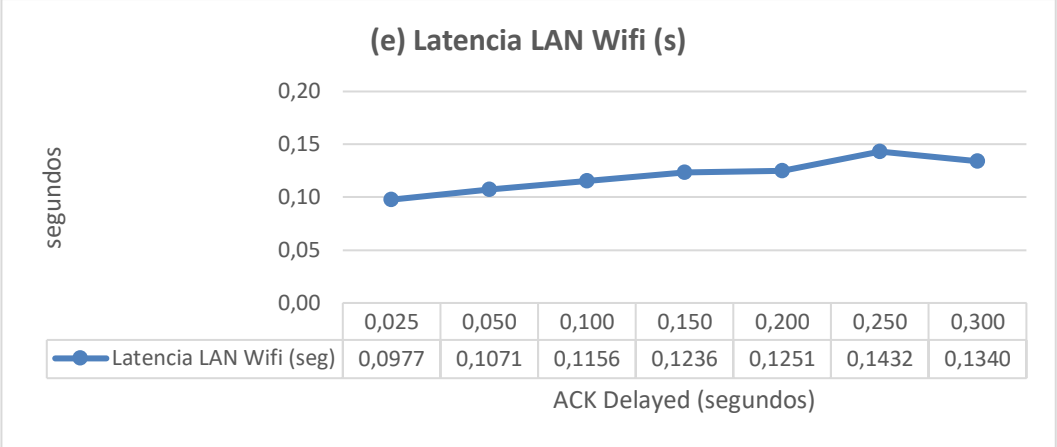
La figura 28 (b) indica que, el throughput registra su punto mas alto cuando el ACK Delayed está configurado a 0,100 segundos; con valores menores o mayores a éste, su valor tiende a disminuir.



La figura 28 (c) indica que el valor mínimo de retransmisiones registradas se da cuando el ACK Delayed está configurado en 0,100 y 0,250 segundos; con valores menores o mayores a estos, la cantidad de retransmisiones tiende a incrementar.



La figura 28 (d) registra el valor más alto para el indicador tráfico recibido cuando el ACK Delayed se configura en 0,100 segundos; con valores menores o mayores a éste, este indicador no supera el máximo registrado.



Para el indicador de latencia, en la figura 28 (e) se registra tendencia a subir cuando el valor del ACK Delayed aumenta; esto tiene una explicación dado que la latencia se calcula una vez que se recibe el ACK correspondiente al segmento, siendo la latencia directamente proporcional.

Conclusión

Por lo registrado en este análisis comparativo, se puede evidenciar que, cuando el ACK Delayed tiene un valor superior o menor a 0,100 segundos, los indicadores se ven afectados de manera negativa respecto a su eficiencia dentro de la transmisión; por ejemplo, el tamaño de ventana, throughput y tráfico recibido, disminuyen y, la cantidad de retransmisiones se incrementan.

Con el valor de ACK Delayed = 0,100 segundos, se tuvo los mejores resultados en cada uno de los indicadores evaluados.