Departament d'Enginyeria Telemàtica

UNIVERSITAT POLITÈCNICA DE CATALUNYA


**Universidad Politécnica de Cataluña**
**Departamento de Ingeniería Telemática**


# TESIS DOCTORAL

# *Diferenciación de servicios y mejora de la supervivencia en redes ad hoc conectadas a redes fijas*


**Autora: Mari Carmen Domingo Aladrén**

**Director: David Remondo Bueno**

**Tutora: Cristina Cervelló i Pastor**


**Año: 2005**

*"Hay que tener perseverancia y sobre todo confianza en ti mismo. Hay que creer que se está dotado para alguna cosa y que esta cosa hay que obtenerla cueste lo que cueste".*

*Marie Curie*

# *Agradecimientos*

Cuando comencé esta tesis no sabía dónde me metía...

Yo era tres años más joven e inexperta y el mundo todavía debía de hacer frente a muchos sucesos...No había estallado la guerra de Irak, acabábamos de 'estrenar' el Campus de Castelldefels y por aquel entonces todavía vivía mi tío...

A lo largo de estos tres años ha pasado y me ha pasado de todo...mientras esta tesis iba poco a poco cobrando forma...adquiriendo vida...

Gracias a esta tesis he aprendido lo que es hacer una tesis. Hacer una tesis es un desafío a la inteligencia, a la imaginación, al ingenio, a la perseverancia… en busca siempre de la mejor respuesta. Hacer una tesis es padecer, pasarse noches en vela pensando cuál es la mejor solución a aquella pregunta que se te ha planteado...Hacer una tesis es gozar al ver que el tiempo y esfuerzo invertidos se convierten en resultados que te hacen sentirte realizada. Hacer una tesis es estudiar, pensar, investigar, discutir, debatir, analizar, interpretar, confiar, intentar contribuir con una minúscula aportación a que avance un gigante llamado ciencia…

Gracias a esta tesis he ido a congresos, he visitado lugares y he conocido a personas maravillosas que me han aportado conocimientos y puntos de vista distintos, haciéndome cambiar de perspectiva. He hecho amistades con profesores de universidad de sitios muy dispares, a quienes quiero agradecer su simpatía y cordialidad: el señor Petre Dini, de Cisco Systems, Estados Unidos, que me ha invitado a participar como miembro del comité técnico de dos congresos que se celebrarán en Lisboa y Chicago, Pedro M. Ruiz, de la Universidad de Murcia, a quien agradezco su ayuda desinteresada, Víctor Sempere, de la Universidad Politécnica de Valencia, con quien continuo en permanente colaboración investigando para una Cicyt común y muy especialmente a Rui Prior, de la Universidad de Oporto, por sus brillantes aportaciones y consejos.

Gracias a mis padres he podido hacer esta tesis...algo que resulta evidente, pues les debo la vida, así que... ¿Qué menos que dedicarles la tesis y agradecerles su apoyo emocional e incondicional?

Dedico esta tesis a Jaime...por la inmensa suerte que he tenido de que en esta gran rifa que es la vida me tocara como hermano...

Dedico esta tesis muy especialmente a mi tío, porque su recuerdo se mantiene vivo en mí y desearía que hubiera vivido para verlo.

También se la dedico a mis amigas de toda la vida: ¡Mónica y Alexandra, sois fantásticas y os deseo lo mejor de lo mejor!

Gracias a mi director David Remondo he podido realizar esta tesis: le agradezco su interés y apoyo prestado.

Gracias a Olga León por su colaboración en los inicios de la investigación.

Gracias al Departamento de Ingeniería Telemática por el soporte material y la ayuda que me han brindado.

Gracias también a todos los miembros del tribunal por su dedicación a la lectura y evaluación de esta tesis.

Cuando empecé esta tesis no sabía dónde me metía...Ahora que sí lo sé... ¡Cuánto me alegro de haberla realizado!

# *Resumen*

La comunicación entre redes ad hoc y redes basadas en infraestructura resulta esencial para poder extender Internet más allá de su alcance tradicional, a aquellas áreas hasta ahora inaccesibles, permitiendo la utilización de servicios Web y otras muchas aplicaciones en todo momento y lugar.

En esta tesis doctoral se abordan dos difíciles retos: intentar proporcionar calidad de servicio extremo a extremo en la comunicación entre una red ad hoc y una red fija, y alargar la supervivencia de la red ad hoc para que dicha comunicación sea lo más estable y duradera posible.

Para lograr alcanzar estos objetivos, se ha realizado primeramente un estudio exhaustivo tanto de los modelos de calidad de servicio como de los protocolos de encaminamiento existentes para redes ad hoc aisladas. Fruto de dicho estudio ha surgido una primera contribución que consiste en el diseño e implementación de un protocolo de encaminamiento para la mejora de la supervivencia en una red ad hoc aislada.

A partir de esta base se ha podido abordar la diferenciación de servicios en redes ad hoc conectadas con redes fijas; como consecuencia de esta investigación se ha desarrollado una segunda contribución que consiste en el diseño y evaluación de un modelo de diferenciación de servicios que se basa en la cooperación para el mantenimiento de la calidad de servicio entre ambas redes.

Finalmente, mediante una tercera contribución, se ha conseguido mejorar la supervivencia de una red ad hoc conectada a una red basada en infraestructura con el diseño e implementación de un protocolo de encaminamiento específicamente creado para tal efecto. Además, se ha demostrado que la incorporación de este protocolo de encaminamiento en una red ad hoc que utiliza un modelo de calidad de servicio basado en la interacción entre la red ad hoc y la red IP fija, no sólo alarga la supervivencia de la red ad hoc sino que además evita un aumento de la congestión y mejora la diferenciación de servicios entre ambas redes.

Las simulaciones exhaustivas realizadas sirven para comparar todas estas contribuciones con otras propuestas anteriores, demostrando su efectividad y rendimiento.

Las contribuciones presentadas en esta tesis doctoral tienen una singular importancia, pues hasta la fecha no se ha desarrollado ningún modelo de calidad de servicio que permita la interacción y favorezca la cooperación entre una red ad hoc y una red IP fija con el fin de proporcionar calidad de servicio extremo a extremo. Las

contribuciones que se aportan demuestran que sí que es posible la diferenciación de servicios entre una red ad hoc y una red IP fija; además, prueban que resulta imprescindible la cooperación e integración de los modelos de calidad de servicio de ambas redes para lograrlo. Este trabajo resulta pionero en estos aspectos y sirve para abrir una nueva línea de investigación con el fin de promover la comunicación entre redes ad hoc y redes fijas.

# Índice

*1 Introducción.....1*

1.1 Fundamentos: las redes ad hoc.....2

1.2 Motivación.....4

1.2.1 Motivación 1: Proporcionar calidad de servicio entre una red ad hoc y una red IP fija.....4

1.2.2 Motivación 2: Mejorar la supervivencia de una red ad hoc conectada a una red IP fija.....6

1.3 Objetivos.....7

1.4 Aportaciones originales de la tesis doctoral.....9

1.5 Estructura de la tesis doctoral.....10

1.6 Publicaciones relacionadas con la tesis doctoral.....11

*2 Modelos de calidad de servicio en redes ad hoc aisladas.....15*

2.1 Protocolos a nivel de la capa MAC para diferenciar servicios...18

2.1.1 La capa MAC (Medium Access Control).....18

2.1.1.1 Protocolo DCF del MAC IEEE 802.11.....23

2.1.2 Clasificación de los mecanismos de QoS a nivel de la capa MAC para IEEE 802.11.....28

2.1.2.1 Soporte a la QoS basado en prioridades.....29

2.1.2.1.1 Espacio entre tramas.....30

2.1.2.1.2 Algoritmo de backoff.....31

2.1.2.1.3 Ejemplos de mecanismos con soporte a la QoS basados en prioridades.....32

2.1.2.1.3.1 El esquema DENG.....32

2.1.2.1.3.2 IEEE 802.11e o Enhanced DCF (EDCF).....33

2.1.2.1.3.3 Adaptative Enhanced DCF (AEDCF).....36

2.1.2.1.3.4 Los algoritmos Virtual MAC (VMAC) y Virtual Source (VS).....37

2.1.2.1.3.5 Blackburst.....39

2.1.2.2 Soporte a la QoS basado en usar una disciplina de servicio justa (fair scheduling).....40

2.1.2.2.1 Espacio entre tramas.....41

2.1.2.2.2 Algoritmo de backoff.....41

2.1.2.2.3 Ejemplos de mecanismos con soporte a la QoS basados en una disciplina de servicio justa.....42
2.1.2.2.3.1 Distributed Deficit Round Robin (DDRR).....42
2.1.2.2.3.2 Distributed Fair Scheduling (DFS).....43
2.1.2.2.3.3 Distributed Weighted Fair Queuing (DWFQ).....44
2.1.3 Comparación entre los diversos mecanismos de QoS para IEEE 802.11.....46
2.2 El modelo INSIGNIA.....47
2.3 El modelo FQMM.....58
2.4 La arquitectura de Servicios Diferenciados.....67
2.4.1 La arquitectura de Servicios Diferenciados aplicada a redes ad hoc.....71
2.4.1.1 El algoritmo RED.....74
2.4.1.2 El algoritmo RIO o RIO-C.....76
2.5 El modelo SWAN.....77
2.6 Comparación entre distintos modelos de calidad de servicio y elección de un modelo de calidad de servicio para redes ad hoc aisladas.....86
2.7 Conclusiones.....90
***3 Protocolos de encaminamiento en redes ad hoc aisladas.....93***
3.1 Protocolos de encaminamiento proactivos, reactivos e híbridos.94
3.1.1 Protocolos de encaminamiento proactivos.....95
3.1.1.1 Destination-sequenced Distance Vector (DSDV).....96
3.1.1.2 Optimized Link State Routing (OLSR).....100
3.1.2 Protocolos de encaminamiento reactivos.....101
3.1.2.1 Dynamic Source Routing (DSR).....102
3.1.2.2 Ad hoc On-Demand Distance Vector (AODV).....109
3.1.3 Protocolos de encaminamiento híbridos.....119
3.1.3.1 Zone Routing Protocol (ZRP).....119
3.2 Protocolos de encaminamiento best-effort y con calidad de servicio.....121
3.2.1 Protocolos de encaminamiento best-effort.....121
3.2.2 Protocolos de encaminamiento con calidad de servicio.....121

3.2.2.1 Protocolo de encaminamiento con calidad de servicio 'basado en ticket' (Ticket-based).....123
3.2.2.2 AODV con QoS.....125
3.3 Encaminamiento y disponibilidad de energía.....129
3.4 Comparación entre distintos protocolos de encaminamiento y elección de un protocolo de encaminamiento para redes ad hoc aisladas…….....134
3.5 Contribución: Desarrollo del protocolo de encaminamiento SEADSR para la mejora de la supervivencia en redes ad hoc aisladas ……....……....136
3.5.1 Explicación teórica.....136
3.5.1.1 Descripción de SEADSR.....136
3.5.1.2 Análisis de SEADSR.....139
3.5.2 Simulaciones.....142
3.5.2.1 Escenario de simulación.....142
3.5.2.1.1 Análisis de las simulaciones.....143
3.6 Conclusiones.....146
***4 Modelos de calidad de servicio en redes ad hoc conectadas a redes fijas.....149***
4.1 Estado actual de la investigación.....149
4.2 Contribución: Desarrollo del modelo de calidad de servicio DS-SWAN para redes ad hoc conectadas a redes fijas….....150
4.2.1 Explicación teórica.....150
4.2.1.1 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija.....151
4.2.1.1.1 Diversas versiones del modelo DS-SWAN para tráfico enviado desde la red ad hoc hacia la red fija.....157
4.2.1.2 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red fija hacia la red ad hoc.....161
4.2.2 Simulaciones.....163
4.2.2.1 Escenario de simulación básico.....163
4.2.2.1.1 Análisis de las simulaciones para tráfico enviado desde la red ad hoc hacia la red fija.....167

4.2.2.1.2 Análisis de las simulaciones con respecto a la escalabilidad para tráfico enviado desde la red ad hoc hacia la red fija..174
4.2.2.1.3 Análisis de las simulaciones con tráfico best-effort TCP para tráfico enviado desde la red ad hoc hacia la red fija..180
4.2.2.1.4 Análisis de las simulaciones para tráfico enviado desde la red fija hacia la red ad hoc y estudio de la escalabilidad...183
4.3 Conclusiones…….....188
***5 Protocolos de encaminamiento en redes ad hoc conectadas a redes fijas.....191***
5.1 Estado actual de la investigación.....191
5.1.1 Mecanismo básico para la conexión de redes ad hoc con Internet …….....191
5.2 Contribución: Desarrollo del protocolo de encaminamiento SD-AODV para la mejora de la supervivencia y la cooperación en el mantenimiento de la calidad de servicio en redes ad hoc conectadas a redes fijas.....194
5.2.1 Explicación teórica.....194
5.2.2 Simulaciones.....198
5.2.2.1 Escenario de simulación.....198
5.2.2.1.1 Análisis de las simulaciones para tráfico enviado desde la red ad hoc hacia la red fija.....199
5.2.2.1.2 Análisis de las simulaciones para tráfico enviado desde la red fija hacia la red ad hoc.....204
5.3 Conclusiones..... 208
***6 Conclusiones…..…….....211***
***7 Líneas futuras……..……215***
***Bibliografía……....217***
***Glosario de acrónimos……..……231***

# *Índice de Figuras*

Fig. 1.1. Red ad hoc.....3
Fig. 1.2. Ejemplo de aplicación de redes ad hoc.....5
Fig. 1.3. Ejemplo de aplicación de redes ad hoc.....5
Fig. 2.1. Ejemplo de aplicación de Bluetooth.....19
Fig. 2.2. Ejemplo de funcionamiento de la función de coordinación distribuida.....23
Fig. 2.3. Reconocimientos positivos en DCF.....25
Fig. 2.4. Problema del terminal escondido.....25
Fig. 2.5. Efectividad del protocolo de handshake RTS/CTS.....27
Fig. 2.6. Problema del terminal expuesto.....27
Fig. 2.7. Mecanismos de QoS distribuidos a nivel de la capa MAC para IEEE 802.11.....29
Fig. 2.8. EDCF propuesto. TXOP (Transmission Opportunity): Intervalo de tiempo en el que una estación tiene derecho a transmitir, definido mediante un inicio de transmisión y una duración máxima.....34
Fig. 2.9. Relación temporal para EDCF.....35
Fig. 2.10. Blackburst. Estaciones de alta prioridad intentando acceder al medio cuando éste está ocupado.....40
Fig. 2.11. El mecanismo DDRR.....42
Fig. 2.12. Ejemplo 1 de funcionamiento del esquema DFS.....44
Fig. 2.13. Ejemplo 2 de funcionamiento del esquema DFS.....44
Fig. 2.14. Modelo de gestión de flujos INSIGNIA en un nodo/router móvil.....49
Fig. 2.15. El campo de opciones de IP INSIGNIA.....50
Fig. 2.16. Establecimiento de una reserva.....51
Fig. 2.17. Reserva rápida.....52
Fig. 2.18. Reencaminamiento del flujo de datos y restauración rápida.....53
Fig. 2.19. Reencaminamiento y degradación.....54
Fig. 2.20. Adaptación: Aumento de un flujo (Scaling up).....57
Fig. 2.21. Adaptación: Reducción de un flujo (Scaling down).....58
Fig. 2.22. Escenario 1.....59
Fig. 2.23. Escenario 2.....59
Fig. 2.24. Arquitectura del modelo FQMM.....60
Fig. 2.25. Componentes para priorizar servicios en la arquitectura del modelo FQMM.....64
Fig. 2.26. Componentes para diferenciar servicios en la arquitectura del modelo FQMM.....66
Fig. 2.27. Una región DS.....67
Fig. 2.28. Elementos básicos de la arquitectura de Servicios Diferenciados.....68

Fig. 2.29. Módulos principales para el tratamiento del tráfico.....68
Fig. 2.30. Ejemplo de arquitectura DiffServ con Servicio Olímpico.....71
Fig. 2.31. Sistema de colas definido a nivel de la capa MAC.....73
Fig. 2.32. Cola RED (Random Early Detection).....75
Fig. 2.33. Probabilidad de descarte RED (Random Early Detection).....75
Fig. 2.34. Cola RIO-C (RED con In/Out) (Coupled).....76
Fig. 2.35. Probabilidades de descarte RIO.....77
Fig. 2.36. Pila de protocolos en cada nodo de la red ad hoc.....77
Fig. 2.37. Modelo SWAN.....78
Fig. 2.38. Campos Differentiated Services Codepoint y ECN en IP.....79
Fig. 2.39. Control de admisión.....81
Fig. 2.40. Los paquetes de prueba 'petición' son empujados hasta la capa SWAN en cada nodo intermedio.....82
Fig. 2.41. Mensaje de prueba petición/respuesta.....82
Fig. 2.42. Regulación.....84
Fig. 2.43. Mensaje de regulación.....84
Fig. 2.44. (Izquierda) Goodput medio para un flujo TCP medido en Kbps en función de la movilidad. (Derecha) Throughput medio de un flujo UDP en función de la movilidad [92].....89
Fig. 2.45. Retardos medios extremo a extremo sufridos por los paquetes de un flujo TCP (izquierda) y de un flujo UDP (derecha).....90
Fig. 3.1. Clasificación de los protocolos de encaminamiento en redes ad hoc.....95
Fig. 3.2. Red ad hoc donde existe movilidad.....98
Fig. 3.3. Relays multipunto.....100
Fig. 3.4. Descubrimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento DSR para enviar datos desde un nodo origen S hasta un nodo destino D (1).....106
Fig. 3.5. Descubrimiento de Ruta (2).....107
Fig. 3.6. Descubrimiento de Ruta (3).....107
Fig. 3.7. Descubrimiento de Ruta (4).....107
Fig. 3.8. Descubrimiento de Ruta (5).....108
Fig. 3.9. Descubrimiento de Ruta (6).....108
Fig. 3.10. Descubrimiento de Ruta (7).....108
Fig. 3.11. Mantenimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento DSR para enviar datos desde un nodo origen S hasta un nodo destino D.....109
Fig. 3.12. Descubrimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento AODV para enviar datos desde un nodo origen S hasta un nodo destino D (1)..116
Fig. 3.13. Descubrimiento de Ruta (2).....116

Fig. 3.14. Descubrimiento de Ruta (3).....117
Fig. 3.15. Descubrimiento de Ruta (4).....117
Fig. 3.16. Descubrimiento de Ruta (5).....117
Fig. 3.17. Descubrimiento de Ruta (6).....118
Fig. 3.18. Descubrimiento de Ruta (7).....118
Fig. 3.19. Descubrimiento de Ruta (8).....118
Fig. 3.20. Ejemplo de Descubrimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento ZRP…….....120
Fig. 3.21. Ejemplo de una red ad hoc que utiliza el protocolo de encaminamiento 'basado en ticket'….....124
Fig. 3.22. Ejemplo de una red ad hoc que utiliza el protocolo de encaminamiento AODV con QoS con la extensión de retardo máximo….....127
Fig. 3.23. Ejemplo de una red ad hoc que utiliza el protocolo de encaminamiento AODV con QoS con la extensión de ancho de banda mínimo.....128
Fig. 3.24. Ejemplo de una red ad hoc que utiliza un protocolo de encaminamiento que tiene en cuenta la capacidad de batería.....131
Fig. 3.25. Función de retardo $\tau(C)$ aplicada sobre un paquete de RREQ que llega a un nodo intermedio para determinar el retardo adicional que se introducirá sobre dicho paquete……..….....138
Fig. 3.26. Red ejemplo. Se muestran los valores de energía.....140
Fig. 3.27. Energía con DSR, SEADSR y MMBCR para la red de la Fig. 3.26 ( $C_{\max}$ = 40 J)...141
Fig. 3.28. Red ejemplo. Se muestran los valores de energía.....141
Fig. 3.29. Energía con DSR, SEADSR y MMBCR para la red de la Fig. 3.28 ( $C_{\max}$ = 60 J)...142
Fig. 3.30. Fallo de los nodos debido al agotamiento de sus reservas de energía en función del tiempo. Se muestran los intervalos de confianza del 90%.....144
Fig. 3.31. Throughput en función del tiempo.....145
Fig. 3.32. Histograma del retardo de paquetes de datos.....145

Fig. 4.1. Escenario propuesto.....151
Fig. 4.2. Envío de mensajes de QoS_PERDIDA en el escenario propuesto.....153
Fig. 4.3. Ejemplo de red.....156
Fig. 4.4. Envío de mensajes de QoS_PERDIDA en el escenario propuesto.....162
Fig. 4.5. Continuación del envío de mensajes de QoS_PERDIDA en el escenario propuesto..162
Fig. 4.6. Entorno de simulación.....164
Fig. 4.7. Funciones del router frontera.....166
Fig. 4.8. Retardo extremo a extremo para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN

(Casos 2 - 3)……..….....168
Fig. 4.9. Throughput para el tráfico best-effort: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3)
…….168
Fig. 4.10. Retardo a nivel de la capa MAC para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3)….....169
Fig. 4.11. Jitter para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).....170
Fig. 4.12. Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).....172
Fig. 4.13. Número de paquetes perdidos en el router frontera de ingreso para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3)….....172
Fig. 4.14. Número de paquetes de VoIP enviados que llegan al router frontera de ingreso: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).....173
Fig. 4.15. Tasa de pérdida de paquetes en el router frontera de ingreso para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3)…….....173
Fig. 4.16. Retardo máximo extremo a extremo del tráfico de VoIP: “DS-SWAN – 20 nodos” (Caso 1) vs. “DS-SWAN – 40 nodos” (Caso 2).....176
Fig. 4.17. Throughput para el tráfico CBR: “DS-SWAN – 20 nodos” (Caso 1) vs. “DS-SWAN – 40 nodos” (Caso 2).....176
Fig. 4.18. Retardo extremo a extremo para el tráfico de VoIP.....177
Fig. 4.19. Jitter para el tráfico de VoIP.....177
Fig. 4.20. Throughput para el tráfico CBR.....178
Fig. 4.21. Retardo máximo extremo a extremo del tráfico de VoIP: “DS-SWAN – 20 nodos” (Caso 1) vs. “DS-SWAN – 40 nodos” (Caso 2)….....179
Fig. 4.22. Throughput para el tráfico CBR: “DS-SWAN – 20 nodos” (Caso 1) vs. “DS-SWAN – 40 nodos” (Caso 2)…….....179
Fig. 4.23. Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2)….....182
Fig. 4.24. Jitter para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).....182
Fig. 4.25. Throughput para el tráfico best-effort: DS-SWAN (Caso 1) vs. SWAN (Caso 2)....183
Fig. 4.26. Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2)…….....184
Fig. 4.27. Jitter para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).....185
Fig. 4.28. Throughput para el tráfico best-effort: DS-SWAN (Caso 1) vs. SWAN (Caso 2)....185
Fig. 4.29. Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).....186
Fig. 4.30. Retardo máximo extremo a extremo del tráfico de VoIP: “DS-SWAN – 20 nodos” (Caso 1) vs. “DS-SWAN – 40 nodos” (Caso 2).....187

Fig. 4.31. Throughput para el tráfico CBR: "DS-SWAN – 20 nodos" (Caso 1) vs. "DS-SWAN – 40 nodos" (Caso 2).....187
Fig. 5.1. Escenario de interconexión.....192
Fig. 5.2. Arquitectura de los protocolos.....192
Fig. 5.3. Ejemplo de red.....196
Fig. 5.4. Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2)…….....199
Fig. 5.5. Throughput para el tráfico best-effort: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2) …….……..…….....200
Fig. 5.6. Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).....201
Fig. 5.7. Tasa de pérdida de paquetes en el router frontera de ingreso para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).....202
Fig. 5.8. Jitter para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).....202
Fig. 5.9. Número de nodos caídos en la red ad hoc: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2)..... ….....203
Fig. 5.10. Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2)……..205
Fig. 5.11. Jitter para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).....205
Fig. 5.12. Throughput para el tráfico best-effort: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2)……...206
Fig. 5.13. Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).....207

# *Índice de Tablas*

Tabla 2.1. Clases de prioridad Deng. $B$ representa el tiempo de backoff en número de slots temporales, $\rho$ es una variable aleatoria en el intervalo (0,1), $i$ representa el procedimiento de backoff i-ésimo para esta trama y $\lfloor x \rfloor$ representa el mayor entero menor o igual a x…….....33
Tabla 2.2. Mapeo entre la clase de tráfico (prioridad de usuario) y la categoría de acceso (AC). .....34
Tabla 2.3. Comparación de diversos mecanismos de soporte a la QoS para WLANs.....47
Tabla 2.4. Roles de los nodos aplicando el modelo FQMM.....60
Tabla 2.5. Tabla de codepoints AF de DiffServ.....…….....70
Tabla 2.6. Diferencias entre la arquitectura de los modelos DiffServ y SWAN.....88
Tabla 3.1. Tabla de encaminamiento para el nodo M4.....98
Tabla 3.2. Tabla de encaminamiento del nodo M4 de actualización.....98
Tabla 3.3. Tabla de encaminamiento para el nodo M4 (actualizada).....99
Tabla 3.4. Tabla de encaminamiento del nodo M4 de actualización enviada en el mensaje de encaminamiento incremental.....99
Tabla 4.1. Valores de parámetros para las simulaciones.....154
Tabla 4.2. Valores de parámetros para las simulaciones.....163
Tabla 4.3. Parámetros del tráfico de voz.....165

# *1 Introducción*

El trepidante mundo de las telecomunicaciones se ha visto vigorosamente revolucionado con el desarrollo e investigación de nuevas aplicaciones y sistemas inalámbricos cuya finalidad es permitir la comunicación en cualquier momento y lugar. Los dispositivos inalámbricos pronto se convertirán en la manera preferida de comunicación y acceso a la información.

Con el término 'pervasive computing' u 'omnipresencia de las computadoras' nos referimos a un entorno donde las aplicaciones virtuales estarán presentes en todas partes y las infraestructuras de ordenadores serán inherentes al ser humano, facilitándole cualquier tarea imaginable.

A medida que las redes inalámbricas vayan evolucionando, los clientes solicitarán aplicaciones de red incluso en casos donde no exista una infraestructura de red propia. Los usuarios de terminales móviles con dispositivos inalámbricos compatibles entre sí deberán ser capaces de establecer una red de corta duración que les permita satisfacer sus necesidades de comunicación en un momento determinado, es decir, deberán poder implementar una red ad hoc.

Pero las redes ad hoc no sólo tendrán utilidad como redes aisladas e independientes. Su papel será primordial al ser empleadas en los 'hotspots' o lugares de interés (hoteles, aeropuertos, centros comerciales, etc.) donde existe una alta concentración de personas con necesidad de establecer al mismo tiempo comunicaciones de voz y/o datos. Para poder cubrir las necesidades de dichos usuarios se suele usar el estándar de tecnología WLAN (Wireless Local Area Networks) denominado IEEE 802.11.

Cuando las WLAN con IEEE 802.11 operan en el modo basado en infraestructura, la red está constituida por al menos un punto de acceso conectado a la infraestructura de la red cableada y a un conjunto de estaciones inalámbricas.

En cambio, cuando las WLAN con IEEE 802.11 operan en el modo ad hoc, lo que tenemos es un conjunto de estaciones inalámbricas que se comunican directamente entre sí, sin la presencia de puntos de acceso, evitándose de esta forma su colapso a la hora de atender a un número tan elevado de usuarios (no todo el tráfico debe de estar dirigido al punto de acceso, tal y como sucede con las WLAN operando en el modo basado en infraestructura).

Estas propiedades convierten a las redes ad hoc en redes altamente flexibles y rápidas de desarrollar, pues no necesitan para su funcionamiento una infraestructura propia. Por este motivo, se han convertido en la manera más natural de extender

Internet más allá de su alcance tradicional, a aquellas áreas hasta ahora inaccesibles, permitiendo la utilización de servicios Web en todo momento y lugar. Gracias a la coexistencia y cooperación entre las redes ad hoc y las redes IP fijas, que nos permitirán acceder a Internet, será posible desarrollar nuevas e innovadoras aplicaciones, aprovechando también las ya existentes.

Un ejemplo futurista serán aquellas redes ad hoc que se establecerán espontáneamente entre vehículos en una carretera para poder intercambiar información, localizarse e influenciarse mutuamente. Estas redes serán muy útiles en caso de accidentes, donde se podría enviar un mensaje de alerta desde un coche al resto de vehículos para que escogiera una ruta alternativa a la del accidente y de este modo pudiera prevenirse la congestión. Internet también podría prestar conjuntamente sus servicios a los usuarios de estas redes ad hoc 'improvisadas' mediante informaciones actualizadas del estado de las carreteras.

Dada la enorme importancia que pueden llegar a tener las comunicaciones inalámbricas y en especial las redes ad hoc, éstas deben estar preparadas para ofrecer una cierta calidad de servicio, QoS (Quality of Service) a aplicaciones tales como las multimedia. Resulta muy complicado establecer calidad de servicio en una red ad hoc debido a sus particularidades; por este motivo, la provisión de calidad de servicio entre redes ad hoc y redes IP fijas se ha convertido en un desafío tan difícil como atrayente.

En esta tesis doctoral se han centrado todos los esfuerzos en establecer una comunicación en las mejores condiciones posibles entre una red ad hoc y una red IP fija, pero para lograr alcanzar el objetivo propuesto ha sido preciso comenzar estudiando qué es una red ad hoc.

## *1.1 Fundamentos: las redes ad hoc*

Puede definirse una red ad hoc [1] como aquella que establece una comunicación espontánea entre terminales fijos y móviles o sólo móviles, siempre y cuando exista la posibilidad física de lograrlo.

Las redes ad hoc *(Véase la Fig. 1.1)* están formadas por dos o más dispositivos que son capaces de comunicarse entre sí sin la necesidad de recurrir a una infraestructura de red preexistente, con lo cual no son requeridas estaciones base ni cables ni routers fijos. Dichas redes pueden estar constituidas por grupos de terminales móviles independientes y basados en radio enlaces, aunque también cabría la posibilidad de

que alguno de estos dispositivos estuviera conectado a un sistema celular o a una red fija.

Las redes ad hoc son adaptativas y están habilitadas para configurarse a sí mismas, prescindiéndose de la intervención de un administrador del sistema.

En la práctica, las redes ad hoc podrían disponer desde decenas hasta centenares de nodos de comunicaciones capaces de cubrir alcances radio de 30 a 100 metros en interiores y de 100 hasta 300 metros en exteriores.

**Fig. 1.1.** Red ad hoc.

En las redes ad hoc es posible que dos nodos inalámbricos se puedan comunicar entre sí, incluso cuando se hallan fuera de su alcance radio, gracias a la presencia de nodos intermedios que actuarán como routers y reenviarán los paquetes de datos de la fuente al destino. Una red donde la comunicación entre dos estaciones se consigue mediante el reenvío de datos a través de otros nodos intermedios, recibe el nombre de red inalámbrica multisalto.

Las redes ad hoc se caracterizan por tener topologías dinámicas, donde los nodos se mueven libremente de manera arbitraria y en un tiempo impredecible y pueden estar constituidas por enlaces unidireccionales (comunicación en un único sentido) o bidireccionales (comunicación en ambos sentidos).

En una red ad hoc [2] se pueden realizar tareas de encaminamiento con el fin de optimizar la calidad de servicio.

Las redes ad hoc presentan restricciones de ancho de banda motivadas por la capacidad variable de sus enlaces inalámbricos y, como consecuencia, en muchas ocasiones se produce congestión. De hecho, el throughput de las comunicaciones inalámbricas es mucho menor que la tasa máxima de transmisión radio debido a causas tales como la contienda en el acceso múltiple, fading, ruido y condiciones de interferencia.

## *1.2 Motivación*

Las motivaciones por las cuales se ha desarrollado particularmente esta tesis doctoral son las siguientes:

- *Proporcionar calidad de servicio entre una red ad hoc y una red IP fija.*
- *Mejorar la supervivencia de una red ad hoc conectada a una red IP fija.*

A continuación se describen con mayor profundidad ambas motivaciones.

### *1.2.1 Motivación 1: Proporcionar calidad de servicio entre una red ad hoc y una red IP fija*

Para analizar con detenimiento esta motivación, resulta indispensable presentar el concepto de calidad de servicio, QoS (Quality of Service).

La calidad de servicio [3] puede definirse como aquella habilidad que posee una red para ofrecer un servicio requerido por alguna aplicación de red específica, estableciendo algún tipo de control sobre el retardo extremo a extremo, las pérdidas, el jitter y/o el ancho de banda.

La calidad de servicio se basa en el concepto de que las propias aplicaciones pueden indicar o incluso negociar sus requisitos específicos con la red [4]. No todas las aplicaciones necesitarán que la red les proporcione por igual los mismos niveles de garantía o de QoS. Las aplicaciones de tiempo real (tales como voz o vídeo) tendrán estrictos requisitos de calidad de servicio temporales, puesto que si los paquetes pertenecientes a dichas aplicaciones no llegan en un momento determinado a su destino, su contenido ya no tendrá valor y la transmisión será incorrecta.

Se ha desarrollado mucha literatura científica [5], [6], [67], [68] en los últimos años acerca de cómo proporcionar calidad de servicio en Internet, que es una red basada en el protocolo IP, que utiliza una técnica de transmisión de almacenamiento y reenvío (store-and-forward), sin ofrecer retardos acotados de llegada de paquetes y donde el tráfico, que circula a ráfagas, puede perderse fácilmente en los routers por una sobrecarga de sus buffers. Internet solamente se presta a hacer voluntariamente lo que pueda para que los paquetes puedan alcanzar su destino (servicio best-effort), pero no ofrece de por sí ninguna garantía a las aplicaciones de tiempo real de que los paquetes llegarán puntualmente a sus destinos. Las arquitecturas IntServ (Integrated

Services) [5], [6] y DiffServ (Differentiated Services) [67], [68] pretenden solucionar este déficit.

Si proporcionar calidad de servicio en una red IP fija tiene sus complicaciones, conseguir ofrecer calidad de servicio en una red ad hoc se convierte en un reto tan extremadamente difícil como atrayente.

En una red ad hoc resulta particularmente complicado proporcionar una cierta calidad de servicio porque tanto la topología como capacidad de los enlaces varían dinámicamente. Además, en los entornos inalámbricos la existencia de fading provoca que las tasas de pérdidas de paquetes y las variaciones de retardo sean mucho mayores y variables en comparación con las redes fijas.

Por todos estos motivos, se ha llegado a cuestionar si resulta viable proporcionar calidad de servicio a una red con estas características; no obstante, se han realizado y se están dedicando muchos esfuerzos para conseguirlo; prestigiosos investigadores de todo el planeta se hallan actualmente entregados a este propósito.

En un principio, la investigación en el campo de las redes ad hoc se centró fundamentalmente en desarrollar redes aisladas e independientes, que pudieran desempeñar su labor en determinados escenarios destacados:

- *Redes en catástrofes naturales (Véase la Fig. 1.2) (huracanes, inundaciones, terremotos, incendios, etc.), es decir, en zonas que carecen de infraestructura.*

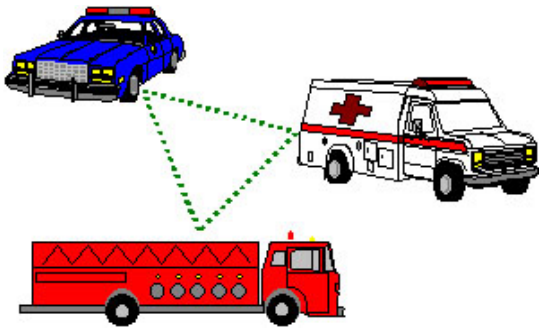


**Fig. 1.2.** Ejemplo de aplicación de redes ad hoc.

- *Redes en operaciones militares en zonas donde tampoco haya infraestructura (Véase la Fig. 1.3).*

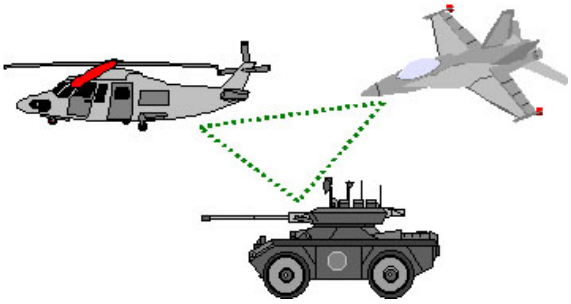


**Fig. 1.3.** Ejemplo de aplicación de redes ad hoc.

- *Redes en áreas remotas o muy escasamente pobladas, donde no salga a cuenta instalar redes con infraestructura.*

Sin embargo, este tipo de redes se halla sujeto a entornos muy restringidos. Además, el tráfico no local de las redes ad hoc viajará muy probablemente a través de redes basadas en infraestructura y la manera de que las redes ad hoc puedan acceder a Internet será también a través de redes con infraestructura.

Por todas estas razones, la investigación actual se ha centrado muy recientemente en cómo conseguir la convivencia e interacción para facilitar la integración entre redes ad hoc y redes IP fijas.

Si la red ad hoc está conectada a una red IP fija, que le proporciona acceso a Internet, será crucial y esencial el desarrollo de un modelo de calidad de servicio que facilite la interacción entre ambas redes para poder transportar el tráfico de tiempo real.

Se convierte pues en un importante reto el conseguir que este modelo de QoS sea capaz de acomodarse tanto a la red ad hoc como a la red IP fija, pudiendo incluso favorecer la cooperación entre distintos modelos de calidad de servicio de cada una de las redes para que la diferenciación de servicios extremo a extremo sea posible.

