

**IMPLEMENTACION DE UN MODELO DE CALIDAD DE  
SERVICIO SOBRE EL PROTOCOLO DE ENRUTAMIENTO  
AODV USANDO LA ESTIMACION DEL ANCHO DE BANDA  
DISPONIBLE EN CADA NODO**

TESIS DE MAESTRIA PRESENTADA POR:  
**ING. JUAN EDUARDO ELJACH ORTEGA**

ASESOR:  
**NESTOR MISAEL PEÑA TRASLAVIÑA, Ph.D**



UNIVERSIDAD DE LOS ANDES  
DEPARTAMENTO DE INGENIERIA ELECTRICA Y ELECTRONICA  
PROGRAMA MAGISTER  
BOGOTA  
AGOSTO DE 2006

Agradezco

A mi familia por su apoyo incondicional y su constante voz de aliento.

A mis amigos y compañeros por su colaboración incondicional.

A mi asesor por sus consejos, por su tiempo y por su empeño en lograr de nosotros lo mejor.

## CONTENIDO

INTRODUCCION.....	VI
1. FUNDAMENTOS SOBRE REDES MANET.....	1
1.1 Protocolo de Control de Acceso al Medio IEEE802.11.....	2
1.2 Protocolos de Enrutamiento.....	4
1.2.1 Protocolos Proactivos.....	6
1.2.2 Protocolos Reactivos.....	7
1.3 Modelos de Calidad de Servicio.....	14
2. QOS DE ENRUTAMIENTO.....	19
2.1 Modelo Implementado sobre AODV.....	19
2.2 Estimación de Ancho de Banda Disponible Sobre la MAC 802.11.....	24
3. SIMULACIONES.....	30
3.1 Prueba de la Estimación de Ancho de Banda Disponible.....	31
3.2 Prueba de Desempeño del Modelo de QoS de Enrutamiento.....	32
3.2.1 Escenarios de Simulación.....	32
3.2.2 Selección de Parámetros del Modelo.....	33
3.2.3 Resultados Para Redes Estáticas.....	36
3.2.4 Resultados Para Redes Móviles.....	44
4. CONCLUSIONES Y RECOMENDACIONES.....	49
5. REFERENCIAS BIBLIOGRAFICAS.....	49
Anexo 1. Código del algoritmo de estimación de ancho de banda.....	55
Anexo 2. Archivos de configuración para los escenarios de simulación.....	67

## LISTA DE FIGURAS

- Figura 1. Tiempos de espera para acceder al canal en la MAC 802.11.
- Figura 2. Estrategia de enrutamiento en LANMAR.
- Figura 3. Estrategia de enrutamiento en AODV.
- Figura 4. Formato del mensaje RREQ.
- Figura 5. Formato del mensaje RREP.
- Figura 6. Formato del mensaje RERR.
- Figura 7. Arquitectura de QoS en nodos fuente.
- Figura 8. Estrategia de QoS en nodos intermedios.
- Figura 9. Estrategia *Cross Layer*.
- Figura 10. Código usado para el envío de información de ancho de banda desde la *MAC* al protocolo AODV.
- Figura 11. Campos de RREQ con QoS.
- Figura 12. Asignación de tareas según el nodo.
- Figura 13. Diagrama de flujo del algoritmo de QoS de enrutamiento.
- Figura 14. Transición de estados de la *PHY*.
- Figura 15. Extracción de tiempos de estado.
- Figura 16. Escenario de simulación para la estimación de ancho de banda.
- Figura 17. Estimación del ancho de banda disponible en cada nodo.
- Figura 18. Número total de paquetes recibidos para diferentes valores de  $\alpha$
- Figura 19. Retardo punto a punto promedio para diferentes valores de  $\alpha$
- Figura 20. Número total de paquetes recibidos según el ancho de banda mínimo
- Figura 21. Retardo punto a punto promedio según el ancho de banda mínimo
- Figura 22. Razón de envío de paquetes para el escenario de 50 nodos.
- Figura 23. Retardo punto a punto promedio para el escenario de 50 nodos.
- Figura 24. Razón de envío de paquetes para el escenario de 80 nodos.

Figura 25. Retardo punto a punto promedio para el escenario de 80 nodos.

Figura 26. Razón de envío de paquetes para el escenario de 100 nodos.

Figura 27. Retardo punto a punto promedio para el escenario de 100 nodos.

Figura 28. Razón de envío de paquetes para el escenario de 50 nodos con movilidad.

Figura 29. Retardo punto a punto promedio para el escenario 50 nodos con movilidad.

Figura 30. Razón de envío de paquetes para el escenario de 80 nodos con movilidad.

Figura 30. Razón de envío de paquetes para el escenario de 80 nodos con movilidad.

Figura 31. Retardo punto a punto promedio para el escenario de 80 nodos con movilidad.

Figura 32. Razón de envío de paquetes para el escenario de 100 nodos con movilidad.

Figura 33. Retardo punto a punto promedio para el escenario de 100 nodos con movilidad.

## INTRODUCCION

El mundo ha empezado a darse cuenta del potencial económico y comercial que tienen las redes inalámbricas sin infraestructura, como lo son las redes Ad Hoc. Sin embargo, para dar este salto al uso comercial, es importante garantizar un nivel razonable de calidad de servicio (QoS). Para tal efecto, es necesario establecer un protocolo de QoS que se encargue de controlar los cuatro factores que influyen en dicho nivel de servicio: la administración del ancho de banda, QoS de enrutamiento, control de admisión y control de congestión [1].

Esta investigación aborda el problema de la estimación de ancho de banda disponible a partir de la razón de utilización del canal de radio, cuyo enfoque permite considerar los efectos de nodos vecinos sobre la estimación. Tener en cuenta el ancho de banda disponible en cada nodo se vuelve fundamental debido a la escasez de este recurso en este tipo de redes [2]. Además, implementa un modelo de QoS de enrutamiento sobre el protocolo AODV basado en la obtención del ancho de banda disponible de cada nodo. El desarrollo y la posterior implementación fueron realizados en un software de simulación de redes conocido como Qualnet® [3]-[4].

## 1. FUNDAMENTOS SOBRE REDES MANET

Una red móvil Ad Hoc (MANET) es un conjunto de nodos que se comunican de manera inalámbrica, sin la necesidad de un proveedor del servicio. Cada nodo tiene la facultad de un *Router*, siendo capaces de enrutar paquetes a través de la red hasta llegar a su destino, con la ayuda de protocolos de enrutamiento. Debido a la movilidad de los nodos, su topología suele ser arbitraria y su forma puede cambiar en cualquier momento [5]. Además de su movilidad, estas redes están expuestas a los problemas típicos de las redes inalámbricas, tales como ruido, interferencia y desvanecimiento de la señal, convirtiéndose en un obstáculo a la hora de garantizar un nivel de calidad de servicio óptimo. Una visión general acerca de las redes Ad Hoc, sus inconvenientes, sus limitantes y sus posibles usos se presenta en [6].

Este tipo de redes suelen ser muy útiles en actividades de alta movilidad y localidades de baja infraestructura de comunicaciones como las unidades militares y de auxilio durante un operativo o una emergencia. Debido a las características móviles de este tipo de redes, recursos como ancho de banda, consumo de energía, capacidad de almacenamiento, capacidad de procesamiento y velocidad de transmisión se vuelven temas críticos que deben ser tenidos en cuenta al momento de diseñar la red.

La mayoría de las investigaciones se encaminan a integrar este tipo de redes con el resto de redes existentes [7], con el objeto de que sean transparentes al usuario. Este objetivo plantea una serie de temas de investigación y desarrollo en el cual se hace necesario trabajar para obtener un aprovechamiento máximo con un desempeño óptimo de las MANET. El protocolo de enrutamiento es uno de los

factores más importantes para lograr dicha transparencia al usuario, por lo que se han presentado varias soluciones para enfrentar este problema.

Los modelos de enrutamiento en las MANET, como en toda red de comunicaciones, tiene como objetivo primario establecer la ruta más eficiente entre dos nodos; pero en MANET se hace mayor énfasis en que el consumo de recursos de red y sobrecarga debe ser lo mínimo posible, sin comprometer la confiabilidad de la red [6] y [8].

Existen dos formas de evaluar el desempeño de un modelo, una es la implementación de escenario real de prueba (*testbed*) el cual presenta resultados reales, pero suele ser complicado en términos de costos e implementación. La otra forma, un poco más adecuada si no se cuentan con los recursos necesarios para implementar un *testbed*, es la simulación; en telecomunicaciones existen varias herramientas de simulación de redes, NS-2 [9] y Qualnet® [3] y [4], entre otras.

### 1.1 Protocolo de Control de Acceso al Medio IEEE802.11

El estándar IEEE 802.11 es una norma aplicada al control de acceso al medio de una red inalámbrica con una cobertura limitada, también conocidas como redes WLAN (*Wireless Local Area Network*) [10] y [11]. A nivel de la capa MAC (*Medium Access Control*), el estándar se vale de dos funcionalidades básicas: DCF (*Distributed Coordination Function*) y PCF (*Point Coordination Function*). Como la función adicional PCF sólo sirve para redes con cierta infraestructura [11], no es aplicable al caso de las MANET, por lo que se hará mayor énfasis sobre la función DCF.

El método fundamental de acceso en el IEEE MAC 802.11 es una DCF conocida como CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*), la cual se implementa en cada nodo [11]. Este método procura evitar colisiones entre



paquetes enviados por diferentes fuentes dentro del mismo rango de transmisión, de tal forma que la pérdida de paquetes debido a estas colisiones se reduzca considerablemente. El tiempo en que la red opera en modo DCF se conoce como CP (*Congestion Period*).

Básicamente, lo que hace este método es examinar el canal de radio para poder obtener el estado del mismo cada vez que se solicite; un nodo que necesite enviar datos a través del canal pide información acerca del estado del mismo y dependiendo del estado en ese momento, el nodo fuente decide si puede o no enviar su información mediante la función CSMA/CA.

El estándar MAC 802.11 considera cuatro estados posibles del canal de radio, de los cuales tres se consideran estados de actividad del canal o estados ocupados:

1. *Busy State* (enviando o recibiendo paquetes).
2. *Carrier Sensing Channel Busy* (hay nodos vecinos transmitiendo).
3. *Virtual Carrier Sensing Busy* (enviando o recibiendo RTS o CTS).
4. *Idle State* (un estado libre).

Si un nodo desea enviar información a través del canal debe esperar hasta que se desocupe el mismo, es decir, que se encuentre en el estado libre. Además, según el método de acceso CSMA/CA, una vez que se percibe libre el canal, el nodo debe esperar un tiempo especificado por el tiempo mínimo entre tramas (*frames*) para poder enviar su información, este tiempo se conoce como IFS (*Interframe Space*).

El momento en que queda libre el canal se considera el de mayor probabilidad de colisiones debido a que varios nodos pueden estar censando el canal y tratando de enviar paquetes al mismo tiempo, por lo que el método de acceso tiene en cuenta también un tiempo de espera que se le asigna de manera aleatoria a cada nodo, este tiempo se conoce como CW (*Contention Window*) y la parte aleatoria la

genera un algoritmo implementado dentro de esta ventana creando un número aleatorio de TS (*Time Slots*), esta implementación se conoce como algoritmo de *Backoff* tal y como se muestra en la figura 1.

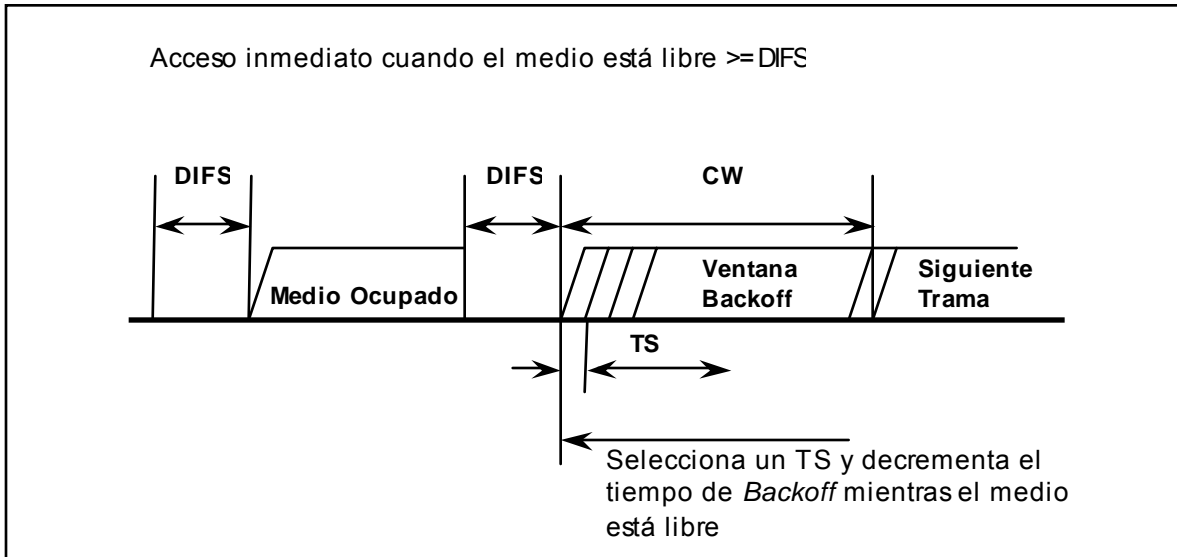


Figura 1. Tiempos de espera para acceder al canal en la MAC 802.11, tomado de [11].

Cada uno de los TS de la figura 1 representa un nodo solicitando acceso al medio, el algoritmo de *Backoff* selecciona un TS de manera aleatoria y decreta el tiempo de *Backoff* mientras el medio está libre. Esta estrategia evita colisiones de paquetes enviados por los nodos que compiten por el acceso al medio.

## 1.2 Protocolos de Enrutamiento

Con el fin de disminuir los errores y la pérdida de paquetes en las sesiones de tráfico de una MANET, se han diseñado protocolos enfocados a la selección de rutas óptimas discriminando entre los miembros de la red, los que tengan mejores condiciones para establecer la ruta [12]. Por lo general, los algoritmos de enrutamiento se centran en buscar las rutas con menor número de saltos al

destino como los presentados en [13]-[15], pero esto no siempre es suficiente, se hace necesario revisar periódicamente la presencia de los nodos debido a cambios en la topología física de la red; la desconexión o desaparición de un nodo puede causar una ruptura de enlace que debe ser tomada en cuenta por el protocolo de enrutamiento.

Actualmente, existen dos enfoques respecto a los protocolos de enrutamiento, cada enfoque presenta ventajas y desventajas con respecto a las MANET, las cuales se mencionan mas adelante. Los protocolos más usados se presentan en la tabla 1.

PROTOCOLOS DE ENRUTAMIENTO	
PROACTIVOS	REACTIVOS
DSDV ( <i>Destination Sequenced Distance Vector</i> ) [16]	DSR ( <i>Dynamic Source Routing</i> ) [17]
FSR ( <i>Fisheye State Routing</i> ) [18]	AODV ( <i>Ad Hoc On Demand Distance Vector</i> ) [13] y [14]
OLSR ( <i>Optimized Link State Routing Protocol</i> ) [19]	TORA ( <i>Temporally-Ordered Routing Algorithm</i> ) [20]
LAN-MAR ( <i>Landmark Ad Hoc Routing Protocol</i> ) [1], [15] y [21]	ABR ( <i>Associativity Based Routing</i> ) [22]

Tabla 1. Protocolos de enrutamiento para redes Ad Hoc.

En [12] se presenta una clasificación de la mayoría de protocolos de enrutamiento usados en redes Ad Hoc según la estrategia, y proporciona una breve descripción de algunos de ellos. Además, presenta una tabla comparativa de *FSR*, *OLSR*, *AODV* y *DSR*. Todos los protocolos mencionados usan como métrica de ruta el camino más corto.

### 1.2.1 Protocolos Proactivos

Estos protocolos manejan lo que se conoce como tablas de enrutamiento, las cuales se transmiten y actualizan a través de la red, es decir, los nodos miembros de la red tienen conocimiento de la topología de la misma en todo momento [23]. Cuando un nodo necesita enviar paquetes, simplemente consulta su tabla de enrutamiento y el protocolo se encarga de seleccionar la mejor ruta. Esta facultad presenta ciertas ventajas y desventajas que pueden ser sensibles en una MANET.

El tener un conocimiento de la topología de la red en todo momento presenta ventajas razonables en redes alambradas, donde los recursos no son tan limitados ya que se disminuye el tiempo de transmisión de paquetes considerablemente puesto que se tiene conocimiento de las rutas existentes. En el caso de las MANET, esta facultad puede convertirse en una desventaja, ya que se están consumiendo recursos bastante limitados en este tipo de redes como lo es el ancho de banda disponible [23]. Es por eso que se hace necesario evaluar las condiciones de la red antes de seleccionar una estrategia de enrutamiento.

En este trabajo, de todos los protocolos proactivos, se estudió LANMAR como opción de enrutamiento debido a su facultad para escalar redes de miles de nodos, por esta razón se presenta con mayor detalle.

#### 1.2.1.1 LANMAR (*Landmark Ad Hoc Routing Protocol*)

Este protocolo organiza grupos de nodos como se muestra en la figura 2, con el fin de lograr una mayor escalabilidad. Su estrategia de enrutamiento involucra un protocolo de enrutamiento plano (FSR [18] en [1] y [15] para la conexión local). Cada grupo selecciona dinámicamente un *Landmark* entre sus miembros, siguiendo algunas reglas de selección; este *Landmark* se encarga de recoger la información de cada uno de los miembros de su grupo, así como la información proveniente de otros *Landmark*.

Cuando se requiere una conexión entre nodos de distintos grupos, es cuando entra en juego la estrategia de LANMAR, utilizando lo que se conoce como “*Landmark Distance Vector*”, los cuales son utilizados para propagar información referente a la tabla de enrutamiento de cada grupo. El *Landmark* seleccionado propaga su “*Landmark Distance Vector*” a los *landmark* vecinos, de tal forma que se obtiene un conocimiento total de las distintas rutas. Una implementación de prueba (*Test-bed*) de este protocolo se presenta en [15].