### *1.2.2 Motivación 2: Mejorar la supervivencia de una red ad hoc conectada a una red IP fija*

Los terminales móviles en las redes ad hoc se alimentan típicamente con baterías de capacidades muy diversas, dependiendo del tipo de dispositivo móvil.

La tecnología de las baterías ha experimentado un progreso muy lento si lo comparamos con los resultados alcanzados por la tecnología de circuitos integrados, donde se ha logrado un crecimiento espectacular en la velocidad de las comunicaciones. Con el fin de que los nodos en la red ad hoc tarden más tiempo en agotar sus capacidades de batería, será necesario desarrollar protocolos que concedan una especial importancia a la conservación de la energía y que busquen minimizar el consumo de la potencia de transmisión en las comunicaciones como criterio de diseño para la optimización de los sistemas inalámbricos.

Las redes ad hoc [130] deben ser diseñadas de acuerdo con estos criterios. Se está llevando a cabo una importante investigación con el fin de desarrollar nuevos protocolos (tanto de encaminamiento como a nivel de la capa de red o de acceso al medio [139]) que sean capaces de alargar el tiempo de vida de las baterías de una red ad hoc con el fin de mejorar su supervivencia.

Si tenemos en cuenta que las redes ad hoc no solamente existirán en entornos aislados, sino que también se integrarán, coexistiendo con otras redes (ya sean fijas o celulares), este factor también resulta decisivo en este tipo de escenarios. Debe fomentarse el uso de técnicas que permitan alargar lo máximo posible el tiempo de vida de los nodos de una red ad hoc, porque de esta forma se está fomentando la comunicación entre las redes ad hoc y otras redes, como las redes IP fijas. Favorecer esta comunicación significa invertir esfuerzos en que sea lo más fiable y duradera posible.

## *1.3 Objetivos*

El objetivo vital de esta tesis doctoral es el de hacer factible una comunicación en las mejores condiciones posibles entre una red ad hoc y una red IP fija.

Se han realizado numerosos estudios de investigación en torno a redes ad hoc aisladas y también en torno a redes IP fijas, pero la literatura existente es escasísima cuando de lo que se trata es de estudiar y analizar la interconexión y comunicación entre redes ad hoc y redes IP fijas.

Conseguir que las redes ad hoc puedan comunicarse en las mejores condiciones posibles con las redes basadas en infraestructura, significa lograr alcanzar los dos objetivos siguientes:

- *Intentar proporcionar calidad de servicio extremo a extremo en la comunicación entre una red ad hoc y una red fija*
- *Alargar la supervivencia de la red ad hoc para que la comunicación sea lo más estable y duradera posible*

Los dos objetivos anteriormente expuestos resultan fundamentales porque:

- *Las redes ad hoc únicamente pueden acceder a Internet a través de uno o más dominios de acceso; de ahí viene la necesidad de crear un modelo de calidad de servicio para la interacción entre ambas redes.*
- *Los requisitos de calidad de servicio son muy críticos cuando el tráfico viaja a través de distintos dominios y necesitan ser estudiados con detenimiento; particularmente el tráfico de la red ad hoc tendrá mayor problema para acceder a los recursos y disponer de ellos a lo largo del tiempo.*

Con el fin de lograr los dos objetivos anteriores, se ha decido desarrollar las tareas siguientes:

- *Se han estudiado las principales características de las redes ad hoc.*

- *Se ha estudiado toda la problemática existente en el diseño e implementación de redes ad hoc.*
- *Se han clasificado y estudiado los modelos de calidad de servicio existentes en redes ad hoc aisladas, y se han comparado entre sí.*
- *Se ha seleccionado un modelo de calidad de servicio para una red ad hoc aislada con el objetivo de utilizarlo posteriormente cuando exista una red ad hoc conectada a una red IP fija.*
- *Se han estudiado los modelos de calidad de servicio para redes IP fijas.*
- *Se ha seleccionado un modelo de calidad de servicio para una red IP fija con el objetivo de utilizarlo posteriormente cuando exista una red ad hoc conectada a una red IP fija.*
- *A partir del estudio de los modelos de calidad de servicio en redes ad hoc aisladas se ha desarrollado un nuevo modelo de calidad de servicio para ser aplicado en una red ad hoc conectada a través de un gateway a una red IP fija.*
- *Se han realizado simulaciones para estudiar, comparar el rendimiento del nuevo modelo de calidad de servicio con otros ya existentes y verificar su superioridad.*
- *Se han clasificado y estudiado los protocolos de encaminamiento en redes ad hoc aisladas, y se han comparado entre sí.*
- *Se han analizado propuestas que relacionan el encaminamiento con la disponibilidad de recursos energéticos para los nodos de una red ad hoc aislada.*
- *Se ha propuesto un nuevo protocolo de encaminamiento que cuando encamine tenga en cuenta la conservación de la energía en una red ad hoc aislada.*
- *Se han realizado simulaciones para estudiar y comparar el rendimiento del nuevo protocolo de encaminamiento propuesto con otros ya existentes.*
- *A partir del estudio de los protocolos de encaminamiento que alargan la supervivencia de la red en redes ad hoc aisladas, se ha propuesto un protocolo de encaminamiento que alargue la supervivencia de una red ad hoc interconectada con una red IP fija.*
- *Se han realizado simulaciones para estudiar y comparar el rendimiento del nuevo protocolo de encaminamiento propuesto con otros ya existentes.*

Los dos objetivos presentados anteriormente quedan unidos en uno solo, pues el nuevo protocolo de encaminamiento propuesto para una red ad hoc conectada a una red IP fija contribuye, junto al modelo de calidad de servicio diseñado para tal efecto, no sólo a alargar la supervivencia de la red ad hoc sino también a mejorar la calidad de servicio y por tanto a comunicar en las mejores condiciones posibles ambas redes.

## *1.4 Aportaciones originales de la tesis doctoral*

En esta tesis doctoral se presentan una serie de aportaciones originales:

- *Diseño e implementación del protocolo de encaminamiento SEADSR para la mejora de la supervivencia en redes ad hoc aisladas.*

  Se ha desarrollado un protocolo de encaminamiento que es una mejora del protocolo de encaminamiento DSR y que a la hora de tomar decisiones de encaminamiento tiene en cuenta la capacidad sobrante de las baterías de los terminales en una red ad hoc aislada con el objetivo de alargar la supervivencia de la red en cuestión.

- *Desarrollo y evaluación de un modelo de calidad de servicio para redes ad hoc conectadas a redes IP fijas.*

  Numerosos estudios tratan el problema de cómo intentar proporcionar calidad de servicio en redes ad hoc aisladas. Sin embargo, este tipo de modelos desarrollados resultan insuficientes cuando se trata de proporcionar calidad de servicio entre una red ad hoc y una red basada en infraestructura. De acuerdo con nuestros conocimientos, no se ha desarrollado hasta la fecha ningún modelo de calidad de servicio que permita la interacción y favorezca la cooperación entre una red ad hoc y una red IP fija con el fin de proporcionar calidad de servicio extremo a extremo. En este sentido el trabajo desarrollado en esta tesis doctoral resulta fundamental, pues en él se propone dejar de considerar la calidad de servicio en una red ad hoc y la calidad de servicio en una red IP fija como dos problemáticas independientes. La provisión de calidad de servicio extremo a extremo entre ambas redes debe contemplarse como un objetivo global y no basta con desarrollar modelos de calidad de servicio que traten de diferenciar servicios independientemente en cada una de las redes de forma separada: Con la ayuda del modelo de QoS creado en esta tesis doctoral se demuestra que resulta imprescindible la cooperación e interacción entre los modelos de calidad de servicio de ambas redes para lograrlo. Este trabajo resulta por tanto pionero en estos aspectos y sirve para

abrir una nueva línea de investigación con el fin de promover la comunicación entre redes ad hoc y redes fijas.

- *Diseño e implementación de un protocolo de encaminamiento para la mejora de la supervivencia y la cooperación en el mantenimiento de la calidad de servicio en redes ad hoc conectadas a redes fijas*

  Numerosos estudios tratan el problema de cómo mejorar la supervivencia en redes ad hoc aisladas. Sin embargo, de acuerdo con nuestros conocimientos no se ha desarrollado hasta la fecha ningún protocolo de encaminamiento que trate de mejorar la supervivencia en una red ad hoc interconectada con una red IP fija.

  Numerosos estudios hablan de maximizar el tiempo de vida de una red ad hoc. Sin embargo, según nuestros conocimientos no se ha desarrollado hasta la fecha ningún protocolo de encaminamiento que utilice como retroalimentación para su correcto funcionamiento información relacionada con el mantenimiento de los parámetros de calidad de servicio que le proporciona la red ad hoc. El protocolo de encaminamiento desarrollado no sólo basa su funcionamiento en la utilización de dicha información sino que además la usa no sólo con el objetivo de promover una disminución del consumo de energía de los nodos de la red ad hoc, sino también con la intención de mantener la calidad de servicio entre la red ad hoc y la red IP fija.

## *1.5 Estructura de la tesis doctoral*

La tesis doctoral está estructurada de la forma siguiente:

En el Capítulo 2 se presentan los modelos de calidad de servicio fundamentales que existen para redes ad hoc aisladas. Se introducen no solamente los protocolos que diferencian servicios a nivel de la capa MAC del IEEE 802.11, sino también aquellos modelos de calidad de servicio más destacados que actúan en capas superiores como son los modelos INSIGNIA, FQMM, la arquitectura de Servicios Diferenciados aplicada a redes ad hoc y el modelo SWAN. Finalmente, se realiza un análisis comparativo entre todos ellos con el fin de decidir qué modelo de calidad de servicio para redes ad hoc aisladas se adapta mejor para nuestros propósitos y utilizarlo con posterioridad en el análisis de redes ad hoc interconectadas con redes IP fijas.

En el Capítulo 3 se presentan los protocolos de encaminamiento existentes en redes ad hoc aisladas. Se realizan dos clasificaciones de los protocolos de encaminamiento atendiendo a sus diferentes propiedades. También se presentan ejemplos de

protocolos de encaminamiento pertenecientes a alguna de las clases anteriormente establecidas que resultan ser los más destacados. Se desarrolla una sección que trata del encaminamiento y la disponibilidad de la energía y se pasa a explicar con detenimiento la primera contribución de esta tesis doctoral "Desarrollo del protocolo de encaminamiento SEADSR para la mejora de la supervivencia en redes ad hoc aisladas", procediéndose a realizar una explicación teórica y a presentar simulaciones detalladas de la misma. También se realiza un análisis comparativo entre todos los protocolos de encaminamiento existentes con el fin de decidir qué modelo de calidad de servicio para redes ad hoc aisladas se adapta mejor para nuestros propósitos y utilizarlo con posterioridad en el análisis de redes ad hoc interconectadas con redes IP fijas.

En el Capítulo 4 se analiza la necesidad de la existencia de modelos de calidad de servicio en redes ad hoc interconectadas con redes fijas. Seguidamente, se presenta una nueva contribución "Desarrollo del modelo de calidad de servicio DS-SWAN para redes ad hoc conectadas a redes fijas", procediéndose a realizar una explicación teórica y a presentar los resultados detalladamente mediante simulaciones de la misma.

En el Capítulo 5 se muestran aquellos protocolos de encaminamiento que han sido desarrollados con el fin de poder conectar las redes ad hoc a redes fijas, introduciéndose un mecanismo básico para la interconexión de Internet con la redes ad hoc que basa su funcionamiento en el descubrimiento de gateways. Seguidamente, se presenta una nueva contribución "Desarrollo del protocolo de encaminamiento SD-AODV para la mejora de la supervivencia y la cooperación en el mantenimiento de la calidad de servicio en redes ad hoc conectadas a redes fijas".

En el Capítulo 6 se exponen las conclusiones más relevantes y el Capítulo 7 propone cuáles serían aquellas líneas futuras a tener en cuenta.

## *1.6 Publicaciones relacionadas con la tesis doctoral*

A continuación se describen aquellas publicaciones que ha generado esta tesis doctoral:

### *Congresos nacionales*

[P1] M. C. Domingo, D. Remondo y O. León, "Nuevo protocolo de encaminamiento para la mejora de la supervivencia en redes ad hoc", Jitel 2003, IV Jornadas de

Ingeniería Telemática, Las Palmas de Gran Canaria, Sept. 2003, pp. 361-367, ISBN: 84-96131-38-6.

[P2] M. C. Domingo y D. Remondo, “Diferenciación de servicios por clase en redes inalámbricas”, URSI 2003, XVIII Simposium Nacional de la Unión Científica Internacional de Radio, Coruña, Sept. 2003, ISBN 84-9749-081-9.

# *Congresos internacionales*

[P3] M. C. Domingo, O. León and D. Remondo, “On the Extension of Battery Life with Dynamic Source Routing”, Proceedings of IFIP WG6.7 Workshop and EUNICE Summer School on Adaptable Networks and Teleservices, Eunice'2002, Trondheim, Norway, Sept. 2002, pp. 15-19, ISBN 82-993980-5-3.

[P4] M. C. Domingo and D. Remondo, “Per-Flow Service Differentiation via Virtual MAC”, Proceedings of WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, INRIA Sophia-Antipolis, France, March 2003, pp.319-320, ISBN 2-7261-1238-2.

[P5] M. C. Domingo and D. Remondo, “A New Energy-Aware Routing Protocol for Mobile Ad Hoc Networks”, Proceedings of Med-Hoc-Net 2003, Mahdia, Tunisia, June 2003.

[P6] M. C. Domingo, D. Remondo and O. León, “A Simple Routing Scheme for Improving Ad Hoc Network Survivability”, Proceedings of IEEE Global Telecommunications Conference (IEEE GLOBECOM 2003), San Francisco, USA, Dec. 2003, ISBN 0-7803-7975-6.

[P7] M. C. Domingo and D. Remondo, “State of Art in Multi-Hop Ad Hoc Networks”, EUNICE Summer School on Next Generation Networks, Proceedings of Eunice’2003, Budapest, Hungary, Sept. 2003, pp.219-225, ISBN 963-421-576-9.

[P8] M. C. Domingo and D. Remondo, “An Interaction Model between Ad-hoc Networks and Fixed IP Networks for QoS Support”, Proceedings of the Seventh ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (ACM MSWIM 2004), Venice, Italy, Oct. 2004, pp. 188-194, ISBN 1-58113-953-5.

[P9] M. C. Domingo and D. Remondo, “Analysis of VBR VoIP Traffic for Ad Hoc Connectivity with a Fixed IP Network”, Proceedings of IEEE Vehicular Technology Conference (IEEE VTC 2004-Fall), Los Angeles, USA, Sept. 2004, ISBN 0-7803-8522-5.

[P10] M. C. Domingo and D. Remondo, “A Cooperation Model between Ad Hoc Networks and Fixed Networks for Service Differentiation”, Proceedings of IEEE Wireless Local Networks (IEEE WLN 2004), held in conjunction with LCN, Tampa, Florida, USA, Nov. 2004, pp. 692-693, ISBN 0-7695-2260-2.

## *Revistas internacionales*

[P11] M. C. Domingo and D. Remondo, “An Improved Service Differentiation Scheme for VBR VoIP in Ad-Hoc Networks Connected to Wired Networks”, Service Assurance with Partial and Intermittent Resources (SAPIR 2004), Fortaleza, Brazil, vol. 3126 of Lecture Notes in Computer Science, Berlin, 2004, Springer Verlag, pp. 301-310, ISBN 3-540-22567-6.

[P12] M.C. Domingo and D. Remondo, “Quality of Service Support in Wireless Ad Hoc Networks Connected to Fixed DiffServ Domains”, IFIP TC6 9th International Conference, Personal Wireless Communications (PWC 2004), Delft, The Netherlands, vol. 3260 of Lecture Notes in Computer Science, Berlin, 2004, Springer Verlag, pp.262-271, ISBN 3-540-23162-5.

[P13] M C. Domingo and D. Remondo, “An Interaction Model and Routing Scheme for QoS Support in Ad Hoc Networks Connected to Fixed Networks”, International Workshop on Quality of Future Internet Services (QofIS 2004), Barcelona, Spain, vol. 3266 of Lecture Notes in Computer Science, Berlin, 2004, Springer Verlag, pp. 74-83, ISBN 3-540-23238-9.

[P14] M. C. Domingo and D. Remondo, “An Improved Resource Allocation Scheme for VBR VoIP Support in Ad Hoc Networks Connected to Fixed IP Networks”, Special Issue on “Wireless Ad Hoc and Sensor Networks” of the Journal of Internet Technology (JIT), Vol. 6 (2005), No. 1, January 2005, pp. 93-100, ISSN 1607-9264.

[P15] M. C. Domingo and D. Remondo, “An Interaction Model for QoS Support in Ad Hoc Networks Connected to Fixed IP Networks”, Accepted for publication in the Special Issue on “Mobile Systems, E-commerce and Agent Technology” of the International Journal of Wireless and Mobile Computing (IJWMC).

[P16] M. C. Domingo and D. Remondo, “A Cooperation Model and Routing Protocol for QoS Support in Ad Hoc Networks Connected to Fixed IP Networks”, Accepted for publication in Service Assurance with Partial and Intermittent Resources (SAPIR 2005), Lisbon, Portugal, 2005, IEEE Computer Society.

[P17] M. C. Domingo and D. Remondo, “Interworking of Ad Hoc Networks and Fixed IP Networks for Quality of Service Support”, Submitted for publication in the Special Issue on “Quality of Future Internet Services” of Journal of Computer Communications (COMCOM).

# *Libros*

[P18] M. C. Domingo and D. Remondo, “Analyzing Voice Transmission between Ad Hoc Networks and fixed IP Networks providing end-to-end Quality of Service”, Accepted for publication in “Wireless Networks and Mobile Computing” (edited by Ding-Zhu Du and Guoliang Xue), in Book Series “Network Theory and Applications”, Springer Verlag.

# 2 Modelos de calidad de servicio en redes ad hoc aisladas

La diferenciación de servicios consiste en distinguir entre diferentes clases de tráfico tratando a cada clase de forma distinta de acuerdo con su nivel de prioridad. La diferenciación de servicios resulta fundamental para el correcto funcionamiento de algunas aplicaciones.

Diferentes aplicaciones tendrán distintos requisitos de calidad de servicio; así por ejemplo, habrá aplicaciones que necesitarán garantías de que los datos enviados llegarán a sus destinos (fiabilidad), mientras que otras solicitarán que la información enviada llegue a sus destinos como máximo en un determinado intervalo de tiempo (requisitos temporales). En redes fijas se trata de satisfacer los requisitos solicitados por las aplicaciones y ahora se está intentando poder conseguir también en la medida de lo posible satisfacer las demandas de las distintas aplicaciones en redes ad hoc.

Podría definirse la calidad de servicio, QoS (Quality of Service), como el nivel de servicio que la red ofrece al usuario [55].

La provisión de calidad de servicio persigue básicamente dos propósitos [61]:

- *Mejor conservación de la información por la red.*
- *Mejor aprovechamiento de los recursos de la red.*

Los requisitos que las aplicaciones solicitarán a la red para su correcto funcionamiento están relacionados con los denominados parámetros de calidad de servicio. Los parámetros más destacables de las aplicaciones multimedia son el ancho de banda, el retardo y la variación del retardo o jitter, mientras que en redes ad hoc el tiempo de vida de las baterías será además un parámetro decisivo en aplicaciones diseñadas para la comunicación de distintos dispositivos móviles.

La provisión de calidad de servicio en redes ad hoc resulta ser un problema complejo y a la vez un interesante desafío [79] debido a las propiedades intrínsecas de este tipo de redes, tales como:

- *Capacidad baja y altamente variable de los enlaces inalámbricos.*

  Cuando las ondas se propagan a través del canal radio en el medio inalámbrico se ven perjudicadas por la atenuación, la propagación multicamino y la interferencia debido a la presencia de nodos vecinos o fuentes de radiación electromagnética en las mismas bandas de trabajo. Existirá variabilidad en los enlaces incluso en una red ad hoc con nodos estáticos.

- *Topologías altamente dinámicas donde los enlaces se rompen frecuentemente.*

  Las redes ad hoc pueden incorporar mecanismos de control de admisión diseñados con el fin de comprobar si existen suficientes recursos en los nodos intermedios a lo largo de una ruta entre una fuente y un destino para poder establecer una sesión de tiempo real con la calidad de servicio deseada. Sin embargo, las sesiones con QoS admitidas de esta forma deben restablecerse frecuentemente debido a las roturas de enlaces, con el consecuente riesgo de que ciertas sesiones con requisitos de QoS estrictos no pueden mantener para algunos paquetes los niveles de QoS esperados.

- *Protocolos de acceso al medio.*

  Algunos protocolos de acceso al medio a nivel de la capa MAC aleatorios (como la función de coordinación distribuida, DCF (Distributed Coordination Function) del IEEE 802.11 [10]), no ofrecen la diferenciación de servicios necesaria.

- *Capacidad limitada de recursos*

  Recursos limitados como el espacio de almacenamiento, la capacidad de procesamiento y otros más críticos, como el ancho de banda o el tiempo de vida de las baterías, afectan significativamente a la provisión de calidad de servicio.

Las propiedades comentadas sobre las redes ad hoc indican que no solamente será imposible proporcionar una calidad de servicio 'hard' o 'dura' (estricta) en este tipo de redes, sino que incluso será complicado proporcionar garantías estadísticas en un entorno de alta movilidad debido a la falta de información concisa, instantánea y predecible acerca de los estados de la red [8]. En consecuencia, muchas de las soluciones desarrolladas para redes fijas no son válidas y necesitan ser adaptadas [99].

A la hora de diseñar un modelo de calidad de servicio para una red ad hoc, deben seleccionarse una serie de características [55]:

- *Reserva de recursos mediante 'estado duro' (hard-state) o 'estado suave' (soft-state)*

  Si la reserva de recursos para una sesión con calidad de servicio es de 'estado duro', deben reservarse recursos desde la fuente hacia el destino a través de los nodos intermedios correspondientes y deben liberarse explícitamente cuando la ruta no se encuentra disponible. En cambio, si los recursos se reservan mediante 'estado suave', la reserva se efectúa temporalmente y permanece activa a lo largo del tiempo solamente si está en

uso y se reciben paquetes del flujo antes de que expire el temporizador asociado la reserva.

- *Modelo de calidad de servicio basado en estados o bien sin estados*

  Un modelo de calidad de servicio basado en estados mantiene información de estado local o global, mientras que si se trata de un modelo sin estados no se mantiene este tipo de información en los nodos. Si el modelo de calidad de servicio es sin estados se reduce la carga de la red, aunque también resulta más complicado el poder proporcionar calidad de servicio

- *Calidad de servicio 'dura' o 'suave'*

  Se dice que se ofrece una calidad de servicio 'dura' (hard) cuando se garantizan los requisitos de calidad de servicio de una conexión durante toda la sesión y, en cambio, la calidad de servicio ofrecida es suave si estos requisitos no pueden llegar a garantizarse durante toda la sesión (es decir, la garantía no es estricta sino estadística).

La calidad de servicio que proporciona una red ad hoc no depende de una capa del modelo de referencia ISO/OSI dedicada a este propósito, sino del esfuerzo coordinado de varias capas [56]. La investigación realizada demuestra que si todas las capas en cada nodo de una red ad hoc comparten información e interaccionan entre sí, resulta mucho más eficiente la comunicación y es más fácil que la red pueda adaptarse a las características variables del medio inalámbrico. Este método de interacción entre capas se denomina diseño 'cross layer' y será el método recomendado.

En este capítulo se presentan los modelos de calidad de servicio más destacados para redes ad hoc aisladas, estudiándose en profundidad sus peculiaridades y características. Primeramente se introducen aquellos protocolos que diferencian servicios a nivel de la capa MAC y más tarde se describen otros esquemas de calidad de servicio que actúan también en capas superiores, tales como los modelos INSIGNIA, FQMM, la arquitectura de servicios diferenciados aplicada a redes ad hoc y el protocolo SWAN.

Termina el capítulo con una comparación entre los distintos modelos de calidad de servicio y la elección de aquel modelo de calidad de servicio que resulta ser el más adecuado para redes ad hoc aisladas.

# *2.1 Protocolos a nivel de la capa MAC para diferenciar servicios*

En un canal broadcast inalámbrico, el protocolo a nivel de la capa MAC determina qué estación será la siguiente en transmitir en la contienda entre varios nodos por el acceso al medio.

Es posible distinguir entre mecanismos a nivel de la capa MAC centralizados y distribuidos. Sin embargo, en esta tesis doctoral solamente se desarrollan los mecanismos distribuidos porque los centralizados por definición no resultan adecuados para redes ad hoc. En cambio, los mecanismos a nivel de la capa MAC distribuidos son más flexibles en cuanto a topología y más robustos.

En las siguientes secciones se hace un repaso de las diversas tecnologías existentes para la implementación de redes ad hoc, seleccionándose una de ellas para la provisión de calidad de servicio en este tipo de redes.

## *2.1.1 La capa MAC (Medium Access Control)*

Los más importantes estándares de comunicaciones con capacidad para implementar redes ad hoc son los siguientes:

- *Bluetooth*

  Es una tecnología [9] para radioenlaces diseñada *(Véase la Fig. 2.1)* para que una amplia variedad de ordenadores, periféricos y dispositivos móviles como teléfonos y PDAs (Personal Digital Assistants) puedan establecer comunicación e intercambiar información entre sí a través de enlaces de corto alcance, donde en su implementación básica cada dispositivo tiene un rango de transferencia de hasta 10 m, si bien algunos dispositivos en el mercado tendrán rangos mayores (hasta 100 m) [9]. Bluetooth [23] fue diseñada en un principio como un simple medio de eliminar cables entre dispositivos, aunque más tarde ha sido descubierta su utilidad para la creación de redes personales PANs (Personal Area Networks) e incluso WLANs (Wireless Local Area Networks). Una PAN es una red formada por una gran variedad de dispositivos que se comunican entre sí mediante cables o a través del medio inalámbrico a cortas distancias, que se centra básicamente en una persona y trata de cubrir sus necesidades. En cambio, una WLAN es una red de área local (LAN) que utiliza ondas de radio en lugar de cable para transmitir datos entre dispositivos a distancias intermedias (centenares de metros). Bluetooth

nunca será puramente una WLAN porque estas últimas redes alcanzan una distancia y velocidades mayores, pero sí puede complementar esta tecnología.

**Fig. 2.1.** Ejemplo de aplicación de Bluetooth.

Bluetooth es un estándar abierto que opera en la frecuencia de 2,4 GHz a velocidades de hasta casi 1 Mbps, aunque existe una nueva especificación para operar a 2,1 Mbps y se espera que las próximas versiones alcancen los 10 Mbps.

Hasta ocho dispositivos se pueden comunicar entre sí dentro de lo que se denominaría una piconet, que sería una red formada por dos o más dispositivos Bluetooth compartiendo el mismo canal físico. Dentro de una piconet un dispositivo actuará como máster y los otros harán el papel de esclavos. Varias piconets pueden unirse entre sí para constituir una scatternet.

- *IEEE 802.11*

  Con objeto de conseguir que las *WLANs* fueran ampliamente aceptadas, el IEEE (Institute of Electrical and Electronics Engineers) aprobó en 1997 la especificación IEEE 802.11 [10] como el estándar norteamericano WLAN. La primera versión proporcionaba velocidades de 1 y 2 Mbps, y operaba en las frecuencias de 2,4 GHz. Especificaba la capa de enlace de datos y la capa física. Posteriormente fueron definidas nuevas versiones del estándar:

  - *IEEE 802.11b*: Surgió casi inmediatamente después de la anterior; soporta velocidades hasta 11 Mbps también en la banda de 2,4 GHz. Actualmente es el sistema más extendido.
  - *IEEE 802.11a*: Aprobada a la vez que el IEEE 802.11b; soporta velocidades hasta 54 Mbps en la banda de 5 GHz.
  - *IEEE 802.11g*: Es la extensión del IEEE 802.11b en la banda de los 2,4 GHz que soporta altas velocidades. Es compatible con IEEE 802.11b. Puede operar hasta a 54 Mbps, pero se espera que en la práctica alcance velocidades de unos 24 Mbps.

- *IEEE 802.11e*: Introduce mecanismos de calidad de servicio (QoS) para ofrecer servicios efectivos de Voz sobre IP y servicios de streaming multimedia sobre los estándares a, b y g del IEEE 802.11 *(Véase la sección 2.1.2.1.3.2. IEEE 802.11e o Enhanced DCF (EDCF), pág. 33).*
- *IEEE 802.11h*: Proporciona mejoras del IEEE 802.11a que facilitarán su coexistencia con otros estándares en la banda de 5 GHz como HiperLAN/2 (explicado a continuación).
- *IEEE 802.11i*: Trata de mejorar la seguridad, que se ha convertido en un factor muy importante a medida que las *WLAN*s han ido ganando en popularidad.
- *IEEE 802.11f:* Especifica cómo puede realizarse el traspaso de clientes entre dos puntos de acceso usando el protocolo IAPP (Inter Access Point Protocol).

❖ *HiperLAN*

Es la abreviatura de "High-Performance Radio Local Area Network". HiperLAN es un conjunto de estándares de comunicaciones *WLAN* que ha sido desarrollado por el ETSI (European Telecommunications Standards Institute).

Hay cuatro tipos de HiperLAN:

- HiperLAN/1: Es un estándar [11] que proporciona comunicaciones hasta 20 Mbps en la banda de 5 GHz. Crea redes de datos inalámbricas flexibles, sin la necesidad de que exista una infraestructura.
- HiperLAN/2: Estándar [12] que compite con IEEE 802.11a pues soporta velocidades de hasta 54 Mbps en la banda de 5 GHz. Está definido principalmente para operar en modo infraestructura, donde un punto de acceso controla la red inalámbrica. Esta topología centralizada se combina con la capacidad de comunicación directa de los terminales; a diferencia de otros protocolos centralizados donde dos terminales asociados al mismo punto de acceso no se pueden comunicar directamente entre sí, HiperLAN/2 permite la comunicación directa entre dos dispositivos dentro del mismo rango.
- HiperLAN/3 o HiperAccess: Estándar [13] que trabaja en la banda de los 5 GHz soportando aplicaciones multimedia a una alta velocidad (hasta 25 Mbps).

- HiperLAN/4 o HiperLink: Estándar [14] que trabaja en la banda de los 17 GHz soportando aplicaciones multimedia a muy alta velocidad (hasta 155 Mbps).

❖ *UWB*

Es una nueva tecnología de radio llamada Banda Ultra Ancha, UWB (Ultra Wide Band) [15], [16]. Específicamente, UWB ha sido definida como una tecnología radio con un espectro que ocupa un ancho de banda mayor que el 20 por ciento de la frecuencia central, o como mínimo 500 MHz.

Los sistemas UWB tienen una densidad espectral de potencia extremadamente baja. Resulta posible reutilizar el espectro solapando emisiones UWB de densidades espectrales de potencia extremadamente bajas en bandas espectrales asignadas a otras tecnologías, favoreciendo su aprovechamiento. Con esta tecnología se espera obtener una mayor densidad de comunicaciones inalámbricas que en el caso de otras tecnologías que estuvieran conviviendo en un mismo área porque dichas tecnologías podrían interferirse mientras que con UWB esto no sucede.

De entre las tecnologías de comunicaciones inalámbricas más importantes con capacidad para implementar redes ad hoc, en esta tesis doctoral se ha seleccionado el IEEE 802.11 [17]. Las razones han sido las siguientes:

❖ *La tecnología Bluetooth restringe muchas rutas posibles del encaminamiento en una scatternet.*

Todos los dispositivos situados en una misma piconet han de comunicarse entre sí a través del máster. Por otro lado, la especificación de Bluetooth no incluye ninguna capa de encaminamiento multisalto para scatternet [18]. Los paquetes de datos pueden ser transmitidos desde una piconet a otra a través de un dispositivo compartido, el cual funciona como bridge. El bridge puede actuar como máster en una piconet y como esclavo en la otra, o bien como esclavo en ambas. Como la comunicación entre dispositivos pertenecientes a piconets distintas a de pasar forzosamente a través de los bridges y la comunicación entre dos dispositivos de la misma piconet a de pasar forzosamente a través del máster, se impide la utilización de muchas rutas directas, limitándose de forma considerable el encaminamiento.

❖ *La tecnología HiperLAN todavía no ha sido comercializada*, si bien la ETSI publicó las especificaciones de HiperLAN/2 en mayo del 2001 y la Federal Communications Commission (FCC) [19] adoptó el día 21 de noviembre de

2003 una serie de normas que proponen cuantificar y gestionar la interferencia entre distintos servicios para el caso de HiperLAN.

- *La tecnología UWB todavía no ha sido comercializada*, si bien la FCC [20] estableció en febrero de 2003 unas normas para autorizar el desarrollo de la tecnología UWB.

En cambio, el estándar IEEE 802.11 y en concreto la versión IEEE 802.11b ha sido ampliamente aceptada y está muy extendida. Además, en junio de 2003 se aprobó la especificación IEEE 802.11g como estándar, con la intención de que extendiera la capa física de IEEE 802.11b para poder soportar tasas de hasta 54 Mbit/s en la banda de los 2,4 GHz. Por otro lado, durante la segunda mitad del 2002 empezaron a aparecer en el mercado las primeras familias de productos *WLAN* basados en la tecnología de alta velocidad IEEE 802.11a, que se están comercializando con éxito. Actualmente se está trabajando en el estándar IEEE 802.11h que debe mejorar el IEEE 802.11a añadiendo licencias de regulación de interiores y exteriores en la banda de los 5 GHz para Europa. Otro estándar que ha quedado completamente especificado es el IEEE 802.11f, el cual define el Inter-Access Point Protocol (IAPP), que sirve para hacer compatible el traspaso entre puntos de acceso (access points) de diferentes vendedores.

El estándar IEEE 802.11 define dos modos de operación posibles: La función de coordinación distribuida, DCF (Distributed Coordination Function) y la función de coordinación puntual, PCF (Point Coordination Function) [21], [23]. Ésta última permite proporcionar diferenciación de servicios para soportar aplicaciones de tiempo real, pero resulta bastante ineficiente y compleja de implementar [27]. Asimismo, el modo PCF requiere un punto de control central (punto de acceso), con lo que resulta inviable trabajar en el modo ad hoc, que es distribuido. Por otro lado, cuando el tráfico se envía a ráfagas, el modo DCF resulta más flexible y eficiente. En consecuencia, la mayoría de tarjetas inalámbricas no soportan el protocolo PCF [87]. El modo DCF parece ser más ventajoso que el PCF, pero como no ofrece calidad de servicio, se han propuesto toda una serie de modificaciones del estándar DCF para conseguir que soporte diferenciación de servicios.

En esta tesis doctoral se ha seleccionado por tanto el protocolo DCF para hacer un estudio en profundidad. Veámoslo.

### *2.1.1.1 Protocolo DCF del MAC IEEE 802.11*

El esquema básico de DCF es el protocolo de acceso múltiple por detección de portadora con prevención de colisiones, CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) *(Véase la Fig. 2.2* [22]*)* [21]. El protocolo CSMA/CA consiste en que una estación escucha el canal con el fin de detectar si se está efectuando alguna transmisión y espera un periodo de tiempo denominado intervalo de backoff en el caso de que el canal haya estado ocupado cuando el paquete que debe ser enviado ha llegado a la capa MAC de la estación o bien cuando la estación ha intentado acceder al medio para enviar el paquete. Se pretende evitar las colisiones antes de que sucedan porque en una red ad hoc resulta extremadamente difícil detectar las colisiones en el aire y además resulta más eficiente prevenirlas.

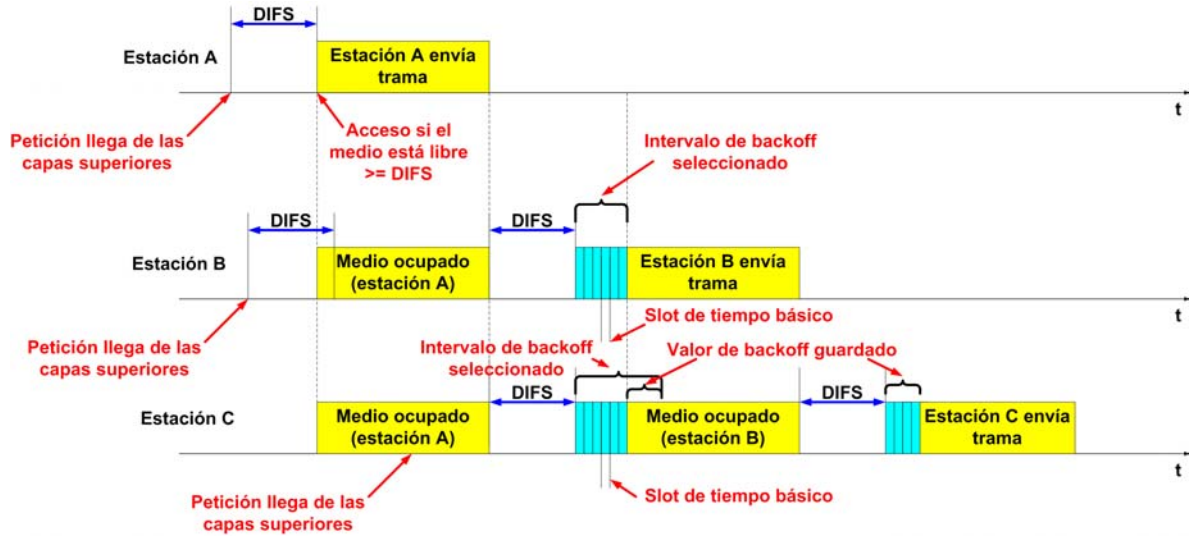



**Fig. 2.2.** Ejemplo de funcionamiento de la función de coordinación distribuida.

Cuando una estación desea enviar una trama [23], primero debe sondear el medio. Si el canal está inactivo durante un intervalo de tiempo igual o mayor que un espacio entre tramas DCF, DIFS (DCF Inter-Frame Space), entonces la estación puede enviar la trama; en caso contrario la estación permanece a la espera de poder transmitir sondeando el medio. En el momento en que dicha estación percibe que el medio vuelve a estar inactivo, espera un DIFS más; si el medio se ocupa durante este intervalo de tiempo, la estación vuelve a quedar a la espera de que el medio se desocupe sondeándolo; sin embargo, si el medio continúa inactivo, la estación pasa a esperar entonces un tiempo aleatorio llamado tiempo de backoff. El intervalo de tiempo de backoff se calcula de la forma siguiente:

$$T_{backoff} = f(0, CW) \times T_{slot}, \tag{2.1}$$

donde $T_{slot}$ representa el slot de tiempo escogido según la capa física, $CW$ simboliza la ventana de contención (Contention Window) y $f(0, CW)$ es un entero pseudoaletorio sacado a partir de una distribución uniforme en el intervalo $[0, CW]$.