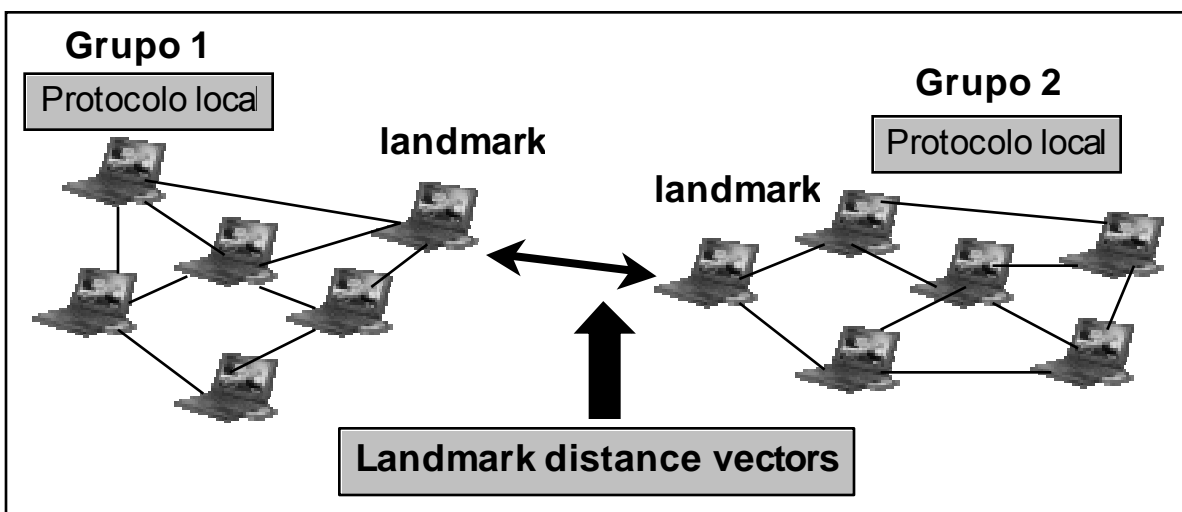


Figura 2. Estrategia de enrutamiento en LANMAR.

### 1.2.2 Protocolos Reactivos

Estos protocolos fueron diseñados específicamente para redes inalámbricas con el fin de enfrentar el problema de recursos limitados. Su funcionamiento difiere de los preactivos en que no manejan tablas de enrutamiento y por consiguiente, no tienen conocimiento de la topología de la red en todo momento, como su nombre lo indica, solo solicitan la información de enrutamiento cuando necesitan enviar información.

Esta estrategia de enrutamiento consume menos ancho de banda que su contraparte, pero aumenta el retardo de transmisión de paquetes ya que tiene que esperar a encontrar una ruta disponible.

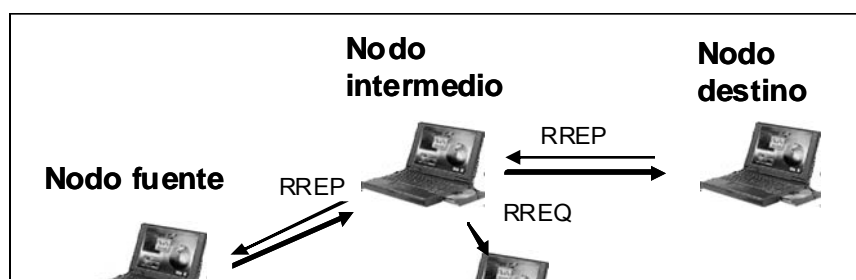
De todos los protocolos reactivos, se consideró AODV como estrategia de enrutamiento por su buen desempeño en redes de cientos de nodos.

#### 1.2.2.1 AODV (*Ad Hoc On Demand Distance Vector*)

Este protocolo se vale de paquetes de control que envía a la red con cierta información consecuentemente con una tabla de enrutamiento, para evaluar la condición de cada ruta existente en búsqueda de cambios en la topología de la red. El protocolo notifica al conjunto de nodos afectados para que ellos puedan eliminar la ruta mediante el "lost link". Este protocolo funciona según las especificaciones de [13] y su funcionamiento se explica en [14].

AODV tiene un funcionamiento parecido a DSR, ambos protocolos difieren en que AODV usa lo que se conoce como "*Destination Sequence Number*" para liberarse de los *loops* que se presentan en DSR [17] Entre dos rutas disponibles se escoge la de mayor "*Destination Sequence Number*".

AODV difunde vía *Broadcast* unos paquetes de descubrimiento llamados Mensaje de Saludo (*HELLO MSG*), con el cual cada nodo anuncia su presencia en la red. Estos mensajes son recibidos por los nodos vecinos que se encuentran dentro del rango de transmisión del nodo que los difunde.



### Figura 3. Estrategia de enrutamiento en AODV.

La figura 3 muestra la estrategia de enrutamiento destacando la dirección de los paquetes de control.

#### 1.2.2.1.1 PAQUETE DE CONTROL RREQ (*ROUTE REQUEST*)

Este paquete de control se difunde vía *Broadcast* por toda la red inicialmente por el nodo fuente, es decir, el que desea obtener una ruta activa para enviar su información. La información de enrutamiento que contiene es la siguiente:

- *Type*: 1 para RREQ.
- *J (Join Flag)*: Esta bandera está reservada para *Multicast*.
- *R (Repair Flag)*: Esta bandera está reservada para *Multicast*.
- *G (Gratuitious RREP Flag)*: Esta bandera indica si se debe enviar un *Gratuitious RREP* via *unicast* al nodo especificado en el campo *Destination IP Address*.
- *D (Destination Only Flag)*: Esta bandera indica que solo el destino puede responder a este mensaje.
- *U (Unknown sequence number)*: Indica que el *Destination Sequence Number* es desconocido.





Este paquete de control puede ser generado tanto por un nodo intermedio como por el mismo destino, diferenciándose uno del otro por la dirección IP, por el número de secuencia de destino (*Destination Sequence Number*), el cual se coloca al máximo valor para el nodo destino y por el número de saltos, el cual se coloca en cero (0) para el nodo destino. Los únicos nodos intermedios que pueden enviar RREP, son aquellos que poseen rutas activas. La información que contienen estos paquetes es la siguiente:

- *R (Repair Flag)*: Esta bandera está reservada para *Multicast*.
- *A (Acknowledgment required)*: Esta bandera solicita un RREP-ACK cuando está activada, ver [13].
- *Reserved*: Una bandera que envía un cero y que se ignora en la recepción.
- *Prefix Size*: Si es diferente de cero, indica que el siguiente salto puede ser usado por cualquier nodo con el mismo prefijo de enrutamiento, según el *Prefix Size*, como el destino requerido.
- *Hop Count*: Es el número de saltos desde el nodo fuente hasta el nodo destino.
- *Destination IP Address*: Es la dirección IP del nodo destino.
- *Destination Sequence Number*: Es el número de secuencia del destino asociado a la ruta establecida.
- *Originator IP Address*: Es la dirección IP del nodo que originó el RREQ para el cual se estableció la ruta.
- *Lifetime*: Es el tiempo en milisegundos durante el cual los nodos que reciben un RREP consideran la ruta como válida.

El mensaje RREP envía la información de enrutamiento siguiendo el formato presentado en [13], y que se muestra en la figura 5.

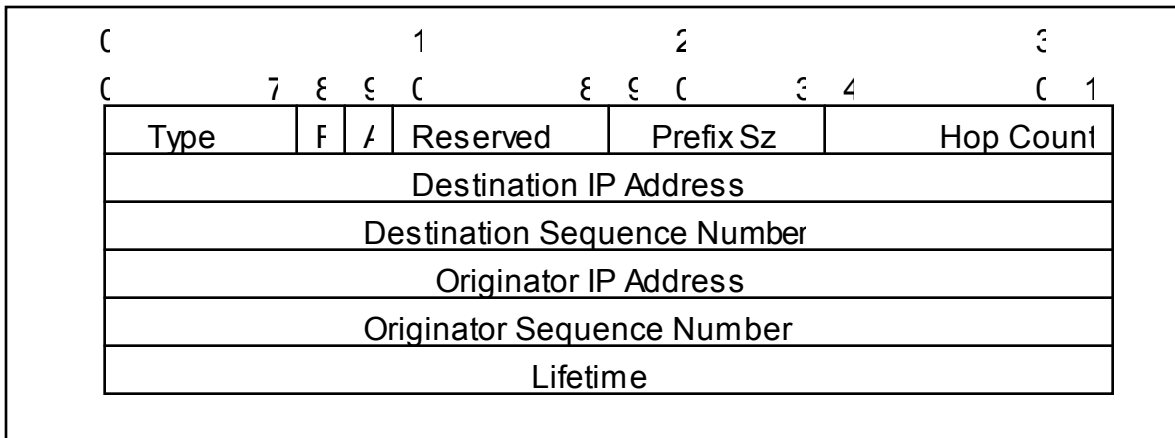


Figura 5. Formato del mensaje RREP, tomado de [13].

### 1.2.2.1.3 PAQUETE DE CONTROL RERR (*ROUTE ERROR*)

Este paquete de control se envía cuando existe algún cambio en la topología de la red, cuando desaparece un nodo intermedio o el mismo nodo fuente. AODV toma ciertas medidas según el caso. En primera instancia, deshabilita la ruta esperando que el error en dicha ruta se arregle, dándole un tiempo de espera. Si definitivamente el nodo se ha desconectado de la red, el protocolo procede a eliminar la ruta del todo. La información que contienen estos paquetes es la siguiente:

- *N (No Delete Flag)*: Cuando se activa, indica que se ha reparado un enlace y no se debe borrar la ruta.
- *Reserved*: Una bandera que envía un cero y que se ignora en la recepción.
- *DestCount*: Es el número de destinos inalcanzables incluidos en el mensaje; debe haber al menos 1.
- *Unreachable Destination IP Address*: Es la dirección IP del nodo que se ha vuelto inalcanzable.
- *Unreachable Destination Sequence Number*: Es el número de secuencia del destino en la tabla de enrutamiento asignado al destino listado en el “*Unreachable Destination IP Address*”.

El mensaje RERR envía la información de enrutamiento siguiendo el formato presentado en [13], y que se muestra en la figura 6.

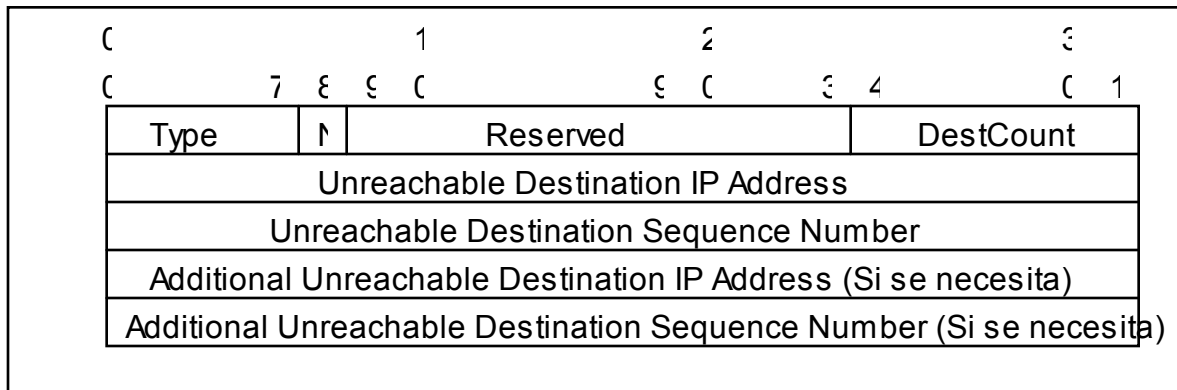


Figura 6. Formato del mensaje RERR, tomado de [13].

#### 1.2.2.1.4 CARACTERISTICAS DEL PROTOCOLO

El protocolo busca rutas sólo cuando se necesita, es decir, cuando no existe una ruta activa conocida por los nodos de la red. Mientras exista una ruta activa conocida para algún destino, ésta va a ser utilizada sin necesidad de la intervención del protocolo. Los números de secuencia ayudan a mantener información actualizada de las rutas disponibles.

Por motivos de eficiencia, solo se mantiene información de un salto en la ruta, en vez de la ruta completa. Esto ayuda a disminuir el retardo de transmisión de paquetes generado por el proceso de descubrir rutas activas.

En el caso que se requiera de una comunicación bidireccional entre los nodos fuente y destino, el que origina el RREQ debe habilitar la bandera G que está relacionada con el "*gratuitous RREP*", el cual permite guardar toda la información recogida por los nodos intermedios para ser utilizada por el mismo nodo destino.

### 1.3 Modelos de Calidad de Servicio (QoS)

En general, según [1], los modelos de QoS tienen en cuenta cuatro factores para obtener un buen desempeño de la red:

1. Estimación del ancho de banda disponible.
2. Control de Admisión de Llamadas (CAC).
3. QoS de enrutamiento.
4. Control de congestión.

Todos los modelos buscan disminuir la pérdida de paquetes debido a los nodos cuellos de botella entre otros factores. Esto se vuelve fundamental cuando se quiere transmitir datos en tiempo real tales como VOIP (Voz sobre IP) o Video, datos que requieren cierto grado de QoS.

Debido a las características de las MANET, es comprensible que garantizar un alto grado de QoS más que complejo, puede resultar imposible. Sin embargo el esfuerzo de los investigadores alrededor del mundo se ha centrado en este problema. En la actualidad existen varias estrategias enfocadas en la implementación de arquitecturas capaces de garantizar cierto grado de QoS. La mayoría son estrategias basadas en las utilizadas por redes alambradas como el protocolo Servicios Integrados (*IntServ*) [24], poco adecuadas debido a su alto consumo de ancho de banda disponible [2].

Otro caso es el de Servicios Diferenciados (*DiffServ*) [25], que se adecua mejor al ambiente de las redes MANET ya que es un modelo ligero comparado con *IntServ*. Aunque se han creado modelos híbridos que extraen lo mejor de cada modelo.

Entre las arquitecturas de QoS desarrolladas específicamente para las redes MANET están: SWAN [26], INSIGNIA [27], CEDAR [28] y FQMM [29]. Un análisis sobre algunos de estos modelos se presenta en [2].

Existe una estrategia que utiliza un enfoque diferente, extrae las capacidades de las redes neuronales para mejorar la forma como se enrutan los datos. La estrategia presentada en [30]-[36] utiliza la técnica de “Paquetes Cognitivos”, paquetes inteligentes que son capaces de tomar decisiones concernientes a la mejor ruta, basándose en un algoritmo de aprendizaje.

El modelo de QoS de enrutamiento implementado en [37], al igual que el presentado en este trabajo, utiliza la técnica *Cross Layer* para la estimación de ancho de banda disponible y su posterior utilización en el protocolo de enrutamiento. El modelo se implementó sobre el protocolo de enrutamiento AODV, y se utilizaron los mensajes “*HELLO*” para difundir la información del ancho de banda disponible por la red.

La prueba del modelo se realizó sobre un escenario de 50 nodos estáticos, con 5 sesiones de tráfico, variando la carga para analizar su influencia sobre el desempeño de la red. Los resultados obtenidos muestran una mejora de alrededor del 60% sobre el protocolo AODV, para valores de carga de 0.5Mbps. El desempeño del modelo se acerca al del protocolo AODV a medida que se disminuye la carga.

El modelo de QoS presentado en [1] emplea la misma técnica *Cross Layer* para implementar QoS de enrutamiento, con la diferencia que éste utiliza el protocolo de enrutamiento utilizado LANMAR, adecuado para redes de gran tamaño debido a su funcionalidad de grupos con protocolo de enrutamiento interno (Fisheye). Las figuras 7 y 8 muestran la distribución de funciones de la arquitectura según el nodo, ya sea nodo fuente o un nodo intermedio.

Como se aprecia en la figura 7, los nodos fuente discriminan el tipo de flujo con el fin de efectuar el control de admisión, regulan la tasa de transmisión de los flujos *best-effort* dependiendo del estado de la red dándole prioridad al flujo de tiempo

real, luego efectúan la estimación de ancho de banda desde la capa *MAC* y toman una decisión de enrutamiento.

La información del ancho de banda disponible es calculada por todos los nodos de la red como se muestra en las figuras 7 y 8, posteriormente, ésta es transmitida entre todos los nodos de cada grupo por el protocolo interno *FSR*. Luego se comparte esta información con todos los grupos mediante el Landmark Distance Vector, para lograr el QoS de enrutamiento.

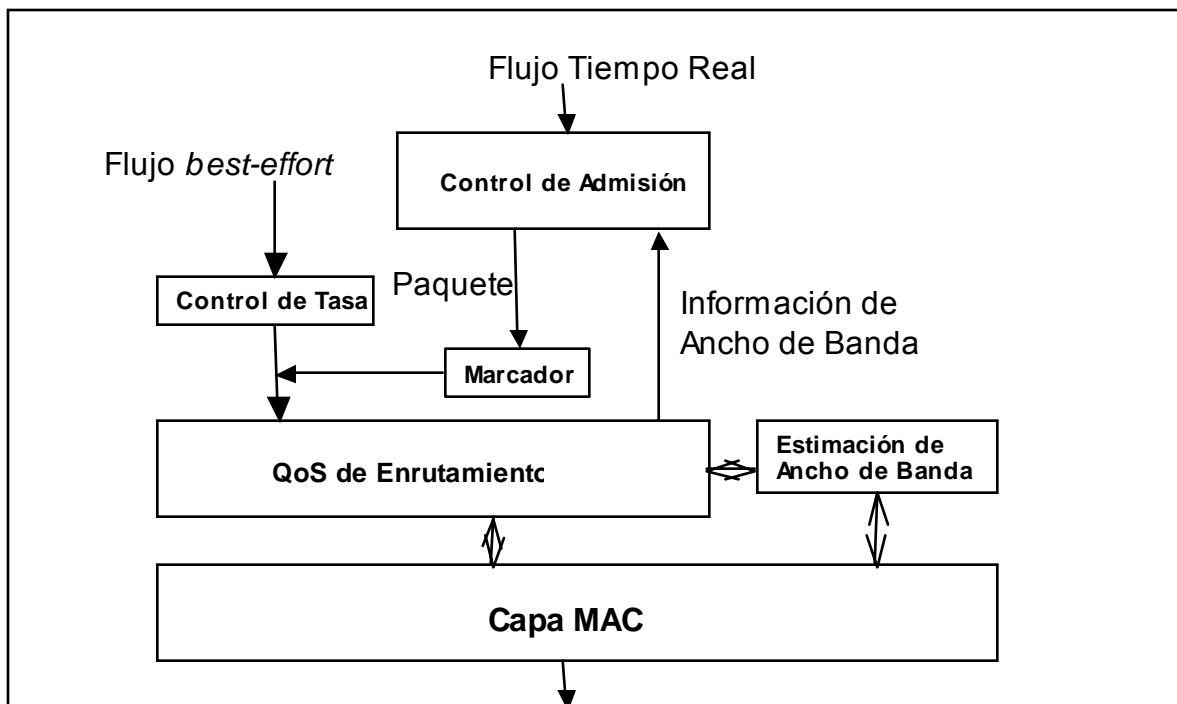


Figura 7. Arquitectura de QoS en nodos fuente, tomado de [1].

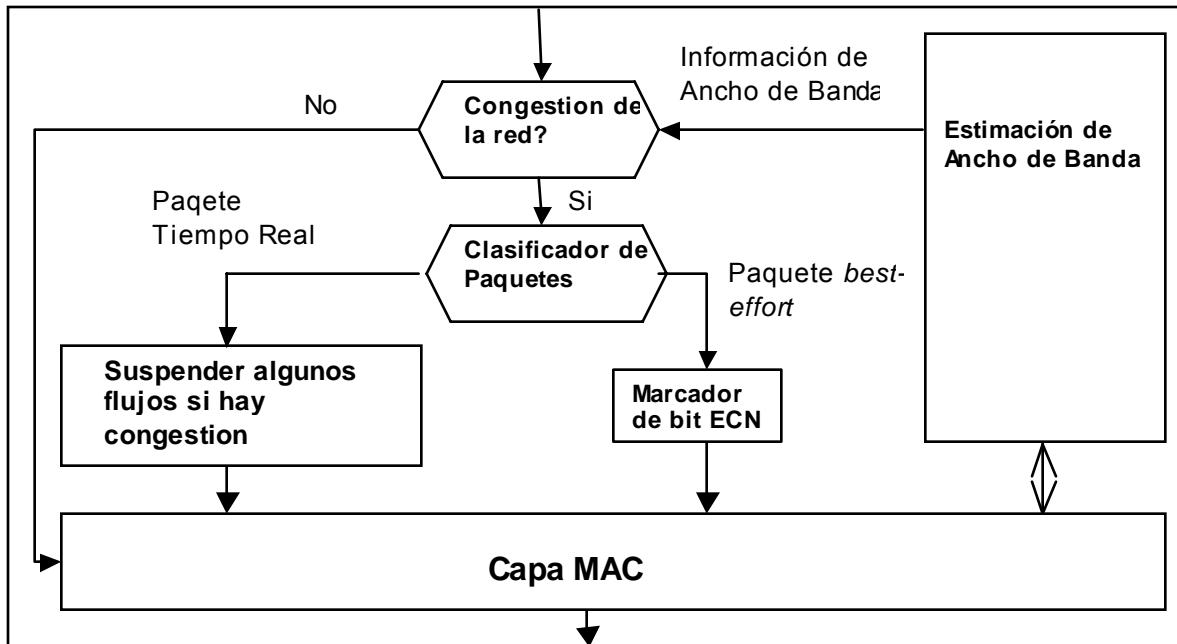


Figura 8. Estrategia de QoS en nodos intermedios, tomado de [1].

Si se desea lograr una buena escalabilidad, se debe usar un protocolo con alguna forma de estructura jerárquica como lo es LANMAR. Se sabe que los protocolos reactivos funcionan muy bien para redes pequeñas de hasta cientos de nodos, pero tienen problemas para escalar redes de miles de nodos bajo un ambiente móvil [1], aunque presentan un mejor desempeño que los protocolos proactivos planos [1], tales como FSR [F].

La información del ancho de banda disponible de cada nodo es incluida en el protocolo de enrutamiento, ya que la arquitectura de QoS se basa en este parámetro para definir su nivel óptimo de servicio. Para lograr esto, es necesario modificar la implementación tanto del protocolo interno (FSR) como del global (LANMAR), adhiriéndole ciertas características de QoS en su tabla de enrutamiento.

Para lograr un Control de Admisión de Llamadas (CAC por sus siglas en inglés), la arquitectura de QoS implementada en [1] revisa la tabla de enrutamiento para

determinar el alcance al destino, si el destino no está al alcance el flujo es rechazado. Los valores de ancho de banda disponible de los nodos se comparan para determinar el ancho de banda mínimo y máximo, de esta forma, si un flujo requiere un ancho de banda mayor al máximo establecido, este será rechazado. Por el contrario, si el flujo requiere un ancho de banda menor que el mínimo establecido, el flujo será admitido. Si el flujo requiere un ancho de banda que cae entre el ancho de banda mínimo y máximo, se envía un paquete de prueba para determinar si es posible utilizar esa ruta, si no hay problemas, el flujo es aceptado.

El protocolo también aplica un control de congestión, utilizando banderas ECN (*Explicit Congestion Notification*) para identificar los flujos *best-effort*. Los bits ECN se activan cuando se detecta congestión en la red, cuando esto ocurre, se envía la notificación a los nodos fuente, los cuales reducen la tasa de transmisión a los flujos *best-effort*, mientras que a los de tiempo real se les aplica el control de admisión. Este control de congestión es efectuado por los nodos intermedios, como se muestra en la figura 8, aceptando o rechazando flujos de tiempo real según las condiciones de la red.

Para un escenario de 100 nodos estáticos, con 10 sesiones de tráfico, el modelo mejora el desempeño del protocolo LANMAR en un 20% sobre la razón de envío de paquetes. En cuanto al retardo punto a punto promedio, el desempeño es el mismo que el del protocolo original.

En [38] se presenta un estudio completo sobre modelos de QoS, cuyo objetivo principal es ofrecer un nivel de QoS sobre redes mixtas, redes que tengan una componente inalámbrica y un componente alambrado para proveer el acceso a Internet. La investigación presenta un recuento de varias estrategias de enrutamiento para las redes Ad Hoc aisladas, y al mismo tiempo menciona varios modelos conocidos para proveer QoS, tanto a nivel de la MAC, como en el mismo protocolo de enrutamiento.



## 2. QOS DE ENRUTAMIENTO

El modelo implementado se basó en la estrategia presentada en [1].

El modelo utiliza el método *Cross Layer* (Capas Cruzadas), el cual aprovecha las capacidades de dos capas de la pila de protocolos de cada nodo logrando una interacción entre ambas como se muestra en la figura 9. Esto hace posible la implementación del protocolo de QoS de enrutamiento en el funcionamiento normal de los protocolos utilizados.

El modelo busca mejorar el desempeño del protocolo de enrutamiento AODV, mediante la adición de información sobre el ancho de banda disponible de cada nodo en sus parámetros de enrutamiento, cuya estimación se realiza en la capa *MAC*, ya que ésta posee la información necesaria para dichos cálculos.

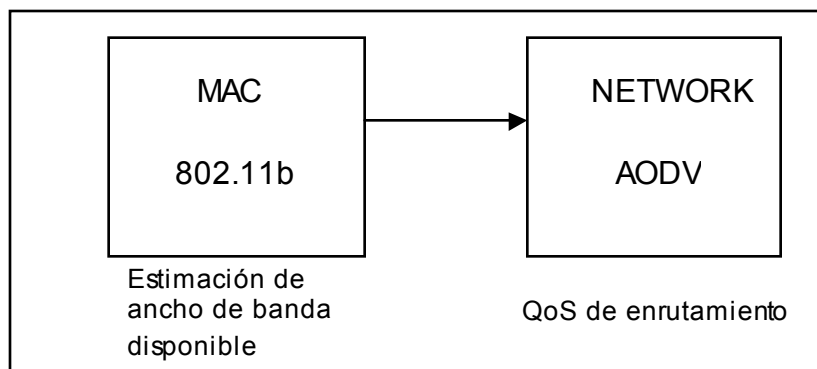


Figura 9. Estrategia Cross Layer.

### 2.1 Modelo Implementado sobre AODV

Se ha probado que los protocolos de enrutamiento reactivos tienen un mejor desempeño en redes pequeñas de hasta cientos de nodos, y lo opuesto ocurre

para redes de miles de nodos [1]. Como este trabajo se enfocó en el estudio de redes pequeñas, se consideró AODV como protocolo de enrutamiento debido a su buen desempeño en este tipo de redes.

El algoritmo implementado sobre AODV básicamente utiliza la información proveniente de la *MAC*, es decir, el ancho de banda disponible de cada nodo, para tomar decisiones de enrutamiento. La información calculada en la *MAC* se envía al protocolo AODV mediante mensajes predeterminados de Qualnet® así:

- Se crean dos nuevas variables en la estructura *Message* que se encuentra en el archivo "*message.h*", ubicado en la carpeta "*addons\seq*" instalada en Qualnet®. Las nuevas variables "*BWtnode*" y "*BWt*" deben guardar la información de la estimación del ancho de banda como se muestra en la figura 10.
- En la carpeta "*include*" instalada en Qualnet®, se modifica el archivo "*api.h*" con el fin de incluir un nuevo evento que indica el envío de la información del ancho de banda estimado desde la *MAC* hacia el protocolo AODV, el cual se encuentra en la capa de red. El nuevo mensaje añadido a la lista de eventos de Qualnet® relacionada con el protocolo AODV en el archivo "*api.h*" se llama "*MSG\_AODV\_AvailableBW*" y se invoca como se muestra en la figura 10.

Estos mensajes son recibidos por una función implementada en "*aodv.cpp*" llamada "*AodvHandleProtocolEvent*", a esta rutina se le agrega un nuevo evento para simular la llegada del ancho de banda estimado en la *MAC*, al cual se le ha llamado "*MSG\_AODV\_AvailableBW*".

Para aprovisionar al protocolo AODV con características de QoS, se modificó, inicialmente, la función que maneja los eventos del protocolo como se explicó previamente. Luego se modificó la estructura de los paquetes de control RREQ,

dotándolos con campos de QoS usando como métrica el ancho de banda disponible. A los paquetes RREQ se le adicionó un nuevo campo correspondiente al requerimiento de ancho de banda mínimo por parte del nodo fuente. La información que llevan los paquetes RREQ se muestra en la figura 11.

```

Message* newMsg; // Crea un nuevo mensaje en la estructura Message.
// Reserva espacio en memoria para el mensaje de tipo MSG_AODV_AvailableBW.
newMsg = MESSAGE_Alloc(node,
                        NETWORK_LAYER,
                        ROUTING_PROTOCOL_AODV,
                        MSG_AODV_AvailableBW);

newMsg->BWtnode = Nodestr;
newMsg->BWt = BWstr;
// Coloca el mensaje en la pila de eventos de Qualnet.
MESSAGE_Send(node,newMsg,delay_msg);

```

Figura 10. Código usado para el envío de información de ancho de banda desde la MAC al protocolo AODV.

Número de Saltos
ID de RREQ
Dirección IP de Destino
Número de Secuencia de Destino
Dirección IP de fuente
Ancho de Banda Mínimo

Figura 11. Campos de RREQ con QoS

Para lograr un nivel de QoS, se hace necesario colocar un valor de ancho de banda mínimo disponible [1] y [37], el cual es definido por el nodo fuente como ya se explicó, basado en el tipo de flujo que necesite enviar. Cuando un nodo necesita enviar datos, solicita un RREQ, definiendo sus requerimientos de ancho de banda. Los nodos intermedios comparan su ancho de banda disponible con el ancho de banda requerido, y si no cumplen con los requerimientos, son

deshabilitados de tal forma que el protocolo busque otra ruta. La asignación de tareas según el tipo de nodo se muestra en la figura 12.

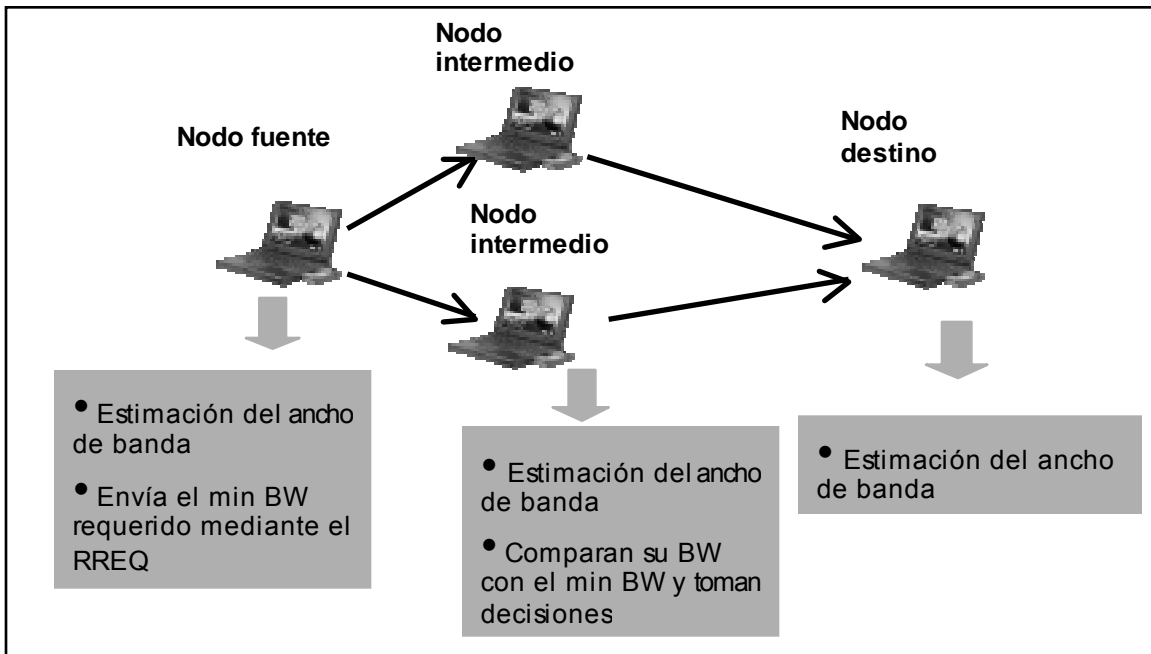


Figura 12. Asignación de tareas según el nodo

Como el protocolo AODV sólo actúa cuando no existe una ruta al destino y el RREP se envía sólo en éste caso, se evalúa el ancho de banda del siguiente salto dentro de una ruta, si éste no cumple con los requerimientos de ancho de banda, será deshabilitado.

Para la implementación del algoritmo de selección de rutas con QoS en el archivo "aodv.cpp" que se encuentra en la carpeta "network" instalada en Qualnet®, se modificó la función "AodvGetNextHop" adicionando el algoritmo mostrado en la figura 13.

Para deshabilitar la ruta se utiliza una función incluida en "aodv.cpp" llamada "AodvDisableRoute". En vez de eliminar la ruta de una vez, ésta se deshabilita, permitiéndole que sea tenida en cuenta en un futuro enlace siempre y cuando cumpla con los requerimientos de ancho de banda.

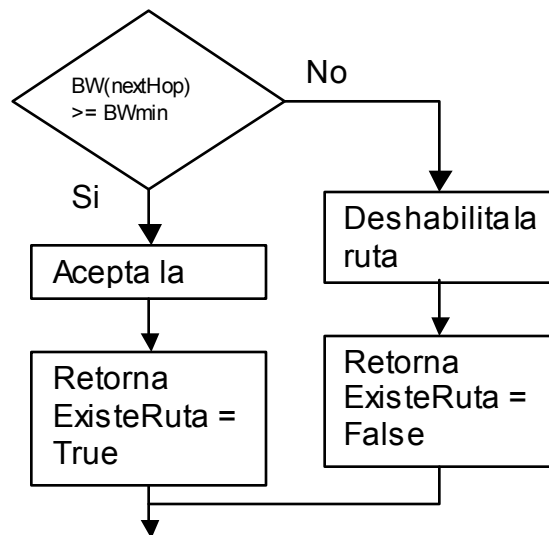


Figura 13. Diagrama de flujo del algoritmo de QoS de enrutamiento

Al igual que en el archivo “*mac\_802\_11.cpp*”, es necesario crear variables que almacenen la información del ancho de banda para cada nodo, de tal forma que pueda ser extraída e identificada mediante el número de identificación de cada nodo. Este número de identificación puede estar dado por el definido en la estructura de cada nodo o por cualquier otra referencia según el caso. Cuando se recibe la información de ancho de banda proveniente de la *MAC*, se usa el número de identificación del nodo que solicitó su información de ancho de banda.

La información del número del nodo que llega de la *MAC* se guarda en una variable de tipo *int* llamada “*BWnode*”, la cual se utiliza como apuntador para guardar la información de ancho de banda estimada en la variable correspondiente al nodo que la solicitó. Para extraer la información del ancho de banda, guardado en las variables respectivas, a la hora de tomar decisiones de enrutamiento, se utiliza el apuntador común de Qualnet® para los nodos llamado “*node->nodeld*”, el cual está definido en la estructura “*Node*” implementada en el archivo “*addons\seqnode.h*” instalado en Qualnet®.

## 2.2 Estimación de Ancho de Banda Disponible Sobre la MAC 802.11

Con el fin de lograr cierto grado de QoS en las redes móviles Ad Hoc, se utiliza como herramienta fundamental la estimación de ancho de banda disponible. Cada nodo estima su ancho de banda disponible constantemente. Esta información es utilizada por el modelo de QoS de enrutamiento para la aceptación o rechazo de flujos que soliciten una ruta.

Para lograr la estimación del ancho de banda disponible en una red inalámbrica que utiliza el protocolo MAC 802.11 [11] puede resultar complejo debido a varios factores relacionados con la herramienta de control de acceso al medio CSMA/CA, la cual evita colisiones en un ambiente de acceso aleatorio, donde los nodos tienen que competir por el acceso al canal.