El intervalo de backoff se usa para inicializar un temporizador de backoff, el cual es disminuido por cada estación mientras el medio esté inactivo y en cambio se congela si se detecta la transmisión de otra estación. Si el medio permanece inactivo durante un intervalo de backoff, el temporizador de backoff disminuye en una unidad para cada slot de tiempo transcurrido. Cuando el temporizador de backoff expira, la estación accede inmediatamente al medio y transmite la trama. Sin embargo, si el medio queda ocupado antes de que el temporizador de backoff haya expirado, la estación guarda el valor del tiempo de backoff residual (calculado como el intervalo de backoff escogido menos el tiempo de backoff consumido). Entonces, dicha estación debe esperar a que el medio se desocupe, esperar un DIFS y después el tiempo de backoff residual que haya guardado.

Puede suceder que dos o más estaciones comiencen a transmitir en el mismo slot de tiempo y se produzca una colisión. Con el fin de reducir la probabilidad de colisiones consecutivas, una estación dobla su $CW$ después de haber intentado transmitir infructuosamente hasta alcanzar un valor máximo ($CW_{\max}$). Las estaciones que han provocado una colisión entre tramas esperan un DIFS y después vuelven a seleccionar un nuevo intervalo de backoff teniendo en cuenta que el valor de $CW$ se ha doblado.

Existe un número máximo de reintentos de retransmisión para cada trama, de tal forma que si el número de colisiones que ha sufrido una estación al intentar transmitir una trama ha alcanzado este límite, entonces la estación deja automáticamente de intentar la transmisión de esa trama.

Después de una transmisión exitosa, la $CW$ se reinicializa a $CW_{\min}$, el valor inicial de $CW$.

Una vez la estación ha transmitido una trama de datos, deberá esperar que se le envíe un reconocimiento positivo o Acknowledgement (ACK) conforme la trama de datos enviada ha sido recibida correctamente *(Véase la Fig. 2.3).* La estación receptora enviará el ACK a la emisora después de haber estado esperando un intervalo de tiempo SIFS (Short IFS), que tiene una duración menor que un DIFS para que entretanto el resto de estaciones no puedan acceder al medio y se prevenga así una colisión entre tramas.

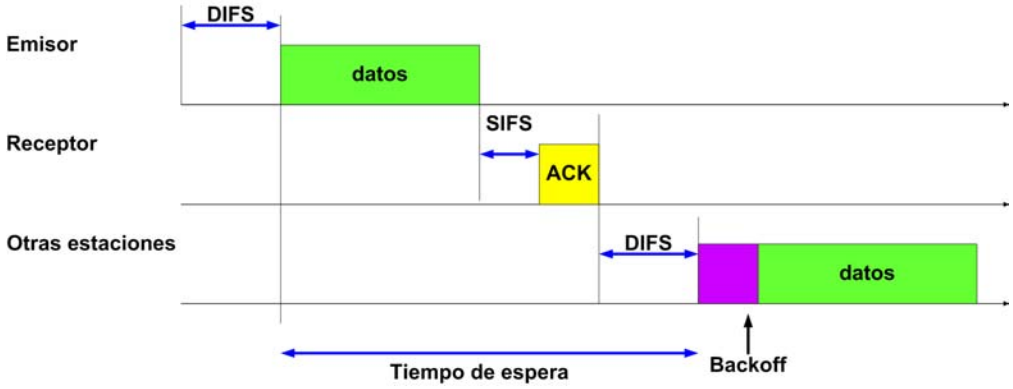



**Fig. 2.3.** Reconocimientos positivos en DCF.

Cuando los nodos de la red ad hoc deseen acceder al medio, pueden encontrarse con alguno de los siguientes problemas, que el protocolo CSMA/CA no es capaz de evitar:

- *Problema del terminal escondido:*

  Surge cuando dos nodos *(Véase la Fig. 2.4)* que se hallan fuera de su alcance radio (el uno respecto al otro) intentan enviar a la vez información al mismo nodo receptor, pudiéndose producir una colisión de los datos que no es detectable por CSMA/CA. Este problema produce una pérdida de eficiencia en cuanto a retardo y ancho de banda. Para poder evitar la colisión, todos los nodos vecinos del receptor deberían tener conocimiento de si el canal se encuentra ocupado. Esto se combate utilizando un protocolo de reconocimiento (handshake) para reservar el canal, mediante el cual el nodo emisor envía una trama RTS (Request To Send) indicando al nodo receptor que desea enviar datos. El nodo receptor puede permitir la comunicación enviando una trama CTS (Clear To Send).

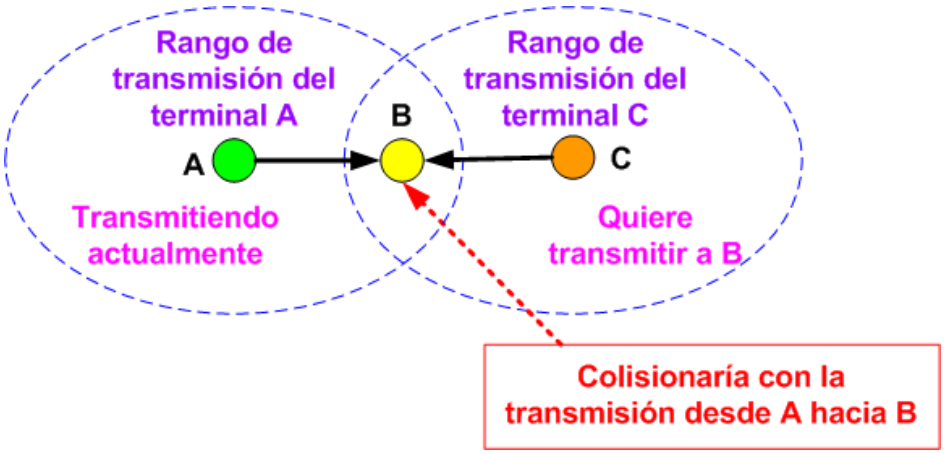



**Fig. 2.4.** Problema del terminal escondido.

Cuando las tramas RTS y CTS se envíen en modo broadcast a todos los nodos vecinos dentro de su alcance radio, los nodos vecinos estarán informados de que el medio estará ocupado y dejarán de transmitir, evitando colisiones.

A pesar de ello, continúa siendo posible que dos estaciones transmitan sus respectivas tramas RTS al mismo tiempo, de forma que colisionen en el nodo receptor, pero ello es menos grave que la colisión de datos.

El análisis que acaba de efectuarse es correcto si se supone que alcance radio y rango de transmisión es lo mismo (sin que haya rango de interferencia), pero en realidad no es así.

Cuando hablamos de alcance radio, deberíamos definir tres rangos distintos (desde el punto de vista del nodo transmisor):

- *Rango de transmisión:*

  Rango dentro del cual un paquete es recibido correctamente.

- *Rango sensible a la portadora:*

  Rango dentro del cual el transmisor comprueba si el medio está libre (poniéndose en modo receptor) y por lo tanto puede ser usado para enviar información.

- *Rango de interferencia:*

  Rango dentro del cual las estaciones receptoras serán interferidas por el transmisor y podrían sufrir una pérdida de datos.

Idealmente, la opción de envío RTS/CTS puede eliminar la mayor parte de las interferencias. Sin embargo, han aparecido estudios [24] que demuestran que la opción de envío RTS/CTS no consigue resolver enteramente el problema del terminal escondido. En teoría, cuando una estación A manda un RTS para poder enviar tramas a una estación B, la estación B enviará un CTS para autorizar la transmisión de tramas y al mismo tiempo cualquier nodo vecino que esté dentro de su rango de transmisión y escuche el medio, al recibir la trama CTS paralizará hasta que sea preciso cualquier intento de transmisión. El problema que puede haber es que un nodo puede no estar dentro del rango de transmisión de la estación B, de forma que no recibirá la trama CTS (por ejemplo, el nodo C en la *Fig. 2.5)*. En cambio, sí que pueden encontrarse dentro del rango de interferencia del receptor (estación B) *(Véase la Fig. 2.5)*, de tal manera que si la estación B está recibiendo un paquete y si el terminal escondido C decide en ese momento iniciar una transmisión, se producirá una colisión.

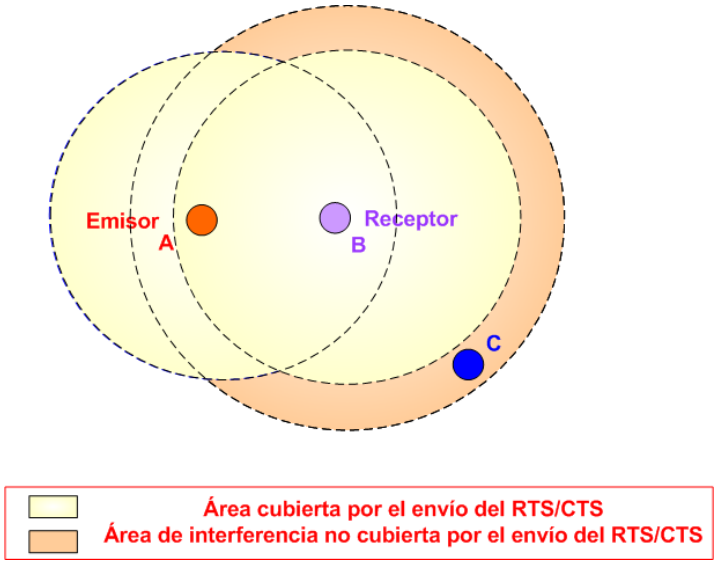



**Fig. 2.5.** Efectividad del protocolo de handshake RTS/CTS.

Para solucionar el problema se ha propuesto [24] que se envíe un CTS únicamente si la potencia de recepción del RTS se halla por encima de un determinado umbral, porque eso significará que el nodo que ha enviado el RTS no está muy distante y así la zona de interferencia será menor (y la calidad del enlace será mejor). El umbral escogido debe ser mayor que el umbral que un nodo requiere para poder recibir con éxito un paquete. El problema de utilizar este mecanismo es que así se reduce el rango de transmisión efectivo.

❖ *Problema del terminal expuesto:*

Es un problema complementario del anterior y sucede cuando una estación B quiere enviar sus tramas a una estación A y al mismo tiempo una estación C decide enviar sus tramas a una estación D *(Véase la Fig. 2.6)*.

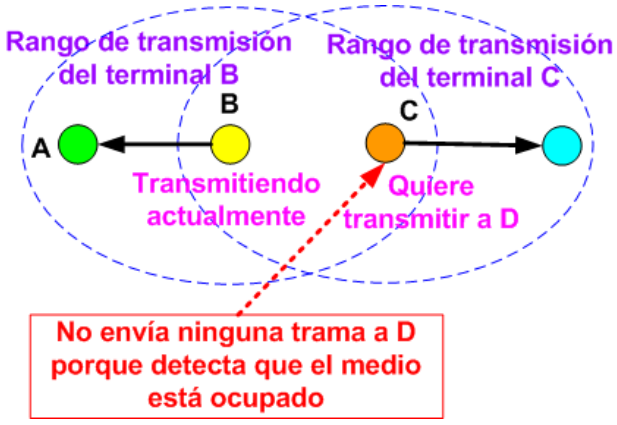



**Fig. 2.6.** Problema del terminal expuesto.

Pero C es un terminal expuesto a B; C escucha el medio y comprueba que está ocupado porque se están enviando tramas de la estación B a la A; entonces C cree que si decide transmitir en esos momentos se producirá una colisión y evita hacerlo. De este modo no se hace efectiva la transmisión de paquetes de la estación C a la D, cuando sí que se podían haber enviado con éxito, siempre y cuando C estuviera fuera del alcance de A. Una solución al problema del terminal expuesto consiste en utilizar antenas directivas [25].

### *2.1.2 Clasificación de los mecanismos de QoS a nivel de la capa MAC para IEEE 802.11*

El protocolo DCF del MAC IEEE 802.11 es simple, fácil de implementar y adecuado para la mayoría de las aplicaciones de datos. Sin embargo, DCF soporta únicamente servicios best-effort, resultando incapaz de ofrecer unas ciertas garantías de calidad de servicio.

Las aplicaciones de tiempo real tales como Voz sobre IP o audio/videoconferencias, requieren ciertos parámetros de calidad de servicio básicos (tales como ancho de banda suficiente, retardos y jitter acotados) para funcionar, si bien pueden tolerar ciertas pérdidas. No obstante, en el modo DCF, todas las estaciones que comparten el mismo canal radio están compitiendo por los mismos recursos sin que existan prioridades [26] o se diferencien servicios. Por lo tanto, no existirá ninguna garantía de que las estaciones con tráfico más prioritario puedan satisfacer sus requisitos de calidad de servicio. Se ha comprobado mediante simulaciones [27] que si se trata de transmitir voz usando el protocolo DCF su calidad se ve degradada y el jitter resulta ser excesivo.

Debido a estas razones diversos autores han propuesto distintas modificaciones del DCF del IEEE 802.11 para introducir diferenciación de servicios.

La *Fig. 2.7* muestra una clasificación jerárquica [28] de los mecanismos de QoS distribuidos a nivel de la capa MAC para IEEE 802.11.

Aunque solamente nos interesarán las WLAN operando mediante el modo ad hoc, esta clasificación también es válida para las WLAN operando en el modo basado en infraestructura.

Se toma el modo DCF del IEEE 802.11 como protocolo distribuido de acceso al medio y se realizan una serie de modificaciones para poder diferenciar servicios que darán lugar a toda una serie de protocolos que se estudiarán en las secciones siguientes.

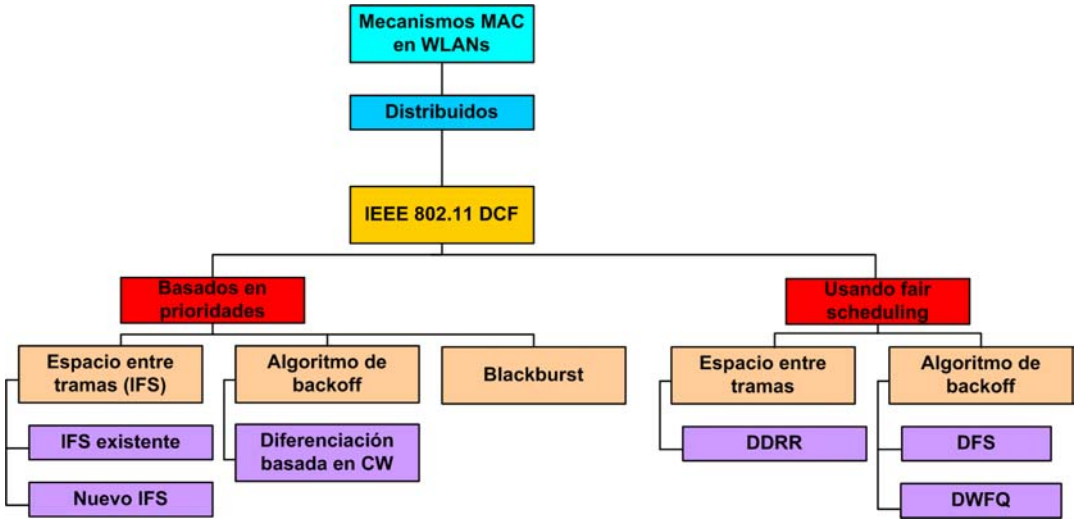



**Fig. 2.7.** Mecanismos de QoS distribuidos a nivel de la capa MAC para IEEE 802.11.

## *2.1.2.1 Soporte a la QoS basado en prioridades*

Algunos mecanismos de QoS definidos proponen diferenciar servicios permitiendo que aquellas clases de tráfico más prioritario puedan acceder al canal con anterioridad. Existen diversas maneras para que una estación pueda acceder rápidamente al medio para enviar una trama *(Véase la Fig. 2.7):*

- *Espacio entre tramas:*

  El espacio entre tramas, IFS (Inter Frame Space) es menor.

- *Algoritmo de backoff:*

  La ventana de contención, CW (Contention Window) es menor y por lo tanto también resulta menor el intervalo de backoff esperado (en media).

- *Usando el protocolo Blackburst (Véase la sección 2.1.2.1.3.5 Blackburst, pág. 39).*

En [29], [30], [31] y [32] se discuten diversos métodos basados en prioridad para proporcionar diferenciación de servicios. La idea básica que subyace detrás de todos ellos es la modificación de varios parámetros del modo DCF del MAC IEEE 802.11 para la obtención de distintas prioridades de tráfico:

- *Función de incremento de Backoff:*

  Cada clase de prioridad puede usar una función de incremento de backoff diferente.

- $CW_{\min}$ *:*

  Cada clase de prioridad tiene un valor mínimo de ventana de contención distinto.

- *IFS:*

  A cada clase de prioridad se le asigna un espacio entre tramas diferente.

- *Longitud de trama máxima:*

  A cada clase de prioridad se le permite transmitir una trama con un tamaño máximo en concreto.

Los mecanismos de soporte a la calidad de servicio basados en prioridades resultan ser injustos (unfair) porque las clases de tráfico más prioritarias acaparan el canal y previenen el acceso del tráfico menos prioritario. A continuación se estudian con mayor profundidad las tres maneras distintas que se han presentado con anterioridad (espacio entre tramas, algoritmo de backoff, protocolo Blackburst) para poder distinguir a nivel de la capa MAC entre diversas prioridades de tráfico.

## *2.1.2.1.1 Espacio entre tramas*

La idea consiste en distinguir entre diferentes prioridades de tráfico asignando un espacio entre tramas menor a aquel tráfico que cuenta con una prioridad mayor. Así se consigue que las tramas de alta prioridad consuman menos tiempo en intentar acceder al medio.

Entre los distintos mecanismos de QoS que modifican el espacio entre tramas para poder ofrecer diferenciación de servicios, distinguiremos los siguientes *(Véase la Fig. 2.7):*

- *Utilización de valores de IFS existentes:*

  Consiste en usar valores de IFS que ya se hallan disponibles a partir del estándar IEEE 802.11 para diferenciar entre el tráfico de alta y de baja prioridad.

  Tal es el caso del esquema Deng [33] *(Véase la sección 2.1.2.1.3.1 El esquema DENG, pág. 32)*, que consigue distinguir entre tráfico de alta y de baja prioridad empleando los intervalos de tiempo PIFS (PCF Inter Frame Space) y DIFS respectivamente (PIFS tiene una duración menor que DIFS). La ventaja de utilizar el esquema Deng es que el tráfico de alta prioridad (voz, vídeo) es capaz de satisfacer sus requisitos de QoS hasta que la carga se hace muy elevada y la desventaja es que aumenta el retardo de acceso al medio y las pérdidas de paquetes en condiciones de mucha carga. Además, las estaciones de baja prioridad generarán tiempos de backoff mayores incluso cuando no existan estaciones de alta prioridad que deseen acceder al medio.

- *Utilización de valores de IFS nuevos:*

EDCF [34] *(Véase la sección 2.1.2.1.3.2 IEEE 802.11e o Enhanced DCF (EDCF), pág. 33)* es un ejemplo de un protocolo que prefiere definir nuevos valores de IFS en lugar de utilizar los ya existentes en el estándar IEEE 802.11. Un problema de este mecanismo es que como el nuevo espacio entre tramas definido AIFS (Arbitration IFS) es más largo (o igual) que un DIFS, una estación que use el modo DCF tradicional obtendrá mayor prioridad (o igual) que aquellas que hayan incorporado el mecanismo EDCF.

Los mecanismos que utilizan diferentes espacios entre tramas para diferenciar servicios, se combinan en muchas ocasiones con el uso de distintos algoritmos de backoff, de tal forma que el efecto de diferenciación puede incrementarse pero también puede eliminarse según sea el cálculo del intervalo de backoff que deba efectuarse. Esto puede llegar a suceder en el caso del esquema Deng cuando existen colisiones.

## *2.1.2.1.2 Algoritmo de backoff*

El algoritmo de backoff puede definirse como un número entero de slots temporales que una estación móvil debe esperar después de un IFS antes de acceder al medio para transmitir si el medio ha estado ocupado anteriormente.

La idea consiste en crear diferentes prioridades de tráfico modificando el algoritmo de backoff. Entre los distintos mecanismos de calidad de servicio que modifican el algoritmo de backoff para poder ofrecer diferenciación de servicios, distinguiremos los siguientes *(Véase la Fig. 2.7):*

- *Diferenciación basada en CW:*

  Si tenemos dos clases de prioridad para el tráfico $i$ y $j$, donde $i < j$, se seleccionan los valores $CW_{\min}$ y $CW_{\max}$ para las tramas de alta y de baja prioridad de forma que $CW_{\min,i} > CW_{\min,j}$ y $CW_{\max,i} > CW_{\max,j}$.

  Se asigna a las tramas de baja prioridad valores de $CW_{\min}$ y $CW_{\max}$ mayores que en el caso de las tramas de alta prioridad porque así tenderán a seleccionar intervalos de backoff mayores en media y así se conseguirá que las tramas de alta prioridad consigan acceder con anterioridad al medio.

  Diversos autores [36], [41] han propuesto varios mecanismos para modificar los valores máximo y mínimo de la ventana de contención.

  En [88], [90] se consigue una buena diferenciación de servicios entre dos clases si se selecciona para el tráfico de alta prioridad una $CW_{\min}$ cuyos valores

estén situados dentro del rango [8,32] y una $CW_{\max}$ de 64, y para el tráfico de baja prioridad una $CW_{\min}$ cuyos valores estén situados dentro del rango [32,128] y una $CW_{\max}$ de 1024. Las simulaciones efectuadas en [87] demuestran que el retardo de los paquetes pertenecientes al tráfico de alta prioridad es claramente inferior al de la otra clase de tráfico, lo cual demuestra que es posible separar servicios de forma efectiva ajustando los intervalos de backoff al modificar los límites de las ventanas de contención.

Una desventaja de la utilización de los algoritmos de backoff es que como el proceso de backoff es exponencial binario, la probabilidad de que una estación tenga que esperar para poder transmitir si su trama ha sufrido una colisión y se ha doblado el valor de $CW$ aumenta en proporción directa con el tiempo que ya ha estado esperando y naturalmente esta propiedad perjudica a las aplicaciones con requisitos temporales como las de tiempo real.

Además, el uso de los algoritmos de backoff favorece que exista una cierta aleatoriedad descontrolada a la hora de acceder al medio y que aumente la variabilidad del throughput y el retardo del tráfico, con lo cual la modificación de dichos algoritmos no siempre resulta la manera más adecuada de conseguir diferenciar servicios.

## *2.1.2.1.3 Ejemplos de mecanismos con soporte a la QoS basados en prioridades*

A continuación se presentan diversos mecanismos concretos con soporte a la QoS basados en prioridades.

## *2.1.2.1.3.1 El esquema DENG*

El esquema DENG [33] está basado en prioridad y utiliza valores de IFS existentes.

Es un método de acceso al medio que consiste en modificar el protocolo CSMA/CA para soportar cuatro clases de prioridad. Se puede proporcionar acceso prioritario al medio inalámbrico utilizando diferentes espacios entre tramas y tamaños de ventana de backoff dependiendo del tipo de servicio asignado al paquete que debe ser enviado. Si dos estaciones desean transmitir una trama [35] y utilizan un espacio entre tramas distinto, a la estación más prioritaria se le habrá asignado probablemente un

espacio entre tramas menor para que pueda transmitir anteriormente con mayor probabilidad.

En este esquema se definen cuatro clases de prioridad que nos permiten establecer dos intervalos de backoff y dos espacios entre tramas distintos *(Véase la Tabla 2.1).*

| ***Prioridad*** | ***IFS*** | ***Algoritmo de backoff*** |
|---|---|---|
| **0** | **DIFS** | $B = \frac{2^{2+i}}{2} + \left\lfloor \rho \times \frac{2^{2+i}}{2} \right\rfloor$ |
| **1** | **DIFS** | $B = \left\lfloor \rho \times \frac{2^{2+i}}{2} \right\rfloor$ |
| **2** | **PIFS** | $B = \frac{2^{2+i}}{2} + \left\lfloor \rho \times \frac{2^{2+i}}{2} \right\rfloor$ |
| **3** | **PIFS** | $B = \left\lfloor \rho \times \frac{2^{2+i}}{2} \right\rfloor$ |

**Tabla 2.1.** Clases de prioridad Deng. $B$ representa el tiempo de backoff en número de slots temporales, $\rho$ es una variable aleatoria uniformemente distribuida en el intervalo (0,1), $i$ representa el procedimiento de backoff $i$-ésimo para esta trama y $\lfloor x \rfloor$ representa el mayor entero menor o igual a x.

## *2.1.2.1.3.2 IEEE 802.11e o Enhanced DCF (EDCF)*

El IEEE 802.11e o DCF Mejorado (Enhanced DCF) (EDCF) [36] está basado en prioridad, utiliza valores de IFS nuevos y diferenciación basada en CW.

Proporciona acceso diferenciado DCF al medio inalámbrico para ocho clases de tráfico o prioridades de usuario, UPs, (User Priorities). Tal y como se muestra en la *Fig. 2.8*, cada estación tendrá cuatro colas o categorías de acceso, ACs (Access Categories) para poder implementar las ocho clases de tráfico [37] que han sido definidas. Por este motivo, una o más clases de tráfico están asignadas a una misma cola AC *(Véase la Tabla 2.2)*. El número de ACs es inferior al de clases de tráfico para reducir la complejidad (contienda por el acceso al medio), teniendo en cuenta que resulta muy poco probable que ocho aplicaciones quieran transmitir tramas simultáneamente.

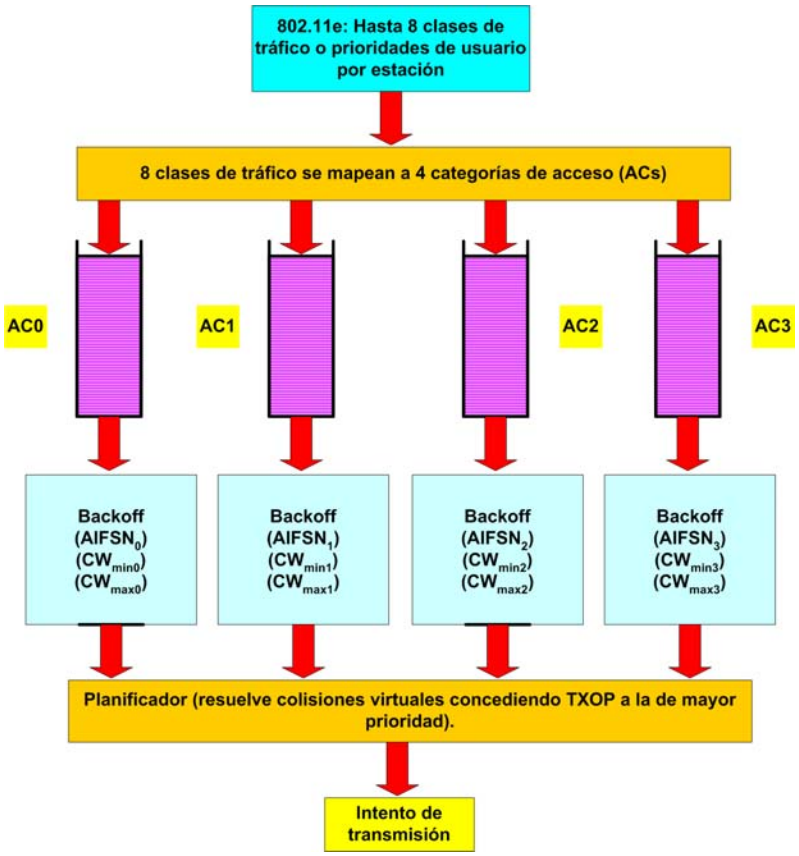



**Fig. 2.8.** EDCF propuesto. TXOP (Transmission Opportunity): Intervalo de tiempo en el que una estación tiene derecho a transmitir, definido mediante un inicio de transmisión y una duración máxima.

| Prioridad de usuario (UP) | 802.11e AC (Access Category) | Tipo de Servicio |
|---|---|---|
| 2 | 0 | Best-effort |
| 1 | 0 | Best-effort |
| 0 | 0 | Best-effort |
| 3 | 1 | Prueba de vídeo |
| 4 | 2 | Vídeo |
| 5 | 2 | Vídeo |
| 6 | 3 | Voz |
| 7 | 3 | Voz |

**Tabla 2.2.** Mapeo entre la clase de tráfico (prioridad de usuario) y la categoría de acceso (AC).

Las ACs usan espacios entre tramas distintos, denominados Arbitration Inter-Frame Spaces (AIFSs). La *Fig. 2.9* muestra una relación temporal del esquema EDCF.

El AIFS [AC] se calcula como [26]:

$$AIFS[AC] = AIFSN[AC] \times T_{slot} + SIFS, \quad (2.2)$$

donde $T_{slot}$ representa el slot de tiempo escogido por la capa física y AIFSN (Arbitration Inter Frame Space Number) toma el valor de 1 o 2. Cuando en la ecuación *(2.2)* AIFSN = 1, las colas de alta prioridad AC1, AC2 y AC3 pasan a tener un valor de AIFS igual a PIFS ( $PIFS = T_{slot} + SIFS$ ). En cambio, si en la ecuación *(2.2)* AIFSN = 2, la cola de baja prioridad AC0 tendrá un valor de AIFS igual a DIFS ( $DIFS = 2xT_{slot} + SIFS$ ). Si llega una trama a una cola de AC vacía y el medio permanece inactivo durante un $AIFS[AC] + T_{slot}$, la trama se transmite inmediatamente. En cambio, si el canal está ocupado, la trama que llegue a la AC deberá de esperar a que el canal se desocupe y después deberá esperar un tiempo $AIFS[AC] + T_{slot}$. Así, la categoría de acceso AC con un valor de AIFS menor tendrá una prioridad mayor.

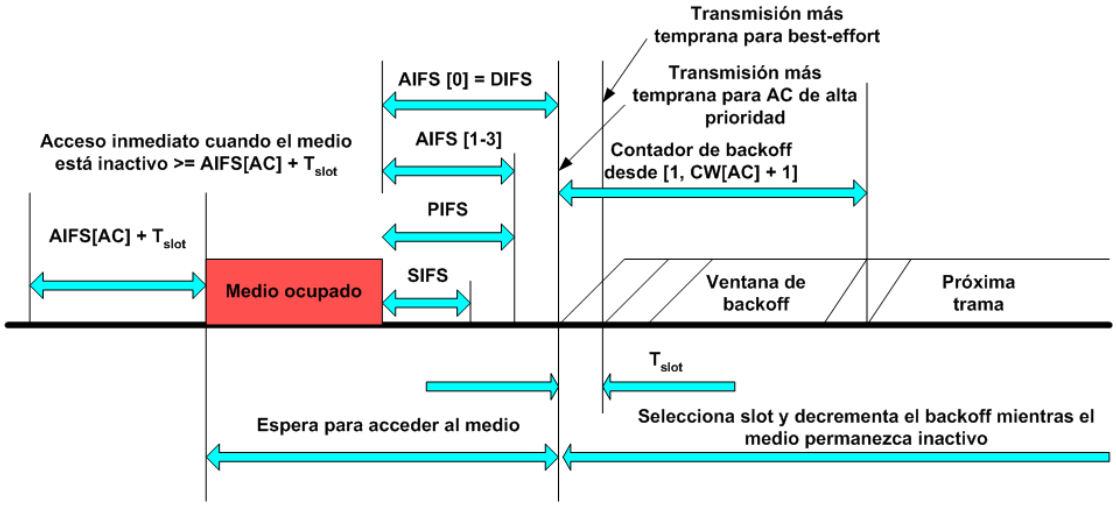



**Fig. 2.9.** Relación temporal para EDCF.

Además, cada AC tiene diferentes tamaños de ventana de contención [38]. Las ACs que tengan valores menores de $CW$ acostumbrarán a disminuir sus intervalos de backoff y en consecuencia tardarán menos tiempo en acceder al medio. Si los intervalos de backoff de dos ACs en una misma estación expiran a la vez, la trama de la cola más prioritaria es la que conseguirá acceder con anterioridad al medio [39]. De esta forma se evita una colisión virtual. La estación con una trama que también hubiera colisionado pero que hasta ahora no ha podido transmitir, doblará su CW y entrará en un proceso de backoff.

Por otro lado, se puede mejorar la eficiencia [40] permitiendo que una estación transmita varios paquetes (packet bursting o ráfagas de paquetes) sin necesidad de

volver a competir por el acceso al medio siempre y cuando no se exceda un tiempo de operación máximo denominado TXOPLimit que no será mayor al tiempo de transmisión de la trama de duración máxima para que el jitter no aumente excesivamente.

## *2.1.2.1.3.3 Adaptative Enhanced DCF (AEDCF)*

El Adaptative Enhanced DCF (AEDCF) [42] está basado en prioridad, utiliza valores de IFS nuevos y diferenciación basada en CW.

Es una modificación del protocolo EDCF que consigue establecer diferentes clases de prioridad para acceder al medio inalámbrico. Después de una transmisión exitosa, el mecanismo EDCF reinicializa la ventana de contención de la clase $i$ correspondiente a $CW_{\min}[i]$ sin tener en cuenta las condiciones de red. Basándose en el hecho de que cuando se produce una colisión es muy probable que en un periodo corto vuelva a suceder otra, AEDCF [43] actualiza el valor de $CW$ más lentamente (no reinicializa la ventana de contención a $CW_{\min}[i]$ después de una transmisión exitosa). Para lograrlo, se calcula primero la tasa de colisión estimada como:

$$f_{act}^{j} = \frac{E(g_j[p])}{E(h_j[p])}, \tag{2.3}$$

donde $j$ se refiere al periodo $j$-ésimo, $E(g_j[p])$ hace referencia al número de colisiones de una estación $p$ durante un periodo $j$ y $E(h_j[p])$ es el número total de paquetes que la estación $p$ ha enviado durante el mismo periodo $j$.

Asimismo, se asigna a cada clase $i$ un factor multiplicativo, $MF$ (Multiplicative Factor) para asegurarse de que existe diferenciación:

$$MF[i] = \min((1 + (i \times 2)) \times f_{media}^{j},\ \alpha), \tag{2.4}$$

donde:

$$f_{media}^{j} = (1 - \alpha) \times f_{act}^{j} + \alpha \times f_{media}^{j-1}, \tag{2.5}$$

donde $\alpha$ representa el factor de estimación suave ($\alpha = 0{,}8$). A la clase de más alta prioridad deberá corresponderle el $MF$ más pequeño.

Entonces, la ventana de contención se actualizará tras una transmisión exitosa de acuerdo con la ecuación siguiente:

$$CW_{nueva}[i] = \max(CW_{\min}[i],\ CW_{antigua}[i] \times MF[i]) \tag{2.6}$$

En AEDCF [26] se introduce un factor de persistencia (Persistence Factor) $PF[i]$ después de cada colisión para acentuar el efecto de diferenciación. Por lo tanto, el cálculo de la nueva ventana de contención después de haberse producido una colisión quedaría como:

$$CW_{nueva}[i] = \min\left(CW_{\max}[i],\ CW_{antigua}[i] \times PF[i]\right) \quad (2.7)$$

Cada clase tendrá un valor de $PF$ diferente, correspondiendo valores de $PF$ pequeños a clases de prioridad alta.

Así AEDCF nos asegura diferentes prioridades ajustando el tamaño de $CW$ de cada clase de tráfico de acuerdo con los requisitos de las aplicaciones y las condiciones de red.

## *2.1.2.1.3.4 Los algoritmos Virtual MAC (VMAC) y Virtual Source (VS)*

Los algoritmos Virtual MAC (VMAC) y Virtual Source (VS) están basados en prioridad y utilizan diferenciación basada en CW.

Como hemos visto, es posible modificar el algoritmo IEEE 802.11 para proporcionar diferenciación de servicios, pero incluso así no se puede garantizar el comportamiento individual de los distintos tipos de tráfico. Por este motivo, el algoritmo denominado Fuente Virtual, VS (Virtual Source) [87], será muy útil en aquellas situaciones donde se necesite estimar si el medio es capaz de soportar nuevas demandas de servicio en un entorno inalámbrico considerando las condiciones locales y la interferencia debido a efectos externos u otras causas.

El algoritmo VS se compone de:

- *El protocolo Aplicación Virtual, VA (Virtual Application):*

  La Virtual Application genera paquetes virtuales de tiempo real como por ejemplo paquetes virtuales de voz a tasa constante. Se les coloca un 'timestamp' o tiempo de generación de los paquetes en su cabecera y son introducidos en un buffer virtual. Entonces los paquetes compiten por el acceso al medio de la misma forma en que lo harían los paquetes reales, pero con la diferencia de que no son transmitidos en realidad.