Esta aleatoriedad obliga a que la actividad de los nodos vecinos puede ser tomada en cuenta en la estimación del ancho de banda disponible de cada nodo [10] y por consiguiente, es necesario tener en cuenta la ocupación del canal de radio. El protocolo MAC 802.11 [11], que funciona dentro de la capa MAC monitorea constantemente los estados de la capa física (PHY). Según el estándar, se consideran cuatro estados posibles entre los cuales uno se considera estado libre y los tres restantes se consideran ocupados. Los estados que se consideran estados de ocupación son:

1. Estado ocupado, que ocurre cuando se está enviando o recibiendo datos.
2. Estado de monitoreo de la portadora del canal, que ocurre cuando los nodos cercanos están ocupando el canal.
3. Estado de monitoreo virtual de la portadora del canal, que ocurre cuando se envían o reciben paquetes de control (RTS o CTS).

Para la implementación de la estimación del ancho de banda disponible, es necesario monitorear el cambio de la *PHY* entre cada uno de los estados mencionados, enfocándose específicamente en las transiciones de cualquiera de los estados de ocupación al estado libre y viceversa como se muestra en la figura 14.

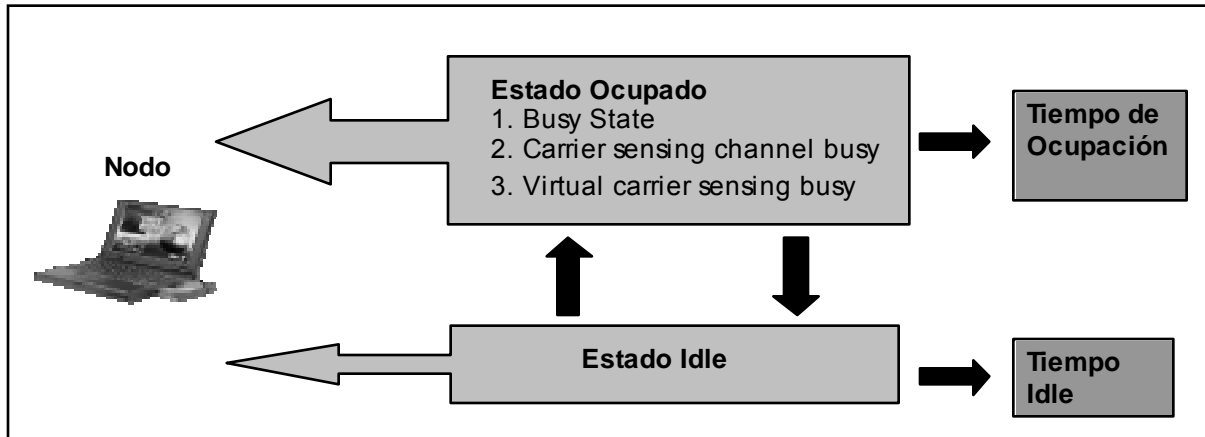


Figura 14. Transición de estados de la *PHY*

Para implementar la modificación en el archivo "*mac\_802\_11.cpp*" incluido en Qualnet®, se agregaron dos nuevas funciones: Una primera función, "*MAC\_ReportChannellnit*", en la cual se declaran e inicializan todas las nuevas variables. Como la estimación de ancho de banda se efectúa de manera autónoma para cada nodo, todas las variables relacionadas a cada nodo deben ser declaradas como arreglos de tamaño relativo al número total de nodos. El número total de nodos se extrae de la estructura del nodo declarada en el archivo *node.h*, la cual se encuentra disponible desde cualquier archivo dentro de Qualnet®.

El hecho de que Qualnet® sea un simulador discreto de eventos, dificulta la adquisición de los tiempos de estado libre u ocupado del canal, obligando a

adoptar otra medida como la de tener en cuenta los tiempos de transición de un estado a otro para un posterior cálculo de los tiempos totales en cada estado.

La segunda función, “*MAC\_ReportChannelTimeAndBW*”, que contiene el algoritmo de estimación de ancho de banda disponible se divide en dos partes fundamentales. En una primera instancia, el algoritmo graba todos los tiempos en los cuales el canal de radio se encuentra ocupado o desocupado, considerando los tiempos de simulación como umbrales o tiempos de transición de un estado a otro. A partir de este razonamiento, se puede determinar el tiempo que cada nodo percibe el canal ocupado o desocupado, teniendo en cuenta para tal efecto, los tiempos de transición actual y anterior.

A continuación se presentan las ecuaciones utilizadas en el algoritmo de estimación del ancho de banda, para calcular los tiempos libres y de ocupación del canal de radio.

$$TiempoOcupado = t - t_0 \quad 1)$$

Donde  $t_0$  representa el tiempo de transición inicial y  $t$  el tiempo transición actual, como se muestra en la figura 15.

$$TiempoTotalOcupado+ = TiempoOcupado \quad 2)$$

De igual manera para los tiempos en estado libre se tiene:

$$TiempoLibre = t_1 - t \quad 3)$$

En este caso,  $t$  se considera el tiempo de transición inicial y  $t_1$  es el tiempo de transición actual, como se muestra en la figura 15.

$$TiempoTotalLibre+ = TiempoLibre \quad 4)$$

El periodo  $T$  es la suma de los tiempos totales de ambos estados.

El código donde se implementan estas ecuaciones se presenta en el Anexo 1.



Estos tiempos de transición se toman cada vez que hay un cambio de estado de la PHY, como se muestra en la figura 14, los cambios de estado son monitoreados y registrados por una función específica del estándar IEEE 802.11 incluido, llamada “*Mac802\_11ReceivePhyStatusChangeNotification*”. Hay que tener en cuenta que en condiciones iniciales solo se obtiene el primer tiempo de transición  $t_0$ , luego no se puede determinar ningún tiempo de estado. Es claro que se deben tener dos tiempos de transición para obtener un tiempo de estado como se aprecia en la figura 15.

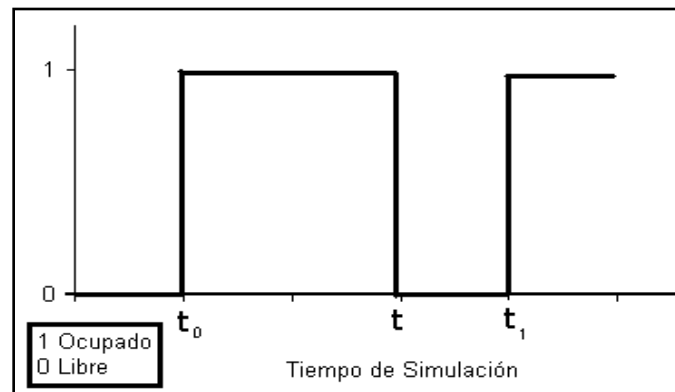


Figura 15. Extracción de tiempos de estado

De esta manera es posible conocer los tiempos que dura cada nodo, ya sea en estado libre o en estado ocupado. Estos tiempos se utilizan para obtener la razón de utilización del canal de cada nodo en la segunda parte del algoritmo. Una vez se ha obtenido la información suficiente y se han cumplido las condiciones de control del flujo del algoritmo, se procede a la segunda instancia por medio de la cual se obtiene el valor estimado del ancho de banda del nodo.

Para esta parte, al igual que la primera instancia del algoritmo, se considera una condición inicial en la cual se obtiene la razón de utilización del canal inicial ( $R_{t-1}$ ), en este momento no se realiza ningún otro cálculo. La razón de utilización se calcula mediante la siguiente ecuación:

$$R = \frac{\text{Tiempo- Estado _ Ocupado}}{T} \quad 5)$$

Para suavizar las lecturas obtenidas de la utilización del canal, las mediciones actuales se relacionan con mediciones anteriores mediante una constante de suavización  $\alpha$  que puede tomar valores entre 0 y 1 [1]. Esta relación se efectúa mediante la siguiente ecuación:

$$R_t = \alpha \cdot R_{t-1} + (1 - \alpha) \cdot R \quad 6)$$

La constante de suavización  $\alpha$  no solo indica el peso que se le quiere dar a las mediciones actuales o anteriores, además, limita el valor de la razón de utilización a valores entre 0 y 1.

Al variar  $\alpha$ , se logró establecer que dándole un peso nulo a las mediciones anteriores, se obtienen peores resultados, en términos de la razón de envío de paquetes y el retardo punto a punto, en comparación a los resultados obtenidos para valores de  $\alpha$  diferentes de 0. Los valores intermedios no generan diferencias entre sí, aunque si difieren con los resultados obtenidos para un  $\alpha$  de 1, pues lo superan en cuanto a la razón de envío de paquetes pero no en lo referente al retardo punto a punto. Se escogió un valor de  $\alpha$  de 0.4, dándole prevalencia a la razón de envío de paquetes sobre el retardo punto a punto. La razón de utilización del canal suavizada  $R_t$  es utilizada para calcular el ancho de banda disponible así:

$$BW_t = W \cdot (1 - R_t) \quad 7)$$

Donde  $W$  representa el ancho de banda nominal que para el caso del estándar IEEE 802.11 es de 2Mbps.

El cálculo del ancho de banda disponible se realiza cada vez que hay un cambio de estado en la *PHY* y se ha cumplido el tiempo máximo de lectura de cambios de estado. Este tiempo de muestreo depende del tráfico presente en la red, el tiempo entre paquetes de una sesión de tráfico determina la frecuencia de estimación.

El código donde se implementan estas ecuaciones se presenta en el Anexo 1.

El modelo implementado utiliza el método *Cross Layer* [37], para sacar provecho de las capacidades de dos capas de la pila de protocolos de cada nodo, logrando una interacción entre ambas. En este caso particular, la *MAC* envía la información del ancho de banda calculada en el protocolo seleccionado (IEEE 802.11b) a la capa de red donde se encuentra el protocolo de enrutamiento, el cual hace uso de la misma para efectuar un control de admisión de los flujos que solicitan una ruta. Para lograr el envío de información desde el estándar IEEE802.11 de la *MAC* al protocolo AODV, se utilizó el mensaje API llamado "MESSAGE\_Send", esta es una función incluida en el archivo "message.h", ubicado en la carpeta "addons\seq" instalada en Qualnet®. Básicamente lo que hace es colocar un evento en la pila de eventos del simulador, para que sea ejecutado, simulando el envío de un mensaje a su destino.

### 3. SIMULACIONES

Para realizar las pruebas de desempeño del modelo implementado, se utilizó la herramienta de simulación de redes Qualnet®. Todos los resultados son comparados con el desempeño del protocolo AODV sin QoS, tomando como parámetro de eficiencia el número total de paquetes recibidos, la razón de envío de paquetes y el retardo punto a punto para cada escenario.

La razón de envío de paquetes equivale a la relación entre la cantidad de paquetes recibidos correctamente sobre el total de paquetes enviados. Este parámetro permite medir en porcentajes, la eficiencia de una red en cuanto a la pérdida de paquetes.

El retardo punto a punto promedio es el tiempo que toman todos los paquetes en llegar al nodo de destino, desde el nodo fuente, incluyendo los tiempos en cola, los tiempos de búsqueda y transmisión.

Para todos los escenarios se utilizó el estándar MAC802.11b en CSMA/CA con DCF y PHY802.11, con una tasa de datos (*data rate*) de 2Mbps a una frecuencia de transmisión de 2.4GHz. El alcance de transmisión de cada nodo es de 339m, con una sensibilidad de recepción de -89 dBm para 2Mbps. El modelo de pérdida por el camino (*pathloss*) utilizado es *TWO-RAY*. La antena está dispuesta a una altura de 1.5m. Cada escenario usa ciertos valores por predeterminados de Qualnet®. Los parámetros particulares de configuración de cada escenario se describen junto con la presentación de los mismos.

### 3.1 Prueba de la Estimación de Ancho de Banda Disponible

El escenario diseñado tiene un área de  $500 \times 500 \text{m}^2$ , con 4 nodos dispuestos como se muestra en la figura 16. Un tráfico CBR que envía 100 paquetes de 512 bytes cada segundo, el tráfico empieza a los 6s. El tiempo total de simulación es de 30s.

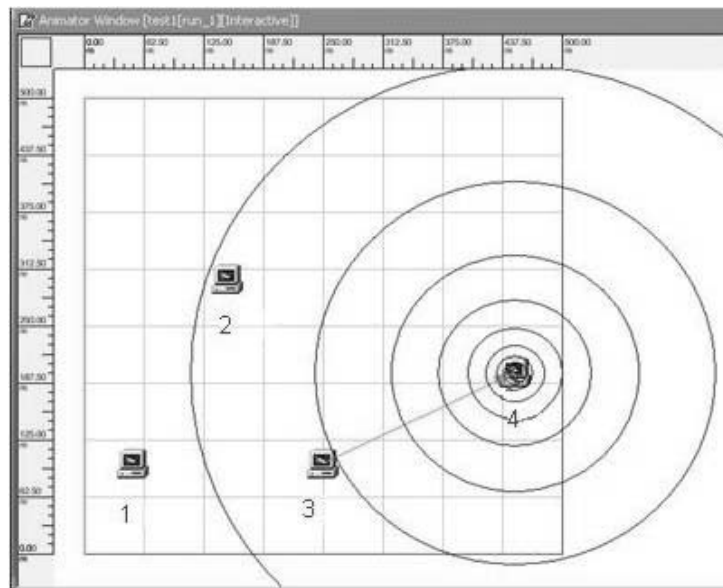


Figura 16. Escenario de simulación para la estimación de ancho de banda.

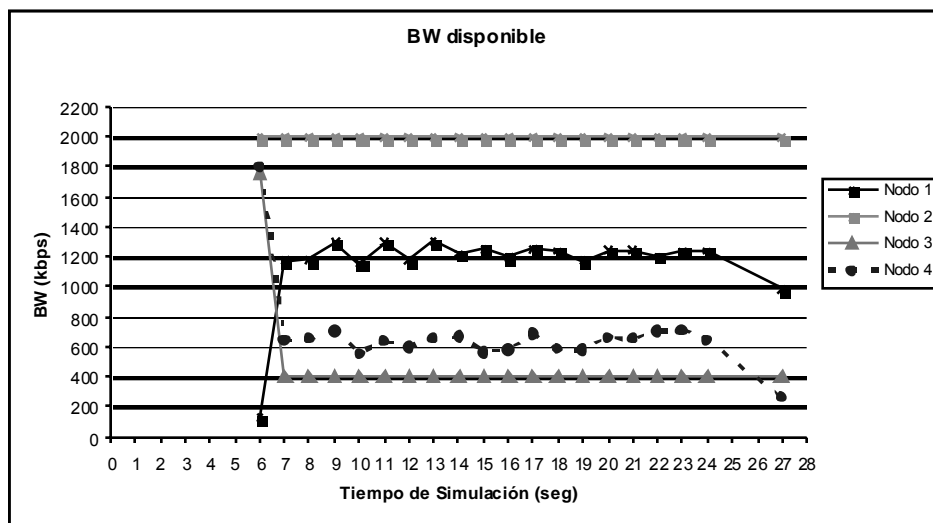


Figura 17. Estimación del ancho de banda disponible en cada nodo.

Cada serie en la figura 17 representa el ancho de banda disponible de cada nodo durante 30 segundos de simulación. El tráfico empieza a los 6 segundos y toma la ruta del nodo 3. El nodo 2 no interviene en la comunicación, por lo que tiene completa disposición para enrutar información, en cuanto a ancho de banda se refiere. El nodo 3, por ser el nodo intermedio, presenta un menor ancho de banda disponible, lo cual es razonable ya que su estimación de ancho de banda disponible tiene en cuenta las actividades que realizan sus nodos vecinos, que en este caso, son los nodos fuente y destino, por lo tanto percibe una mayor ocupación del canal.

### 3.2 Prueba de Desempeño del Modelo de QoS de Enrutamiento

Para evaluar el desempeño del modelo implementado, se observó el efecto causado por la variación de la constante de suavización  $\alpha$  y el ancho de banda mínimo requerido, en términos del número total de paquetes recibidos y el retardo punto a punto de los paquetes.

Después de seleccionar los valores adecuados para los parámetros del modelo, se procedió a evaluar su desempeño comparándolo con el protocolo original AODV.

#### 3.2.1 Escenarios de Simulación

Se diseñaron 3 escenarios con 50, 80 y 100 nodos, con los siguientes parámetros:

- Nodos uniformemente distribuidos.
- Un área de simulación de  $1500 \times 1500 \text{m}^2$ .
- El tiempo de simulación es de 300s.

- Se simulan entre 6 y 10 sesiones de tráfico CBR. La configuración de los archivos de aplicación se muestran para cada escenario.
- Cada sesión envía un total de 2344 paquetes de 512 bytes cada 25.6 ms.
- Cada sesión tiene una duración de 60s.
- El modelo de movilidad utilizado es *Random Waypoint*, con velocidades entre 1m/s y 10m/s, con pausas cada 2s.

Los archivos de configuración de cada escenario se presentan en el Anexo 2.

### 3.2.2 Selección de Parámetros del Modelo

Se realizaron varias simulaciones para determinar el valor adecuado del parámetro  $\alpha$ , que determina la dependencia de la medición de la razón de utilización del canal actual con la medición anterior. La prueba se realizó para valores de  $\alpha$  de 0, 0.2, 0.4, 0.6, 0.7, 0.9 y 1, sobre el escenario de 50 nodos con 6 sesiones de tráfico. Los resultados se muestran en las figuras 18 y 19.