- *Una cola de interficie:*

  Entre la aplicación y la capa MAC hay una cola de interficie donde las ráfagas de paquetes pueden ser suavizadas.

- *El protocolo MAC Virtual, VMAC (Virtual MAC):*

Con el algoritmo VMAC [88] es posible estimar estadísticas a nivel MAC para los paquetes virtuales que hacen referencia a la QoS tales como el retardo, la variación del retardo, el número de colisiones y las pérdidas. Por ejemplo, midiendo el retardo a nivel de la capa MAC para una conexión virtual puede decidirse si una conexión real en las mismas condiciones puede aceptarse o debe ser rechazada. El cálculo del resto de parámetros sirve para terminar de decidir en virtud de los resultados si será posible o no proporcionar la calidad de servicio que una sesión de tiempo real requiera para ser establecida.

Los parámetros que hacen referencia a la QoS para las diferentes clases de tráfico [89] son estimados mediante la monitorización pasiva del canal, con el fin de no introducir ninguna carga adicional. No obstante, la probabilidad de colisión será estimada. Se considera que ha habido una colisión cuando como mínimo otro terminal utiliza el mismo slot de tiempo para transmitir.

Los paquetes de una aplicación particular sufren el siguiente retardo: Retardo debido a la paquetización (tiempo requerido para llenar un paquete con información de la fuente), retardo causado por el periodo de espera en las colas y retardo a nivel de la capa MAC debido a la contienda por el acceso al medio inalámbrico.

Con la ayuda del Virtual Source es posible estimar el retardo total de los paquetes, mientras que el algoritmo Virtual MAC es el responsable de estimar el retardo a nivel de la capa MAC. Si, por ejemplo, está activada la opción de envío RTS/CTS para reducir el problema del terminal escondido, el retardo *d* de un paquete a nivel de la capa MAC se calcula como [87], [88]:

$$d = t_{espera} + t_{RTS} + t_{CTS} + t_{paquete} + t_{ACK} + 3t_{SIFS} + 3\tau, \tag{2.8}$$

donde $\tau$ es el retardo máximo de propagación y $t_{espera}$ hace referencia al periodo de tiempo que el paquete ha estado esperando desde su llegada a la cola hasta que finalmente el paquete RTS ha podido ser transmitido (incluyéndose tanto el tiempo de backoff e IFS como posibles resoluciones de colisión).

La tasa de bits de datos $p_{tasa\_bits\_datos}$ puede ser calculada como:

$$p_{tamaño} \times p_{tasa} = p_{tasa\_bits\_datos} = const., \tag{2.9}$$

donde $p_{tamaño}$ representa el tamaño del paquete a nivel de la capa de aplicación y $p_{tasa}$ es el tiempo entre llegada de paquetes de la aplicación.

Puede observarse que hay una negociación entre retardo de paquetización y retardo a nivel de la capa MAC porque las tasas de paquete altas (retardo de paquetización

menor) operan con paquetes de datos más pequeños y causan más colisiones, de forma que el retardo MAC será mayor. Por otro lado, tener tasas de paquetes bajas (retardo de paquetización mayor) significa tener paquetes de datos mayores y supone una disminución del retardo a nivel de la capa MAC.

La curva de retardo virtual $d(p_{tasa})$ nos proporciona el retardo medio de los paquetes virtuales generados por el algoritmo VS a una tasa particular de generación de paquetes $p_{tasa}$. Cualquier dispositivo móvil de la red ad-hoc ejecuta el algoritmo VS tomando diferentes valores para $p_{tasa}$, de forma que la curva de retardo pueda ser construida.

Cuando se tienen las curvas del retardo y de la variación del retardo (jitter) en función del tiempo entre llegada de paquetes, un nodo particular ya puede seleccionar un valor óptimo para la tasa de paquetes y otro para el tamaño del paquete con el fin de que el retardo y la variación del retardo (jitter) que la aplicación requiera puedan ser satisfechos.

Se consigue una buena diferenciación de servicios entre dos clases usando los algoritmos VS y VMAC si se selecciona para el tráfico de alta prioridad una $CW_{\min}$ cuyos valores estén situados dentro del rango [8,32] y una $CW_{\max}$ de 64 y para el tráfico de baja prioridad una $CW_{\min}$ cuyos valores estén situados dentro del rango [32,128] y una $CW_{\max}$ de 1024.

## *2.1.2.1.3.5 Blackburst*

Blackburst [44] está basado en prioridad.

Es un esquema MAC distribuido cuyo principal objetivo es el de minimizar el retardo para tráfico de tiempo real. Las estaciones de baja prioridad acceden al medio usando el estándar DCF del MAC IEEE 802.11.

Los nodos que transmiten paquetes de tiempo real (estaciones de alta prioridad) planifican su próximo intento de transmisión cada un cierto número de segundos expresado mediante intervalos de tiempo constantes e iguales denominados $t_{sch}$. Dichos nodos tienen la habilidad de 'atascar' el medio con pulsos de energía llamados black burst. Si un nodo con tráfico de alta prioridad planifica un tiempo de acceso para el momento actual *(Ver Fig. 2.10)* y advierte que el canal ha permanecido inactivo durante un intervalo entre tramas PIFS (PCF Inter Frame Space), el nodo comienza a transmitir una trama [45]. Si el medio está ocupado, el nodo debería esperar hasta que

el canal quedara inactivo, después esperaría durante un PIFS y entonces entraría en un periodo de contención black burst, enviando un black burst para 'atascar' el canal durante un cierto periodo de tiempo. La longitud del black burst se calcula como un determinado número de black slots y depende de cuanto tiempo ha estado esperando la estación para acceder al medio. Después de transmitir el black burst, el nodo espera durante un intervalo de observación para ver si otro nodo transmite un black burst mayor, implicando en este caso que dicho nodo ha tenido que estar esperando durante un periodo de tiempo mayor para acceder al canal y por lo tanto le toca acceder antes. Si al sondear el medio se comprueba que éste está libre, entonces el nodo podrá transmitir su paquete; en caso contrario, la estación esperará a que el canal vuelva a estar inactivo y entrará en un nuevo periodo de contención black burst. Después de una transmisión exitosa de trama, la estación planifica el próximo intento de emisión al cabo de $t_{sch}$. Así, los flujos de tiempo real permanecen sincronizados y tanto el retardo como el jitter para el tráfico de tiempo real disminuyen. La principal desventaja de utilizar este esquema es que requiere para su correcto funcionamiento que el tráfico de alta prioridad acceda al medio de manera regular durante intervalos planificados (tasa constante).

La *Fig. 2.10* ilustra el funcionamiento del esquema Blackburst.

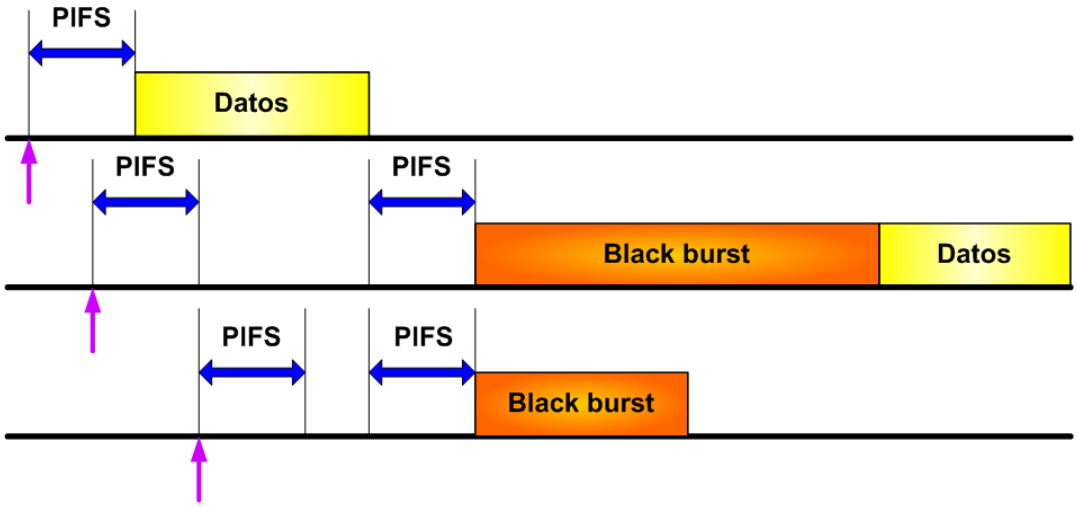



**Fig. 2.10.** Blackburst. Estaciones de alta prioridad intentado acceder al medio cuando éste está ocupado.

## *2.1.2.2 Soporte a la QoS basado en usar una disciplina de servicio justa (fair scheduling)*

Algunos mecanismos de QoS definidos proponen utilizar un encolamiento justo para que las distintas clases puedan acceder al medio. Los algoritmos de planificación justa

tratan de repartir con equidad los recursos de la red entre flujos en proporción a un peso asignado a cada flujo.

Existen dos maneras para que una estación pueda enviar su trama de manera justa *(Véase la Fig. 2.7):*

- *Espacio entre tramas:*

  El espacio entre tramas, IFS, se combina con un encolamiento justo.

- *Algoritmo de backoff:*

  Los algoritmos de backoff se combinan con un encolamiento justo.

Estudiemos con mayor profundidad las dos maneras distintas que existen para poder realizar un encolamiento justo a nivel de la capa MAC, distinguiendo entre diversas prioridades de tráfico.

## *2.1.2.2.1 Espacio entre tramas*

La idea consiste en combinar un encolamiento justo con diferentes espacios entre tramas.

Como mecanismo de este tipo los autores en [46] proponen Distributed Deficit Round Robin (DDRR) *(Véase la sección 2.1.2.2.3.1 Distributed Deficit Round Robin (DDRR), pág. 42)*.

## *2.1.2.2.2 Algoritmo de backoff*

La idea consiste en combinar los algoritmos de backoff con un encolamiento justo.

Entre los distintos mecanismos de QoS que modifican el algoritmo de backoff para poder combinar distintos intervalos de backoff con el encolamiento justo, distinguiremos los siguientes:

- *Distributed Fair Scheduling* [49] *(Véase la sección 2.1.2.2.3.2 Distributed Fair Scheduling (DFS), pág. 43).*

  Este esquema mantiene la $CW$ fija, pero mapea la clase de tráfico al intervalo de backoff.

- *Distributed Weighted Fair Queuing* [53] *(Véase la sección 2.1.2.2.3.3 Distributed Weighted Fair Queuing (DWFQ), pág. 44).*

  Esta técnica mapea la clase de tráfico a la $CW$ .

# *2.1.2.2.3 Ejemplos de mecanismos con soporte a la QoS basados en una disciplina de servicio justa*

A continuación se presentan diversos mecanismos concretos con soporte a la QoS basados en una disciplina de servicio justa.

## *2.1.2.2.3.1 Distributed Deficit Round Robin (DDRR)*

Está basado en una disciplina de servicio justa, utiliza distintos valores de IFS.

El algoritmo Distributed Deficit Round Robin (DRRR) [46] se basa en la disciplina de servicio Deficit Round Robin (DRR) [47] aplicada a la capa MAC del IEEE 802.11 para intentar proporcionar calidad de servicio. Para diferenciar servicios combina una disciplina de servicio justa con la utilización de diferentes espacios entre tramas.

Cada estación divide su tráfico en varias clases en concordancia con los niveles de calidad de servicio deseados. Cada clase de tráfico tendrá unos requisitos de throughput que determinarán cuál será la tasa de servicio cuanto (cuántico) que se le adjudicará. Cada clase de tráfico mantiene un contador de déficit (deficit counter) con los cuantos acumulados y puede transmitir siempre y cuando el valor asignado a este contador sea positivo. Si una clase tuviera por ejemplo un throughput de 100 Kbit/s esto le daría derecho a poder obtener cuantos de 100 Kbit/s. Cuando un paquete se sirve, al 'deficit counter' se le resta el valor del tamaño del paquete. El contador de déficit se mapea a un valor de espacio entre tramas apropiado (si el contador de déficit es grande el IFS asignado será pequeño), tal y como se muestra en la *Fig. 2.11.*

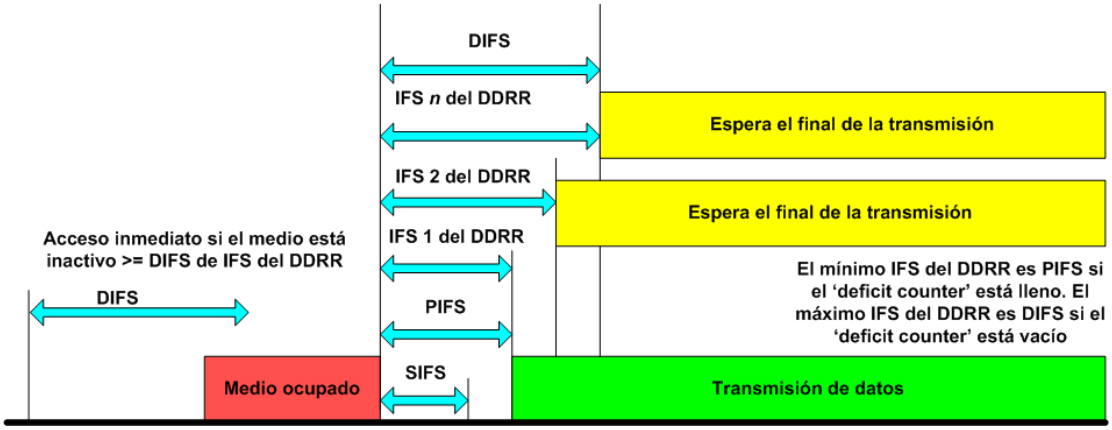



**Fig. 2.11.** El mecanismo DDRR.

Si una estación escucha el medio y está inactivo, espera un IFS y, si el medio continúa estando desocupado, transmite la trama. En cambio, si la segunda vez que escucha el medio, éste está ocupado, la estación espera a que se desocupe

sondeándolo, entonces espera un nuevo IFS y luego transmite. Este esquema no usa el algoritmo de backoff debido a las fluctuaciones de throughput y retardo que su uso aporta en consecuencia. En vez de eso, el valor de IFS resultante se multiplica por un número aleatorio entre 1 y un valor $\beta > 1$ para reducir colisiones entre estaciones con el mismo contador de déficit.

En [48] se proponen mejoras del algoritmo DDRR y se realizan nuevas simulaciones para compararlo con otros algoritmos de encolamiento justo.

## *2.1.2.2.3.2 Distributed Fair Scheduling (DFS)*

El esquema Distributed Fair Scheduling (DFS) está basado en una disciplina de servicio justa y utiliza distintos intervalos de backoff.

Con este esquema se introduce encolamiento justo en el medio inalámbrico [49]. Este mecanismo está basado en aplicar SCFQ (Self-Clocked Fair Queueing) al medio inalámbrico [50]. En DFS, se pone en marcha siempre un proceso de backoff antes de transmitir una trama. El protocolo propuesto asigna ancho de banda en proporción a los pesos de los diferentes tipos de flujos que comparten el canal, usando una función lineal, exponencial o bien la raíz cuadrada para calcular el intervalo de backoff. Dicha función dependerá del tamaño del paquete y del peso del flujo. Si se usa una función lineal, el intervalo de backoff $B$ calculado para las diferentes clases de prioridad será directamente proporcional al tamaño del paquete $L$ dispuesto para ser enviado (justicia) e inversamente proporcional al peso del flujo $P$:

$$B = \left\lfloor \rho \times \left\lfloor \frac{f \times L}{P} \right\rfloor \right\rfloor, \tag{2.10}$$

donde $f$ representa un factor de escalado que se usa para traspasar el intervalo de backoff a una escala adecuada y $\rho$ es una variable aleatoria uniforme en el intervalo [0,9, 1,1] que se introduce para prevenir colisiones entre tramas.

Las *Fig. 2.12* y *Fig. 2.13* representan dos ejemplos de funcionamiento del esquema DFS.

El peso está asociado al throughput, de tal forma que una clase de tráfico de alta prioridad tendrá asignado un peso mayor que una clase de tráfico de baja prioridad para poder enviar un throughput mayor. Así, aquellas estaciones con pesos menores generarán intervalos de backoff mayores que aquellas estaciones con pesos mayores.

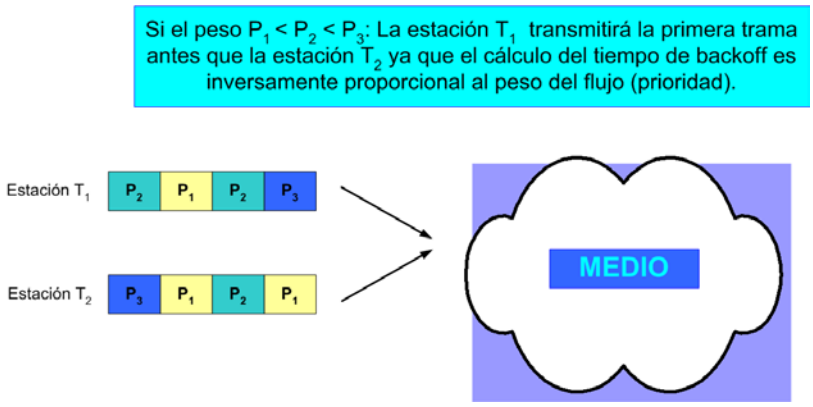



**Fig. 2.12.** Ejemplo 1 de funcionamiento del esquema DFS.

Los flujos con tamaño de paquete menor serán enviados más a menudo gracias a que también se incluye el tamaño del paquete en el cálculo del intervalo de backoff, añadiéndose justicia (fairness) al procedimiento de selección de dicho intervalo. En caso de colisión debe calcularse un nuevo intervalo de backoff usando el algoritmo de backoff original implementado por el protocolo IEEE 802.11 DCF.

El problema de utilizar este protocolo es su complejidad, pues las simulaciones realizadas demuestran que el throughput de una clase de tráfico será extremadamente sensible a la elección de los tamaños de las tramas y los pesos de los flujos, con lo cual será muy complicado mapear el nivel de QoS requerido al peso asociado necesario. Por otro lado, al tratarse de un algoritmo que modifica el intervalo de backoff, añade un periodo de tiempo extra en el acceso de las tramas al medio.

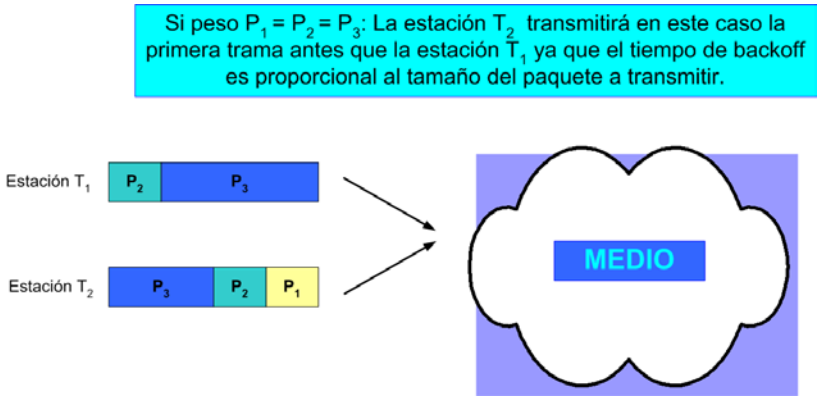



**Fig. 2.13.** Ejemplo 2 de funcionamiento del esquema DFS.

## *2.1.2.2.3.3 Distributed Weighted Fair Queuing (DWFQ)*

El algoritmo DWFQ (Distributed Weighted Fair Queuing) [53] está basado en una disciplina de servicio justa y utiliza diferenciación basada en CW.

Se han propuesto varios algoritmos [51], [52], [53] para poder diferenciar entre distintas clases de tráfico modificando los valores de la ventana de contención (diferenciación basada en $CW$).

ARME (Assured Rate MAC Extension), es una extensión del protocolo MAC de lEEE 802.11b que soporta diferenciación de servicios. Existen dos tipos de servicio (Assured Rate Service y best-effort) que acceden al canal con el modo DCF pero utilizando distintas ventanas de contención ($CWs$) [51]. Las estaciones con tráfico best-effort usan la ventana de contención definida por el estándar IEEE 802.11. En cambio, el cálculo de la $CW$ para las estaciones con tráfico Assured Rate es dinámico y se modifica para alcanzar el throughput deseado. El tamaño de la ventana de contención es inversamente proporcional al throughput. El valor de la ventana de contención disminuirá si el throughput estimado es menor que el deseado, mientras que en el caso contrario el valor de $CW$ aumenta ligeramente. De esta forma cambia la prioridad de una estación y también su throughput a la hora de enviar sus datos al medio. DIME (DiffServ MAC Extension) es una extensión para ARME donde se considera también el tráfico Expedited Forwarding y reutiliza el espacio entre tramas de la función de coordinación puntual de forma distribuida [52].

Los mismos autores proponen usar el algoritmo DWFQ (Distributed Weighted Fair Queuing) [53]. En el modo DCF, el ancho de banda que recibe un flujo depende de su ventana de contención: Cuanto menor sea la $CW$, mayor será su throughput. En DWFQ se pretende asignar a cada flujo un ancho de banda proporcional a su peso. Para ello se calcula para cada flujo de una estación un cociente $L_i = R_i / W_i$, donde $R_i$ hace referencia al throughput estimado para el flujo $i$ y $W_i$ hace referencia a su peso. El cociente de un flujo se incluye en la cabecera de cada paquete que una estación envía al medio. Una estación que reciba un paquete proveniente de otra, comparará el cociente $L_i$ incluido en la cabecera del paquete recibido con el del flujo que esté enviando y actuará sobre su $CW$ aumentándola o disminuyéndola según sea mayor o menor el $L_i$ de su flujo en comparación con el del resto de estaciones. Los pesos sirven para que exista una cierta diferenciación entre clases y anchos de banda asignados. Sin embargo, el proceso de modificación de la ventana de contención implica necesariamente aumentar la variabilidad del throughput y del retardo.

Cada nodo $i$ puede mantener localmente varios flujos con pesos $W_1,...,W_n$ en colas separadas y con la ayuda de un planificador weighted fair queuing debe escogerse atendiendo al peso de cada flujo el orden en que los paquetes serán transmitidos. Así

se distribuye el ancho de banda del nodo entre sus flujos proporcionalmente a sus pesos.

### *2.1.3 Comparación entre los diversos mecanismos de QoS para IEEE 802.11*

Los mecanismos de calidad de servicio a nivel de la capa MAC para IEEE 802.11 descritos se basan en esquemas que han sido adaptados para mapear los requisitos de QoS utilizados en redes fijas a parámetros de la capa MAC del IEEE 802.11.

Estos mecanismos propuestos no son capaces de garantizar los niveles de calidad de servicio deseados debido a las condiciones dinámicas de la red ad hoc, pero sí que proporcionan una cierta diferenciación de servicios.

Los mecanismos de calidad de servicio basados en prioridades no son justos en la gran mayoría de las ocasiones y debe recurrirse a esquemas de encolamiento justo para satisfacer este criterio. Además, introducen tiempos de espera adicionales al modificar los espacios entre tramas y los intervalos de backoff. La elección de intervalos de backoff aleatorios incrementa también la variabilidad del throughput y del retardo.

En la *Tabla 2.3* se comparan los diferentes esquemas de soporte a la QoS existentes de acuerdo con los siguientes criterios:

- *Justicia:*

  Correlación existente entre la prioridad asignada (peso o tasa de cuanto) y el throughput medido en una estación.

- *Carga de señalización:*

  Se cuantifica comprobando cuál es el número de mensajes intercambiados y también mirando si es necesario incluir algún campo adicional en la cabecera a nivel de la capa MAC para enviar los paquetes o midiendo el tiempo medio de espera requerido antes de poder transmitir una trama.

- *Complejidad:*

  Hace referencia a los requisitos de cálculo que requiera un mecanismo con soporte a la QoS y la necesidad de intercambio de mensajes.

- *Throughput agregado*
- *Variación del throughput*
- *Retardo medio en el acceso al canal*

| Mecanismo de soporte a la calidad y de servicio | Esquemas basados en prioridad | | | Uso de una disciplina de servicio justa | | |
|---|---|---|---|---|---|---|
| | IFS | DIFERENCIACIÓN BASADA EN CW | BLACKBURST | DDRR | DFS | DWFQ |
| **Justicia** | Baja | Baja | Baja | Alta | Alta | Media |
| **Señalización** | Baja | Baja | Baja | Baja | Media | Alta |
| **Complejidad** | Baja | Baja | Baja | Baja | Media | Alta |
| **Throughput** | Medio | Medio | Alto | Alto | Alto | Medio |
| **Variación del throughput** | Alta | Media | Media | Baja | Baja | Alta |
| **Retardo de acceso** | Alto | Alto | Medio | Bajo | Alto | Alto |

**Tabla 2.3.** Comparación de diversos mecanismos de soporte a la QoS para WLANs.

## *2.2 El modelo INSIGNIA*

INSIGNIA [54] es un modelo de calidad de servicio distribuido específicamente diseñado para ser aplicado en redes ad hoc móviles, donde tanto la topología de red como la conectividad entre nodos y las condiciones del canal radio varían dinámicamente con el tiempo. INSIGNIA [55] distingue entre aplicaciones best-effort, aquellas que requieren una calidad de servicio básica (base QoS) y las que necesitan para su correcto funcionamiento de una calidad de servicio mejorada (enhanced QoS, como por ejemplo el vídeo). Algunas aplicaciones de QoS mejorada requieren un ancho de banda máximo pero son capaces de adaptarse a unas reservas de ancho de banda menores, por debajo de las cuales resultan inoperativas. Otras aplicaciones de QoS mejorada no pueden funcionar con reservas de ancho de banda inferiores a las requeridas normalmente y si no existen recursos suficientes en la red terminan viéndose degradadas a best-effort [58]. INSIGNIA trata de proporcionar diferenciación de servicios extremo a extremo desde la fuente al destino de un flujo de datos, reservando una serie de recursos para el tráfico de tiempo real mediante señalización en banda (in-band). El término señalización 'in-band' significa que la información de control es transportada junto a los datos dentro de los paquetes. En entornos altamente dinámicos como es el caso de las redes ad hoc, esta señalización resulta mucho más conveniente que la señalización fuera de banda (out-of-band), en la cual los paquetes de datos y de control viajan separadamente por la red, usualmente a través de distintos canales. El motivo es que en los sistemas de señalización 'out-of-

band' (tales como el protocolo RSVP) cuando se producen cambios en la topología de red, los nodos intermedios a lo largo de la ruta antigua son avisados explícitamente para que puedan liberar aquellos recursos que tenían reservados; pero si dichos nodos quedan fuera del alcance radio resulta imposible la liberación de recursos. La señalización de INSIGNIA [56] es considerada como el primer protocolo de señalización con QoS específicamente diseñado para la reserva de recursos en una red ad hoc.

La reserva de recursos para las sesiones de tiempo real se realiza manteniendo un sistema de estados por flujo, donde las sesiones pueden ser establecidas, restauradas, adaptarse a circunstancias cambiantes de la red y también pueden ser rechazadas. Este sistema de estados por flujo se denomina 'suave' (soft-state) en contraposición a los sistemas de estados 'duros' (hard-states) de las redes fijas, donde se mantiene tanto la ruta como la reserva de recursos entre la fuente y el destino durante toda la sesión. Por el contrario, en los sistemas de estados 'suaves' cuando un paquete de datos llega a un nodo donde no se ha producido una reserva de recursos, se pone en marcha un mecanismo denominado control de admisión para realizar la reserva y una vez dicha reserva ha sido admitida comienza a funcionar un temporizador de estado suave para el flujo establecido. Los sucesivos paquetes de datos que atraviesen este nodo sirven para reinicializar el temporizador de estado suave y de esta forma mantener la reserva. Si no se recibe ningún paquete de datos durante un cierto intervalo de tiempo (el de la duración del temporizador), entonces el temporizador de ese nodo expira y los recursos reservados para ese flujo pasan entonces a ser liberados.

Otra característica fundamental de este modelo de calidad de servicio es que separa el encaminamiento de la señalización y el envío de paquetes.

Los componentes del modelo de calidad de servicio INSIGNIA se ilustran en la *Fig. 2.14* y pasan a comentarse a continuación [57]:

- *Módulo de envío de paquetes:*

  Es el encargado de clasificar los paquetes entrantes para reenviarlos al módulo apropiado:

  - *Los mensajes de encaminamiento se envían al módulo relacionado con el protocolo de encaminamiento.*
  - *Los mensajes de señalización se procesan mediante el módulo de señalización de INSIGNIA.*
  - *Los paquetes de datos pueden entregarse localmente o bien reenviarse al planificador que usará una disciplina de servicio para*

*que sean retransmitidos hacia el siguiente salto en dirección a su destino.*

- *Módulo del protocolo de encaminamiento:*

  Es capaz de cambiar las rutas en consonancia con los cambios de topología que se produzcan en la red ad hoc. Refleja todos los cambios que se producen en la tabla de encaminamiento, para que los módulos de señalización y de envío de paquetes estén al corriente. Permite utilizar correctamente tanto protocolos de encaminamiento proactivos como reactivos.

- *Señalización 'in-band':*

  Gracias a este tipo de señalización es posible establecer, restaurar, adaptar y liberar con rapidez las sesiones de tiempo real.

- *Control de admisión:*

  Se encarga de realizar la reserva de ancho de banda para un flujo determinado. Una vez se ha efectuado una reserva debe reavivarse periódicamente utilizando un mecanismo de estados 'suave' durante el envío de paquetes de datos.

- *Disciplina de servicio de paquetes:*

  Permite la utilización de una gran variedad de disciplinas de servicio para gestionar el envío de paquetes.

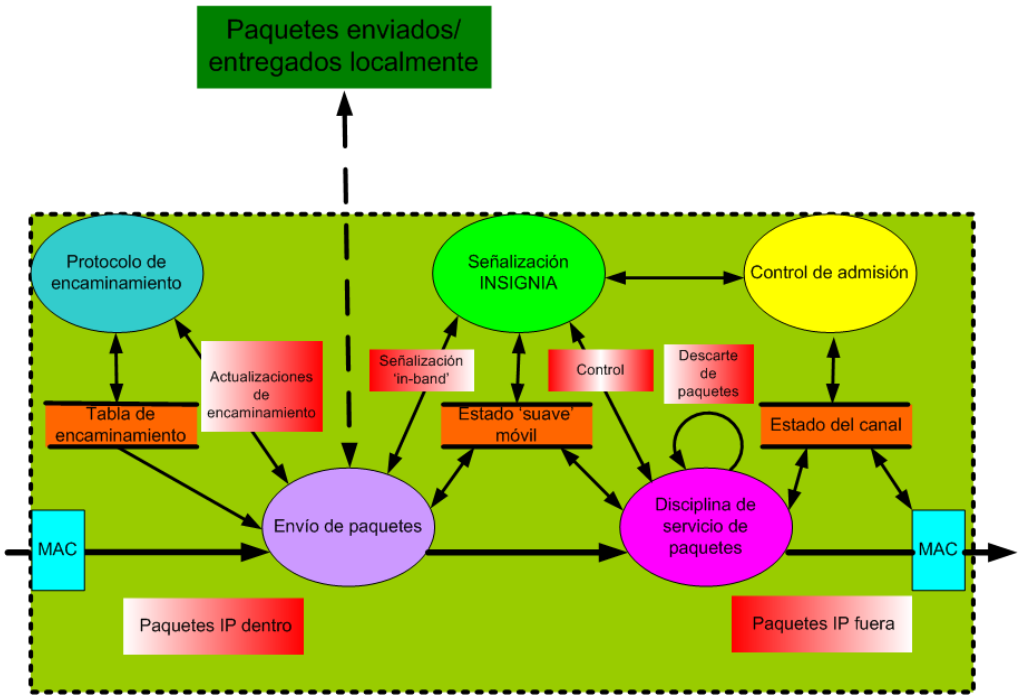



**Fig. 2.14.** Modelo de gestión de flujos INSIGNIA en un nodo/router móvil.

❖ *Capa MAC (Medium Access Control):*

Proporciona acceso al medio compartido con calidad de servicio para tráfico de tiempo real y servicios best-effort. INSIGNIA ha sido diseñado para que fuera compatible con cualquier protocolo de acceso al medio, pudiendo operar sobre distintas tecnologías de la capa de enlace de datos.

INSIGNIA [58] fue diseñado para que pudiera adaptarse a las condiciones dinámicas de la red, consumiendo muy poco ancho de banda en la señalización. Para lograrlo, este sistema soporta una serie de comandos o instrucciones denominados 'reserva rápida', 'restauración rápida' y 'adaptación'. Estos comandos se codifican en el campo de opciones de la cabecera IP, el cual está compuesto a su vez por los campos siguientes *(Véase la Fig. 2.15)*:

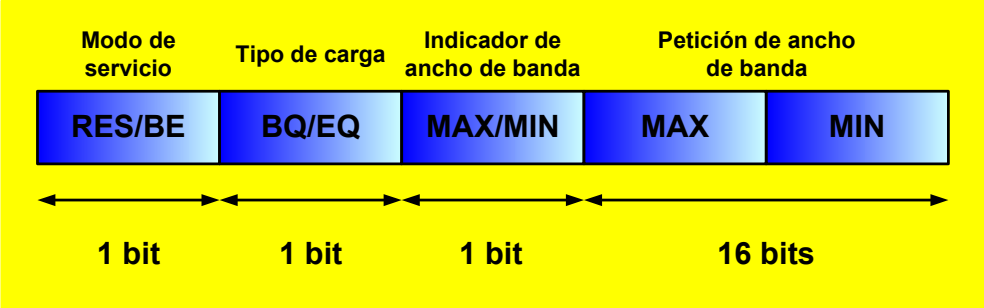



**Fig. 2.15.** El campo de opciones de IP INSIGNIA.

Modo de servicio, tipo de carga, indicador de ancho de banda y petición de ancho de banda.

A continuación paso a explicar cada uno de los distintos comandos del protocolo:

❖ *Reserva rápida* [59]

La fuente modifica los valores del campo opciones en la cabecera IP de los paquetes de petición de reserva, poniendo el modo de servicio a RES (reservation), el tipo de carga a calidad de servicio básico, BQ (base QoS) o bien calidad de servicio mejorada, EQ (enhanced QoS), el indicador de ancho de banda a MAX/MIN (según sea máximo (para una calidad de servicio mejorada o EQ) o mínimo (para una calidad de servicio básica o BQ) el número de recursos que se desean reservar) y la petición de ancho de banda debe solicitar unos requisitos de ancho de banda válidos.

A continuación, los paquetes de petición de reserva serán enviados a los nodos intermedios a lo largo de la ruta, los cuales se encargarán de ejecutar el control de admisión, reservando recursos y estableciendo un estado por flujo en cada nodo intermedio situado entre la fuente y el destino. El nodo fuente continuará enviando paquetes de petición de reserva hasta que el

nodo destino complete esta fase enviándole información acerca del estado por flujo de la reserva incluida dentro del denominado informe de calidad de servicio (QoS report).

La *Fig. 2.16* muestra un ejemplo de establecimiento de reserva para un flujo entre una fuente $M_s$ y un destino $M_D$. Se envían paquetes con tipo de carga EQ (altos requisitos de calidad de servicio) o BQ (requisitos de calidad de servicio básicos) para reservar recursos a lo largo de la ruta y finalmente se consigue completar la fase estableciendo una reserva de recursos máxima para poder enviar todos los paquetes del flujo.

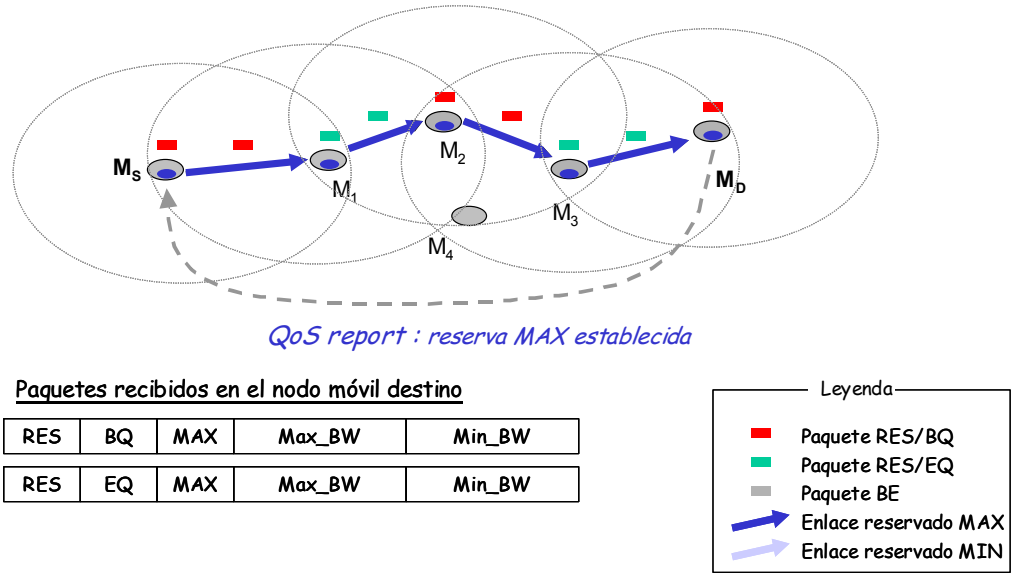



**Fig. 2.16.** Establecimiento de una reserva.

La *Fig. 2.17* muestra una petición de reserva rápida para un flujo adaptativo. El nodo fuente $M_s$ envía una petición de reserva de recursos máxima al primer nodo intermedio a lo largo de la ruta ($M_1$), el cuál ejecuta el control de admisión y reserva los recursos solicitados (si se encuentran disponibles), procediendo más tarde a reenviar el paquete de petición de reserva al siguiente nodo intermedio ($M_2$). Este procedimiento se repite salto a salto hasta que por fin el paquete de petición de reserva alcanza el nodo $M_D$ de destino. Para poder comprobar el estado de la reserva por flujo el nodo destino verifica los valores de los campos 'modo de servicio', 'tipo de carga' e 'indicador de ancho de banda' del paquete recibido. En el ejemplo que nos ocupa observamos que el nodo $M_2$ actúa como nodo con ancho de banda restrictivo o 'cuello de botella', de forma que sólo es posible reservar un ancho de banda mínimo entre los nodos $M_2$ y $M_3$. El nodo destino recibe un paquete de petición de reserva y comprueba los valores del campo de opciones de IP.

Si el indicador de ancho de banda tiene un valor MAX, esto significa que ha sido exitosa la reserva de recursos entre la fuente y el destino para poder proporcionar una calidad de servicio básica y mejorada. En cambio, si al indicador de ancho de banda se le ha asignado un valor de MIN, esto indica que solamente será posible ofrecer una calidad de servicio básica a lo largo de la ruta. En nuestro caso lo que ocurre es que el nodo $M_2$ (que es el 'cuello de botella') se ve incapaz de proporcionar una calidad de servicio mejorada debido a la falta de recursos y cambia el valor del campo 'indicador de ancho de banda' del paquete de reserva recibido a MIN, así como el modo de servicio de los paquetes con tipo de carga EQ (calidad de servicio mejorada) de RES a BE (Best-effort). Al destino le basta con recibir un único paquete de petición de reserva para completar la fase de reserva de un flujo. Por lo tanto, el resultado final será que se ha realizado una reserva máxima de recursos entre la fuente y el nodo con ancho de banda más restrictivo (cuello de botella) (entre $M_S$ y $M_2$ en la *Fig. 2.17*), mientras que no ha sido posible reservar recursos entre el nodo que actúa como cuello de botella y el destino.

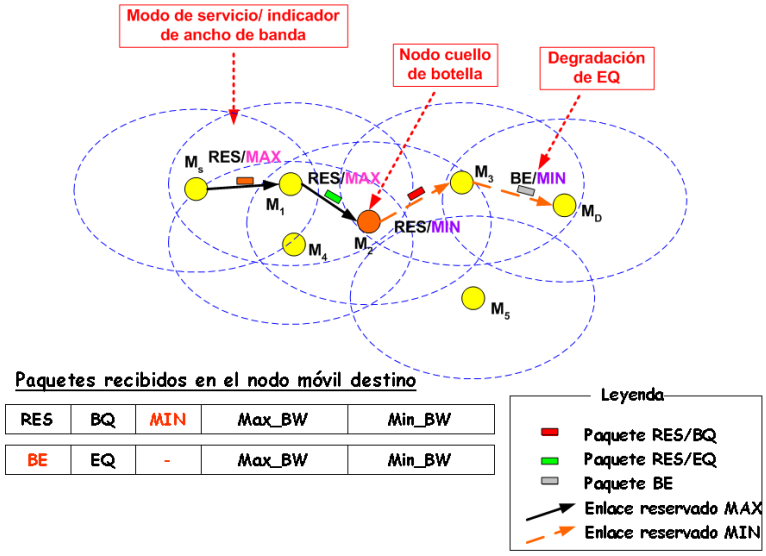



**Fig. 2.17.** Reserva rápida.

Por lo tanto, únicamente ha tenido lugar una 'reserva parcial de recursos', entendiéndose que entre $M_S$ y $M_2$, la reserva de recursos ha sido máxima, mientras que entre $M_2$ y $M_D$ se ha producido una reserva de recursos mínima que no sirve para satisfacer los elevados requisitos de paquetes con tipo de carga EQ, que se van degradados a best-effort. El nodo destino $M_D$ informará a la fuente acerca del estado de la reserva enviando un informe de calidad de servicio. La recepción de un paquete BE/EQ/MIN o RES/BQ/MIN indica que los paquetes de calidad de servicio han sido degradados. En concreto, los

paquetes de calidad de servicio mejorada han sido degradados a best-effort. Nótese que los paquetes best-effort no requieren que se realice una reserva de recursos para ellos. Cuando la fuente reciba dicho informe y compruebe que la reserva de recursos ha sido solamente parcial, podrá decidir anularla enviando paquetes con el tipo de carga EQ pero un modo de servicio BE. Así se anularía la reserva de recursos entre los nodos $M_S$ y $M_2$.

❖ *Restauración rápida* [60]

Una vez el control de admisión ha sido aceptado y se han reservado recursos para un flujo, se pone en marcha un temporizador en cada nodo intermedio a lo largo de la ruta entre la fuente y el destino que estará asociado con el estado de cada flujo y se actualizará, inicializándose cada vez que un nodo reciba algún paquete perteneciente a dicho flujo. Si pasa el tiempo y no se reciben paquetes pertenecientes a dicho flujo, el temporizador de cada nodo expirará y automáticamente se liberarán los recursos asignados. Esto es lo que termina sucediendo en la *Fig. 2.18*, donde el nodo $M_2$ cambia su posición y es necesario establecer una nueva ruta para enviar los paquetes desde el nodo fuente $M_S$ al nodo destino $M_D$ a través de los nodos intermedios $M_1$-$M_4$-$M_3$. Como no se enviarán más paquetes de datos a $M_2$, éste terminará por liberar los recursos que tenía asociados para el flujo que ha sido preciso volver a encaminar.

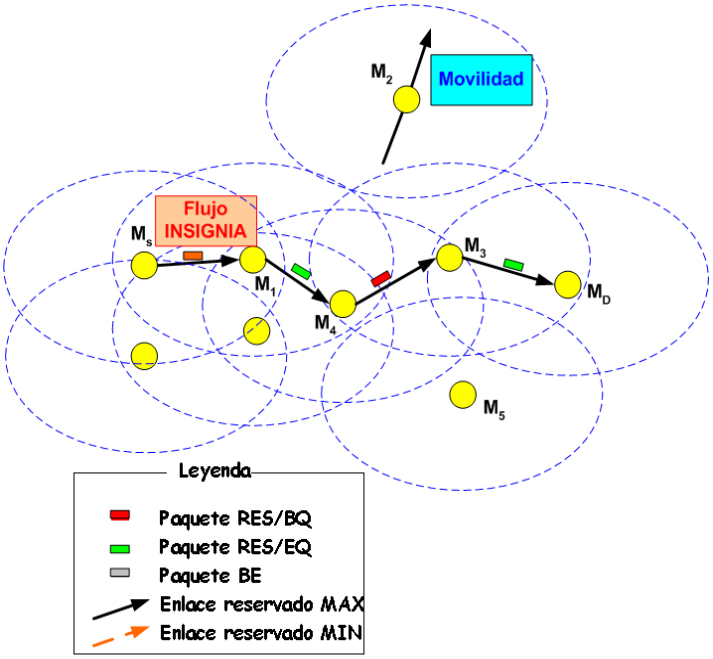



**Fig. 2.18.** Reencaminamiento del flujo de datos y restauración rápida.

En muchas ocasiones es necesario poner en marcha el proceso denominado 'restauración rápida' para poder restablecer las reservas de

recursos de aquellos flujos para los cuales ha sido preciso buscar una nueva ruta debido a la movilidad de alguno o varios de sus nodos intermedios. Es preciso, por tanto, que el algoritmo de encaminamiento encuentre una nueva ruta y que los nuevos nodos intermedios a lo largo de dicha ruta ejecuten el control de admisión, reservando recursos para el flujo. La restauración se denominará 'inmediata' en el caso de que el protocolo de encaminamiento ya disponga en su caché de una ruta nueva para enviar paquetes de la fuente al destino, con lo cual el flujo reconducido recupera inmediatamente el estado de su reserva original; esto significa que un flujo en modo de reserva máximo o mínimo se restaura al momento a su modo de reserva anterior (máximo o mínimo, según fuera el caso). Si no hay ninguna ruta almacenada en la caché del nodo, el protocolo de encaminamiento pondrá en marcha un procedimiento de Descubrimiento de Ruta y el tiempo de restauración del flujo estará entonces relacionado con la velocidad a la cual el protocolo de encaminamiento es capaz de descubrir una ruta.

Un ejemplo de restauración rápida sería el de la *Fig. 2.18*, donde el nodo $M_2$ queda fuera del alcance radio de sus nodos vecinos debido a la movilidad.

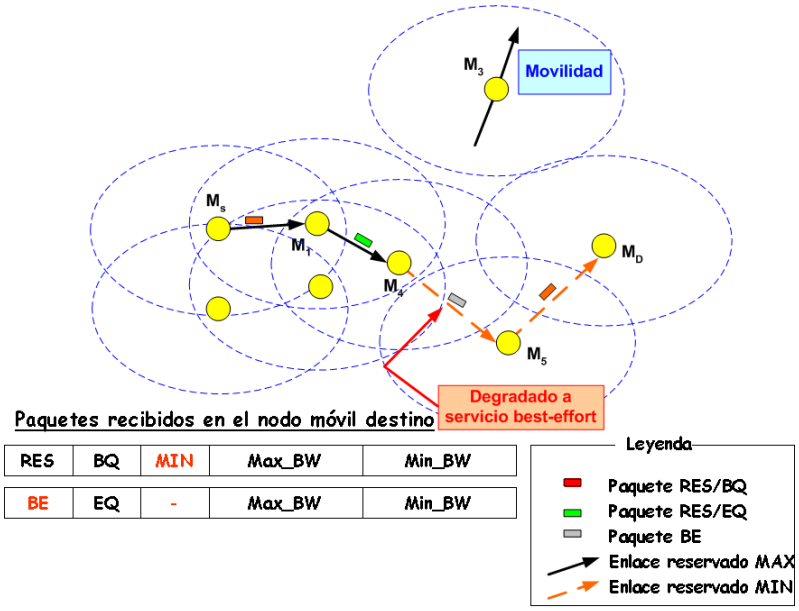



**Fig. 2.19.** Reencaminamiento y degradación.

Entonces el nodo $M_1$ utiliza el protocolo de encaminamiento con el objetivo de encontrar una nueva ruta hacia el destino. Cuando el nodo $M_4$ recibe los paquetes comprueba si se ha realizado una reserva de recursos para ellos consultando una tabla de estado suave (soft-state) por flujo. Si no se han reservado recursos entonces el nodo pondría en marcha el procedimiento de control de admisión para hacerlo. Cuando los paquetes llegan a $M_3$ se detecta

que ya existe una reserva anterior para ellos porque la ruta anterior atravesaba este nodo y la reserva ahora se mantiene.

Los temporizadores de estado suave (soft-state) garantizan que el nodo $M_3$ mantenga su reserva de recursos para el flujo que ha sido reconocido porque los paquetes se siguen reenviando a través de este nodo intermedio, mientras que en el nodo $M_2$ se liberan los recursos asignados a dicho flujo al expirar el temporizador. Cuando se reconduce un flujo a través de un nodo que carece de recursos suficientes para hacer posible la reserva, el flujo es degradado a paquetes 'best-effort'. Debido a esto, los nodos siguientes que reciban aquellos paquetes que ahora se consideran best-effort no intentarán reservar recursos ni mantener un estado de reserva asociado a dicho flujo. Lo que sucederá entonces es que el estado de reserva asociado al flujo expirará en estos nodos, pasando automáticamente a liberarse aquellos recursos que estaban asociados. Será posible la restauración de una reserva en el caso de que se liberen recursos en un nodo intermedio con ancho de banda restrictivo o 'cuello de botella' (como por ejemplo el nodo $M_4$ en la *Fig. 2.19*) o bien se vuelvan a encaminar los paquetes del flujo a través de una nueva ruta que en esta ocasión sí que disponga de recursos para completar la reserva. A este tipo de restauración se le denomina 'restauración degradada'. La 'restauración degradada' consiste en que un flujo que se ha visto reconducido se ve degradado durante un periodo de tiempo *T* para pasar más tarde a recuperar su estado de reserva original. Pueden ocurrir dos sucesos:

- *Un flujo con modo de reserva máximo opera en el modo de reserva mínimo o best-effort y recupera su modo de reserva máximo después de un cierto intervalo de tiempo.*
- *Un flujo con el modo de reserva mínimo opera en el modo best-effort y recupera su modo de reserva mínimo después de un cierto intervalo de tiempo.*

Por otro lado, un flujo sufrirá una 'degradación permanente' si no se puede restablecer la reserva de recursos original y permanece degradado a lo largo de toda la sesión.

En la *Fig. 2.19* se observa que el enlace a través de la nueva ruta $M_4$-$M_5$ sólo puede proporcionar a los paquetes con el tipo de carga EQ el modo de servicio best-effort; por lo tanto, la restauración únicamente puede ser degradada o permanente. Ante esta situación, la aplicación debe decidir si le conviene más seguir enviando su tráfico con el tipo de carga EQ como si fuera best-effort o bien cesar el envío de paquetes. El nodo destino podría

enviar una orden para que la fuente dejara de transmitir los paquetes en vez de enviarlos como tráfico best-effort, si no va a sacar ningún provecho de dicho envío.

- *Adaptación*

  Los informes de calidad de servicio (QoS reports) que el destino envía a la fuente para asesorarla acerca del estado de un flujo, son utilizados en el proceso de adaptación. Los informes de QoS se envían periódicamente para completar la fase de reserva o comprobar cuál es el estado de la calidad de servicio que se está ofreciendo a una aplicación, si bien también es posible que sean enviados cuando se necesiten (por ejemplo, cuando deben tomarse decisiones relacionadas con la adaptación). La periodicidad con que se envían los informes de calidad de servicio dependerá de la sensibilidad de la aplicación. Para poder enviar información acerca de la calidad de servicio de un flujo, el nodo destino medirá determinados parámetros (pérdidas de paquetes, throughput, retardo, etc.) e inspeccionará los valores del campo opciones en la cabecera IP (examinando, por ejemplo, el indicador de ancho de banda). Una vez un nodo destino ha monitorizado un flujo, debe decidirse qué acciones hay que emprender para que el flujo pueda adaptarse en concordancia con una política de adaptación específica para la aplicación a la cual el flujo pertenece y que ha sido definida por el usuario.

  Durante la fase de restauración del estado de un flujo, se invoca el proceso denominado control de admisión para realizar la reserva de recursos de un flujo. Si cambia la calidad de servicio proporcionada a un flujo, el destino lo notará y podrá emprender alguna acción, seleccionando un mecanismo de adaptación.

  El sistema de señalización INSIGNIA soporta tres posibles mecanismos de adaptación, que son enviados en forma de comandos por parte del destino a la fuente para que actúe y se fundamentan en los informes de QoS:

  - *Reducción de un flujo (scale-down):*

    Se pide a la fuente que envíe los paquetes del flujo como best-effort en vez de QoS mejorada, o bien que envíe los paquetes del flujo como best-effort en vez de QoS básica y QoS mejorada. Se trata de enviar todo el tráfico como best-effort para anular de esta forma cualquier reserva parcial de recursos que existiera.

  - *Dejar de transmitir un flujo:*

    Se pide a la fuente que deje de enviar paquetes del flujo con QoS mejorada o bien con QoS básica y QoS mejorada.

- *Aumento de un flujo (scale-up):*

  Se pide a la fuente que inicie una reserva de recursos para un flujo con calidad de servicio básica y/o mejorada.

En la *Fig. 2.20* se ilustra el proceso de aumento de un flujo. En este caso, debido por ejemplo a la movilidad o la dinámica de la red, ahora el nodo $M_2$ es capaz de realizar una reserva de recursos máxima en vez de mínima y así lo señala cambiando el indicador de ancho de banda de los paquetes MIN a MAX, antes de proceder a enviarlos en dirección al destino. El nodo $M_2$ no reserva recursos para estos paquetes, pero sí que informa al destino de que puede ser mejorada la calidad de servicio del flujo (aunque no garantizada al cien por cien). El destino alertará a la fuente de la situación enviándole un informe de QoS y la fuente podrá actuar en consecuencia (basándose en la política de adaptación de la aplicación) enviando en este caso paquetes EQ con un modo de servicio de RES.

La *Fig. 2.21* en cambio muestra un caso de reducción de un flujo. Debido a la movilidad del nodo $M_2$ es necesario buscar una nueva ruta para reconducir un flujo.

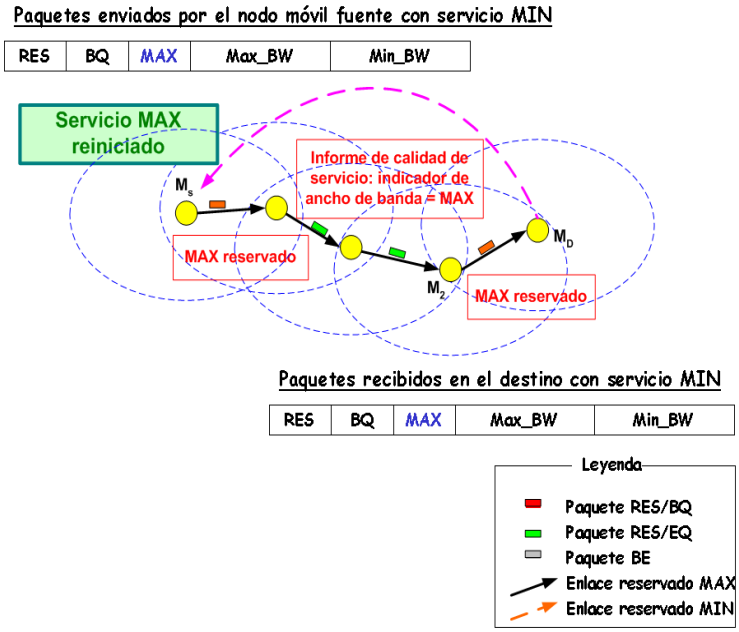



**Fig. 2.20.** Adaptación: Aumento de un flujo (Scaling up).

La nueva ruta atravesará el nodo intermedio $M_3$, el cual no dispone de recursos suficientes para mantener una reserva de recursos máxima. El destino avisará a la fuente mediante un informe de calidad de servicio de que se está garantizando la entrega de los paquetes BQ, pero los paquetes EQ a partir del nodo $M_3$ se tratan como si fueran best-effort.

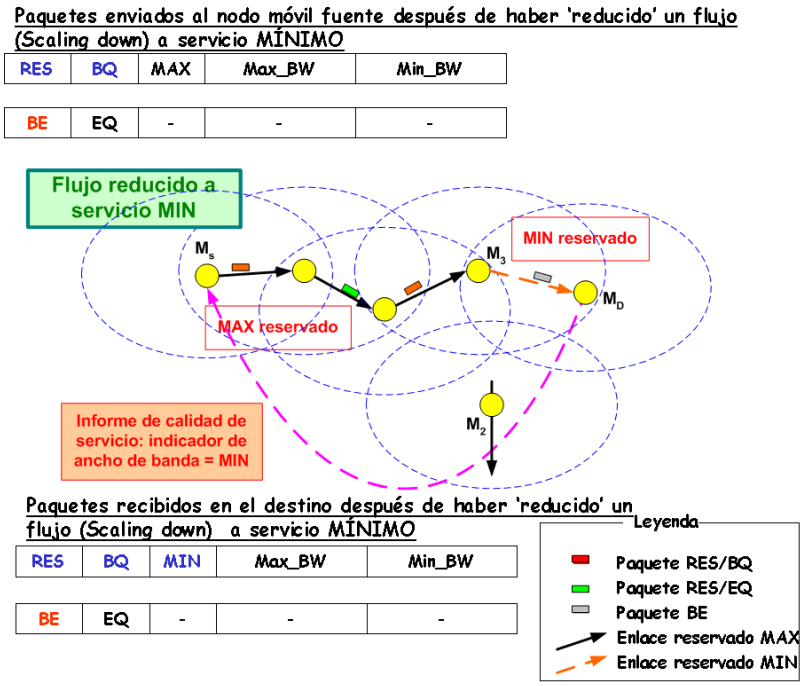



**Fig. 2.21.** Adaptación: Reducción de un flujo (Scaling down).

En vista de los acontecimientos la fuente pasa a enviar los paquetes EQ con el modo de servicio BE (best-effort) con el fin de anular una reserva parcial de recursos que se había establecido entre la fuente $M_S$ y $M_3$, confiando en que más adelante el número de recursos disponibles será suficiente para mejorar la calidad de servicio entre los nodos $M_3$ y $M_D$.

## *2.3 El modelo FQMM*

El modelo FQMM (flexible QoS Model for MANETs (Mobile Ad-hoc Networks)) [61] es el primer modelo de calidad de servicio propuesto expresamente para redes ad hoc (si bien es cierto que el modelo INSIGNIA ha sido desarrollado con anterioridad, pero INSIGNIA se centra sobretodo en la señalización mientras que el modelo FQMM resulta más completo en su totalidad). FQMM ha sido desarrollado para una red no jerárquica de tamaño pequeño o mediano (con menos de 50 nodos).

Siguiendo el paralelismo con la arquitectura DiffServ, define tres tipos de nodos:

- *Nodos de ingreso*

  Son aquellos nodos que actúan como fuentes enviando datos. Serán los encargados de clasificar, monitorizar, marcar y someter a un control de policía a los paquetes.

- *Nodos interiores:*

Son aquellos nodos que actúan como nodos intermedios o routers, reenviando los paquetes de los demás nodos de acuerdo con un PHB específico definido por el campo DSCP *(Véase la sección 2.4 La arquitectura de Servicios Diferenciados, pág. 67).*

❖ *Nodos de egreso:*

Son aquellos nodos que actúan como destinos de los paquetes enviados.

En la *Fig. 2.22* se puede observar el papel que desempeña cada uno de los nodos integrantes de la red ad hoc. El nodo $M_1$ actúa como nodo de ingreso enviando paquetes a través de una ruta que atraviesa los nodos interiores $M_3$, $M_4$ y $M_5$ en dirección hacia el nodo de egreso $M_6$. No obstante, conviene observar que los roles que desempeñan los nodos son dinámicos, pues tal y como se observa en la red ad hoc de la *Fig. 2.23*, el nodo $M_8$, además de ser nodo interior para la conexión $C_1$, pasa a ser nodo de ingreso para la conexión $C_2$. Los roles asignados a cada uno de los nodos se listan en la *Tabla 2.4*.

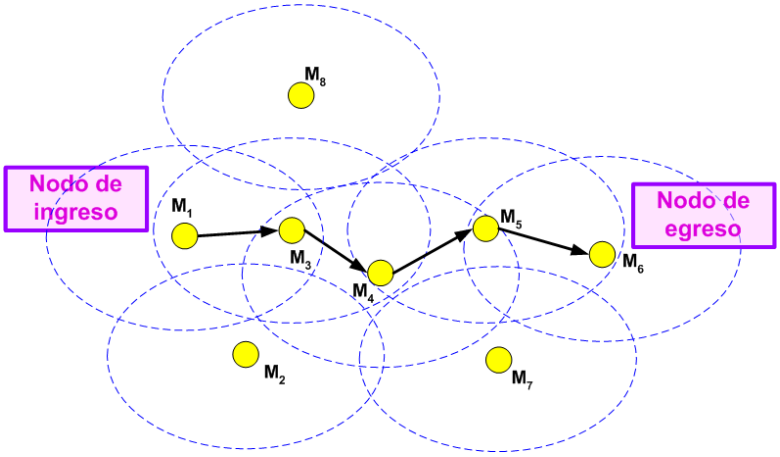



**Fig. 2.22.** Escenario 1.

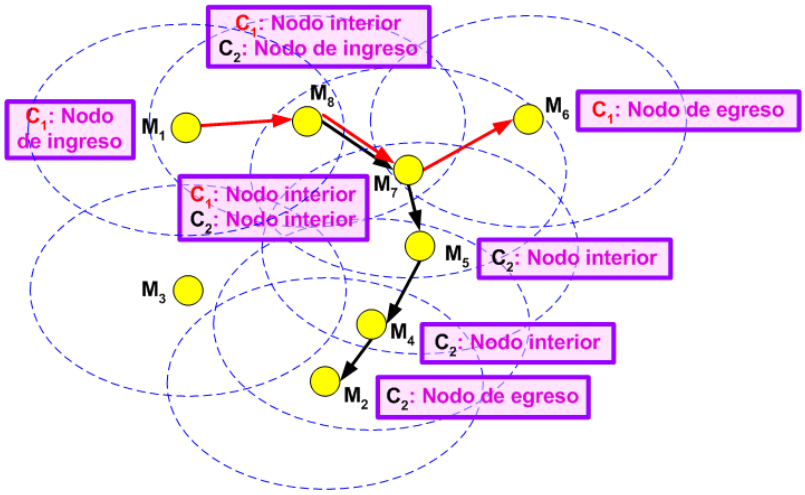



**Fig. 2.23.** Escenario 2.

| Conexión | Nodo de ingreso | Nodo interior | Nodo de egreso |
|---|---|---|---|
| $C_1$ | $M_1$ | $M_8$, $M_7$ | $M_6$ |
| $C_2$ | $M_8$ | $M_7$, $M_5$, $M_4$ | $M_2$ |

**Tabla 2.4.** Roles de los nodos aplicando el modelo FQMM.

En el modelo FQMM [62] se propone un esquema híbrido como política de reserva de recursos que combine la diferenciación de servicios por flujo de IntServ [5], [6] con la diferenciación de servicios por clase de DiffServ [67], [68] . Esto significa que para el tráfico más prioritario la reserva de recursos se hará por flujo, mientras que el resto de tráfico reservará sus recursos por clase.

La reserva de recursos por flujo queda restringida a una cantidad limitada de tráfico (el más prioritario), porque el ancho de banda de una red ad hoc está limitado, si bien sí que es posible que las clases más prioritarias de tráfico se beneficien de una diferenciación por flujo gracias a que la cantidad de tráfico en este tipo de redes es mucho menor que un backbone de Internet.

La arquitectura del modelo FQMM puede contemplarse en la *Fig. 2.24.* El modelo de calidad de servicio FQMM trabaja en la capa IP, que interactúa cooperando con la capa MAC. El plano de envío de datos se sitúa por debajo de la línea de puntos suspensivos, mientras que el plano de control y gestión quedaría ubicado por encima. En el plano de envío de datos se realizan una serie de operaciones para poder reenviar los paquetes, mientras que en el plano de control se utilizan una serie de protocolos con el objetivo de que el plano de envío de datos pueda realizar sus funciones.

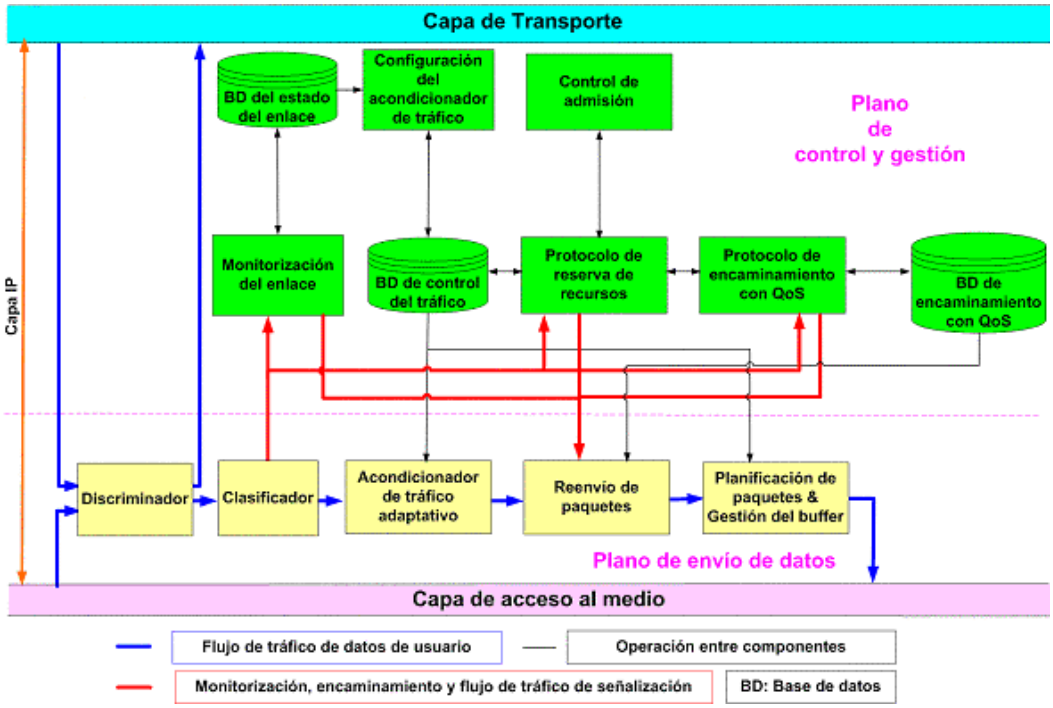



**Fig. 2.24.** Arquitectura del modelo FQMM.

Los módulos de cada plano pueden comunicarse directamente entre sí o bien accediendo a información referente al otro plano a través de las bases de datos.

En el plano de envío de datos distinguiremos los módulos siguientes:

- *Discriminador:*

  Clasifica cada paquete entrante según provenga de la capa de transporte o bien de la capa MAC. Los paquetes que vienen de la capa de transporte y viajan en dirección al medio son enviados al módulo que actúa de clasificador, Los paquetes que provienen de la capa MAC (y por lo tanto del medio) son enviados a la capa de transporte si el nodo actual es su destino o bien se remiten al clasificador si dicho nodo actúa solamente como un nodo intermedio hacia el destino.

- *Clasificador:*

  Envía los paquetes a los módulos 'monitorización del enlace', 'protocolo de reserva de recursos' y 'protocolo de encaminamiento con calidad de servicio', situados en el plano de control y gestión. También es capaz de mapear los paquetes de tráfico a una clase leyendo el campo DSCP situado en la cabecera del paquete. Cada clase estará asociada con un perfil de servicio almacenado en la base de datos de control del tráfico. Para los paquetes pertenecientes a una clase prioritaria, la reserva de recursos se hace por flujo y para el resto el flujo de paquetes correspondiente se agrega a los demás flujos pertenecientes a la misma clase de tráfico para formar un agregado de flujo y pasa a efectuarse un tratamiento por clase.

- *Acondicionador de tráfico adaptativo:*

  Este módulo se halla situado en el nodo de ingreso y es el encargado de controlar y conformar el tráfico generado basándose en un perfil de servicio almacenado en la base de datos de control del tráfico. Está formado por un marcador, un conformador y un descartador. Este módulo se denomina 'adaptativo' porque es capaz de ajustar el perfil de servicio a los estados de los enlaces, adaptándose a la dinámica de la red.

- *Reenvío de paquetes:*

  Reenvía los paquetes entrantes a un puerto de salida, incluso si se trata de paquetes de control procedentes de algún módulo del plano de control y gestión.

- *Planificación de paquetes y gestión del buffer:*

Gestiona la/s cola/s en el puerto de salida. La base de datos de control del tráfico contiene información acerca de la configuración de la disciplina de servicio seleccionada.

En el plano de control y gestión distinguiremos los módulos siguientes:

- *Monitorización del enlace*

  Monitoriza el estado del canal y anota el resultado en la base de datos del estado del enlace. La monitorización del enlace puede ser local o bien abarcar un área de mayor alcance donde el nodo intercambia información con otros nodos.

- *Configuración del acondicionador de tráfico*

  Configura el perfil de servicio de las aplicaciones y lo registra en la base de datos de control del tráfico. Para poder definir el perfil del servicio requerido dicho módulo basa sus decisiones en la información almacenada tanto en la base de datos del estado del enlace como en la base de datos de control del tráfico. La escala de tiempos a la hora de monitorizar el estado de los enlaces y configurar el acondicionador de tráfico debe ser escogida con cautela, pues ejercerá una gran influencia en la calidad de servicio ofrecida a la aplicación.

- *Control de admisión*

  Es el módulo encargado de comparar los recursos disponibles con aquellos que habían sido solicitados y decidir si reservar recursos para un nuevo flujo o un agregado de flujo consultando la política de gestión de recursos que ha sido establecida. El control de admisión es invocado por el protocolo de reserva de recursos para decidir si es viable efectuar la reserva.

- *Protocolo de reserva de recursos*

  Permite la reserva de una serie de recursos de red extremo a extremo para satisfacer la calidad de servicio solicitada. Para ello, lo primero que hace es interactuar con el protocolo de encaminamiento con el fin de que éste establezca una ruta entre la fuente y el destino para poder enviar paquetes; después reserva los recursos necesarios a lo largo de dicha ruta basándose en el resultado del proceso de control de admisión. Este módulo mantiene información acerca del estado de la reserva, ya sea por flujo o bien por clase y actualiza la base de datos de control del tráfico.

- *Protocolo de encaminamiento con calidad de servicio*

  Se encarga de encontrar y mantener una ruta con QoS entre una fuente y un destino para un flujo o un agregado de flujo.

- *Bases de datos*

Las bases de datos almacenan la información necesaria para que los módulos puedan funcionar correctamente. Existen tres:

- *Base de datos del estado del enlace:*

  El módulo 'monitorización del enlace' mantiene la base de datos del estado del enlace, la cual es usada por el módulo 'configuración del acondicionador de tráfico' con el fin de configurarse.

- *Base de datos de control del tráfico:*

  Los módulos 'configuración del acondicionador de tráfico' y 'protocolo de reserva de recursos' mantienen la base de datos de control del tráfico. Dicha base de datos mantiene información acerca del perfil de servicio, los estados de la reserva por flujo/agregado de flujo y otros estados de control utilizados para poder configurar los parámetros del módulo 'acondicionador de tráfico adaptativo' y el módulo 'planificación de paquetes y gestión del buffer'.

- *Base de datos de encaminamiento con QoS:*

  El protocolo de encaminamiento con QoS mantiene la base de datos de encaminamiento con QoS, la cual es utilizada por el módulo 'reenvío de paquetes' para averiguar las rutas.

Una primera evaluación del modelo FQMM ha consistido en establecer una cierta prioridad para determinados tipos de tráfico con respecto al resto [61].

Se suele considerar que el tráfico TCP (Transmission Control Protocol) es el que se usa para servicios de datos, mientras que el tráfico UDP (User Datagram Protocol) se relaciona normalmente con las aplicaciones de tiempo real. Atendiendo a este criterio, es posible establecer tres perfiles de servicio distintos:

- *Distintos niveles de prioridad del tráfico TCP.*
- *Distintos niveles de prioridad del tráfico UDP.*
- *Distintos niveles de prioridad entre los tráficos TCP y UDP: El tráfico UDP tiene siempre prioridad sobre el tráfico TCP* [63].

A continuación se explica el primero de los tres criterios, los cuales se encuentran ampliamente desarrollados en [64]. El primer criterio se concreta definiendo un nivel de alta y otro nivel de baja prioridad para los servicios FTP (File Transfer Protocol). Se consigue simplificar la arquitectura del modelo FQMM *(Véase la Fig. 2.25)*, eliminando los módulos 'protocolo de reserva de recursos', 'protocolo de encaminamiento con QoS' (se dejaría un módulo de encaminamiento a secas), 'control de admisión', 'monitorización del enlace' y la base de datos del estado del enlace. Se necesita

además que el módulo 'configuración del acondicionador de tráfico' y la base de datos de control del tráfico puedan priorizar una clase sobre la otra.

Del plano de datos, no obstante, sí que son necesarios todos los módulos.

Para poder establecer una prioridad de una clase sobre la otra, se han definido dos esquemas en el módulo 'planificación de paquetes y gestión del buffer':

- *Gestión de colas RIO-C (Véase la sección 2.4.1.2 El algoritmo RIO o RIO-C, pág. 76)*

  En este caso, los paquetes de alta prioridad son marcados como 'HIGH' y los paquetes de baja prioridad son marcados como 'LOW' de acuerdo con el perfil de servicio. Preferiblemente se descartan aquellos paquetes marcados como 'LOW' en el caso de que exista congestión.

- *Disciplina de servicio por prioridades (Priority scheduling)*

  Los paquetes de alta prioridad son siempre encolados delante de los paquetes de baja prioridad que ya se encuentran almacenados en el buffer. Así se consigue que los paquetes de alta prioridad sean servidos por el planificador antes que los paquetes de baja prioridad. En caso de congestión y al tratarse de una cola FIFO (First In First Out), si llega un paquete de alta prioridad y se encuentra la cola llena, se descartará el último paquete de baja prioridad de la cola.

Una segunda evaluación del módulo FQMM centra su atención en la diferenciación de servicios en redes ad hoc.