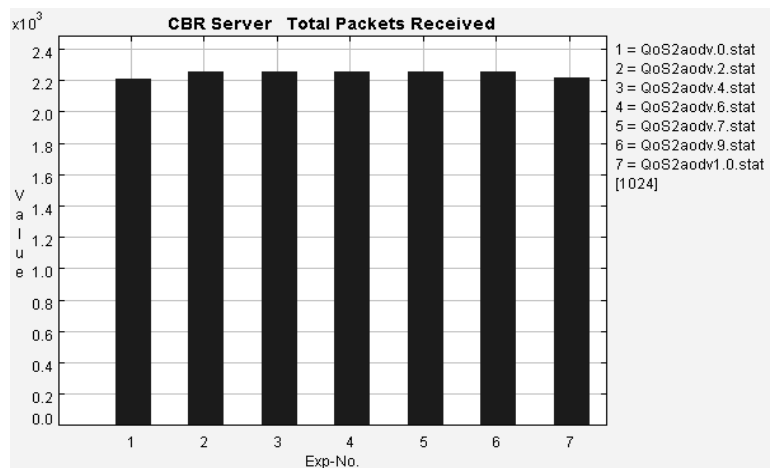


Figura 18. Número total de paquetes recibidos para diferentes valores de  $\alpha$

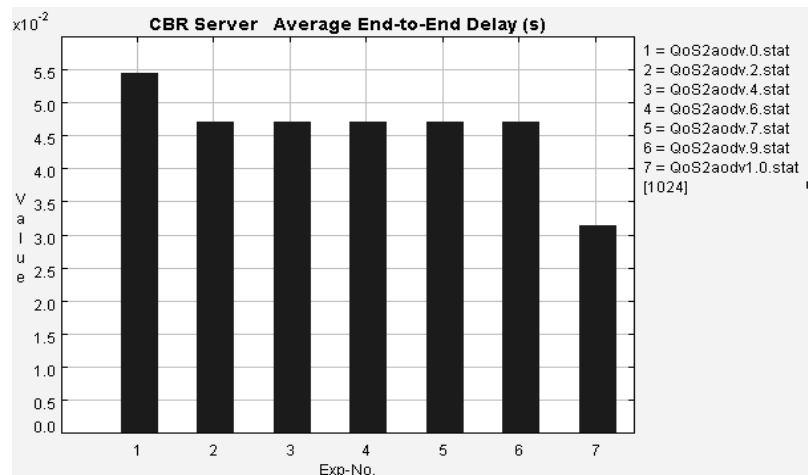


Figura 19. Retardo punto a punto promedio para diferentes valores de  $\alpha$

Los resultados de las figura 18 y 19 muestran que cuando se utiliza la razón de utilización del canal para la estimación del ancho de banda disponible, sin tener en cuenta valores anteriores de éste parámetro, aumenta el retardo punto a punto promedio y disminuye el número total de paquetes recibidos con respecto a los demás valores de  $\alpha$ , razón por la cual se descartó este valor. Para valores intermedios de  $\alpha$ , mejora los resultados obtenidos con  $\alpha$  igual a 0, tanto en el número total de paquetes recibidos como en el retardo punto a punto promedio. Cuando  $\alpha$  vale 1, solo se tiene en cuenta la razón de utilización del canal anterior y no la actual, y comparando los resultados con las demás, mejora el desempeño del modelo en cuanto al retardo punto a punto pero no supera el desempeño para valores intermedios de  $\alpha$ , en cuanto al número total de paquetes recibidos. Por lo anterior, se escogió un valor de  $\alpha$  de 0.4.

Luego se analizó el efecto causado por el valor del ancho de banda mínimo seleccionado, usando el mismo escenario descrito. Los valores de ancho de banda mínimos evaluados fueron 600Kbps, 800Kbps y 900Kbps, todos comparados con el desempeño del protocolo AODV sin QoS, como se muestra en las figuras 20 y 21. Los valores del ancho de banda mínimo se escogieron de acuerdo a los resultados obtenidos en las simulaciones, donde valores menores a los mostrados no tenían ningún efecto en el desempeño de la red. El valor del



ancho de banda mínimo debe tener en cuenta el volumen de tráfico de la red, la cantidad de nodos por área de la red, el número de sesiones de tráfico en la red y el tipo de tráfico en sí.

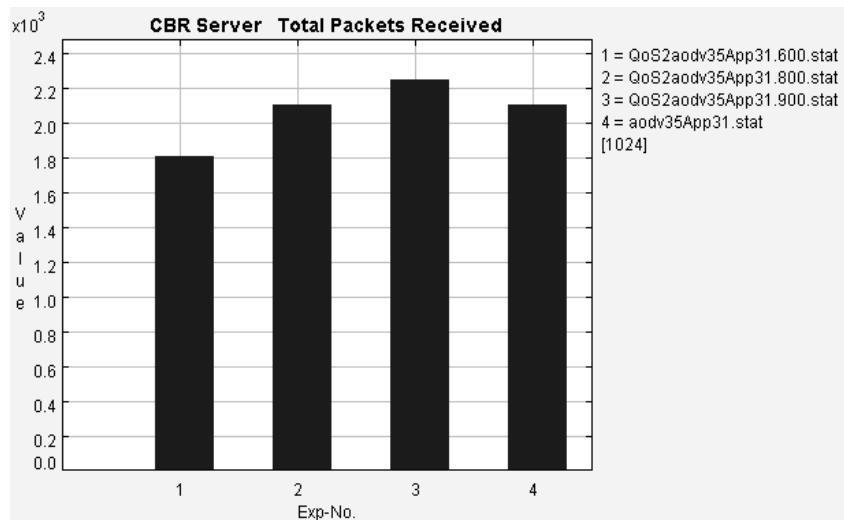


Figura 20. Número total de paquetes recibidos según el ancho de banda mínimo

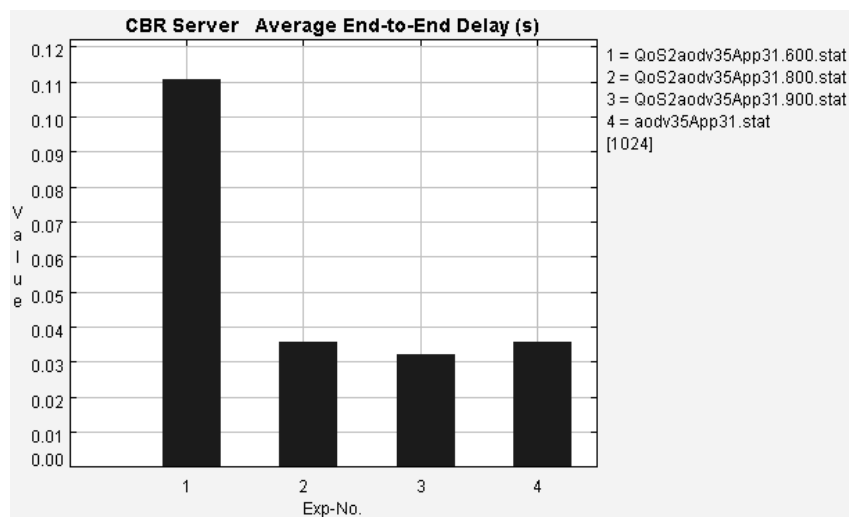


Figura 21. Retardo punto a punto promedio según el ancho de banda mínimo

Los experimentos 1 al 3 de las figuras 20 y 21 representan el desempeño de la red usando el modelo de AODV con QoS, mientras que el experimento 4 representa el protocolo AODV sin QoS.

### 3.2.3 Resultados Para Redes Estáticas

Para el escenario de 50 nodos, se logró una mejora relativa del 6.4% para 6 sesiones de tráfico, sobre la razón de envío de paquetes como se ve en la figura 22 y una disminución del retardo punto a punto promedio del 11.1% con respecto al protocolo original, como se ve en la figura 23. Para 9 y 10 sesiones se obtiene una mejora del 2.9% y 1.9% sobre la razón de envío de paquetes respectivamente y una disminución del 28.2% y 23.4% del retardo punto a punto promedio. El ancho de banda mínimo utilizado en este escenario fue de 800Kbps.

En la tabla 2 se muestran la configuración para 6 sesiones de tráfico para el escenario de 50 nodos, identificando los nodos fuente y destino. También se indica cuando empieza y cuando termina cada sesión.

En las figuras 22 y 23, *QoS2aodv3* representa los resultados obtenidos para la simulación del escenario de 50 nodos, usando el protocolo AODV con QoS, mientras que *aodv3* representa los resultados obtenidos con el protocolo AODV original.

Número de Sesión	Nodo fuente	Nodo destino	Inicia (s)	Termina (s)
1	1	7	2	62
2	49	41	50	110
3	16	3	100	160
4	39	45	190	250
5	9	34	210	27
6	47	15	235	295

Tabla 2. Configuración de 6 sesiones de tráfico para 50 nodos.

De igual manera, para 7 sesiones se muestra la configuración en la tabla 3.

Número de Sesión	Nodo fuente	Nodo destino	Inicia (s)	Termina (s)
1	1	7	2	62
2	49	41	50	110
3	27	32	70	130
4	16	3	100	160
5	39	45	190	250
6	9	34	210	270
7	47	15	235	295

Tabla 3. Configuración de 7 sesiones de tráfico para 50 nodos.

Para 8 sesiones se tiene la configuración de la tabla 4.

Número de Sesión	Nodo fuente	Nodo destino	Inicia (s)	Termina (s)
1	1	7	2	62
2	49	41	50	110
3	27	32	70	130
4	16	3	100	160
5	39	45	190	250
6	8	21	200	260
7	9	34	210	270
8	47	15	235	295

Tabla 4. Configuración de 8 sesiones de tráfico para 50 nodos.

Para 9 sesiones se tiene la configuración de la tabla 5.

Número de Sesión	Nodo fuente	Nodo destino	Inicia (s)	Termina (s)
1	1	7	2	62
2	20	24	20	80
3	49	41	50	110
4	27	32	70	130
5	16	3	100	160
6	50	48	100	160
7	39	45	190	250
8	9	34	210	270
9	47	15	235	295

Tabla 5. Configuración de 9 sesiones de tráfico para 50 nodos.

Número de Sesión	Nodo fuente	Nodo destino	Inicia (s)	Termina (s)
1	1	7	2	62
2	20	24	20	80
3	49	41	50	110
4	27	32	70	130
5	16	3	100	160
6	50	48	100	160
7	39	45	190	250
8	8	21	200	260
9	9	34	210	270
10	47	15	235	295

Tabla 6. Configuración de 10 sesiones de tráfico para 50 nodos.

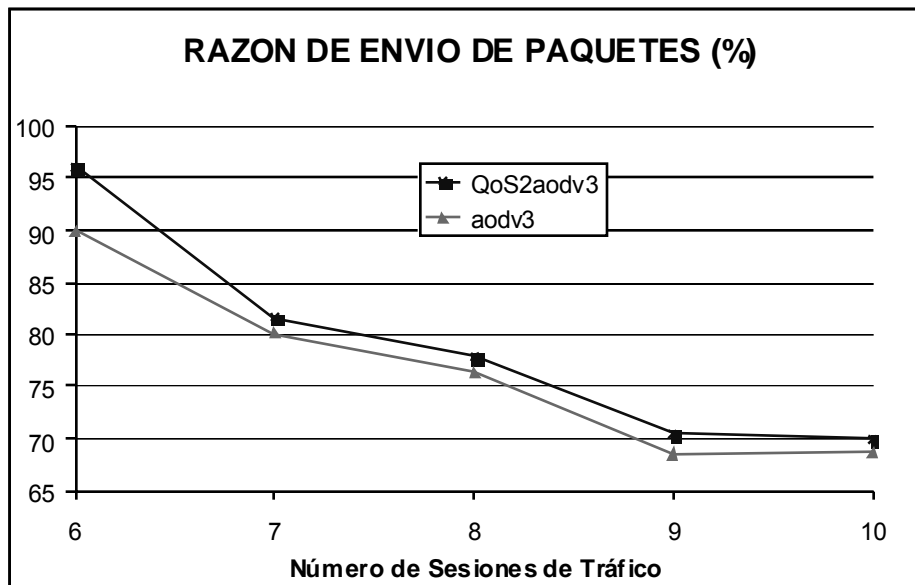


Figura 22. Razón de envío de paquetes para el escenario de 50 nodos.

El experimento se repitió 10 veces con diferentes semillas, obteniendo un 95% de confianza en que la razón de envío de paquetes tiene una desviación de más o menos 3.06% con respecto a la media.

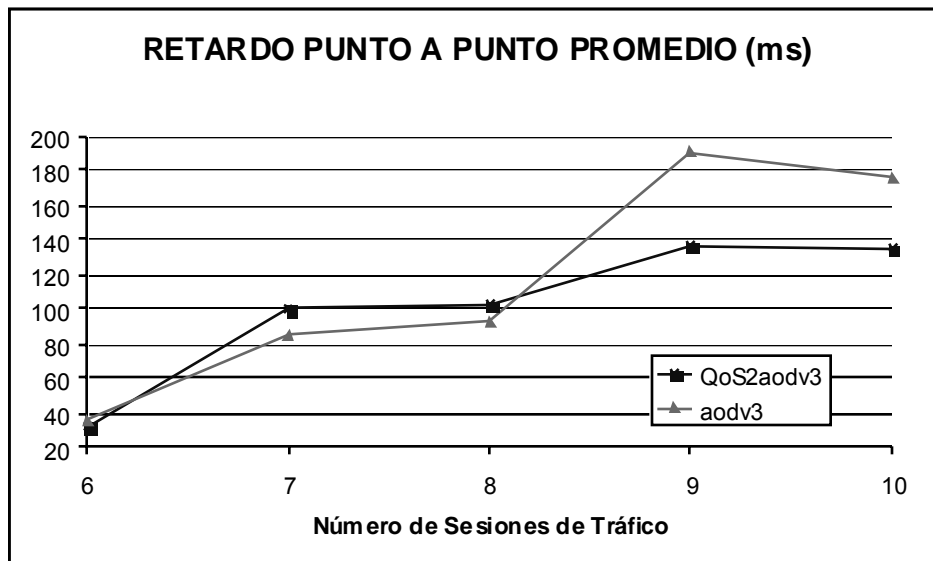


Figura 23. Retardo punto a punto promedio para el escenario de 50 nodos.

En la tabla 7 se muestran la configuración para 10 sesiones de tráfico para el escenario de 80 nodos, identificando los nodos fuente y destino. También se indica cuando empieza y cuando termina cada sesión.

Para el escenario de 80 nodos, se logró una mejora relativa del 3.3% para 6 sesiones de tráfico, sobre la razón de envío de paquetes como se ve en la figura 24 y una disminución del 24.8% sobre el retardo punto a punto promedio como se ve en la figura 25. Para 9 y 10 sesiones se obtiene una mejora del 2.8% y 7.9% sobre la razón de envío de paquetes respectivamente. En este caso, para 9 sesiones de tráfico se obtuvo un aumento del 15.7% del retardo punto a punto promedio, y para 10 sesiones se disminuye un 0.58%. En los resultados se aprecia el sacrificio en cuanto al retardo punto a punto a medida que aumenta la cantidad de nodos y el número de sesiones de tráfico. El ancho de banda mínimo utilizado en este escenario fue de 800Kbps.

Número de Sesión	Nodo fuente	Nodo destino	Inicia (s)	Termina (s)
1	1	7	2	62
2	28	64	20	80
3	49	41	50	110
4	27	32	90	150
5	16	3	100	160
6	50	48	100	160
7	39	45	190	250
8	8	2	200	260
9	9	34	210	270
10	47	15	235	295

Tabla 7. Configuración de 10 sesiones de tráfico para 80 nodos.

En las figuras 24 y 25, *QoS2aodv8* representa los resultados obtenidos para la simulación del escenario de 80 nodos, usando el protocolo AODV con QoS, mientras que *aodv3* representa los resultados obtenidos con el protocolo AODV original.

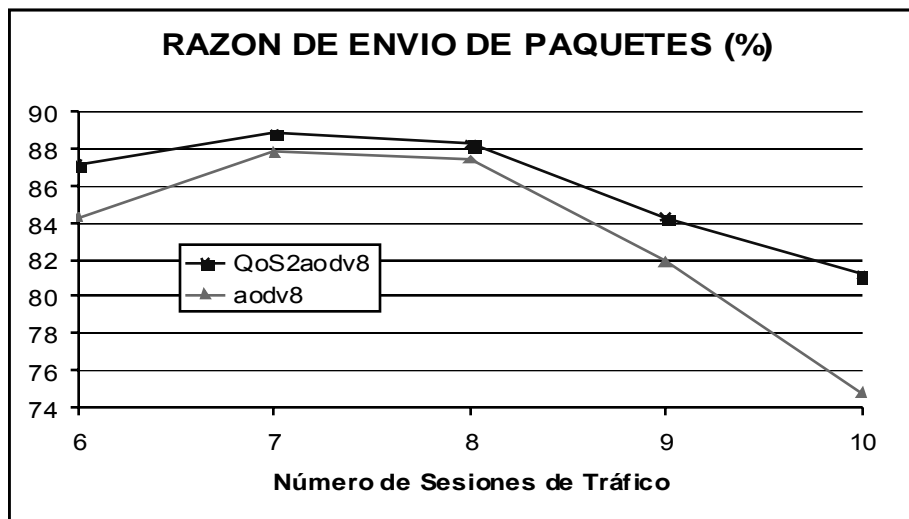


Figura 24. Razón de envío de paquetes para el escenario de 80 nodos.

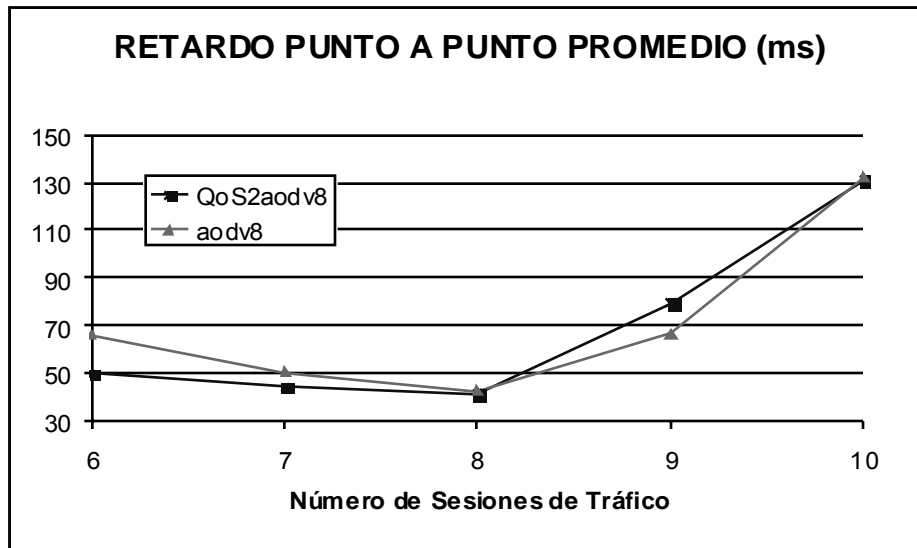


Figura 25. Retardo punto a punto promedio para el escenario de 80 nodos.

El experimento se repitió 10 veces con diferentes semillas, obteniendo un 95% de confianza en que la razón de envío de paquetes tiene una desviación de más o menos 2.8% con respecto a la media.

Para el escenario de 100 nodos, se obtuvo una diferencia negativa del 0.2% para 6 sesiones de tráfico, sobre la razón de envío de paquetes como se ve en la figura 26 y un del retardo punto a punto promedio del 30.4% con respecto al protocolo original, como se ve en la figura 27. Para 9 y 10 sesiones se obtiene una mejora del 2.1% y 10.3% sobre la razón de envío de paquetes respectivamente y una disminución del 26.3% y 48.9.4% del retardo punto a punto promedio. El ancho de banda mínimo utilizado en este escenario fue de 800Kbps.

El experimento se repitió 10 veces con diferentes semillas, obteniendo un 95% de confianza en que la razón de envío de paquetes tiene una desviación de más o menos 0.86% con respecto a la media.

Número de Sesión	Nodo fuente	Nodo destino	Inicia (s)	Termina (s)
1	60	52	3	63
2	22	84	30	90
3	50	78	70	130
4	82	94	100	160
5	71	55	120	180
6	41	2	140	200
7	66	59	190	250
8	67	79	210	270
9	44	81	220	280
10	64	95	235	295

Tabla 8. Configuración para 10 sesiones de tráfico.

En las figuras 26 y 27, *QoS2aodv4* representa los resultados obtenidos para la simulación del escenario de 100 nodos, usando el protocolo AODV con QoS, mientras que *aodv4* representa los resultados obtenidos con el protocolo AODV original.

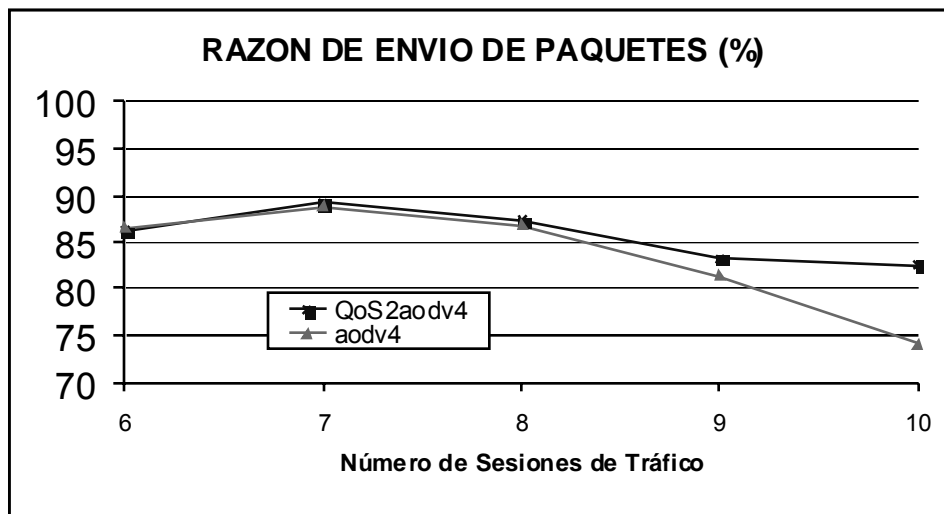


Figura 26. Razón de envío de paquetes para el escenario de 100 nodos.

En los 3 escenarios se puede apreciar el efecto que causa la cantidad de sesiones de tráfico sobre el desempeño de la red. A medida que aumenta el



número de sesiones, disminuye la razón de envío de paquetes y aumenta el retardo punto a punto promedio.

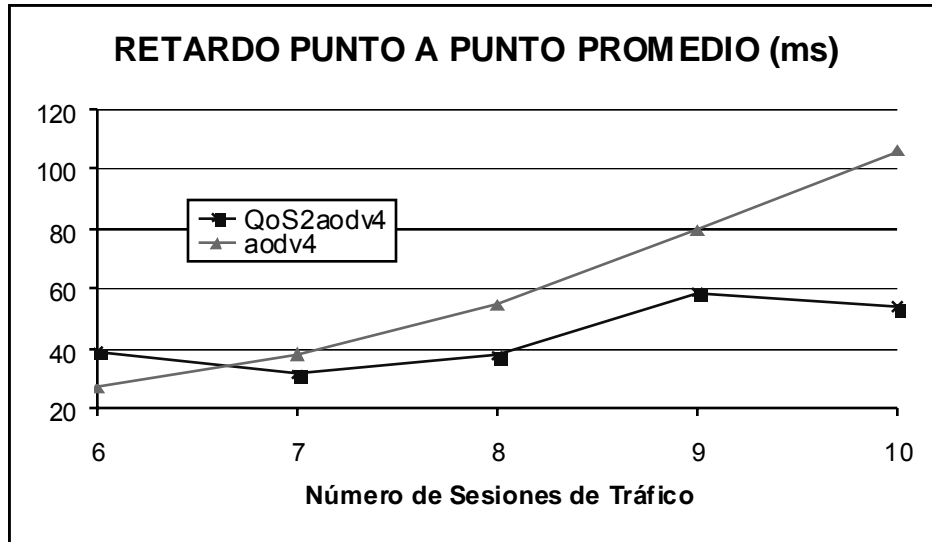


Figura 27. Retardo punto a punto promedio para el escenario de 100 nodos.

El protocolo AODV con QoS funciona mejor cuando existe un mayor tráfico por área, como se ve en las figuras 26 y 27. El funcionamiento depende de la configuración de los escenarios en los que se utiliza, por lo que es conveniente conocer bien el tipo de red que se requiere para realizar una buena elección del protocolo de enrutamiento.

En general, los resultados muestran una tendencia a mejorar el desempeño del protocolo AODV cuando se incluye la facultad de QoS, sobretodo cuando el número de sesiones de tráfico es mayor. Esta tendencia se puede ver especialmente en los escenarios de 80 y 100 nodos, que son los que presentan mayores complicaciones. El modelo implementado evita crear rutas a través de nodos cuello de botella, identificándolos mediante la estimación del ancho de banda disponible.

Para ver resultados obtenidos por investigadores en este campo, refiérase a [1] y [37].

### 3.2.4 Resultados Para Redes Móviles

El modelo de movilidad utilizado es *Random Waypoint*, la velocidad de los nodos varía entre 1m/s y 10m/s, con pausas cada 2s.

Para el escenario de 50 nodos, se logró una mejora relativa del 0.92% para 6 sesiones de tráfico, sobre la razón de envío de paquetes como se ve en la figura 28 y una disminución del retardo punto a punto promedio del 0.36% con respecto al protocolo original, como se ve en la figura 29. Para 9 y 10 sesiones se obtiene una mejora del 0.25% y una diferencia negativa de 1.9% sobre la razón de envío de paquetes respectivamente. En cuanto al retardo punto a punto promedio no hay diferencia tanto para 9 como para 10 sesiones de tráfico. El ancho de banda mínimo utilizado en este escenario fue de 800Kbps.

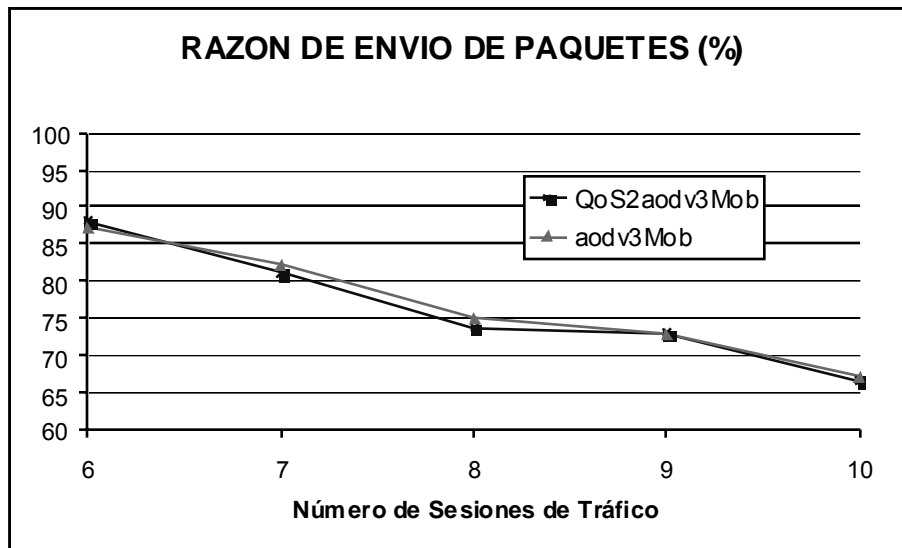


Figura 28. Razón de envío de paquetes para el escenario de 50 nodos con movilidad.

En las figuras 28 y 29, *QoS2aodv3Mob* representa los resultados obtenidos para la simulación del escenario de 50 nodos móviles, usando el protocolo AODV con QoS, mientras que *aodv3Mob* representa los resultados obtenidos con el protocolo AODV original.

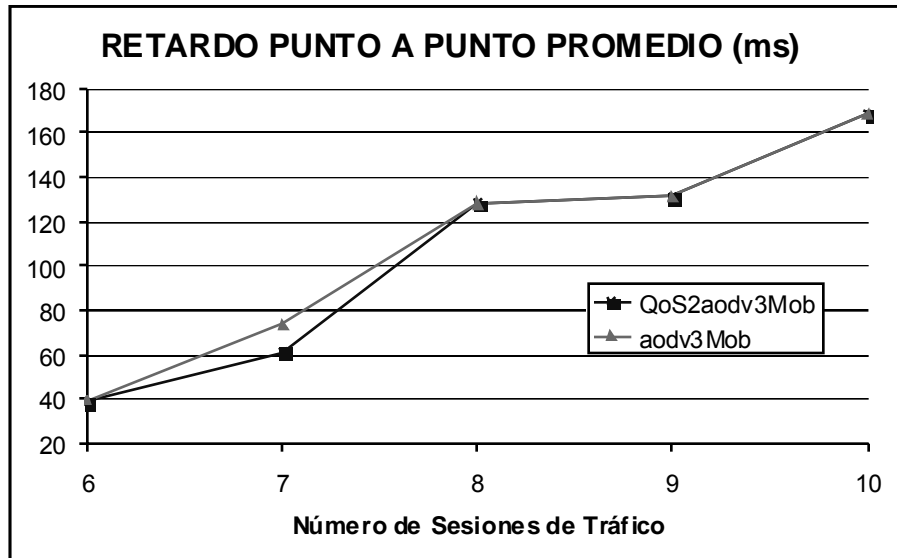


Figura 29. Retardo punto a punto promedio para el escenario 50 nodos con movilidad.

Para el escenario de 80 nodos, se logró una mejora relativa del 0.75% para 6 sesiones de tráfico, sobre la razón de envío de paquetes como se ve en la figura 30 y una disminución del retardo punto a punto promedio del 5.3% con respecto al protocolo original, como se ve en la figura 31. Para 9 y 10 sesiones se obtiene una diferencia negativa del 0.4% y una mejora de 13.2% sobre la razón de envío de paquetes respectivamente. En cuanto al retardo punto a punto promedio, se obtuvo una diferencia negativa de 43.1% para 9 sesiones y una mejora de 54.3% para 10 sesiones de tráfico. El ancho de banda mínimo utilizado en este escenario fue de 800Kbps.

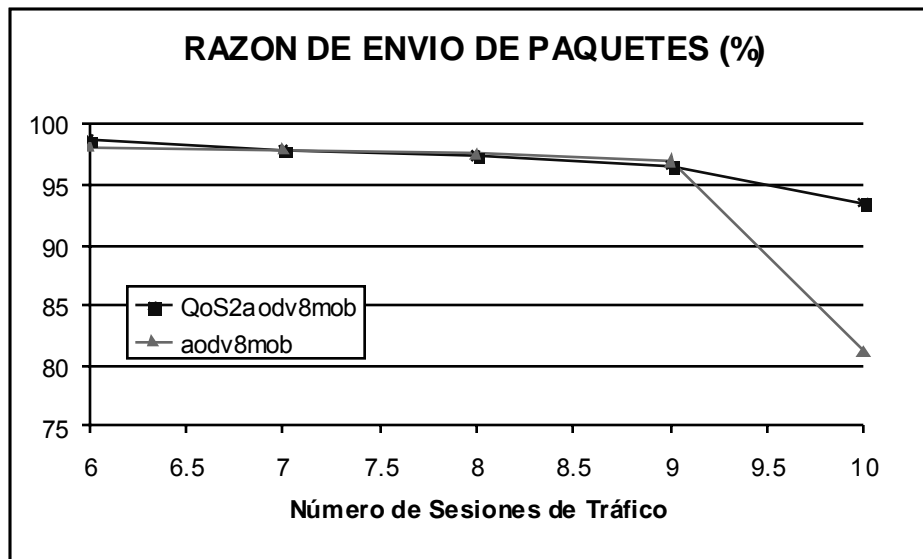


Figura 30. Razón de envío de paquetes para el escenario de 80 nodos con movilidad.

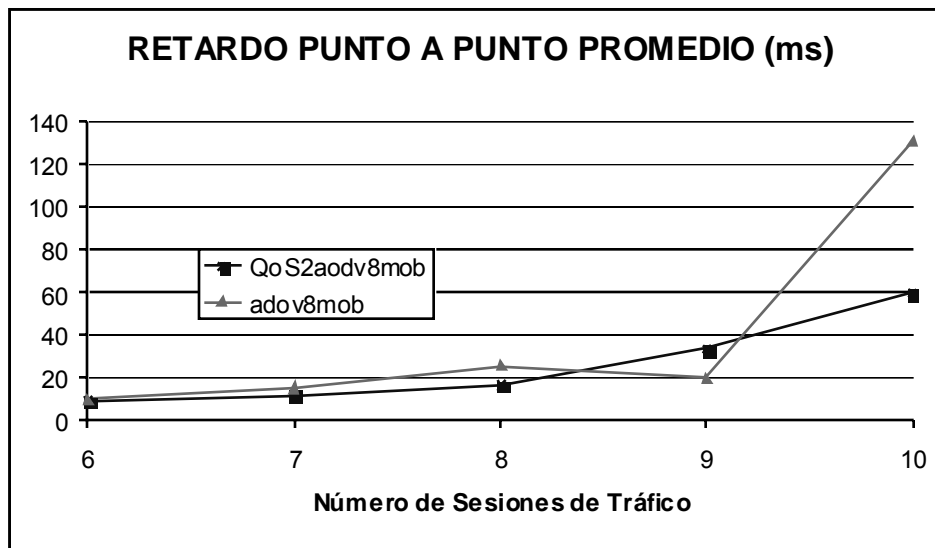


Figura 31. Retardo punto a punto promedio para el escenario de 80 nodos con movilidad.

Para el escenario de 100 nodos, se logró una mejora relativa del 2.1% para 6 sesiones de tráfico, sobre la razón de envío de paquetes como se ve en la figura 32 y un aumento del retardo punto a punto promedio del 26.3% con respecto al protocolo original, como se ve en la figura 33. Para 9 y 10 sesiones se obtiene una

mejora del 1.8% y 3% sobre la razón de envío de paquetes respectivamente. En cuanto al retardo punto a punto promedio se obtiene una diferencia negativa para 9 del 17% y una mejora del 5.8% para 10 sesiones de tráfico. El ancho de banda mínimo utilizado en este escenario fue de 800Kbps.

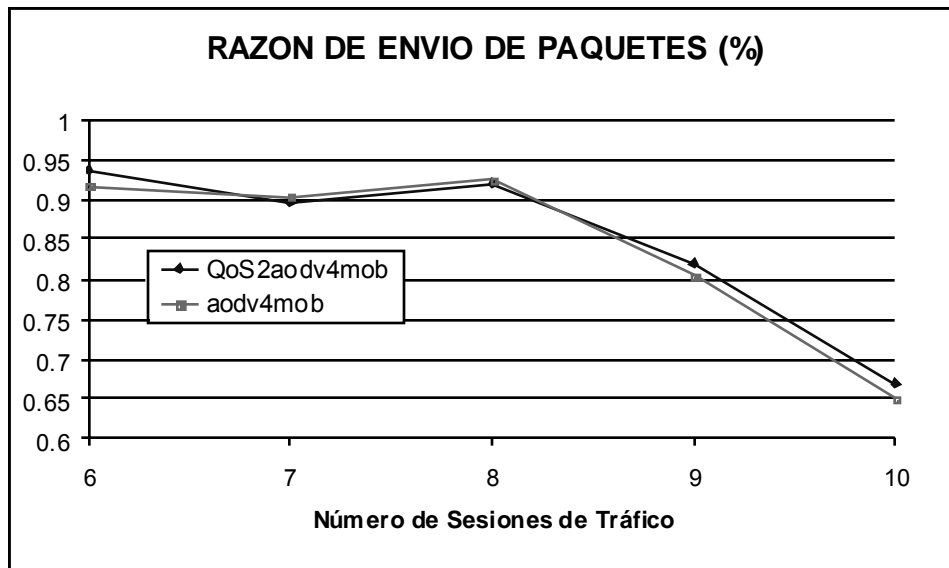


Figura 32. Razón de envío de paquetes para el escenario de 100 nodos con movilidad.

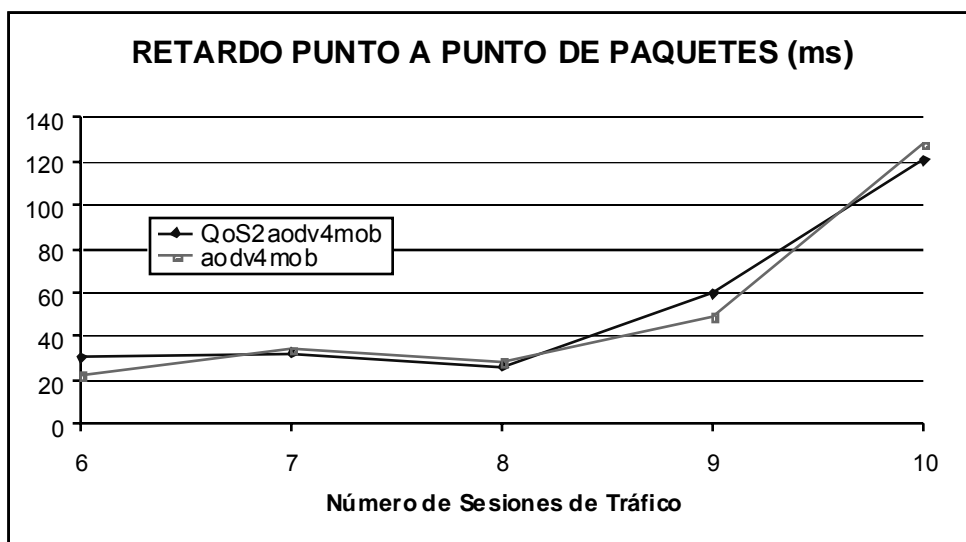


Figura 33. Retardo punto a punto promedio para el escenario de 100 nodos con movilidad.