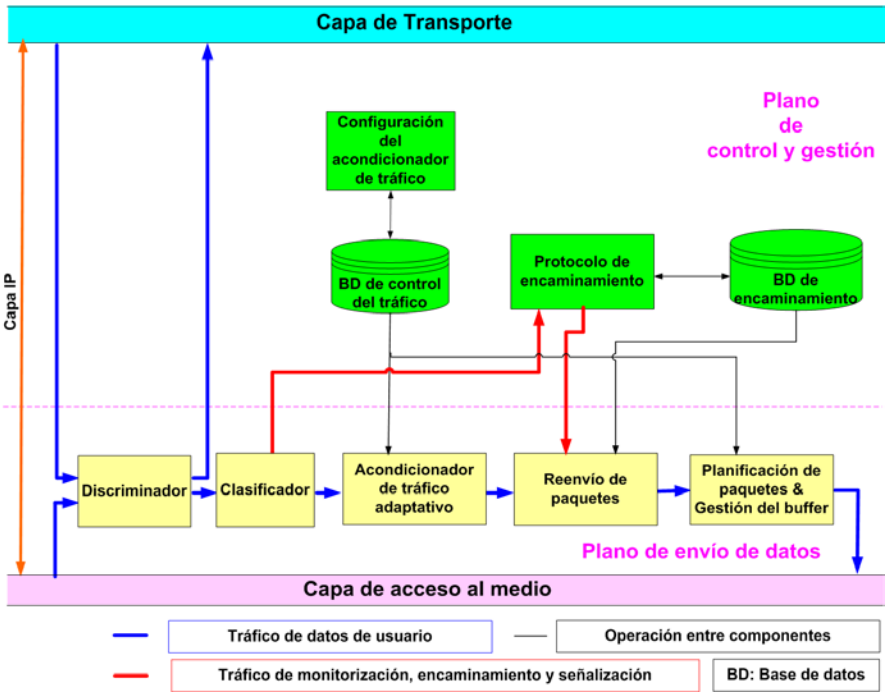



**Fig. 2.25.** Componentes para priorizar servicios en la arquitectura del modelo FQMM.

La diferenciación de servicios consiste en distinguir entre diferentes clases de tráfico tratando a cada clase de forma distinta de acuerdo con su nivel de prioridad. Existen dos modos de diferenciar servicios:

- *Diferenciación de servicios absoluta:*

  Consiste en garantizar unos determinados parámetros de calidad de servicio por usuario o clase, como por ejemplo la tasa de pérdidas, el retardo extremo a extremo, el jitter, el throughput, etc. Un usuario verá como su petición es rechazada si los recursos que solicita no pueden ser proporcionados por la red.

- *Diferenciación de servicios relativa:*

  No consiste en que la red garantice la existencia de recursos para un determinado usuario, sino que en realidad se intenta que exista una relación de prioridad relativa de unas clases respecto de otras.

En [65] los autores estudian la diferenciación de servicios absoluta para una red ad hoc. Los resultados muestran que no se puede mantener una diferenciación de servicios absoluta para una red ad hoc debido a su dinamismo (cambios de topología, variaciones de la capacidad de los enlaces, de la carga, etc.). Por este motivo, se efectúa una evaluación de la diferenciación de servicios relativa aplicada al modelo FQMM.

Se ofrece un perfil de servicio relativo consistente en un porcentaje de la capacidad efectiva del enlace o tasa relativa, seleccionada dentro de un rango entre 0 y 1. Se define la 'capacidad efectiva del enlace' como el ancho de banda disponible para que un nodo pueda enviar información en forma de paquetes (descontando el ancho de banda usado por sus nodos vecinos) y dependerá de factores tales como las restricciones de potencia, la movilidad, la topología, el número de colisiones, el encaminamiento y la carga de tráfico de la red ad hoc. Se cuenta con el módulo 'monitorización del enlace' para poder estimar correctamente este parámetro.

La arquitectura del modelo FQMM ha sido adaptada para poder diferenciar servicios de manera relativa, tal y como se muestra en la *Fig. 2.26.*

Se han eliminado los módulos 'control de admisión', 'protocolo de encaminamiento con calidad de servicio' y la base de datos de encaminamiento con QoS, así como el módulo 'protocolo de reserva de recursos'. El módulo 'planificación de paquetes y gestión del buffer' utiliza el esquema RIO-C para la gestión de colas.

.

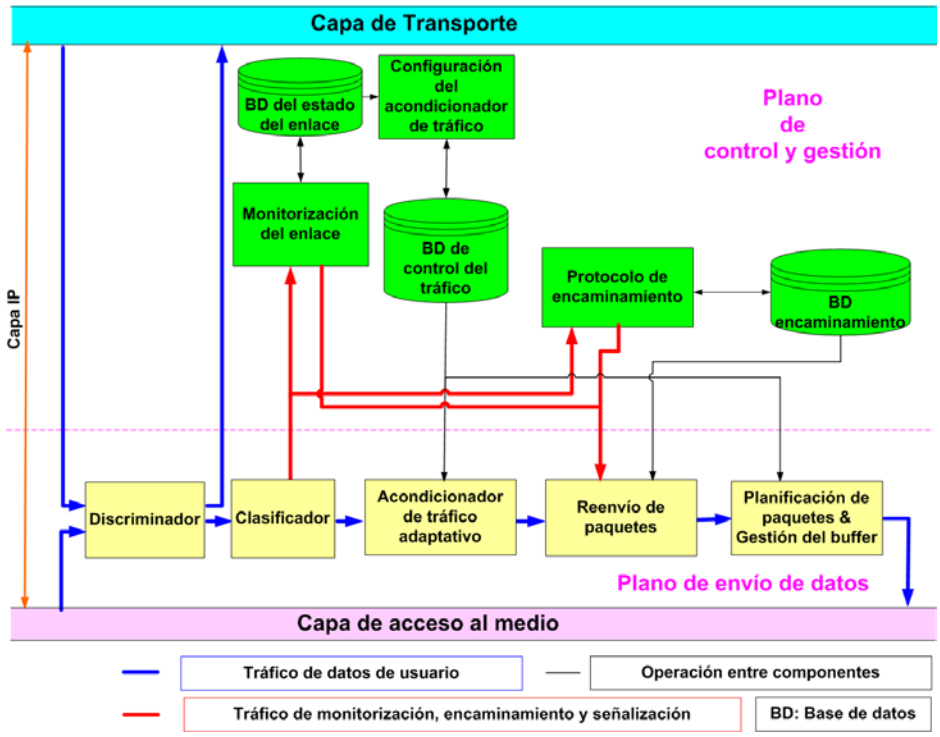



**Fig. 2.26.** Componentes para diferenciar servicios en la arquitectura del modelo FQMM.

El módulo 'acondicionador de tráfico adaptativo' es un token bucket que monitoriza los paquetes y los marca como IN u OUT dependiendo de si cumplen con un perfil de tráfico determinado. Los parámetros del token bucket $\rho$ (tasa de generación de tokens) y $\beta$ (tamaño de la cubeta) se ajustan dinámicamente para una sesión $i$, de acuerdo con las ecuaciones siguientes:

$$\rho_i = \gamma_i * C_t * R, \tag{2.11}$$

$$\beta_i = \gamma_i * C_t * L, \tag{2.12}$$

donde $\gamma_i$ es la tasa relativa de la sesión $i$, $\rho_i$ es la tasa de generación de tokens, $\beta_i$ es el tamaño de la cubeta, $C_t$ es la capacidad efectiva del enlace y *R* y *L* son constantes.

En [66] se proponen dos métodos para calcular la capacidad efectiva del enlace en una red ad hoc donde los nodos se mueven aleatoriamente. El primer método está basado en parámetros y el segundo en medidas.

Se ha escogido la asignación de ancho de banda como parámetro para realizar una diferenciación de servicios relativa entre clases.

En vez de utilizar un protocolo de encaminamiento best-effort y más tarde tratar de reservar recursos a lo largo de la ruta encontrada (lo cual puede convertirse en un éxito o en un fracaso), es mejor utilizar desde el principio un protocolo de

encaminamiento con calidad de servicio que ya tenga en cuenta las restricciones de ancho de banda.

Se considera por tanto que el modelo de calidad de servicio FQMM es flexible básicamente debido a tres motivos fundamentales:

- *Nodos con roles dinámicos*
- *Esquema de reserva de recursos híbrido*
- *Capacidad para combinar de manera flexible los distintos módulos del modelo con el fin de poder ofrecer la calidad de servicio deseada*

## *2.4 La arquitectura de Servicios Diferenciados*

Seguidamente se introduce la arquitectura DiffServ, definiendo sus características principales, para pasar a explicar después como puede adaptarse dicha arquitectura para la diferenciación de servicios en redes ad hoc.

La arquitectura de Servicios Diferenciados, DiffServ (Differentiated Services) ha sido definida por el IETF (Internet Engineering Task Force) [67] [68] con el fin de proporcionar una diferenciación de servicios escalable en Internet y poder así ofrecer distintos servicios y aplicaciones a los usuarios en concordancia con sus necesidades.

Esta arquitectura define una región DS (Differentiated Services) *(Véase la Fig. 2.27)* como un área formada por uno o varios dominios DS (posiblemente bajo la supervisión de diferentes autoridades administrativas).

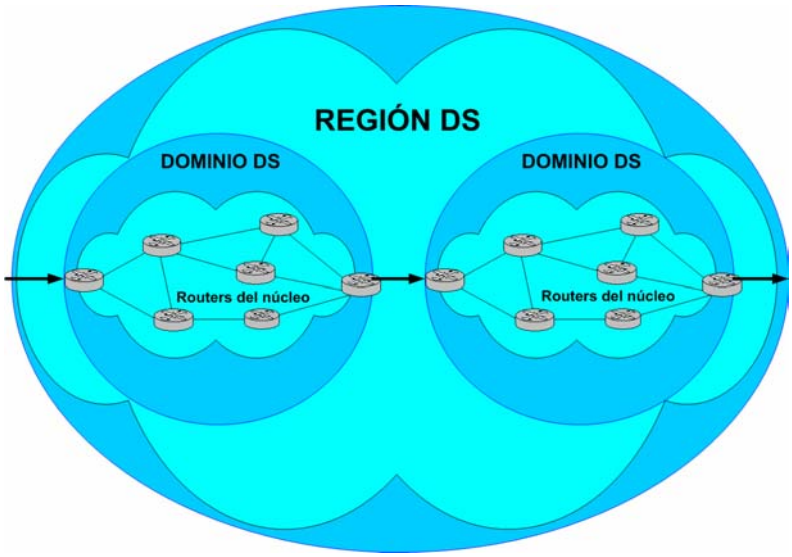



**Fig. 2.27.** Una región DS.

Podemos definir un dominio DS *(Véase la Fig. 2.28)* como una parte contigua de Internet sobre la cual se aplican un conjunto de políticas DS consistentes. Consta de dos partes: la red del núcleo, integrada por los routers del núcleo (core routers) y la red

de acceso, donde un router frontera (un nodo de ingreso DS (DS ingress node) o un nodo de egreso DS (DS egress node)) conecta un dominio DS con otro nodo ubicado en otro dominio DS o bien en un dominio que no es capaz de soportar servicios diferenciados.

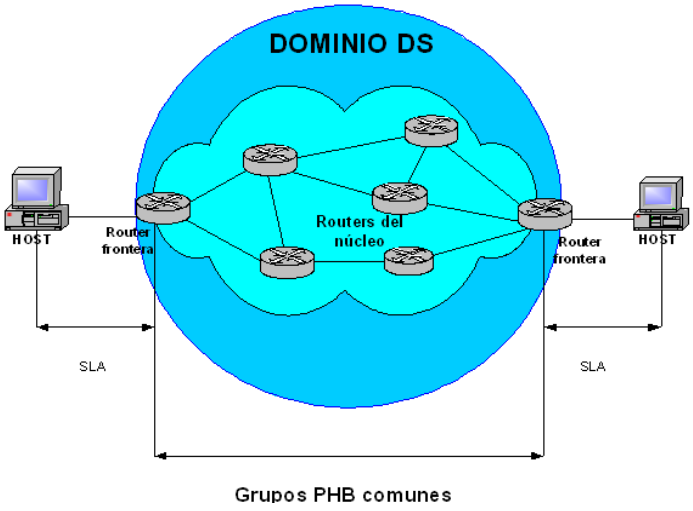



**Fig. 2.28.** Elementos básicos de la arquitectura de Servicios Diferenciados.

Los principales módulos que actúan sobre el tráfico que llega a los routers frontera son un clasificador, un medidor, un marcador, un conformador y un descartador *(Véase la Fig. 2.29)*.

Los routers frontera están ubicados en un extremo de la red y clasifican los paquetes entrantes de acuerdo con una política especificada por el Acuerdo de Nivel de Servicio, SLA (Service Level Agreement). El SLA [69] es una especie de contrato entre el usuario y el proveedor de servicios de Internet, ISP (Internet Service Provider), donde los clientes especifican sus necesidades y aborda no solamente aspectos técnicos (como el acondicionamiento del tráfico o la disponibilidad de servicios), sino también aspectos comerciales (como la tarificación). El Traffic Conditioning Agreement (TCA) es un documento que forma parte del SLA y aborda aspectos tales como la clasificación y las reglas de acondicionamiento que deben ser aplicadas a un flujo de tráfico entrante en la red.

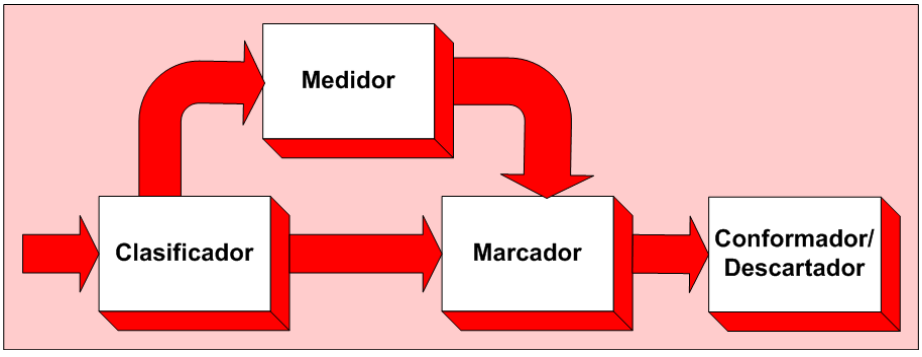



**Fig. 2.29.** Módulos principales para el tratamiento del tráfico.

Un flujo [70] puede ser identificado por la fuente, el destino o cualquier combinación de los campos de la cabecera del paquete (número de puerto, etc.). Esto permite a un router frontera clasificar los paquetes de cada flujo entrante. La misión del clasificador es identificar los flujos de tráfico para aplicarles un tratamiento diferencial en el dominio DS, en función de los perfiles de tráfico configurados. Un medidor pasa a comprobar que los paquetes entrantes de dicho flujo cumplan ciertos parámetros de tráfico. Entonces estos flujos son agregados para formar un pequeño número de diferentes clases de tráfico; esto se logra marcando de una manera determinada el campo Differentiated Services Codepoint (DSCP) en la cabecera IP de manera que la secuencia de seis bits escogida refleje el nivel de servicio deseado. Este campo ocupa los seis primeros bits del campo TOS (Type of Service) en IPv4 o bien del campo Clase de Tráfico en IPv6 *(Véase la Fig. 2.38)*. Los mecanismos de conformación y control de policía son finalmente los encargados de garantizar que el tráfico cumpla las especificaciones de los SLAs.

Al marcar de una manera determinada un paquete se le asocia un PHB (Per-Hop Behavior) que alude a un tratamiento particular que se aplicará a la hora de reenviar el paquete. Los routers del núcleo examinan el codepoint de los paquetes entrantes y deciden reenviar el paquete conforme a su PHB asociado, porque el PHB define la clase de prioridad del paquete. Estos routers no mantienen información de estado por flujo y procesan los paquetes entrantes muy rápidamente puesto que el número de clases de servicio definido mediante el campo DSCP es limitado.

Conviene destacar que la arquitectura de Servicios Diferenciados pretende desplazar la complejidad (clasificación del tráfico, acondicionamiento) a los extremos de la red [71], preservando el interior de dicha red para tratar los paquetes que llegan a los routers del núcleo de una forma concreta, dentro de un conjunto limitado de comportamientos, según el agregado de flujo al que pertenezcan.

Habrá una clase de servicio asociada al PHB EF (Expedited Forwarding) [72] [73], el cual proporciona pocas pérdidas, baja latencia, jitter reducido y servicio de ancho de banda extremo a extremo garantizado. Proporciona el denominado Servicio Premium (Premium Service).

Por otro lado, el PHB AF o Assured Forwarding [74] define un grupo de codepoints que pueden ser usados para especificar cuatro clases de tráfico, cada una de las cuales tiene tres precedencias de descarte *(Véase la Tabla 2.5)*.

Un paquete IP [75] que pertenezca a una clase $i$ AF y que tenga una precedencia *j* será marcado con el codepoint de AF $AF_{ij}$, donde $1 \le i \le N$ y $1 \le j \le M$, siendo *N* el número máximo de clases AF y *M* el número máximo de niveles de precedencia de descarte. Generalmente se definen cuatro clases (*N* = 4) con tres niveles de

precedencia de descarte en cada clase (*M* = 3), pero los proveedores de servicios podrían si quisieran definir más clases AF y niveles de precedencia de descarte para uso local.

| PRECEDENCIA DE DESCARTE | Clase #1 | Clase #2 | Clase #3 | Clase #4 |
|---|---|---|---|---|
| Prec. de descarte baja | (AF11) 001010 | (AF21) 010010 | (AF31) 011010 | (AF41) 100010 |
| Prec. de descarte media | (AF12) 001100 | (AF22) 010100 | (AF32) 011100 | (AF42) 100100 |
| Prec. de descarte alta | (AF13) 001110 | (AF23) 010110 | (AF33) 011110 | (AF43) 100110 |

**Tabla 2.5.** Tabla de codepoints AF de DiffServ.

A un paquete IP se le puede asignar dentro de cada clase AF uno de los tres niveles de precedencia de descarte. El valor de precedencia de descarte se usa en caso de congestión para determinar qué paquetes deben ser descartados o eliminados primero (aquellos que tengan un valor mayor) protegiendo a aquellos paquetes que tengan un valor de precedencia de descarte menor.

A cada clase se le asigna un cierto número de recursos en cada nodo DS tales como espacio de buffer o ancho de banda. Los flujos AF no poseen unas especificaciones de calidad de servicio tan estrictas como las del tráfico EF pero sí que necesitan para su correcto funcionamiento disponer de un ancho de banda mínimo que pueda ser incrementado en el caso de que la red no esté congestionada.

El PHB AF se encarga de garantizar a los clientes un throughput mínimo incluso durante los períodos de congestión en la red y permite que los usuarios consuman más ancho de banda cuando la red transporta poca carga.

Las clases asociadas al PHB AF proporcionan el denominado Servicio Asegurado, AS (Assured Service) [76] [77], lo cual significa que a un usuario de dicho servicio se le asegura que es muy improbable que su tráfico sea descartado, siempre y cuando cumpla con el perfil y no se exceda en capacidad. A los distintos agregados de flujo se les pueden ofrecer diferentes 'forwarding assurances' o garantías de llegada de paquetes a su destino a la hora de realizar el reenvío.

Además, se usan tres de las cuatro clases actualmente definidas del grupo PHB AF para implementar el Servicio Olímpico (Olympic Service) [78] *(Véase la Fig. 2.30)*; el tráfico puede dividirse en las clases Oro (Gold) (se le asigna una porción elevada del enlace), Plata (Silver) (porción intermedia) y Bronce (Bronze) (porción más baja). Se

decide que haya menos paquetes compitiendo por la clase Oro que por la clase Plata, con lo cual la carga para la clase Oro es menor (y así este tráfico será reenviado con un retardo medio menor en comparación con la clase Plata). También se decide que haya menos tráfico compitiendo por la clase Plata que por la clase Bronce, con lo cual la carga para la clase Plata es menor (y así este tráfico será reenviado con un retardo medio menor en comparación con la clase Bronce). Si se desea se puede asignar una precedencia de descarte alta, media o baja a los paquetes pertenecientes a una misma clase.

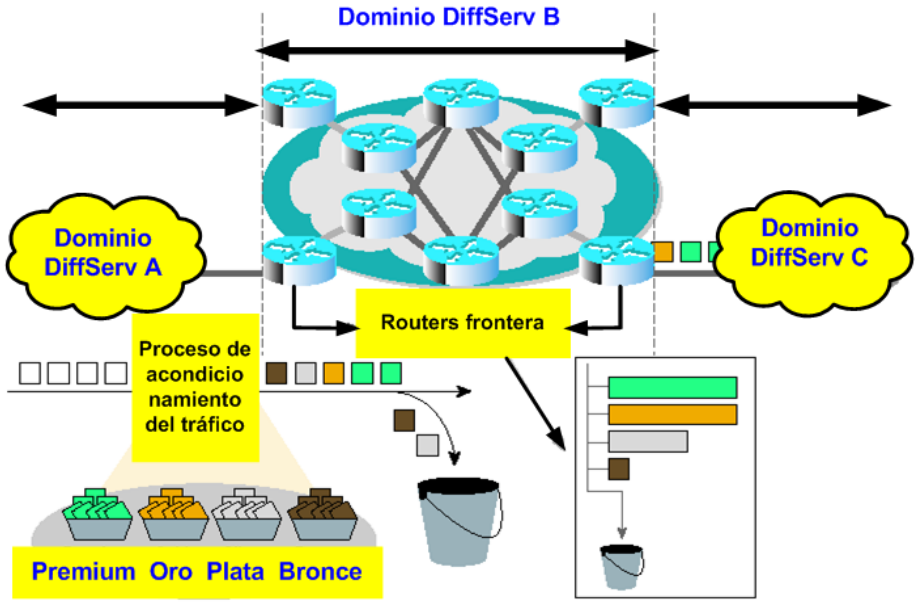



**Fig. 2.30.** Ejemplo de arquitectura DiffServ con Servicio Olímpico.

Finalmente, BE (Best Effort) define una clase en la cual no existe ninguna garantía de calidad de servicio.

## *2.4.1 La arquitectura de Servicios Diferenciados aplicada a redes ad hoc*

Algunas características de la arquitectura DiffServ, tales como la ausencia de un mecanismo de señalización que sobrecargue la red o la falta de reserva de recursos por flujo, parecen convertir a este modelo en uno muy apropiado para su aplicación en redes ad hoc. Además, un modelo de calidad de servicio distribuido sin estados como DiffServ, se adapta mejor al entorno de una red ad hoc debido a sus propiedades inherentes tales como simplicidad, escalabilidad y habilidad para hacer frente a las condiciones dinámicas de la red.

Por todos estos motivos, algunos investigadores han tratado de adaptar DiffServ (originalmente diseñado para redes fijas de alta velocidad) a redes ad hoc inalámbricas.

Al hacerlo, se han encontrado con los siguientes problemas:

- *Cualquier nodo en una red ad hoc puede funcionar como nodo fuente o bien como nodo intermedio (cuando actúa como router reenviando los paquetes de los demás nodos).*

  En consecuencia, cada nodo en una red ad hoc deberá ser capaz de asumir dos roles en paralelo con la arquitectura DiffServ:

  - *router frontera* (cuando envía paquetes como nodo fuente)
  - *router del núcleo* (cuando reenvía paquetes como nodo intermedio) [79]

  Estos dos modos de actuación presentan un coste de almacenamiento alto e incrementan la complejidad de cada nodo.

- *El concepto de SLA entendido como un contrato encargado de especificar los niveles de servicio para los agregados de flujo no existe como tal en una red ad hoc.*

  Debido a esta razón resulta complicado establecer reglas que regulen el tráfico entre nodos móviles pertenecientes a este tipo de redes y se trata más bien de que cada nodo se ocupe por sí mismo de que su tráfico no exceda la capacidad de la red.

Además, las redes ad hoc cuentan con una serie de características que impiden que el modelo de servicios DiffServ sea aplicado tal cual: topología de red dinámica, capacidad limitada de recursos, canal radio compartido y propenso a errores, etc.

Por lo tanto, será necesario adaptar el modelo DiffServ a las redes ad hoc, tal y como han hecho los autores en [80]. Para lograrlo, se ha distinguido entre dos clases de tráfico con distintas prioridades (alta y baja).

Cualquier nodo de la red es capaz de funcionar como router frontera y como router del núcleo. Los paquetes pertenecientes a cada una de las clases de tráfico son marcados en los nodos fuente (que actúan como routers o nodos frontera) y entonces son enviados siguiendo una ruta hacia su destino. Los nodos intermedios a lo largo de dicha ruta realizarán el control de policía, conformando y descartando paquetes cuando sea necesario. Nótese que a los nodos intermedios se les asignan unas funcionalidades que acaban de ser comentadas y que son usualmente realizadas por los routers frontera, pero al mismo tiempo actúan como routers del núcleo, tratando de distinta forma a cada paquete según sea su prioridad. Para conseguirlo, los paquetes

de cada prioridad son encolados separadamente a nivel de la capa MAC utilizando el algoritmo Random Early Detection (RED) [81] con colas basadas en prioridad RIO o RIO-C *(Véase la sección 2.4.1.1 El algoritmo RED, pág. 74 y la sección 2.4.1.2 El algoritmo RIO o RIO-C, pág. 76).*

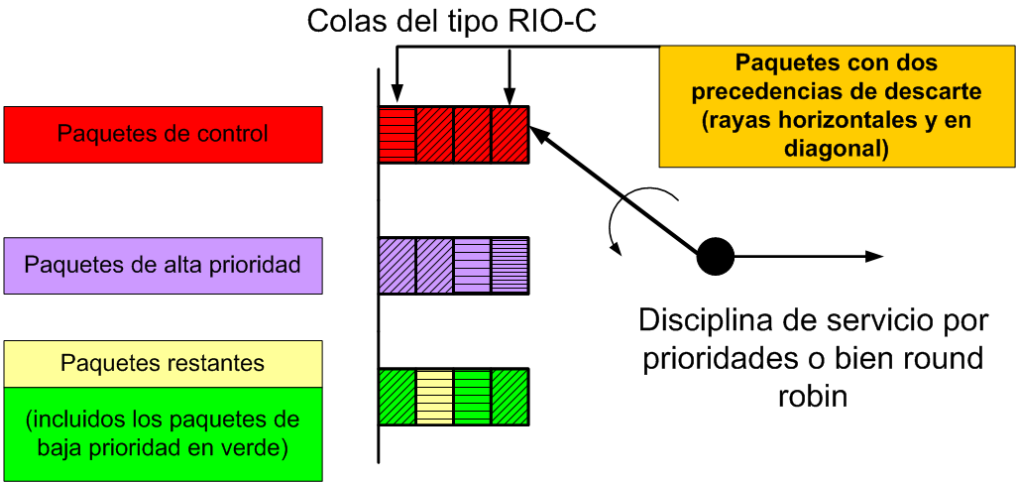



**Fig. 2.31.** Sistema de colas definido a nivel de la capa MAC.

Para poder diferenciar servicios aplicando el modelo DiffServ a redes ad hoc [80], [82], se han definido en cada nodo a nivel de la capa MAC tres colas físicas del tipo RIO-C *(Véase la Fig. 2.31),* las cuales poseen dos niveles de precedencia de descarte. Los paquetes de control que lleguen al nodo serán encolados en la primera cola física, mientras que los paquetes de alta prioridad pasarían a la segunda cola y el resto de paquetes (incluidos aquellos de baja prioridad) lo harían a la tercera. Una vez se haya identificado la clase a la cual el paquete pertenece, se le aplicará una política determinada para ver si cumple o no con un perfil específico y poder de esta forma decidir cuál será su precedencia de descarte. Tanto la longitud de las colas como la probabilidad de descarte de paquetes durante la congestión dependerán de la clase a la cual pertenezca un paquete. Los paquetes de alta prioridad se transmitirán antes que los paquetes de baja prioridad. Los paquetes de alta prioridad son tratados de manera preferente tanto por parte de la disciplina de servicio (cuando se selecciona una disciplina de servicio por prioridades) como cuando se produce congestión (al haber seleccionado unos parámetros para aplicar el algoritmo RIO-C donde los paquetes de baja prioridad son descartados antes que los de alta prioridad).

Se estudia el diferente comportamiento del sistema usando dos planificadores distintos:

- *La disciplina de servicio round robin (round robin scheduling):*

El planificador round robin supone que cada flujo contiene paquetes con el mismo peso y tamaño y sirve los paquetes de cada cola no vacía en un orden cíclico.

- *La disciplina de servicio por prioridades (priority scheduling):*

  Con el planificador por prioridades los paquetes de mayor prioridad son servidos con mayor urgencia que los paquetes de baja prioridad. Cuando llega un paquete de alta prioridad, se inserta antes que todos los paquetes de baja prioridad en la cola. Por lo tanto, los paquetes de alta prioridad son enviados antes que los paquetes de baja prioridad.

## *2.4.1.1 El algoritmo RED*

Seguidamente se explica el algoritmo RED, que ha sido utilizado como mecanismo para la diferenciación de servicios en redes fijas. Se introduce aquí la explicación de este algoritmo para poder más tarde entender el algoritmo RIO o RIO-C, que es una variante de RED con bits In/Out y ha sido utilizado para poder adaptar la arquitectura DiffServ a redes ad hoc *(Véase la sección 2.4.1 La arquitectura de Servicios Diferenciados aplicada a redes ad hoc, pág. 71)*.

RED [81] es un algoritmo de prevención de congestión que sirve para gestionar las colas de un router y eliminar paquetes antes de que éstas se llenen y se produzca congestión.

RED [75] utiliza el nivel de ocupación de la cola como parámetro de entrada a una función que decide si se deberán descartar paquetes. A medida que aumenta la ocupación de la cola, aumenta la probabilidad de descarte de paquetes.

Las *Fig. 2.32* y *Fig. 2.33* muestran un ejemplo de cola RED y su función de probabilidad de descarte asociada:

- *Para una ocupación por debajo de un umbral mínimo, $min_{th}$, los paquetes son encolados normalmente (la probabilidad de descarte es nula).*
- *Por encima de $min_{th}$, la probabilidad de descarte de paquetes aumenta linealmente hasta llegar a la probabilidad $max_p$, que se alcanza cuando se llega a un umbral de ocupación $max_{th}$.*
- *Por encima de $max_{th}$, se descartan todos los paquetes (la probabilidad de descarte es uno).*

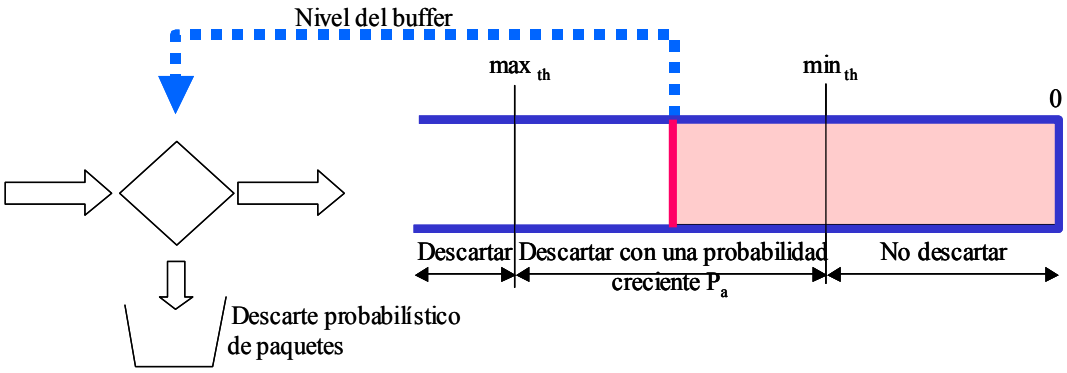



**Fig. 2.32.** Cola RED (Random Early Detection).

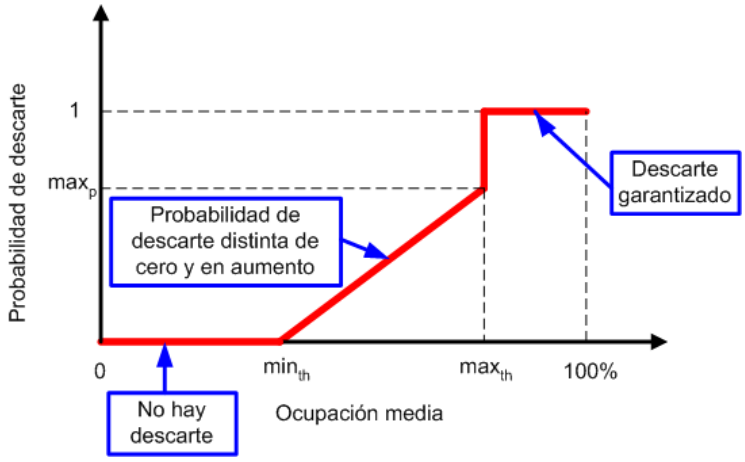



**Fig. 2.33.** Probabilidad de descarte RED (Random Early Detection).

Las tres fases anteriormente comentadas se suelen denominar fase normal, fase de prevención de congestión y fase de control de congestión. Es importante apreciar que RED comenzará a descartar paquetes antes de que la cola se haya llenado.

La ocupación promedio de la cola se recalcula cada vez que llega un nuevo paquete y está basada en un filtro paso bajo, una media móvil ponderada exponencialmente, EWMA (Exponentially Weighted Moving Average) de la ocupación instantánea de la cola.

Se formula de la siguiente manera:

$$Q_{media} = (1 - W_q) \times Q_{media} + Q_{inst} \times W_q , \qquad (2.13)$$

siendo $Q_{media}$ la ocupación promedio, $Q_{inst}$ la ocupación instantánea y $W_q$ el peso de la función de media móvil. El parámetro $W_q$ relaciona la ocupación promedio con la ocupación instantánea; la gracia está en elegir un valor para dicho parámetro que ignore las fluctuaciones a corto plazo para que no se produzcan pérdidas de paquetes innecesarias, pero que sea capaz de reaccionar a tiempo ante niveles de ocupación de la cola demasiado elevados, antes de que la latencia sea excesiva.

## *2.4.1.2 El algoritmo RIO o RIO-C*

Seguidamente se explica el algoritmo RIO o RIO-C, que ha sido utilizado como mecanismo para la diferenciación de servicios en redes fijas y también para poder adaptar la arquitectura DiffServ a ad hoc *(Véase la sección 2.4.1 La arquitectura de Servicios Diferenciados aplicada a redes ad hoc, pág. 71).*

El algoritmo RIO [83], también llamado en su forma original RIO-C (Random Early Detection with In/Out) (Coupled), es una variante de RED con bits In/Out.

Las *Fig. 2.34* y *Fig. 2.35* muestran un ejemplo de cola del tipo RIO-C y las funciones de probabilidad de descarte asociadas. Este algoritmo asume que los paquetes han sido etiquetados por un marcador como 'In' o bien como 'Out', poniendo o no a 1 un bit en la cabecera del paquete dependiendo de si éste se encuentra dentro o fuera del perfil después de haberle aplicado una política específica.

RIO-C [84] utiliza el mismo mecanismo que el algoritmo RED, pero está configurado en dos grupos de parámetros: uno para los paquetes 'In' y otro para los paquetes 'Out'. Cuando un paquete llegue a un router, éste lo marcará con la etiqueta apropiada y entonces le aplicará el grupo de parámetros pertinente. Si se trata de un paquete etiquetado como 'In', el router calculará $avg\_in$, la ocupación promedio de la cola virtual para los paquetes 'In'. En cambio, si se trata de un paquete etiquetado como 'Out', el router calculará $avg\_total$, la ocupación promedio de la cola física para todos los paquetes (tanto los marcados como 'In' como los marcados como 'Out').

La probabilidad de descarte de un paquete etiquetado como 'In' dependerá por tanto de $avg\_in$, mientras que la probabilidad de descarte de un paquete etiquetado como 'Out' dependerá de $avg\_total$.

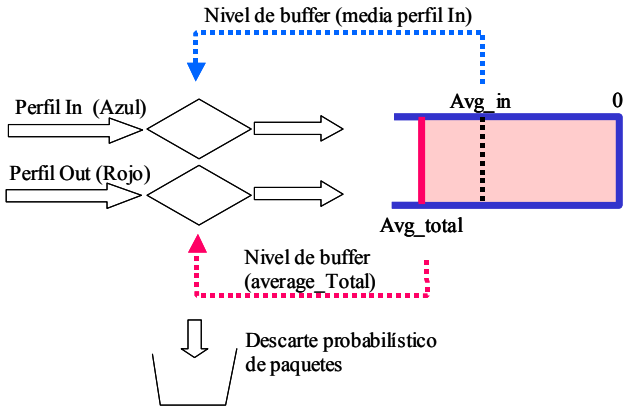



**Fig. 2.34.** Cola RIO-C (RED con In/Out) (Coupled).

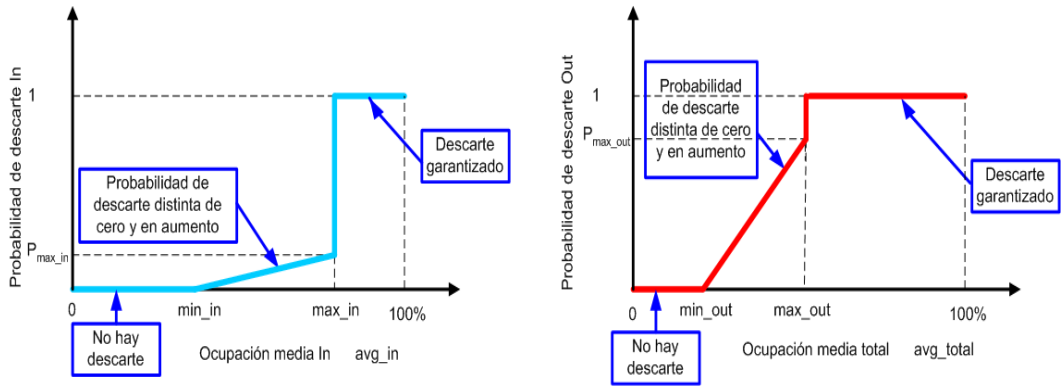



**Fig. 2.35.** Probabilidades de descarte RIO.

El objetivo de RIO-C es descartar preferiblemente aquellos paquetes marcados como 'Out' en el caso de que se produzca congestión. Esto se consigue seleccionando adecuadamente los valores de los parámetros $(\min\_in, \max\_in, P_{\max\_in})$, así como $(\min\_out, \max\_out, P_{\max\_out})$ *(Véase la Fig. 2.35).*

## *2.5 El modelo SWAN*

SWAN (Stateless Wireless Ad Hoc Networks) [85] es un modelo de red sin estados específicamente diseñado para proporcionar diferenciación de servicios en redes ad hoc inalámbricas. Es distribuido y utiliza una capa MAC best-effort. Además, es capaz de distinguir entre el tráfico de tiempo real y el tráfico best-effort. Los autores consideran en sus simulaciones que el tráfico de tiempo real es UDP y el tráfico best-effort es TCP o bien tráfico UDP cuyas sesiones de tiempo real han sido rechazadas.