En general, el algoritmo no mejora significativamente el desempeño del protocolo AODV sin QoS, y esto se debe a que el algoritmo puede estar utilizando valores estimados de ancho de banda disponible, que no corresponden al verdadero valor actual por el cambio constante de la topología de la red. Estos cambios en la topología de la red causan congestión en los enlaces, aumentando la pérdida de paquetes y el retardo promedio considerablemente.

El modelo sólo alivia la congestión causada por aquellos nodos que presentan un ancho de banda disponible insuficiente para garantizar un nivel de QoS, que se convierten en un cuello de botella en el enlace [1].

#### 4. CONCLUSIONES Y RECOMENDACIONES

Se logró implementar una técnica de estimación del ancho de banda disponible en cada nodo, que permite tener en cuenta la influencia de nodos vecinos mediante el cálculo de la razón de utilización del canal de radio. La selección de esta técnica permitió sobrellevar la complejidad referente a la estimación del ancho de banda sobre una red que utiliza el protocolo MAC 802.11, cuyo control de acceso al medio CSMA/CA es de carácter aleatorio y por consiguiente, la ocupación del canal se ve afectada por los nodos vecinos.

Se logró implementar un modelo de QoS de enrutamiento basado en la técnica Cross Layer, usando como herramienta de eficiencia, la estimación del ancho de banda disponible en cada nodo. El algoritmo implementado sobre el protocolo de enrutamiento AODV, incluido en la herramienta de simulación Qualnet®, mejora los resultados que presenta el protocolo original en escenarios sin movilidad, logrando disminuir el retardo punto a punto de paquetes y al mismo tiempo disminuye la pérdida de paquetes. Bajo efectos de movilidad, el modelo no supera los resultados obtenidos por el protocolo AODV sin QoS, ya que el uso de la estimación de ancho de banda disponible solo previene la congestión debida a nodos cuello de botella en las rutas.

Esta investigación comprende una parte importante de la arquitectura de QoS como lo es el QoS de enrutamiento y la estimación de ancho de banda disponible. Futuras investigaciones deberán estar encaminadas a complementar este trabajo mediante la implementación de modelos que involucren un control de admisión y un control de congestión, con el fin de mejorar el desempeño bajo un ambiente de movilidad.

## 5. REFERENCIAS BIBLIOGRAFICAS

- [1] K. Xu, K. Tang, R. Bagrodia, M. Gerla, M. Bereschinsky, (2002). *Adaptive Bandwidth Management and QoS Provisioning in Large Scale Ad Hoc Networks*. Scalable Network Technologies. Recuperado de [www.cs.ucla.edu/NRL/wireless/uploads/kxu-milcom03.pdf](http://www.cs.ucla.edu/NRL/wireless/uploads/kxu-milcom03.pdf)
- [2] Z. Demetrios, (2000). *A Glance at Quality of Services in Mobile Ad-Hoc Networks*. Recuperado de [www.eurecom.fr/~nikaeinn/qos.pdf](http://www.eurecom.fr/~nikaeinn/qos.pdf)
- [3] *QualNet User's Manual*. Recuperado del Sitio Web de Scalable Network Technologies: [www.scalable-networks.com](http://www.scalable-networks.com)
- [4] *QualNet Programmer's guide*. Recuperado del Sitio Web de Scalable Network Technologies: [www.scalable-networks.com](http://www.scalable-networks.com)
- [5] *Background of Ad Hoc Networks*. Recuperado del Sitio Web de National Institute of Standards and Technology: [http://w3.antd.nist.gov/wahn\\_mahn.shtml](http://w3.antd.nist.gov/wahn_mahn.shtml)
- [6] F. Yee Loo, (2004, Jan). *Ad Hoc Network: Prospects and Challenges*. Aoyama Morikawa Laboratory. Recuperado de [www.mlab.t.u-tokyo.ac.jp/~yfoo/Research/MANET\\_Rinkou.pdf](http://www.mlab.t.u-tokyo.ac.jp/~yfoo/Research/MANET_Rinkou.pdf)
- [7] F. Baker, (2002, March). *An outsider's view of MANET*. Internet Engineering Task Force document. Recuperado de <http://www.grc.upv.es/manets/local/brefs/An%20outsider's%20view%20of%20MANET%20-%20draft-baker-manet-review-01.txt>



[8] D. De Couto, D. Aguayo, B. Chambers, R. Morris, (2002). *Performance of Multihop Wireless Networks: Shortest Path is Not Enough*. M.I.T. Laboratory for Computer Science. Recuperado de <http://pdos.csail.mit.edu/papers/grid/hotnets02/paper.pdf>

[9] *The Network Simulator NS-2*. Recuperado del Sitio Web Information Sciences Institute: <http://www.isi.edu/nsnam/ns>.

[10] Francisco López. *El Estándar IEEE 802.11 Wireless LAN*. Universidad Técnica Federico Santa María. Recuperado de <http://greco.dit.upm.es/~david/TAR/trabajos2002/08-802.11-Francisco-Lopez-Ortiz-res.pdf>

[11] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ANSI/IEEE Std 802.11*. Recuperado del Sitio Web de la IEEE: <http://standards.ieee.org/getieee802/download/802.11b-1999.pdf>

[12] Xiaoyan Hong, Kaixin Xu, and Mario Gerla, (2002). *Scalable Routing Protocols for Mobile Ad Hoc Networks*. University of California, Los Angeles, Recuperado de <http://www.cs.ucla.edu/NRL/wireless/uploads/01020231-from-ntmgz-hxy.pdf>

[13] Charles E. Perkins, (2002). *Ad hoc On Demand Distance Vector (AODV)*. Mobile Ad Hoc Networking Working Group, INTERNET DRAFT, Recuperado de <http://www.ietf.org/rfc/rfc3561.txt>

[14] *A Quick Guide to AODV Routing*, Recuperado del Sitio Web de National Institute of Standards and Technology: [http://w3.antd.nist.gov/wctg/aodv\\_kernel/aodv\\_guide.pdf](http://w3.antd.nist.gov/wctg/aodv_kernel/aodv_guide.pdf)

[15] Y.Lee, J. Chen, X. Hong, K. Xu, T. Breyer, M. Gerla, (2003). *Experimental Evaluation of LANMAR, a Scalable Ad-Hoc Routing Protocol*. Recuperado de <http://netlab.cs.ucla.edu/wiki/files/yeng2005wcnc.pdf>

[16] C. E. Perkins, P. Bhagwat, (1993). *Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers*. Computer Science Department, University of Maryland,. Recuperado de <http://www.cs.virginia.edu/~cl7v/cs851-papers/dsdv-sigcomm94.pdf>

[17] D. B. Jonson, D. A. Maltz, Y. Hu, (2004, Julio). *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)*. IETF MANET Working Group. Recuperado de <http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt>

[18] Guangyu Pei, Mario Gerla, Tsu-Wei Chen, (1999). *Fisheye State Routing: A Routing Scheme for Ad Hoc Wireless Networks*. Computer Science Department, University of California, Bell Laboratories, Los Angeles, CA, Recuperado de <http://systems.cs.colorado.edu/~grunwald/MobileComputing/Papers/fisheye-state-routing-in-mobile-ad-hoc-networks.pdf>

[19] P. Jacques, P. Muhlethaler, A. Qayyum, (2001). *Optimized Link State Routing Protocol (OLSR)*. IETF MANET Working Group. Recuperado de <http://hipercom.inria.fr/olsr/draft-ietf-manet-olsr-04.txt>

[20] V. Park, S. Corson, (1998, Agosto). *Temporally-Ordered Routing Algorithm (TORA)*. INTERNET DRAFT. Recuperado de <http://www3.ietf.org/proceedings/98dec/I-D/draft-ietf-manet-tora-spec-01.txt>

[21] X. Hong, N. Nguyen, S. Liu, Y. Teng, (2002). *Dynamic Group Support in LANMAR Routing Ad Hoc Networks*. Computer Science Department, University of California at Los Angeles,. Recuperado de

<http://www.cs.ucla.edu/NRL/wireless/uploads/mwcn2002-hxy.pdf>

[22] Chai-Keon Tho, (1995). *Associativity-Based Routing For Ad Hoc Mobile Networks*. Computer Laboratory, University of Cambridge. Recuperado de [http://citeseer.ist.psu.edu/cache/papers/cs/24964/http:zSzzSzusers.ece.gatech.edu/uzSz~cktohzSzso\\_new.pdf/associativity-based-routing-for.pdf](http://citeseer.ist.psu.edu/cache/papers/cs/24964/http:zSzzSzusers.ece.gatech.edu/uzSz~cktohzSzso_new.pdf/associativity-based-routing-for.pdf)

[23] Z. J. Haas, J. Deng, B. Liang, P. Papadimitratos, S. Sajama, (2002). *Wireless Ad Hoc Networks*. Cornell University, School of Electrical and Computer Engineering, Ithaca, NY. Recuperado de <http://wnl.ece.cornell.edu/Publications/ency01.pdf>

[24] R. Braden, D. Clark, S. Shenker, (1994, Junio). *Integrated Services in the Internet Architecture: an Overview*. Network Working Group. Recuperado de <http://citeseer.ist.psu.edu/cache/papers/cs/5970/ftp:zSzzSzftp.pasteur.frzSzpubzSzcomputingzSzrfczSz16xxzSz1633.pdf/braden94integrated.pdf>

[25] Anita Kasari, (2000, October). *An Architecture for Differentiated Services*. Department of Computer Science, University of Helsinki. Recuperado de <http://www.cs.helsinki.fi/u/kraatika/Courses/QoS00a/kasari.pdf>

[26] G. Ahn, A. Campbell, A. Veres, L. Sun, (2002, Jun). *SWAN: Service differentiation in stateless wireless ad hoc networks*. Proceedings of IEEE INFOCOM 2002. Recuperado de <http://citeseer.ist.psu.edu/ahn02swan.html>

[27] S. Lee, G. Ahn, X. Zhang, and A. Campbell, (2000, Apr). *INSIGNIA: An IP-based quality of service framework for mobile ad hoc networks*. Journal of Parallel and Distributed Computing (JPDC), 60(4).. Recuperado de [www.iks.inf.ethz.ch/education/ss04/seminar/42.pdf](http://www.iks.inf.ethz.ch/education/ss04/seminar/42.pdf)

[28] P. Sinha, R. Sivakumar, V. Bharghavan, (1999, Aug). *CEDAR: A core-extraction distributed ad hoc routing algorithm*. Proceedings of IEEE INFOCOM 1999. Recuperado de [www.cs.cornell.edu/People/egs/615/sivakumar99cedar.pdf](http://www.cs.cornell.edu/People/egs/615/sivakumar99cedar.pdf)

[29] H. Xiao, K. Chua, W. Seah, A. Lo, (2000). *A Flexible Quality of Service Model for Mobile Ad-Hoc Networks*. Department of Electrical Engineering, University of Singapore. Recuperado de [www1.i2r.a-star.edu.sg/~winston/papers/MTC2000Spring-FQMM.pdf](http://www1.i2r.a-star.edu.sg/~winston/papers/MTC2000Spring-FQMM.pdf)

[30] Erol Gelenbe, (2002). *QoS and Sensible Routing Decisions*. Dept. of Electrical & Electronic Engineering, Imperial College, London. Recuperado de [www.ee.imperial.ac.uk/gelenbe/index\\_files/CMS.pdf](http://www.ee.imperial.ac.uk/gelenbe/index_files/CMS.pdf)

[31] E. Gelenbe, R. Lent, Z. Xu, *Networks With Cognitive Packets*. University of Central Florida, Orlando, U.S.A. Recuperado de [http:// doi.ieeecomputersociety.org/10.1109/MASCOT.2000.876422](http://doi.ieeecomputersociety.org/10.1109/MASCOT.2000.876422)

[32] E. Gelenbe, R. Lent, A. Nunez. *Self-Aware Networks and QoS*. University of Central Florida, Orlando, U.S.A. Recuperado de [http:// doi.ieeecomputersociety.org](http://doi.ieeecomputersociety.org)

[33] E. Gelenbe, D. Gabor M. Gellman, R. Lent, P. Liu, P. Su, (2001). *Autonomous Smart Routing for Network QoS*. University of Central Florida, Orlando, U.S.A. Recuperado de [http:// doi.ieeecomputersociety.org](http://doi.ieeecomputersociety.org)

[34] E. Gelenbe, R. Lent, Z. Xu, (2001). *Design and Performance of Cognitive Packet Networks*. University of Central Florida, Orlando, U.S.A. Recuperado de [http:// doi.ieeecomputersociety.org](http://doi.ieeecomputersociety.org)

[35] E. Gelenbe, R. Lent, Z. Xu (2001). *Measurement and Performance of Cognitive Packet Networks*. University of Central Florida, Orlando, U.S.A. Recuperado de [http:// doi.ieeecomputersociety.org](http://doi.ieeecomputersociety.org)

[36] E. Gelembe, R. Lent, (2004, Abril). *Power-Aware Ad Hoc Cognitive Packet Networks*. University of Central Florida, Orlando, U.S.A. Recuperado de [http:// doi.ieeecomputersociety.org](http://doi.ieeecomputersociety.org)

[37] Lei Chen, W. Heinzelman, (2005, March). *QoS-Aware Routing Based on Bandwidth Estimation for Mobile Ad Hoc Networks*. IEEE Journal on Selected Areas in Communications, Vol. 23, No. 3. Recuperado de [www.ece.rochester.edu/research/wcng/papers/chen05\\_jsac.pdf](http://www.ece.rochester.edu/research/wcng/papers/chen05_jsac.pdf)

[38] M. Domingo, (2005). *Diferenciación de Servicios y Mejora de la Supervivencia en Redes Ad Hoc Conectadas a Redes Fijas*. Departamento de Ingeniería de Telemática, Universidad Politécnica de Cataluña. Recuperado de [www.tdx.cesca.es/TESIS\\_UPC/AVAILABLE/TDX-1019105-084529//01Mcda01de01.pdf](http://www.tdx.cesca.es/TESIS_UPC/AVAILABLE/TDX-1019105-084529//01Mcda01de01.pdf)

## Anexo 1. Código del algoritmo de estimación de ancho de banda

Este código se agrega al archivo “*mac\mac\_802\_11.cpp*” instalado en Qualnet®.

Al principio del archivo se deben declarar las variables globales:

```
double* BwT;  
double* tlast;  
double tlast1;  
double tnow;  
double* periodstr;  
double* R;  
double* Rt;  
double* R_last;  
double* TbT;  
double* TiT;  
BOOL initCond;  
BOOL initCond2;  
BOOL* initCond3;  
BOOL calcBW;  
clocktype t;  
clocktype* t_last;  
clocktype t_last1;  
clocktype currentTime;  
clocktype* period;  
clocktype MaxPeriod;  
clocktype PHY_busyTime;  
clocktype* PHY_totalBusyTime;  
clocktype PHY_idleTime;  
clocktype* PHY_totalIdleTime;  
  
char result[MAX_STRING_LENGTH];
```

```

char BWstr1[MAX_STRING_LENGTH];

//-----
//  NAME:          MAC_ReportChannelInit
//  PURPOSE:       Con esta funcion se inicializa el valor de las variables
//                  de MAC_Report_ChannelStatusTime
//
//-----

// Esta función se llama desde la función "Mac802_11Init"

void MAC_ReportChannelInit(Node* node)
{
    int tNodes = node->numNodes;
    int i;

    // Se reserva espacio en memoria para los arreglos.

    TbT = (double*)MEM_malloc(sizeof(double) * tNodes);
    TiT = (double*)MEM_malloc(sizeof(double) * tNodes);
    periodstr = (double*)MEM_malloc(sizeof(double) * tNodes);
    Rt = (double*)MEM_malloc(sizeof(double) * tNodes);
    R_last = (double*)MEM_malloc(sizeof(double) * tNodes);
    R = (double*)MEM_malloc(sizeof(double) * tNodes);
    BWt = (double*)MEM_malloc(sizeof(double) * tNodes);
    t_last = (clocktype*)MEM_malloc(sizeof(clocktype) * tNodes);
    period = (clocktype*)MEM_malloc(sizeof(clocktype) * tNodes);
    PHY_totalBusyTime =
        (clocktype*)MEM_malloc(sizeof(clocktype) * tNodes);
    PHY_totalIdleTime =
        (clocktype*)MEM_malloc(sizeof(clocktype) * tNodes);
    initCond3 = (BOOL*)MEM_malloc(sizeof(BOOL) * tNodes);

    // Se inicializan las variables

    for (i = 0; i < tNodes; i++)
    {
        R[i] = 0;
        R_last[i] = 0;
        t_last[i] = 0;
        Rt[i] = 0;
        BWt[i] = 0;
        PHY_totalBusyTime[i] = 0;
        PHY_totalIdleTime[i] = 0;
        TbT[i] = 0;
        TiT[i] = 0;
        periodstr[i] = 0;
        initCond3[i] = true;
    }

    PHY_busyTime = 0;