En la *Fig. 2.36* se ilustra la pila de protocolos de cada nodo, en la cual se ha incluido la capa SWAN.

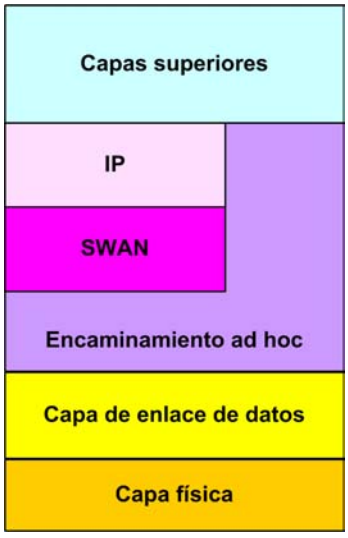



**Fig. 2.36.** Pila de protocolos en cada nodo de la red ad hoc.

Cada nodo dispone de mecanismos para poder clasificar los paquetes, conformarlos (si es necesario) y poner en marcha (en el caso de que sea preciso) un proceso denominado control de admisión.

El clasificador [86] de un nodo se halla situado entre las capas MAC e IP *(Véase la Fig. 2.37)* y es capaz de diferenciar entre paquetes best-effort y de tiempo real. Esto es posible porque cuando se desea enviar un nuevo flujo de paquetes de tiempo real, primero se somete a dichos paquetes a un proceso denominado control de admisión con el fin de determinar si existen recursos suficientes para poder aceptar la nueva sesión de tiempo real.

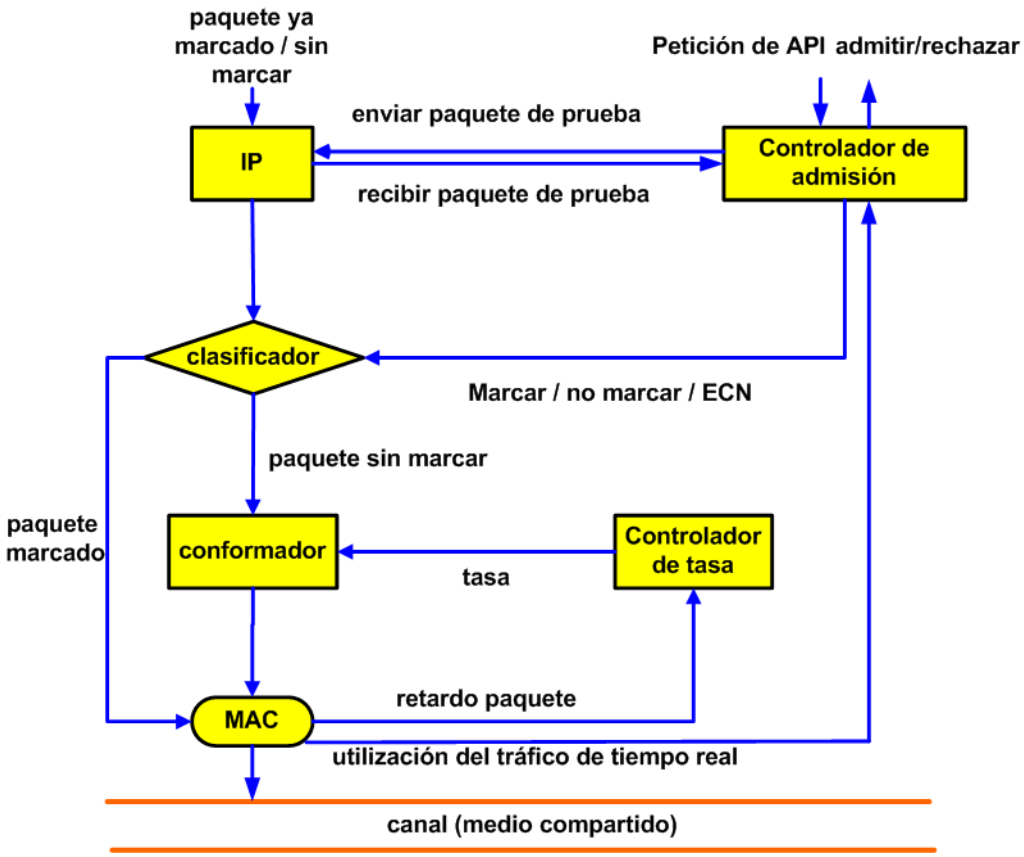



**Fig. 2.37.** Modelo SWAN.

En caso afirmativo, los paquetes de tiempo real son marcados como tales modificando el valor del campo DS (DiffServ) [68] en la cabecera IP de dichos paquetes, tal y como se muestra en la *Fig. 2.38*.

Los seis bits del campo DS determinan un codepoint que sirve para especificar el grado de servicio deseado. El valor del codepoint para los paquetes de tiempo real deberá ser escogido, mientras que el valor recomendado para los paquetes best-effort es la secuencia de bits '000000'.

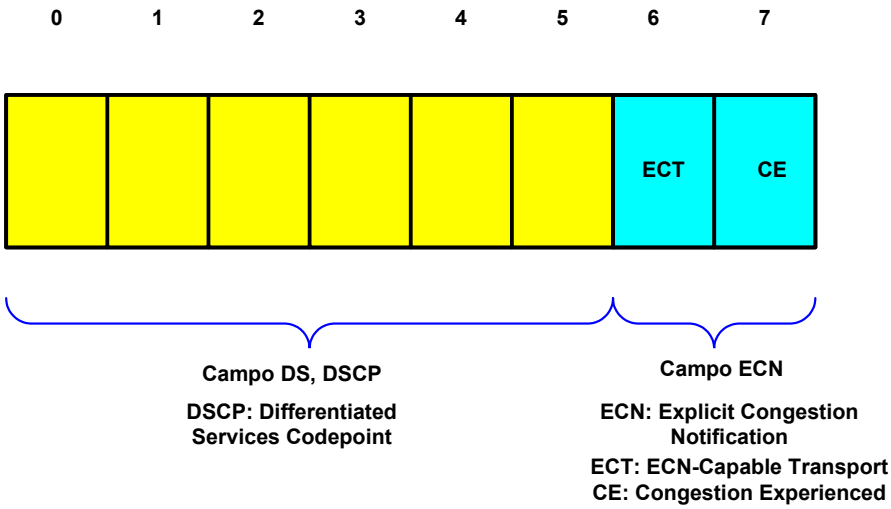



**Fig. 2.38.** Campos Differentiated Services Codepoint y ECN en IP.

El clasificador leerá este campo y sabrá por lo tanto distinguir entre los paquetes UDP de tiempo real que han sido marcados como tales y los paquetes UDP cuya sesión de tiempo real ha sido rechazada y que pasan entonces a ser considerados como tráfico best-effort. El tráfico TCP es tratado directamente como best-effort.

Entonces se retardarán en cada nodo únicamente los paquetes best-effort, utilizándose un conformador de tráfico denominado leaky bucket y de acuerdo con una tasa previamente calculada al haber aplicado el algoritmo de control de tasa AIMD (Additive Increase Multiplicative Decrease). Dicho algoritmo funciona de la siguiente manera:

Cada nodo mide los retardos de los paquetes a nivel de la capa MAC continua e independientemente de tal forma que el controlador de tasa o conformador de tráfico aprovecha dicha información. El retardo de los paquetes a nivel de la capa MAC para el modo de operación función de coordinación distribuida, DCF (Distributed Coordination Function) del IEEE 802.11 puede ser estimado como el tiempo total de espera antes de poder enviar un paquete, al que añadimos el tiempo transcurrido hasta que se recibe un ACK o reconocimiento positivo conforme el paquete de datos ha sido recibido correctamente si no ha habido colisión. Por ejemplo, si está activada la opción de envío RTS/CTS para reducir el problema del terminal escondido, el retardo $d$ de un paquete a nivel de la capa MAC se calcula como [87], [88]:

$$d = t_{espera} + t_{RTS} + t_{CTS} + t_{paquete} + t_{ACK} + 3t_{SIFS} + 3\tau, \tag{2.14}$$

donde $\tau$ es el retardo máximo de propagación. La variable $t_{espera}$ hace referencia al periodo de tiempo que el paquete ha estado esperando desde su llegada a la cola hasta que finalmente el paquete RTS ha podido ser transmitido (incluyéndose tanto el tiempo de backoff e IFS como posibles resoluciones de colisión).

Cada *T* segundos, el algoritmo de control de tasa comprueba si uno o más paquetes sufren retardos a nivel de la capa MAC superiores a un determinado umbral $D_{MAX}$. Si esto es así, el algoritmo reduce la tasa del conformador (y por lo tanto la tasa de transmisión) aplicando una tasa de decremento (decremento multiplicativo del *r* %). En caso contrario, el dispositivo móvil incrementa su tasa de transmisión gradualmente (incremento aditivo con una tasa de incremento de *c* Kbit/s). Si se comprueba que la tasa de transmisión actual (no la que acabamos de calcular) y la tasa de transmisión que hemos calculado (que coincide con la tasa del conformador) son muy distintas, entonces, para que el tráfico best-effort aumente su tasa de transmisión actual gradualmente y evitar así la transmisión de ráfagas incontroladas de tráfico best-effort, el controlador de tasa ajusta la tasa del conformador un *g* % por encima de la tasa actual en el caso de que la diferencia entre ambas tasas sea superior a un *g* % de la tasa actual. El control de la tasa se aplica en cada nodo de forma distribuida y sirve para restringir el ancho de banda del tráfico best-effort, de forma que las aplicaciones de tiempo real puedan utilizar el que necesiten. Por otro lado, gracias al control de tasa, el ancho de banda no usado por las aplicaciones de tiempo real puede ser aprovechado eficientemente por el tráfico best-effort.

El tráfico total (best-effort y de tiempo real) que es transportado a lo largo de un canal local (medio compartido) no puede sobrepasar una cierta tasa umbral para evitar que los paquetes sufran un retardo excesivo y se produzca saturación.

Nótese también que el retardo umbral $D_{MAX}$ debe ser seleccionado tras estudiar en profundidad los requisitos referentes al retardo para las aplicaciones de tiempo real en la red ad hoc donde va a aplicarse el modelo SWAN. Además, conviene remarcar que la tasa del conformador deberá ser ajustada cada *T* segundos, donde *T* tendrá un valor lo suficientemente pequeño como para ser capaz de responder con éxito a los cambios generados como consecuencia del comportamiento dinámico de la red ad hoc.

Por otro lado, se ha comentado que SWAN utiliza un proceso denominado control de admisión [91] basado en la fuente para el tráfico UDP de tiempo real *(Véase la Fig. 2.39)*. Cada nodo monitoriza independiente y localmente al escuchar el canal la tasa de los flujos agregados de tiempo real y esta información es aprovechada durante este proceso. Los nodos utilizan una media móvil ponderada exponencialmente, EWMA (Exponentially Weighted Moving Average), con el fin de poder eliminar fluctuaciones aleatorias en las medidas de las tasas. Además, los autores del modelo SWAN establecen que cuando un nodo desea establecer una nueva sesión de tráfico de tiempo real, el protocolo de encaminamiento AODV [108] se pone en funcionamiento para encontrar una nueva ruta hacia el destino seleccionado. El mecanismo de control

de admisión consiste en el envío de un paquete de prueba 'petición/respuesta' extremo a extremo a lo largo de la ruta encontrada *(Véase la Fig. 2.40)* para estimar la disponibilidad de ancho de banda local en cada nodo intermedio (se sobreentiende que cada nodo estima el ancho de banda local del enlace radio que usaría) y así determinar si es posible o no la aceptación de una nueva sesión de tiempo real extremo a extremo a lo largo de una ruta. El nodo fuente será el encargado de enviar el paquete de prueba 'petición' al nodo destino para estimar la disponibilidad de ancho de banda extremo a extremo. Esta petición será un paquete UDP que contendrá el campo "ancho de banda más restrictivo o 'cuello de botella'" *(Véase la Fig. 2.41).*

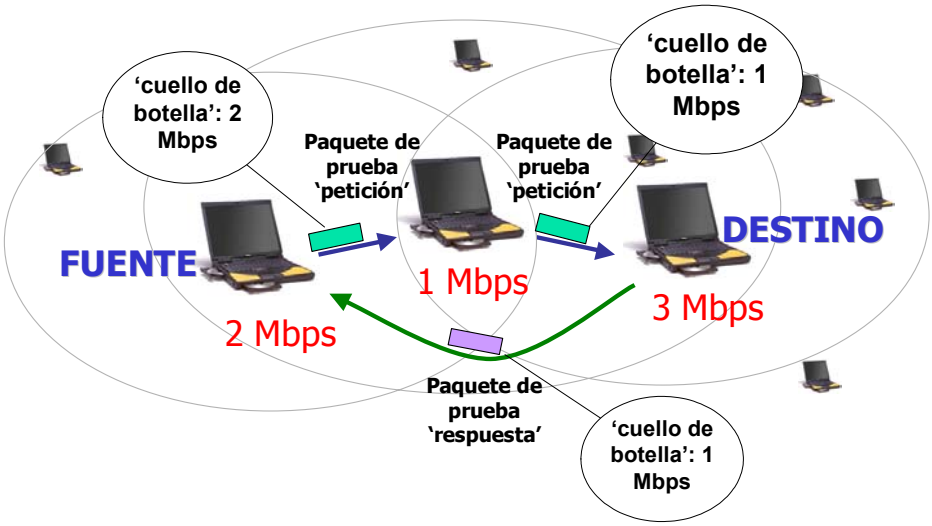



**Fig. 2.39.** Control de admisión.

Todos los nodos intermedios a lo largo de la ruta procesarán el paquete, comprobarán su disponibilidad de ancho de banda y actualizarán el campo "ancho de banda más restrictivo o 'cuello de botella'" si su propio ancho de banda es menor que el valor actual de dicho campo. El ancho de banda disponible en el canal (medio compartido) puede calcularse como la diferencia entre un cierto umbral de admisión y la tasa actual medida para el tráfico de tiempo real. El umbral de admisión para el tráfico de tiempo real está por debajo de la tasa umbral (número de recursos disponibles) para evitar la inanición del tráfico best-effort y permitir así que los tráficos de tiempo real y best-effort puedan compartir el canal eficientemente. Además, el canal debe poder tolerar variaciones de ancho de banda sin necesidad de suprimir las sesiones de tráfico de tiempo real si estas variaciones son menores.

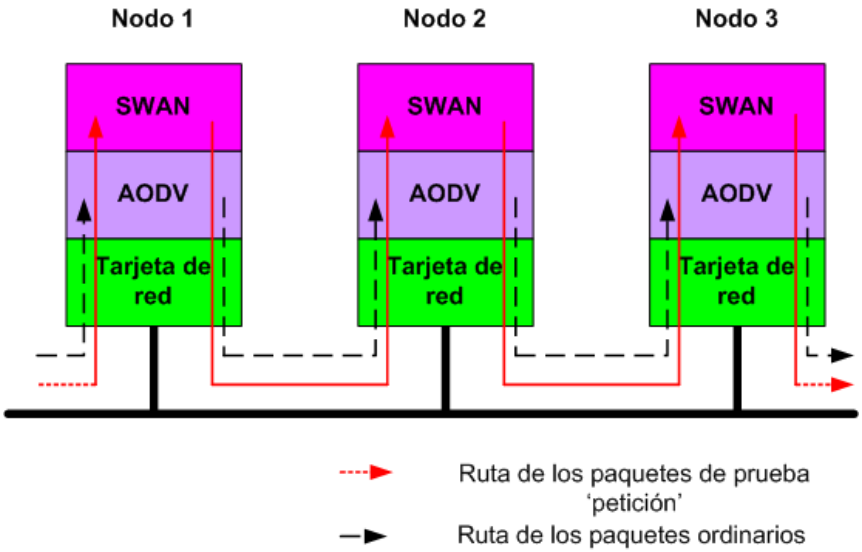



**Fig. 2.40.** Los paquetes de prueba 'petición' son empujados hasta la capa SWAN en cada nodo intermedio.

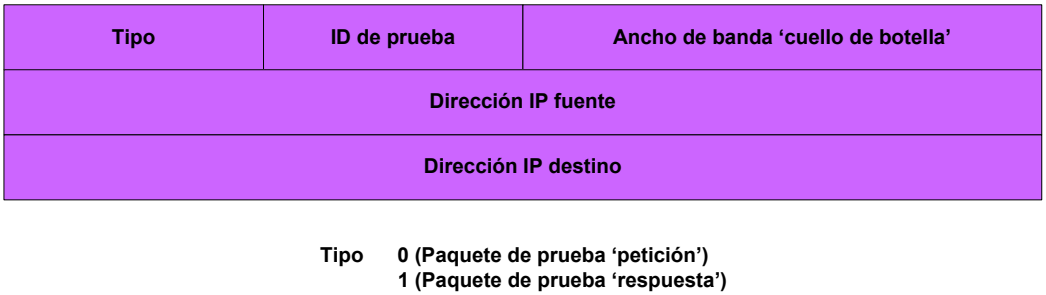



**Fig. 2.41.** Mensaje de prueba petición/respuesta.

Finalmente, el nodo destino recibe el paquete de prueba 'petición' y devuelve a la fuente un paquete de prueba 'respuesta' con una copia del ancho de banda que es el más restrictivo o 'cuello de botella' a lo largo de la ruta. Cuando la fuente recibe dicho paquete compara la disponibilidad del ancho de banda con los requisitos de ancho de banda necesarios para la nueva sesión de tiempo real y así decide si el nuevo flujo de tiempo real puede admitirse o debe rechazarse. Si se decide admitir dicho flujo, los paquetes son entonces marcados como paquetes de tiempo real y no serán conformados en los nodos intermedios por el leaky bucket utilizado para el tráfico best-effort, con lo cual no serán regulados. En caso contrario, dichos paquetes serán tratados como best-effort y sí que serán regulados.

Se dice que el modelo SWAN ofrece una calidad de servicio 'suave' o 'laxa' porque no existe ninguna garantía de que un flujo de tiempo real pueda satisfacer sus requisitos de calidad de servicio durante todo el tiempo de conexión. En ciertas ocasiones será preciso readmitir o rechazar las sesiones de tiempo real establecidas debido a los motivos siguientes:

- *Tanto la carga de tráfico como la topología de red varían dinámicamente, de forma que las sesiones de tiempo real pueden no ser capaces de cumplir con sus requisitos de ancho de banda y retardo, por lo que deberán ser rechazadas o readmitidas de nuevo.*
- *La interferencia de nodos vecinos, la calidad de los enlaces (variable con el tiempo), etc. puede causar fading, el cual reduce la tasa umbral disponible en un canal local compartido y puede también convertir en necesario el restablecer de nuevo o rechazar las sesiones de tiempo real.*

El mecanismo de notificación de congestión explícita, ECN, (Explicit Congestion Notification) denominado 'regulación basada en la red' regula dinámicamente las sesiones de tiempo real admitidas y trata por tanto de restablecerlas en el caso de que sea necesario tal y como se muestra en la *Fig. 2.42* y como se explicará a continuación. Cuando un nodo móvil detecta que la tasa umbral en un canal local compartido es menor que la tasa actual para el tráfico de tiempo real, deduce que se está produciendo congestión (condiciones de sobrecarga) y pone en marcha el mecanismo ECN. Lo que hace este algoritmo es favorecer que los nodos congestionados seleccionen aleatoriamente cada *T* segundos un subconjunto de paquetes de tiempo real que reciben y marquen los bits ECN en la cabecera IP de dichos paquetes. Concretamente, se marcan los bits ECN-Capable Transport y Congestion Experienced *(Véase la Fig. 2.38)*, que son los dos últimos bits del campo TOS (Type of Service) en IPv4 o bien del campo Clase de Tráfico en IPv6. En nuestro caso, el bit del campo ECN-Capable Transport de los paquetes UDP es puesto directamente a 1 y, además, cuando un nodo congestionado decide seleccionar un paquete que pertenece a una sesión de tiempo real y marcarlo, se pondrá también a 1 el bit del campo Congestion Experienced. El destino monitoriza los paquetes con los bits ECN marcados e informa a la fuente enviando un mensaje de regulación como el que se muestra en la *Fig. 2.43.*

Entonces el nodo fuente trata de restablecer la sesión de tiempo real de acuerdo con sus necesidades de ancho de banda enviando nuevamente un paquete de prueba 'petición' al ejecutar el control de admisión. Si no lo consigue, la sesión de tiempo real es rechazada y los paquetes son tratados como best-effort.

Existe otro mecanismo de congestión explícita denominado 'regulación basada en la fuente' que actúa del mismo modo que la 'regulación basada en la red' con la diferencia de que cuando un nodo móvil detecta que hay congestión marca los bits ECN en la cabecera IP de todos los paquetes de tiempo real que recibe. Además, cuando el destino monitorice todos los paquetes con los bits ECN marcados y envíe

mensajes de regulación para informar a la fuente, ésta no tratará de restablecer inmediatamente los flujos de tiempo real, sino que lo hará paulatinamente tras haber esperado un tiempo aleatorio.

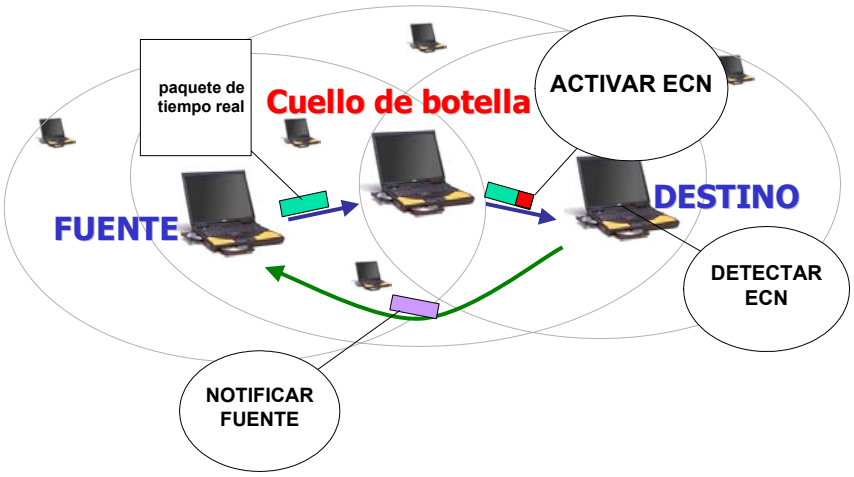



**Fig. 2.42.** Regulación.

| Tipo | ID de mensaje | CU (Currently Unused) (Sin usar actualmente) |
|---|---|---|
| Dirección IP fuente | | |
| Dirección IP destino | | |

**Tipo** **2 (Mensaje de regulación usando el algoritmo basado en la fuente)**
**3 (Mensaje de regulación usando el algoritmo basado en la red)**

**Fig. 2.43.** Mensaje de regulación.

De los dos mecanismos de congestión explícita presentados, el denominado 'regulación basada en la red' consigue un mejor rendimiento de red porque el mecanismo 'regulación basada en la fuente' añade una mayor latencia al introducir un tiempo de espera antes de readmitir o rechazar una sesión de tiempo real. Esta circunstancia provoca que exista una mayor congestión y en consecuencia se rechaza un número mayor de sesiones de tiempo real, con lo que la red termina consumiendo menos recursos de los que posee.

Un nodo móvil detectará que se está produciendo congestión (condiciones de sobrecarga) debido básicamente a tres motivos fundamentales:

- *Fading:*

  Consiste en un desvanecimiento de la señal causado por cambios en las características del camino de propagación con el tiempo debido a la interferencia de un nodo con sus vecinos, al estado de la atmósfera, etc. La consecuencia será una reducción del ancho de banda del canal radio.

- *Movilidad:*

  Debido a la movilidad experimentada por los nodos pertenecientes a una red ad hoc, es posible que ciertos flujos que ya habían sido admitidos con anterioridad se vean obligados a ser reconducidos a través de nuevas rutas con nodos intermedios que carezcan de suficientes recursos como para transportar el tráfico con éxito.

- *Falsa admisión:*

  Este suceso ocurre cuando varios nodos fuente inician a la vez un proceso de control de admisión con la intención de comprobar si existen recursos para sus flujos respectivos. Los nodos intermedios actúan de manera descentralizada sin mantener información de estado por cada flujo, de tal forma que puede ser posible que dichos nodos intermedios sean nodos comunes para los flujos que se desean establecer y contesten afirmativamente a todas las fuentes indicando que sí que hay recursos. Aunque por ejemplo sólo haya recursos para establecer una sesión de tiempo real y no varias a la vez, un nodo intermedio que reciba varias peticiones de establecimiento de sesión deberá contestar que sí existen recursos a todas las peticiones, mientras que no se establezca definitivamente alguna de ellas. Pero lo que puede llegar a suceder es que cada fuente reciba una respuesta afirmativa por parte del control de admisión y decida iniciar su sesión de tiempo real, de manera que todas las sesiones de estos flujos comiencen a la vez y en ese caso los nodos intermedios no cuenten con recursos suficientes para poder garantizar la calidad de servicio de semejante número de sesiones. Con la ayuda del mecanismo de congestión explícita denominado 'regulación basada en la fuente' pretende corregirse este problema distinguiendo entre sesiones de tiempo real antiguas y recientemente establecidas y obligando a la fuente a rechazar aquellas sesiones falsamente admitidas de manera reciente. Dichas sesiones son identificadas al mantenerse una cierta información referente al estado de un flujo. Con la ayuda del mecanismo de congestión explícita denominado 'regulación basada en la red' se pretende también solucionar el problema de la falsa admisión, indicándose en este caso mediante un bit del campo TOS si la sesión de tiempo real es nueva o antigua y convirtiéndose todas las sesiones nuevas marcadas como tales, al cabo de un cierto tiempo, en sesiones antiguas. Los nodos intermedios se encargarán de gestionar este proceso, introduciéndose una cierta complejidad en la red.

Los nodos intermedios no mantienen ninguna información de estados por flujo ni realizan ninguna reserva de recursos, evitándose así la introducción de una señalización compleja y resultando un sistema más simple y escalable.

## *2.6 Comparación entre distintos modelos de calidad de servicio y elección de un modelo de calidad de servicio para redes ad hoc aisladas*

En las secciones anteriores del capítulo 2 se han estudiado en profundidad los modelos de calidad de servicio más destacados para redes ad hoc aisladas. Después de haberlos comparado en busca de aquel esquema que diferenciara servicios más eficientemente, en esta tesis doctoral nos hemos decantado por el modelo SWAN, debido fundamentalmente a los siguientes motivos:

- *Comparación entre los mecanismos de calidad de servicio a nivel de la capa MAC para IEEE 802.11 y el modelo SWAN*

  La ventaja fundamental del modelo SWAN frente a los distintos mecanismos de calidad de servicio a nivel de la capa MAC para IEEE 802.11 es que SWAN no necesita una capa MAC capaz de proporcionar QoS. SWAN utiliza una capa MAC best-effort. Como este modelo puede diferenciar servicios con independencia de la capa MAC que se esté utilizando (algo que no sucede con los mecanismos de calidad de servicio a nivel de la capa MAC para IEEE 802.11), resulta ser el modelo ideal para adaptarse a las capas de enlace de datos y física de los distintos estándares inalámbricos. Así, con SWAN será posible dar soporte a redes heterogéneas con capas MAC correspondientes a distintas tecnologías.

- *Comparación entre los modelos de calidad de servicio INSIGNIA y SWAN*

  La principal ventaja del modelo SWAN frente al modelo INSIGNIA es que SWAN no mantiene ningún estado por flujo en los nodos intermedios a lo largo de la ruta entre la fuente y el destino en la red ad hoc. Por consiguiente, SWAN no utiliza una señalización compleja para establecer, actualizar o eliminar información relacionada con los estados por flujo, al contrario que el modelo INSIGNIA, el cual resulta mucho más complicado.

  Cuando aumenta la movilidad en una red ad hoc, resulta especialmente difícil y problemático manejar información referente a estados. SWAN carece de estos problemas; es un modelo sencillo, robusto y además escalable, a diferencia del modelo INSIGNIA, donde el aumento de nodos en la red ad hoc

puede suponer un conflicto de envergadura. Por otra parte, el modelo SWAN no se ve afectado en su funcionamiento por cambios en la topología o la carga de la red, ni siquiera porque se produzcan roturas de enlaces.

- *Comparación entre los modelos de calidad de servicio FQMM y SWAN*

  La principal ventaja del modelo SWAN frente al modelo FQMM es que la arquitectura de red resulta mucha más sencilla. En FQMM se presentan una serie de módulos que forman parte de su arquitectura y que operarán eficientemente únicamente en el caso de que estén funcionando todos a la vez.

  FQMM utiliza IntServ para proporcionar QoS por flujo al tráfico de más alta prioridad, mientras que usa DiffServ para diferenciar servicios entre el tráfico de más baja prioridad. Por este motivo, FQMM sufrirá las mismas desventajas que el modelo IntServ (señalización compleja y falta de escalabilidad) y también las desventajas propias del modelo DiffServ (es escalable, pero no se garantiza una calidad de servicio extremo a extremo para los flujos individuales).

  Por otro lado, la reserva de recursos en el modelo FQMM se hace por flujo para los paquetes pertenecientes a una clase prioritaria y, para el resto, el flujo de paquetes correspondiente se agrega a los demás flujos pertenecientes a la misma clase de tráfico para formar un agregado de flujo y pasa a efectuarse un tratamiento por clase. Este tratamiento de los paquetes es diferente en el modelo SWAN, donde no se mantiene en los nodos intermedios una reserva de recursos por flujo ni por agregado de flujo, resultando un sistema más manejable y escalable. En FQMM no se especifica cuál es el número de sesiones de tiempo real que pueden mantenerse por flujo, pero si este número crece mucho, el sistema no va a ser capaz de soportarlo; no se podrán mantener los parámetros de calidad de servicio para dichas sesiones y existirán problemas de escalabilidad. Además, los nodos interiores actúan como routers, reenviando los paquetes de los demás nodos de acuerdo con un PHB específico definido por el campo DSCP. Resultará difícil codificar el PHB en el campo DSCP si a las sesiones de tiempo real se les da un tratamiento por flujo, pues el campo DSCP está limitado a 6 bits.

- *Comparación entre los modelos de calidad de servicio DiffServ (aplicado a redes ad hoc) y SWAN*

La tabla siguiente (*Tabla 2.6)* refleja las principales diferencias existentes entre el modelo de calidad de servicio SWAN y la arquitectura DiffServ aplicada a redes ad hoc, tal y como fue desarrollada por los autores en [80].

| SWAN | DiffServ |
|---|---|
| **Soporta dos clases de tráfico: de tiempo real y best-effort.** | **Soporta hasta 16 clases de tráfico (4 bits del DSCP).** |
| **Control de la tasa del tráfico best-effort basándose en la información acerca del retardo de los paquetes en la capa MAC.** | **Control de la tasa del tráfico basándose en la congestión (tamaño de la cola).** |
| **Garantías de QoS 'suaves' o laxas. Los flujos de tiempo real se pueden degradar a best-effort.** | **Se pueden descartar paquetes de los flujos de tiempo real si se produce congestión.** |
| **Conformador sólo para el tráfico best-effort.** | **Sistema de colas con una disciplina de servicio asignada para manejar diversas clases de tráfico.** |
| **Control de admisión para el tráfico de tiempo real.** | **Actualmente no existe control de admisión.** |
| **Mecanismo ECN usado para controlar la tasa del tráfico de tiempo real.** | **Monitorización y control de policía para controlar la tasa del tráfico de tiempo real.** |

**Tabla 2.6.** Diferencias entre la arquitectura de los modelos DiffServ y SWAN.

Con el objeto de estudiar en mayor profundidad las diferencias a la hora de proporcionar calidad de servicio en una red ad hoc utilizando el modelo SWAN o bien la arquitectura DiffServ, los autores en [92] han procedido a realizar simulaciones.

Para ello han comparado el modelo SWAN desarrollado en [91] con el modelo DiffServ desarrollado en [80] por ellos mismos.

En dichas simulaciones, 51 nodos que usan el modelo de movimiento 'random waypoint' se mueven en un área de 300 m x 1500 m. Se distingue entre tráfico best-effort TCP y tráfico de tiempo real UDP. El tráfico TCP está constituido por aplicaciones FTP y tráfico Web, mientras que las aplicaciones de voz y vídeo forman el tráfico UDP. Se seleccionan 4 y 5 nodos para que actúen como fuentes TCP y UDP respectivamente. Se usa DSR [104] como protocolo de encaminamiento. En las simulaciones se calcula el throughput medio así como el retardo extremo a extremo experimentado por los paquetes

pertenecientes a los flujos de tiempo real. También se mira el 'goodput' de los flujos TCP para averiguar qué porción del ancho de banda existente puede ser empleado por el tráfico best-effort.

La *Fig. 2.44* muestra el goodput disponible para una conexión Web y el thoughput medio para una conexión de video. Claramente se observa que el modelo SWAN supera a DiffServ. Para encontrar un motivo acerca del por qué de estas diferencias haría falta recordar que SWAN utiliza un algoritmo de control de tasa para el tráfico best-effort (TCP) con el fin de mantener la calidad de servicio de las conexiones de tiempo real extremo a extremo que actúa dinámicamente, mientras que DiffServ utiliza toda una serie de parámetros que son estáticos a la hora de gestionar las colas mediante RIO-C y utilizar una disciplina de servicio para diferenciar servicios.

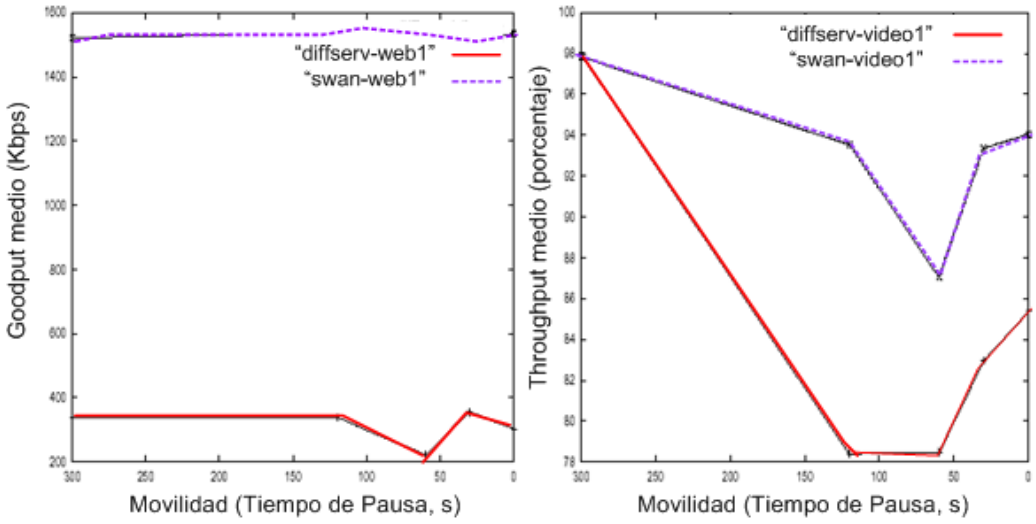



**Fig. 2.44.** (Izquierda) Goodput medio para un flujo TCP medido en Kbps en función de la movilidad. (Derecha) Throughput medio de un flujo UDP en función de la movilidad [92].

Los autores han comprobado que si se utilizan otro tipo de aplicaciones para los flujos de tiempo real o bien se emplean otros protocolos de encaminamiento, SWAN continuará realizando una diferenciación de servicios superior a DiffServ.

La *Fig. 2.45* muestra el retardo extremo a extremo de los paquetes pertenecientes a un flujo TCP y el retardo extremo a extremo de los paquetes pertenecientes a un flujo UDP.

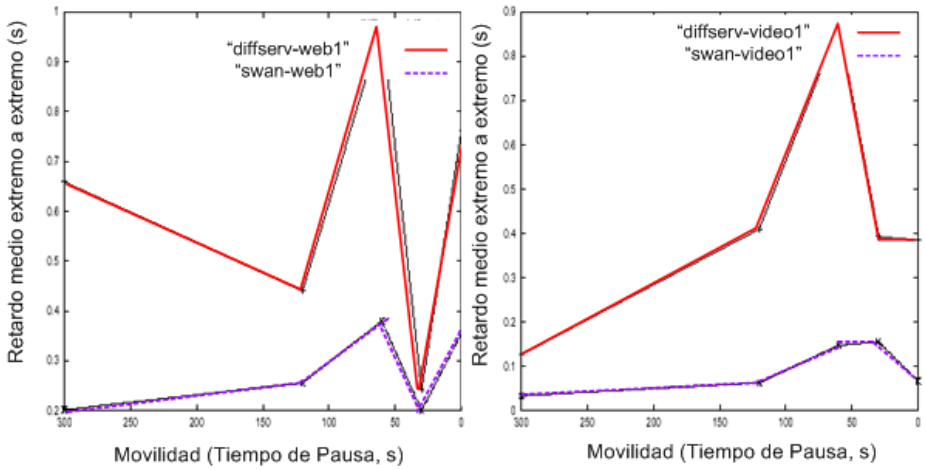



**Fig. 2.45.** Retardos medios extremo a extremo sufridos por los paquetes de un flujo TCP (izquierda) y de un flujo UDP (derecha) [92].