```

```

    PHY_idleTime = 0;
    t = 0;
    t_last1 = 0;

    initCond = true;
    initCond2 = true;
    period = 0;
    MaxPeriod = 0;
}

//-----
//  NAME:      MAC_ReportChannelStateTimeAndBW
//  PURPOSE:   Con esta función se reporta el estado del canal desde la
//             capa MAC, se toman los tiempos en cada estado y se calcula
//             el BW disponible para cada nodo.
//
//  RETURN:    Channel Utilization Ratio
//-----

// Esta función se llama desde:
// "Mac802_11ReceivePhyStatusChangeNotification"
// solo cuando la PHY cambia de estado ocupado a libre y viceversa

static
MAC_ReportChannelStateTimeAndBW(
    Node* node,          // nodo que solicita la estimación
    MacData802_11* M802) // estructura de la MAC
{
    char *stop;
    clocktype delay_msg;
    currentTime = getSimTime(node);
    long tNodes = node->numNodes; //total de nodos en la red
    int cNode = node->nodeId; // número del nodo actual
    int i;
    int interfaceIndex;

    clocktype startTime;

// TIEMPO_INICIO es una constante que se declara en "mac\mac_802_11.h"

    startTime = TIEMPO_INICIO * SECOND;

// Si no el tiempo de simulación es menor que el tiempo de inicio no haga
// nada

    if (currentTime >= startTime)
        {

// Si es primera vez que entra initCond = true
    if (initCond)
        {
            t_last1 = getSimTime(node); // captura el tiempo de simulación

```



```

// anterior

for (i = 0; i < tNodes; i++)
{
    t_last[i] = t_last1; // lo relaciona con el respectivo nodo
}

// MaxPeriod determina hasta cuando se capturan los tiempos de estado
// ocupado o libre

    MaxPeriod = t_last1 + (PERIODO * SECOND);
    TIME_PrintClockInSeconds( MaxPeriod, result);
    initCond = false;

} //if

// Ejecuta la captura de tiempos de estado ocupado o libre

    if (!initCond2)
    {
        t = getSimTime(node);
        TIME_PrintClockInSeconds( t, result);
        tnow = strtod(result,&stop);

// Si el tiempo de simulación es menor a MaxPeriod realice la captura

        if (t <= MaxPeriod)
        {
// Si la PHY cambia a estado libre calcula el tiempo de estado ocupado

            if (Mac802_11PhyStatus(node, M802) == PHY_IDLE)
            {
                PHY_busyTime = t - t_last[cNode-1];
                PHY_totalBusyTime[cNode-1] += PHY_busyTime;

                TIME_PrintClockInSeconds( PHY_totalBusyTime[cNode-1], result);
                TbT[cNode-1] = strtod(result,&stop);
            }

// Si la PHY cambia a estado libre calcula el tiempo de estado libre

else
        {
            PHY_idleTime = t - t_last[cNode-1];
            PHY_totalIdleTime[cNode-1] =
                t - (PHY_totalBusyTime[cNode-1] + t_last1);

            TIME_PrintClockInSeconds( PHY_totalIdleTime[cNode-1], result);
            TiT[cNode-1] = strtod(result,&stop);
        }

// El tiempo actual de simulación es el Nuevo t_last
        t_last[cNode-1] = t;

```

```

    }//if

// Si el tiempo de simulación es mayor o igual a MaxPeriod, calcula la
// utilización del canal y el ancho de banda

    else //if (t >= MaxPeriod)
    {
        PHY_totalIdleTime[cNode-1];

// Si es la primera vez que se calcula entra initCond3 = trae, luego solo
// se calcula R_last

        if (initCond3[cNode-1])
        {
            periodstr[cNode-1] = TbT[cNode-1] + TiT[cNode-1];
            R_last[cNode-1] = TbT[cNode-1]/periodstr[cNode-1];
            initCond3[cNode-1] = false;
        }//if initCond3
    else
    {
        delay_msg = 0;

        for (i = 0; i < tNodes; i++)
        {
            periodstr[i] = TbT[i] + TiT[i];
            R[i] = TbT[i]/periodstr[i];
            Rt[i] = SmoothConstant*(R_last[i]) +
                (1 - SmoothConstant)*R[i]; // utilización
                // del canal actual

            BwT[i] = W*(1 - Rt[i]); //ancho de banda disponible actual
            R_last[i] = Rt[i]; //se actualiza el valor de R_last
        }//for

        MAC_ReportChannelInit(node); // Se inicializan las variables

        calcBW = true; // bandera de control del programa

        initCond3[cNode-1] = false;
    } //if (t >= MaxPeriod)
    }//else
    }//if initCond2

    initCond2 = false;

    if (calcBW)
    {
// Solo se inicializan si se calculó el ancho de banda disponible
        initCond2 = true;
        calcBW = false;
    }
    }//if startTime

} // MAC_ReportChannelStateTimeAndBW

```

## Anexo 2. Archivos de configuración para los escenarios de simulación

A continuación se presenta el archivo “finalqos3.config” que contiene la configuración del escenario de 50 nodos.

```
# ***** QualNet Configuration File *****

# ***** General *****

# ***** General *****
VERSION 3.8
EXPERIMENT-NAME Qualnet
SIMULATION-TIME 300S

# The random number seed is used to initialize part of the seed of
various randomly generated numbers in the simulation. Use different seeds
to see the consistency of the results of the simulation.
SEED 1

# ***** Parallel Settings *****

# Method for assigning nodes to parallel partitions
PARTITION-SCHEME AUTO

# ***** Terrain *****

# The size of the physical terrain in which the nodes are being
simulated.

COORDINATE-SYSTEM CARTESIAN
# The size of the terrain in meters.
TERRAIN-DIMENSIONS ( 1500, 1500 )
# Terrain altitude in meters.
DUMMY-ALTITUDES ( 1500, 1500 )
# If this is set to YES, the simulation terminates when it attempts to
use an elevation not included in the terrain data files. If it is NO, the
execution simply assumes that such elevations are 0.0.
TERRAIN-DATA-BOUNDARY-CHECK YES

# ***** Node Positioning *****

# ***** Nodes *****

# The number of nodes being simulated.
DUMMY-NUMBER-OF-NODES 50
# The node placement strategy.
```

```
NODE-PLACEMENT FILE
NODE-POSITION-FILE C:\qualnet\3.8\gui\scenarios\finalqos3\finalqos3.nodes

# ***** Mobility *****

MOBILITY RANDOM-WAYPOINT
MOBILITY-WP-PAUSE 2S
# meters per second
MOBILITY-WP-MIN-SPEED 1
# meters per second
MOBILITY-WP-MAX-SPEED 10
MOBILITY-POSITION-GRANULARITY 1.0
# If yes, nodes get their altitude coordinate from the terrain file, if
one is specified.
MOBILITY-GROUND-NODE NO

# ***** Wireless Settings *****

# ***** Channel *****

PROPAGATION-CHANNEL-FREQUENCY 2400000000
PROPAGATION-MODEL STATISTICAL
# Signals with powers below PROPAGATION-LIMIT (in dBm) (before the
antenna gain at the receiver) are not delivered.
PROPAGATION-LIMIT -111.0
PROPAGATION-PATHLOSS-MODEL TWO-RAY
PROPAGATION-SHADOWING-MODEL CONSTANT
# in dB
PROPAGATION-SHADOWING-MEAN 4.0
PROPAGATION-FADING-MODEL NONE

# ***** Radio/Physical Layer *****

PHY-MODEL PHY802.11b
PHY802.11-AUTO-RATE-FALLBACK NO
# bandwidth in bps. supported data rates: 1Mbps, 2Mbps, 5.5Mbps, 11Mbps
PHY802.11-DATA-RATE 2000000
PHY802.11b-TX-POWER--1MBPS 15.0
PHY802.11b-TX-POWER--2MBPS 15.0
PHY802.11b-TX-POWER--6MBPS 15.0
PHY802.11b-TX-POWER-11MBPS 15.0
PHY802.11b-RX-SENSITIVITY--1MBPS -93.0
PHY802.11b-RX-SENSITIVITY--2MBPS -89.0
PHY802.11b-RX-SENSITIVITY--6MBPS -87.0
PHY802.11b-RX-SENSITIVITY-11MBPS -83.0
PHY802.11-ESTIMATED-DIRECTIONAL-ANTENNA-GAIN 15.0
PHY-RX-MODEL PHY802.11b
# Channels the radio is capable of listening to.
PHY-LISTENABLE-CHANNEL-MASK 1
# Channels the radio is currently listening to. Can be changed during
run time.
PHY-LISTENING-CHANNEL-MASK 1
# Temperature of the environment in K
```

```
PHY-TEMPERATURE 290.0
PHY-NOISE-FACTOR 10.0
ANTENNA-MODEL OMNIDIRECTIONAL
# antenna gain in dB
ANTENNA-GAIN 0.0
# antenna height in meters
ANTENNA-HEIGHT 1.5
# efficiency of the antenna
ANTENNA-EFFICIENCY 0.8
# antenna mismatch loss in dB
ANTENNA-MISMATCH-LOSS 0.3
# antenna cable loss in dB
ANTENNA-CABLE-LOSS 0.0
# antenna connection loss in dB
ANTENNA-CONNECTION-LOSS 0.2

# ***** MAC Protocol *****

MAC-PROTOCOL MAC802.11
MAC-802.11-DIRECTIONAL-ANTENNA-MODE NO
MAC-802.11-SHORT-PACKET-TRANSMIT-LIMIT 7
MAC-802.11-LONG-PACKET-TRANSMIT-LIMIT 4
MAC-802.11-RTS-THRESHOLD 0
MAC-802.11-PCF-STATISTICS NO
# specifies an additional delay for messages sent by the MAC layer to the
phy layer. Some MAC protocols use a multiple of this value.
MAC-PROPAGATION-DELAY 1US
# must be set to YES if nodes want to overhear packets destined to the
neighboring node.
PROMISCUOUS-MODE YES

# ***** Adaptation Protocols *****

# ***** Adaptation Layer *****

ADAPTATION-LAYER-STATISTICS NO
ATM-STATIC-ROUTE NO

# ***** ARP Specific *****

# ***** ARP Enabled *****

ARP-ENABLED NO

# ***** ARP Specs *****

ARP-TIMEOUT-INTERVAL 20M

# ***** Network Protocols *****

# ***** Network Protocol *****
```

```
NETWORK-PROTOCOL IP
IP-ENABLE-LOOPBACK YES
IP-LOOPBACK-ADDRESS 127.0.0.1
IP-FRAGMENTATION-UNIT 2048
IP-QUEUE-NUM-PRIORITIES 3
IP-QUEUE-PRIORITY-INPUT-QUEUE-SIZE 50000
DUMMY-PRIORITY-QUEUE-SIZE NO
IP-QUEUE-PRIORITY-QUEUE-SIZE 50000
DUMMY-PRIORITY-WISE-IP-QUEUE-TYPE NO
IP-QUEUE-TYPE FIFO
# ECN as presented in RFC 2481. Requires one of the IP-QUEUE-TYPE (RED,
RIO, or WRED). Furthermore, the source and destination nodes must be ECN
enabled.
ECN NO
IP-QUEUE-SCHEDULER STRICT-PRIORITY
ROUTER-BACKPLANE-THROUGHPUT 0

# ***** Routing Protocol *****

DUMMY-ROUTING DYNAMIC
ROUTING-PROTOCOL AODV
# The maximum possible number of hops between two nodes in the network
AODV-NET-DIAMETER 35
# Conservative estimate of the average one hop traversal time for packets
and should include queuing, transmission, propagation and other delays
AODV-NODE-TRAVERSAL-TIME 40MS
# Timeout time for an active route; each time a data packet is received,
the lifetime of that route is updated to this value. A default value of
10 seconds is suggested for error detection through MAC layer message
(like what 802.11 does)
AODV-ACTIVE-ROUTE-TIMEOUT 3S
# The destination of a RREQ replies with AODV-MY-ROUTE-TIMEOUT as the
lifetime of the route.
AODV-MY-ROUTE-TIMEOUT 6S
# Lifetime of a hello message is determined by AODV-ALLOWED_HELLO_LOSS *
AODV-HELLO_INTERVAL
AODV-HELLO-INTERVAL 1S
# Lifetime of a hello message is determined by AODV-ALLOWED_HELLO_LOSS *
AODV-HELLO_INTERVAL
AODV-ALLOWED-HELLO-LOSS 2
# Specifies the number of times AODV will repeat expanded ring search for
a destination if no Route Reply is received within specified amount of
time.
AODV-RREQ-RETRIES 2
# A constant use for calculating the time after which an active route
should be deleted. After timeout of an active route, the route is
finally deleted from the routing table after a time period of (K * max
(AODV-ACTIVE_ROUTE_TIMEOUT, AODV-ALLOWED_HELLO_LOSS * AODV-
HELLO_INTERVAL)), Here K is AODV-ROUTE-DELETION-CONSTANT.
AODV-ROUTE-DELETION-CONSTANT 5
# If the value is set to YES, a node will send a hello message if there
is no broadcast within the last hello interval. Simulation time will
increase depending on the frequency of the hello updates.
```

```
AODV-PROCESS-HELLO NO
# If this value is set to YES, the node will try to locally repair a
broken route, if possible.
AODV-LOCAL-REPAIR NO
# If the source node of a route gets a route error message, it will
initiate a new Route Request for the destination if the value is set to
YES.
AODV-SEARCH-BETTER-ROUTE YES
# Maximum number of packets the message buffer of AODV can hold. If the
buffer fills up, incoming packets for the buffer will be dropped.
AODV-BUFFER-MAX-PACKET 100
# If nothing is specified, buffer overflow will be checked by number of
packets in the buffer. If some value is specified here, incoming packets
will be dropped if the incoming packet size + current size of the buffer
exceeds this value.
AODV-BUFFER-MAX-BYTE 0
AODV-OPEN-BI-DIRECTIONAL-CONNECTION YES
AODV-TTL-START 1
AODV-TTL-INCREMENT 2
AODV-TTL-THRESHOLD 7
HSRP-PROTOCOL NO
# Static routes have priority over those discovered by routing protocols
STATIC-ROUTE NO
# Default routes have less priority than static routes and those
discovered by routing protocols
DEFAULT-ROUTE YES
DEFAULT-ROUTE-FILE
C:\qualnet\3.8\gui\scenarios\finalqos3\finalqos3.routes-default

# ***** MPLS configuration *****

MPLS-PROTOCOL NO

# ***** Transport Layer *****

TCP LITE
TCP-USE-RFC1323 NO
TCP-DELAY-ACKS YES
TCP-DELAY-SHORT-PACKETS-ACKS NO
TCP-USE-NAGLE-ALGORITHM YES
TCP-USE-KEEPALIVE-PROBES YES
TCP-USE-PUSH YES
TCP-MSS 512
TCP-SEND-BUFFER 16384
TCP-RECEIVE-BUFFER 16384

# ***** ATM Layer2 *****

# ***** ATM Layer2 *****

ATM-RED-MIN-THRESHOLD 5
ATM-RED-MAX-THRESHOLD 15
ATM-RED-MAX-PROBABILITY 0.02
```

```
ATM-RED-SMALL-PACKET-TRANSMISSION-TIME 10MS
ATM-QUEUE-SIZE 15000
ATM-SCHEDULER-STATISTICS NO
ATM-LAYER2-STATISTICS NO
ATM-QUEUE-STATISTICS NO

# ***** Traffic and Status *****

# ***** Application Layer *****

# Used to setup applications such as FTP and Telnet. Will be added to
any applications configured manually.
APP-CONFIG-FILE C:\qualnet\3.8\gui\scenarios\finalqos3\finalqos3.app

# ***** Extras *****

# ***** Tracing *****

# Generates trace data compatible with Tracer viewing tool.
PACKET-TRACE NO
ACCESS-LIST-TRACE NO

# ***** Statistics *****

# ***** Statistics *****

# All the statistics are compiled together into a file called
"ExperimentName.stat"(where experiment name is specified on the General
settings) at the end of the simulation.

APPLICATION-STATISTICS YES
TCP-STATISTICS YES
UDP-STATISTICS YES
ROUTING-STATISTICS YES
ICMP-STATISTICS NO
IGMP-STATISTICS NO
EXTERIOR-GATEWAY-PROTOCOL-STATISTICS YES
NETWORK-LAYER-STATISTICS YES
QUEUE-STATISTICS YES
SCHEDULER-STATISTICS YES
MAC-LAYER-STATISTICS YES
PHY-LAYER-STATISTICS YES
MOBILITY-STATISTICS NO
MPLS-STATISTICS NO
MPLS-LDP-STATISTICS NO
RSVP-STATISTICS NO
SRM-STATISTICS NO
DIFFSERV-EDGE-ROUTER-STATISTICS NO
QOSPF-STATISTICS NO
# Network Statistics should be on
ACCESS-LIST-STATISTICS NO
```



```

POLICY-ROUTING-STATISTICS NO
ROUTE-REDISTRIBUTION-STATISTICS NO
SIGNALLING-STATISTICS NO
MOBILE-IP-STATISTICS NO

# ***** Node Specific *****

# ***** Device properties *****

USE-NODE-ICON YES
NODE-ICON C:\qualnet\3.8\gui\scenarios\finalqos3\DEFAULT.GIF

# ***** Router Specs *****

# ***** Router Configuration Specs *****

# ***** Node Orientation *****

AZIMUTH 0
ELEVATION 0

# ***** Parallel Properties *****

# Parallel partition to which to assign node.
PARTITION 0

#-----Default Subnet -----
SUBNET N8-192.0.0.0 { 1 thru 50 } Default

IP-FORWARDING NO
[ 1 thru 50 ] IP-FORWARDING YES

COMPONENT 0 {1 2 3 4 7 8 9 10 11 12 13 14 15 16 17 19 20 21 22 24 25 26
27 28 29 30 31 32 33 34 35 36 37 38 40 42 46 47 49 50 45 43 41 5 44 48 18
39 23 6} 50 750.0 750.0 0.0 1500.0
COMPONENT 0 {1500.0 3000.0

```

Los demás escenarios tienen la misma configuración, la diferencia entre uno y otro radica en los siguientes archivos individuales:

```
NODE-POSITION-FILE C:\qualnet\3.8\gui\scenarios\escenario\escenario.nodes
```

```
DEFAULT-ROUTE-FILE
```

```
C:\qualnet\3.8\gui\scenarios\escenario\escenario.routes-default
```

APP-CONFIG-FILE C:\qualnet\3.8\gui\scenarios\escenario\escenario.app

NODE-ICON C:\qualnet\3.8\gui\scenarios\escenario

#-----Default Subnet -----  
SUBNET N8-192.0.0.0 { 1 thru tnodos } Default

IP-FORWARDING NO  
[ 1 thru tnodos ] IP-FORWARDING YES

COMPONENT 0 { }  
COMPONENT 0 {