SWAN es capaz de mantener un retardo extremo a extremo inferior a 150 ms para el video, lo cual es esencial. Como en DiffServ no se controla, a diferencia del modelo SWAN, la tasa de los paquetes best-effort, aumentará el retardo de los paquetes de tiempo real, que permanecerán largo tiempo almacenados en las colas esperando para poder acceder al medio.

En cambio, el modelo SWAN es capaz de mantener la calidad de servicio del tráfico de tiempo real retardando el acceso a la capa MAC (y por lo tanto al medio) del tráfico best-effort. Si se detecta congestión, las sesiones de tiempo real con problemas deben ser readmitidas o degradadas a best-effort, manteniéndose de esta forma retardos bajos para el resto de flujos.

## *2.7 Conclusiones*

En el Capítulo 2 se han presentado los modelos de calidad de servicio fundamentales que existen para redes ad hoc aisladas. Se han introducido no solamente los protocolos que diferencian servicios a nivel de la capa MAC del IEEE 802.11, sino también aquellos modelos de calidad de servicio más destacados que actúan en capas superiores como son los modelos INSIGNIA, FQMM, la arquitectura de Servicios Diferenciados aplicada a redes ad hoc y el modelo SWAN. Finalmente, se ha realizado un análisis comparativo entre todos ellos con el fin de decidir qué modelo de calidad de servicio para redes ad hoc aisladas se adapta mejor para nuestros propósitos y utilizarlo con posterioridad en el análisis entre redes ad hoc interconectadas con redes IP fijas.

SWAN resulta más adecuado como modelo de calidad de servicio para la diferenciación de servicios en una red ad hoc de acuerdo con las razones expuestas en la sección 2.6. *(Véase la sección 2.6 Comparación entre distintos modelos de calidad de servicio y elección de un modelo de calidad de servicio para redes ad hoc aisladas, pág. 86).* Estas razones son que SWAN utiliza una capa MAC best-effort que sí que le permite, a diferencia de los mecanismos de calidad de servicio a nivel de la capa MAC para IEEE 802.11, dar soporte a redes heterogéneas con capas MAC correspondientes a diferentes tecnologías de estándares inalámbricos. SWAN no mantiene, a diferencia del modelo INSIGNIA, un sistema de estados por flujo, con lo cual resulta mucho menos complejo y además es escalable. Si comparamos SWAN con el modelo FQMM, su principal ventaja será la sencillez y escalabilidad. También se han realizado simulaciones para comparar la arquitectura de Servicios Diferenciados aplicada a redes ad hoc directamente con el modelo SWAN. Los resultados han demostrado que como en DiffServ aplicado a redes ad hoc se utilizan toda una serie de parámetros que son estáticos a la hora de gestionar ciertas colas mediante RIO-C, SWAN supera también en rendimiento a este modelo.

SWAN tiene la ventaja de que puede diferenciar correctamente servicios en redes ad hoc heterogéneas con capas MAC correspondientes a distintas tecnologías. Sin embargo, esto no significa que los mecanismos de calidad de servicio a nivel de la capa MAC para IEEE 802.11 no tengan su utilidad. De hecho, resultaría muy interesante implementar un modelo de calidad de servicio como SWAN en una red ad hoc donde el protocolo a nivel de la capa MAC fuera por ejemplo el IEEE 802.11e para observar la interacción entre ambos protocolos y mejorar la diferenciación de servicios.

# 3 Protocolos de encaminamiento en redes ad hoc aisladas

El diseño de protocolos de encaminamiento para redes ad hoc [94] se ha convertido en un importante desafío debido a la escasez de recursos y a las condiciones cambiantes (tamaño de red, densidad de tráfico, particiones de red, etc.) de este tipo de redes.

Los algoritmos de vector distancia [95] [96] y estado enlace [101] usados tradicionalmente en redes fijas no pueden ser aplicados porque el frecuente intercambio de mensajes de actualización obligaría a consumir una parte importante de los recursos de ancho de banda y capacidad de batería de la red ad hoc. Asimismo, los protocolos de encaminamiento en redes ad hoc deben operar con tasas de error para los enlaces elevadas y topologías muy dinámicas, en contraposición a las redes fijas.

En muchas ocasiones [1], la investigación de los protocolos de encaminamiento se centra en redes homogéneas, donde todos los nodos tienen los mismos recursos y capacidades; pero debería tenerse en cuenta que las redes ad hoc son heterogéneas y existe una gran cantidad de parámetros diversos relacionados con el encaminamiento tales como disponibilidad de energía, movilidad del terminal o tamaño de los buffers.

De hecho, se deberían diseñar protocolos de encaminamiento para redes ad hoc que satisficieran básicamente los siguientes criterios [93] [99]:

- *Señalización mínima*

  La reducción de los mensajes de control ayuda a conservar la capacidad de batería y la comunicación de los nodos.

- *Tiempo de procesamiento mínimo*

  Se requieren algoritmos con cálculos computacionales que no sean excesivamente complejos para disminuir el tiempo de procesamiento y alargar de esta forma el tiempo de vida de la batería.

- *Mantenimiento en condiciones de topología dinámica*

  El algoritmo deberá ser capaz de localizar una nueva ruta rápidamente cuando se rompe un enlace, ya sea debido al fading o a la movilidad.

- *Modo de operación distribuido:*

  Propiedad esencial de las redes ad hoc.

- *Libre de bucles:*

  Se pretende evitar el problema de tener paquetes circulando perdidos por la red.

- *Modo de operación 'sleep' o inactivo*

  Los protocolos de encaminamiento deberán estar preparados para afrontar aquellos periodos de tiempo en los cuales los nodos frenan su actividad y permanecen inactivos para ahorrar energía.

- *Soporte de enlaces unidireccionales*

  Los protocolos de encaminamiento en muchas ocasiones han sido diseñados y funcionan correctamente sólo con enlaces bidireccionales y esto no debería ser así, porque en la práctica podemos encontrarnos con la existencia de enlaces unidireccionales que sean clave para el intercambio de información en redes ad hoc.

Se han diseñado numerosos protocolos de encaminamiento para redes ad hoc atendiendo a estos criterios. En las próximas secciones *(3.1 y 3.2)* se realizan dos posibles clasificaciones de los protocolos de encaminamiento con el fin de dividirlos en una serie de categorías y se introducen ejemplos de cada categoría. La primera clasificación está relacionada con la construcción de rutas bajo demanda o no y la segunda clasificación está vinculada al soporte de calidad de servicio. Estas dos clasificaciones no son las únicas existentes; hubiera sido posible considerar otras clasificaciones atendiendo a si los protocolos de encaminamiento son unicast o multicast, son de camino único o de camino múltiple, etc.

En la sección *3.3* se presenta la relación existente entre el encaminamiento y la disponibilidad de energía.

En la sección *3.4* se establece una comparación entre distintos protocolos de encaminamiento y se elige aquel protocolo de encaminamiento que resulta ser más adecuado para su implementación en redes ad hoc aisladas.

Finalmente, en la sección *3.5* se presenta una primera contribución consistente en el diseño e implementación de un protocolo de encaminamiento para la mejora de la supervivencia en redes ad hoc aisladas.

## *3.1 Protocolos de encaminamiento proactivos, reactivos e híbridos*

En general, podemos clasificar los protocolos de encaminamiento en tres categorías diferentes [25]:

- *Protocolos proactivos o globales o basados en tablas*
- *Protocolos reactivos o bajo demanda*
- *Protocolos híbridos*

En la *Fig. 3.1* se observa esta clasificación de los protocolos de encaminamiento, mencionándose aquellos protocolos específicos más importantes, que son ejemplos de protocolos de encaminamiento de camino único y unicast. Veamos en qué se diferencian los distintos tipos de protocolos de encaminamiento, además de explicar los más destacados.

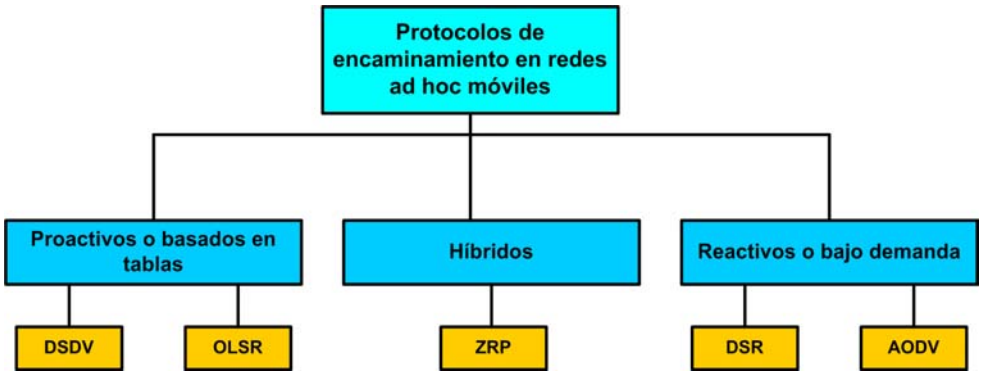



**Fig. 3.1.** Clasificación de los protocolos de encaminamiento en redes ad hoc.

## *3.1.1 Protocolos de encaminamiento proactivos*

Los protocolos de encaminamiento proactivos [25] [94] o basados en tablas son aquellos algoritmos que mantienen información de encaminamiento actualizada de cada nodo a cada nodo de la red. Dicha información se halla almacenada en las denominadas tablas de encaminamiento, las cuales son actualizadas periódicamente y cuando se producen cambios en la topología de la red.

Este tipo de protocolos operará en redes en las cuales se necesite que el procedimiento de Descubrimiento de Ruta no tenga una latencia excesiva y se pueda asumir el funcionamiento de un protocolo de este tipo en cuanto al consumo de recursos tales como ancho de banda y energía.

Los protocolos de encaminamiento proactivos diferirán los unos de los otros entre sí dependiendo de cómo se actualiza la información de encaminamiento, cómo se detecta dicha información, cuál es el número de tablas de encaminamiento y qué información contienen dichas tablas.

Se han propuesto esquemas basándose en dos protocolos de encaminamiento tradicionales diseñados para redes fijas: los algoritmos vector distancia y estado enlace.

A continuación se presentan dos protocolos de encaminamiento proactivos destacados *(Véase la Fig. 3.1)*:

- *Destination-sequenced Distance Vector (DSDV) (Véase la sección 3.1.1.1 Destination-sequenced Distance Vector (DSDV), pág. 96).*

  Está basado en el algoritmo de vector distancia.

- *Optimized Link State Routing (OLSR) (Véase la sección 3.1.1.2 Optimized Link State Routing (OLSR), pág. 100).*

  Está basado en el algoritmo de estado enlace.

## *3.1.1.1 Destination-sequenced Distance Vector (DSDV)*

Destination-sequenced Distance Vector (DSDV) es protocolo de encaminamiento proactivo de vector distancia.

Este protocolo de encaminamiento se basa en el algoritmo clásico de vector distancia o Bellman-Ford [95] [96], el cual ha sido mejorado para evitar bucles. Sirve para encontrar a partir del algoritmo de vector distancia aquella ruta que proporciona la trayectoria más corta posible hacia un destino [56]

Cada nodo dentro de la red ad hoc mantiene una tabla de encaminamiento con la información siguiente para cada destino [97]:

- *Dirección IP destino*
- *Número de secuencia del destino*
- *Próximo salto hacia el destino (dirección IP)*
- *Coste de la ruta hacia el destino (en número de saltos)*
- *Tiempo de instalación: Sirve para eliminar rutas antiguas*

Cada nodo envía periódicamente en modo broadcast su tabla actualizada a sus vecinos [98]:

- *Cada nodo añade su número de secuencia cuando envía su tabla de encaminamiento*
- *Cuando los demás nodos reciben dicha información actualizan sus propias tablas de encaminamiento*

Las tablas de encaminamiento también pueden enviarse si se producen cambios en la topología de la red (creación o rotura de enlaces). En este caso, la información de actualización que viaja en los mensajes de encaminamiento es la siguiente:

- *Dirección IP destino*

- *Coste de la ruta hacia el destino (en número de saltos)*
- *Número de secuencia del destino*

Los nodos utilizan los números de secuencia del destino para poder distinguir entre rutas antiguas y rutas más recientes hacia un mismo destino [99]. Un nodo incrementa su número de secuencia cuando se produce un cambio a nivel local en la topología de sus vecinos (se crea o elimina un enlace). Aquella ruta hacia un destino que tenga asociada el número de secuencia del destino más reciente (el mayor) será la que se considerará válida. En el caso de que existan dos rutas con el mismo número de secuencia del destino hacia un destino, prevalecerá aquella cuyo número de saltos sea menor.

Se usan dos tipos de paquetes de actualización de rutas [94]:

- *Full dump*

  Transportan toda la información contenida en la tabla de encaminamiento de un nodo. Este tipo de paquetes se envía muy raramente.

- *Incremental*

  Este tipo de paquetes transporta únicamente la información contenida en la tabla de encaminamiento de un nodo que ha variado desde que el último paquete 'full dump' fue enviado. Estos paquetes son enviados con mayor frecuencia; así se evita una señalización y un consumo de ancho de banda excesivos debido al envío periódico de tablas de encaminamiento enteras y actualizadas.

Sin embargo, a pesar de la introducción de los paquetes 'incremental', DSDV continúa teniendo problemas debido al exceso de señalización requerida, que crece de acuerdo con $O(N^2)$, siendo $N$ el número de nodos de la red. Por esta razón, el protocolo no será escalable.

DSDV utiliza un sistema para sofocar las fluctuaciones en el encaminamiento. Para evitar que un nodo anuncie un cambio de ruta para encaminar paquetes cuando existe una ruta mejor pero que todavía se está descubriendo, se precisa que cada nodo espere un tiempo fijo antes de anunciar una nueva ruta hacia el destino con un coste menor. El tiempo fijo a esperar será calculado como el tiempo medio que se tarda en conseguir todos los mensajes de actualización de una ruta. Así, un nodo recibirá todos los mensajes con información acerca de cambios en la ruta antes de propagar cualquiera de dichos cambios. De esta manera, los nodos vecinos reducen la utilización del ancho de banda y el consumo de potencia.

El siguiente ejemplo *(Véase la Fig. 3.2)* ilustra una red ad hoc que utiliza el protocolo de encaminamiento DSDV. El nodo $M_4$ tendrá una tabla de encaminamiento como la de la *Tabla 3.1*.

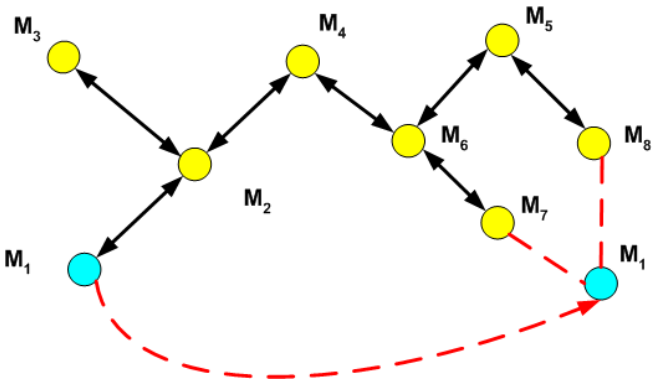



**Fig. 3.2.** Red ad hoc donde existe movilidad.

La información de la tabla de encaminamiento que se enviará en el mensaje de encaminamiento de actualización se ilustra en la *Tabla 3.2*.

| *Destino* | *Próximo salto* | *Coste* | *Número de secuencia* | *Tiempo de instalación* |
|---|---|---|---|---|
| $M_1$ | $M_2$ | 2 | S406_$M_1$ | T001_M4 |
| $M_2$ | $M_2$ | 1 | S128_$M_2$ | T001_M4 |
| $M_3$ | $M_2$ | 2 | S564_$M_3$ | T001_M4 |
| $M_4$ | $M_4$ | 0 | S710_$M_4$ | T001_M4 |
| $M_5$ | $M_6$ | 2 | S392_$M_5$ | T002_M4 |
| $M_6$ | $M_6$ | 1 | S076_$M_6$ | T001_M4 |
| $M_7$ | $M_6$ | 2 | S128_$M_7$ | T002_M4 |
| $M_8$ | $M_6$ | 3 | S050_$M_8$ | T002_M4 |

**Tabla 3.1.** Tabla de encaminamiento para el nodo $M_4$.

| *Destino* | *Coste* | *Número de secuencia* |
|---|---|---|
| $M_1$ | 2 | S406_$M_1$ |
| $M_2$ | 1 | S128_$M_2$ |
| $M_3$ | 2 | S564_$M_3$ |
| $M_4$ | 0 | S710_$M_4$ |
| $M_5$ | 2 | S392_$M_5$ |
| $M_6$ | 1 | S076_$M_6$ |
| $M_7$ | 2 | S128_$M_7$ |
| $M_8$ | 3 | S050_$M_8$ |

**Tabla 3.2.** Tabla de encaminamiento del nodo $M_4$ de actualización.

Si se produce un cambio en la topología de la red y el nodo $M_1$ se mueve hasta alcanzar la nueva posición que se ilustra en la *Fig. 3.2*, la tabla de encaminamiento para el nodo $M_4$ y el mensaje de encaminamiento de actualización serán los que se muestran en la *Tabla 3.3* y la *Tabla 3.4* respectivamente. Solamente existe una entrada de encaminamiento nueva para el nodo destino $M_1$, pero durante este intervalo de tiempo sí que se han recibido nuevos números de secuencia de destinos asociados a otras entradas de la tabla. El nodo $M_4$ deberá enviar un mensaje de encaminamiento 'incremental' para informar a sus vecinos de la variación en la entrada del nodo destino $M_1$ para que puedan enterarse sin necesidad de esperar a que se envíe el siguiente paquete 'full dump' con la actualización completa de la tabla. También se incluyen las variaciones de los números de secuencia del resto de entradas, con lo que al final, como ha variado cada entrada de la tabla, es como si se hubiera enviado toda la tabla completa.

| *Destino* | *Próximo salto* | *Coste* | *Número de secuencia* | *Tiempo de instalación* |
|---|---|---|---|---|
| $M_1$ | $M_6$ | 3 | S516_$M_1$ | T810_M4 |
| $M_2$ | $M_2$ | 1 | S238_$M_2$ | T001_M4 |
| $M_3$ | $M_2$ | 2 | S674_$M_3$ | T001_M4 |
| $M_4$ | $M_4$ | 0 | S820_$M_4$ | T001_M4 |
| $M_5$ | $M_6$ | 2 | S502_$M_5$ | T002_M4 |
| $M_6$ | $M_6$ | 1 | S186_$M_6$ | T001_M4 |
| $M_7$ | $M_6$ | 2 | S238_$M_7$ | T002_M4 |
| $M_8$ | $M_6$ | 3 | S160_$M_8$ | T002_M4 |

**Tabla 3.3.** Tabla de encaminamiento para el nodo $M_4$ (actualizada).

| *Destino* | *Coste* | *Número de secuencia* |
|---|---|---|
| $M_4$ | 0 | S820_$M_4$ |
| $M_1$ | 3 | S516_$M_1$ |
| $M_2$ | 1 | S238_$M_2$ |
| $M_3$ | 2 | S674_$M_3$ |
| $M_5$ | 2 | S502_$M_5$ |
| $M_6$ | 1 | S186_$M_6$ |
| $M_7$ | 2 | S238_$M_7$ |
| $M_8$ | 3 | S160_$M_8$ |

**Tabla 3.4.** Tabla de encaminamiento del nodo $M_4$ de actualización enviada en el mensaje de encaminamiento incremental.

### *3.1.1.2 Optimized Link State Routing (OLSR)*

OLSR [100] es un protocolo de encaminamiento proactivo de estado enlace.

Es un protocolo de encaminamiento punto a punto que se basa en el algoritmo de estado enlace [101].

Cada nodo mantiene una ruta hacia el resto de nodos de la red ad hoc. Los nodos que forman parte de la red ad hoc intercambian periódicamente mensajes acerca de estado enlace, pero se utiliza la estrategia multipoint replaying [102] (MPR) para minimizar tanto el tamaño de los mensajes de encaminamiento como el número de nodos que reenvían en modo broadcast los mensajes de encaminamiento. La estrategia MPR [99] consiste en que cada nodo utiliza mensajes de 'Hello' para descubrir qué nodos se encuentran a un salto de distancia y confecciona una lista. Cada nodo seleccionará un subconjunto de nodos vecinos de dicha lista, que sean capaces de poder llegar a alcanzar todos los nodos que se encuentren a dos saltos de distancia respecto del nodo que está realizando la selección. Por ejemplo, en la *Fig. 3.3* el nodo A selecciona a los nodos B, C, K y N como nodos MPR, porque son capaces de alcanzar a todos los nodos que están a dos saltos de distancia con respecto al nodo A.

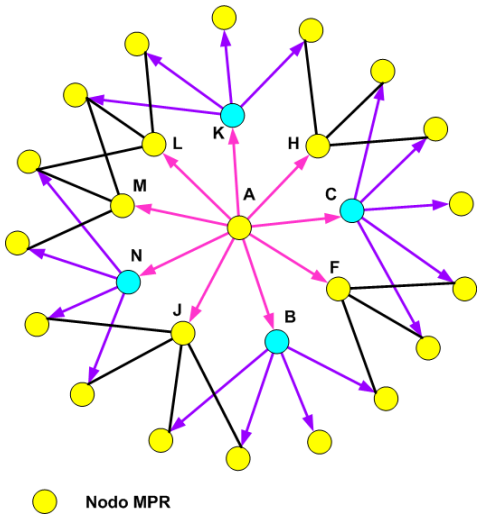



**Fig. 3.3.** Relays multipunto.

Estos nodos vecinos seleccionados serán los únicos encargados de retransmitir los paquetes de encaminamiento y se denominan 'relays multipunto' (retransmisores multipunto). El resto de nodos vecinos procesarán los paquetes de encaminamiento que reciban, pero no los podrán retransmitir.

Cada nodo determina una ruta óptima (en número de saltos) hacia cada destino utilizando la información almacenada (en la tabla de encaminamiento de la topología y en la de sus vecinos) [94] y guarda dicha información en una tabla de encaminamiento

para que esté disponible en el preciso momento en que un nodo desee empezar a enviar datos.

Este protocolo selecciona enlaces bidireccionales para el envío de paquetes [56], prescindiéndose de los enlaces unidireccionales.

## *3.1.2 Protocolos de encaminamiento reactivos*

Los protocolos de encaminamiento reactivos [25], [94] o bajo demanda son aquellos en los cuales los algoritmos de encaminamiento crearán rutas únicamente en el caso de que un nodo fuente necesite enviar información a un nodo destino. Así, se utilizan los recursos de red tales como la energía o el ancho de banda de forma más eficiente que en los protocolos de encaminamiento proactivos, aunque, por otro lado, aumenta el retardo de Descubrimiento de Ruta.

Durante el proceso de Descubrimiento de Ruta, si un nodo fuente desconoce una ruta hacia el destino envía un mensaje de petición de ruta (Route Request) en modo broadcast para obtenerla y recibirá un mensaje de respuesta de ruta (Route Reply), que contendrá la ruta buscada.

Si los enlaces son bidireccionales y por tanto el mensaje que contiene la ruta buscada (Route Reply) puede utilizar la misma ruta que el mensaje de petición de ruta o Route Request, entonces la señalización introducida en el proceso de Descubrimiento de Ruta crece en el peor caso con $O(N+M)$, donde $N$ representa el número de nodos en la red y $M$ el número de nodos en el camino de vuelta con la respuesta de ruta; para enlaces unidireccionales la señalización introducida crece con $O(2N)$.

Los protocolos de encaminamiento reactivos se pueden dividir en dos grupos:

- *Basados en la fuente (source-based)*

  Cada paquete de datos [103] transporta en su cabecera la ruta completa de la fuente al destino, es decir, las direcciones de cada nodo intermedio a lo largo de la ruta desde la fuente al destino. Cada nodo intermedio consultará la cabecera del paquete que le llega para saber por dónde debe reenviarlo. Por lo tanto, ya no hace falta que cada nodo intermedio mantenga una tabla de encaminamiento con información actualizada continuamente mediante el envío periódico de mensajes de encaminamiento, como sucedía con los protocolos proactivos. Como contrapartida, en las redes ad hoc grandes, la probabilidad de que un enlace se rompa crece con el número de nodos y, además, al aumentar con mayor probabilidad el número de nodos intermedios

a lo largo de la ruta, crece también la cabecera del paquete. En consecuencia, los protocolos de encaminamiento basados en la fuente no son recomendables en redes de gran tamaño con muchos saltos y alta movilidad debido a sus dificultades para escalar.

- *Salto a salto (hop-by-hop) o punto a punto (point-to-point)*

  El paquete lleva en su cabecera únicamente la dirección del destino y la dirección del próximo salto, de forma que cada nodo intermedio a lo largo de la ruta en dirección al destino deberá consultar su tabla de encaminamiento para decidir por donde debe reenviar el paquete.

  La ventaja de utilizar este tipo de encaminamiento es que cada nodo intermedio actualiza su tabla de encaminamiento continua e independientemente, de forma que cuando le llega un paquete decide encaminarlo según el estado actual de la red y así las rutas pueden adaptarse más fácilmente a la topología dinámica de este tipo de redes. La desventaja de utilizar este protocolo es la necesidad de que cada nodo intermedio a lo largo de la ruta mantenga su tabla de encaminamiento permanentemente actualizada mediante el intercambio periódico de mensajes de actualización con sus nodos vecinos.

Seguidamente se presentan los dos protocolos de encaminamiento bajo demanda más conocidos y aceptados por toda la comunidad científica debido a sus particulares méritos *(Véase la Fig. 3.1):*

- *Dynamic Source Routing (DSR) (Véase la sección 3.1.2.1 Dynamic Source Routing (DSR), pág. 102).*

  Está basado en la fuente.

- *Ad-hoc On-Demand Distance Vector (AODV) (Véase la sección 3.1.2.2 Ad hoc On-Demand Distance Vector (AODV), pág. 109).*

  Funciona salto a salto.

## *3.1.2.1 Dynamic Source Routing (DSR)*

DSR [104] es un protocolo de encaminamiento reactivo basado en la fuente.

Es un protocolo de encaminamiento bajo demanda, lo cual significa que se crearán rutas únicamente en el caso de que un nodo fuente necesite enviar datos a un nodo destino.

DSR presenta las siguientes características:

- *Señalización de control baja:*

No se realizan actualizaciones periódicas de información relacionada con el encaminamiento, dado su carácter reactivo.

- *Señalización de procesamiento:*

  Los mensajes de DSR serán procesados en un tiempo mayor o menor dependiendo del tamaño de las cabeceras.

- *Prevención de bucles*

  Existe un mecanismo para evitar la formación de bucles.

- *Funciona con enlaces unidireccionales o bidireccionales*

Cada nodo mantiene una caché con rutas válidas que han sido establecidas en el pasado.

El protocolo utiliza dos mecanismos fundamentales: Descubrimiento de Ruta (Route Discovery) y Mantenimiento de Ruta (Route Maintenance).

El funcionamiento del protocolo DSR se ilustra mediante unas imágenes que han sido cedidas amablemente por el profesor Nitin Vaidya.

Cuando un terminal móvil necesita enviar paquetes a un destino concreto, lo primero que deber hacer es consultar su caché para ver si encuentra alguna ruta particular hacia ese destino y, si existe, hace uso de ella enviando paquetes.

Si no encuentra dicha ruta, el nodo fuente inicia un proceso de Descubrimiento de Ruta *(Véanse las Fig. 3.4 y Fig. 3.5),* enviando un paquete broadcast denominado petición de ruta, RREQ (Route Request), con la dirección del nodo fuente, la dirección del nodo destino y un identificador de RREQ. Además, cada RREQ contiene las direcciones de cada nodo intermedio que ha reenviado este paquete almacenadas en el denominado registro de ruta. Cada nodo dentro del alcance de transmisión radio recibe este paquete y comprueba si es el destino o bien si su caché contiene una ruta activa hacia el destino. En estos casos, el nodo retorna un mensaje de petición de respuesta, RREP (Route Reply), que contiene una copia del registro de ruta acumulado en el RREQ. En el caso de que dicho nodo sea el destino, el registro de ruta representa todas las direcciones de los nodos intermedios que el RREQ ha atravesado en su camino desde la fuente hasta ese nodo concreto. Si este nodo no es el destino pero se trata de un nodo particular que conoce una ruta hacia el destino, añade la ruta contenida en su caché al registro de ruta y así genera el RREP de acuerdo con la información que posee.

Si este nodo ni es el destino ni conoce una ruta hacia el destino, entonces consulta si ya ha recibido un paquete con la misma fuente, destino e identificador de RREQ, o bien si su propia dirección ha aparecido en el registro de ruta. En este caso, con el fin de limitar el número de peticiones de ruta que se propagan a través de la red, el nodo

elimina el RREQ *(Véase la Fig. 3.6).* Por lo tanto, los RREQ que se propagan durante el Descubrimiento de Ruta son los que llegan primero a un determinado nodo. Si se elige como ruta óptima la ruta almacenada en el primer RREP que consigue llegar al nodo fuente, se puede afirmar entonces que la ruta seleccionada es la más rápida que hay en ese momento para llegar al destino.

Si no se cumple ninguna de las condiciones anteriores, el nodo retransmite el paquete, añadiendo su propia dirección al registro de ruta *(Véase la Fig. 3.7)* [105].

Cuando un nodo debe devolver el mensaje RREP al iniciador del Descubrimiento de Ruta *(Véase la Fig. 3.8),* lo que hará será consultar su caché en busca de una ruta hacia el nodo fuente y la utilizará en caso de que exista. Por otro lado, si los enlaces son simétricos (bidireccionales), el nodo puede invertir la ruta almacenada en el registro de ruta del RREQ *(Véase la Fig. 3.9).* En el caso de protocolos que requieren un intercambio bidireccional de tramas como el protocolo MAC del IEEE 802.11, debe utilizarse forzosamente la ruta inversa para probar su funcionamiento. Si los enlaces no son bidireccionales, el nodo puede realizar su propio procedimiento de Descubrimiento de Ruta hacia el nodo origen y debe hacer 'piggyback' de su RREP (incluirlo al final) en un nuevo RREQ.

Cuando el nodo fuente recibe el RREP almacena la ruta contenida en su caché con el objetivo de usarla en el posterior envío de paquetes hacia ese destino *(Véase la Fig. 3.10)*. Un nodo fuente puede aprender con la ayuda de un único procedimiento de Descubrimiento de Ruta muchas rutas hacia un destino concreto. Así, cuando deja de estar disponible una ruta, no es necesario iniciar un nuevo procedimiento de Descubrimiento de Ruta para encontrar una disponible; de esta forma aumenta la rapidez de actuación del protocolo de encaminamiento ante los cambios de ruta y disminuye la señalización necesaria para localizar una nueva ruta que sea adecuada.

Los nodos intermedios que reciben tanto los RREQ como los RREP pueden crear o actualizar entradas en sus tablas de encaminamiento hacia cada uno de los nodos a lo largo de la ruta almacenada en estos paquetes.

Asimismo, los nodos cuentan con una opción denominada 'promiscuous listening', que consiste en que pueden recibir y procesar tanto paquetes de control como de datos no destinados a ellos mismos a nivel de la capa MAC. Así pueden aprovechar la información acerca de las rutas contenidas en estos mensajes para aprender gratuitamente información de encaminamiento hacia otros destinos de la red.

DSR permite elegir la métrica que se estime más oportuna para poder enviar paquetes de un nodo origen a un nodo destino. Puede por ejemplo considerarse que el nodo fuente siempre selecciona la primera ruta que le llega hacia ese destino

concreto, pero también podría seleccionarse la ruta más corta en términos de número de saltos o bien en relación a otros criterios.

Cada ruta almacenada en la caché de rutas está asociada a un temporizador [106] para que pueda ser eliminada si no se utiliza durante un cierto periodo de tiempo.

Cuando un paquete sigue una ruta hacia un destino, cada nodo a lo largo de dicha ruta debe asegurarse de que el paquete ha sido recibido por el nodo siguiente del registro de ruta. Este objetivo se consigue gracias al empleo de paquetes de confirmación, como los enviados a nivel de la capa MAC en el estándar IEEE 802.11. Si un nodo intermedio no recibe dichos paquetes *(Véase la Fig. 3.11),* llega a la conclusión de que ha habido algún problema en el enlace y elimina las rutas que contienen dicho enlace de su caché de rutas. Entonces envía un paquete de error de ruta, RERR (Route Error) [25] a cada emisor que ha estado enviando paquetes a dicho nodo vecino desde la última vez en que se recibieron los reconocimientos. Este mecanismo se conoce como Mantenimiento de Ruta y se activa únicamente en el caso de que un nodo fuente esté enviando paquetes a un nodo destino.

Después, el nodo intermedio consulta su caché y, si existe, utiliza una ruta allí contenida para enviar aquellos paquetes que se han encontrado con que la ruta que querían usar no está disponible. Este procedimiento se denomina 'route salvaging' (salvando la ruta) [99]. En caso contrario, el nodo deberá descartar aquellos paquetes almacenados en su cola que seguían una ruta a través del enlace roto.

Cuando se recibe un RERR, el emisor original utiliza otras rutas contenidas en su caché para reenviar los datos o bien inicia un nuevo procedimiento de Descubrimiento de Ruta. La *Fig. 3.11* ilustra el funcionamiento del proceso de Mantenimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento DSR para poder enviar datos desde un nodo origen S hasta un nodo destino D.

Entre todas las versiones existentes de DSR, en esta tesis doctoral se ha seleccionado para analizar y realizar simulaciones en las secciones *3.5.1.1*, *3.5.1.2* y *3.5.2.1.1* aquella en la que se selecciona la ruta más rápida hacia un destino cuando se utiliza el proceso de Descubrimiento de Ruta (se selecciona como ruta óptima la primera ruta que llega almacenada en un RREP al nodo fuente). El motivo de esta elección ha sido que esta versión es la más sencilla de todas. Además, así se evita el haber de recurrir a una función de coste para calcular la ruta óptima e introducir un tiempo adicional de espera durante el procedimiento de Descubrimiento de Ruta para que puedan llegar unas cuantas rutas a la fuente y haber de determinar después entre todas cuál es la óptima. Si se sigue este criterio, con DSR la ruta con el mínimo número de saltos es seleccionada únicamente en el caso ideal de que los paquetes de RREQ sufran el mismo retardo de transmisión en cada enlace y que no se produzcan

retardos por otros motivos (por ejemplo, procesamiento o contienda MAC). En situaciones reales, habrá una alta probabilidad de que DSR seleccione rutas con el mínimo número de saltos, pero no existe ninguna garantía.

Existe una opción del protocolo DSR denominada 'estado por flujo' [107], que sirve para encaminar sin necesidad de que las direcciones IP de los nodos intermedios se hallen especificadas en la cabecera IP de los paquetes que se desea enviar, disminuyendo de esta forma la señalización del protocolo. Después de que un nodo fuente haya descubierto una ruta mediante un procedimiento de Descubrimiento de Ruta tradicional, se establece un estado de envío de paquetes salto a salto en la red. Cada nodo intermedio a lo largo de la ruta será capaz de reenviar el paquete al siguiente salto basándose en su propio conocimiento a nivel local del flujo al cual pertenece el paquete que debe ser encaminado. El primer paquete, que usa encaminamiento basado en la fuente, inicializa el estado por flujo y entonces, mediante este procedimiento es posible encaminar el resto de paquetes sucesivos sin que la ruta completa viaje en la cabecera de cada paquete. El estado que se establece en cada salto es un estado por flujo y expira automáticamente cuando la ruta creada deja de utilizarse.

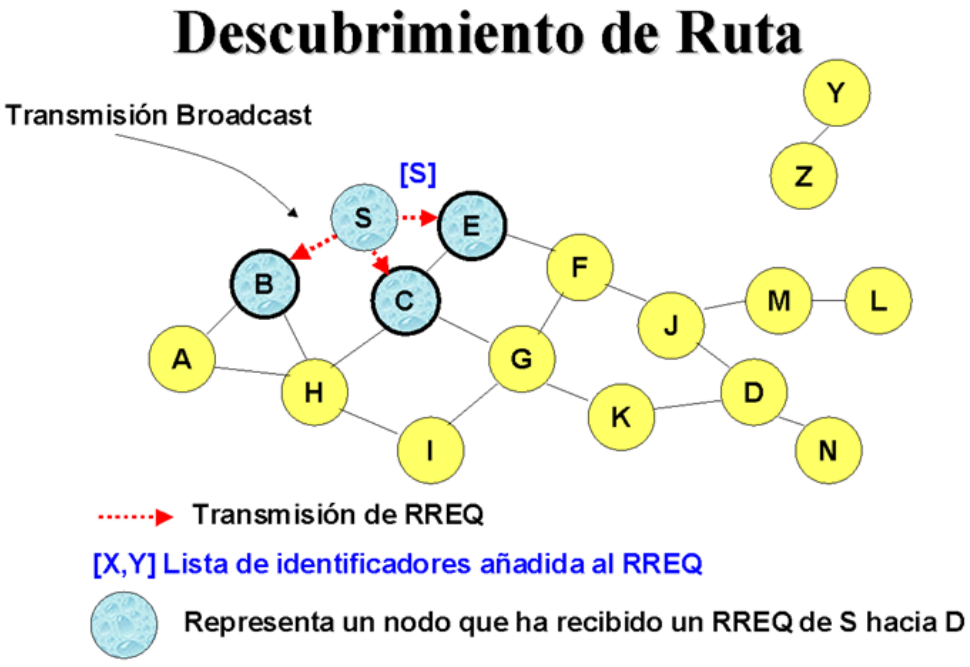



**Fig. 3.4.** Descubrimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento DSR para enviar datos desde un nodo origen S hasta un nodo destino D (1).

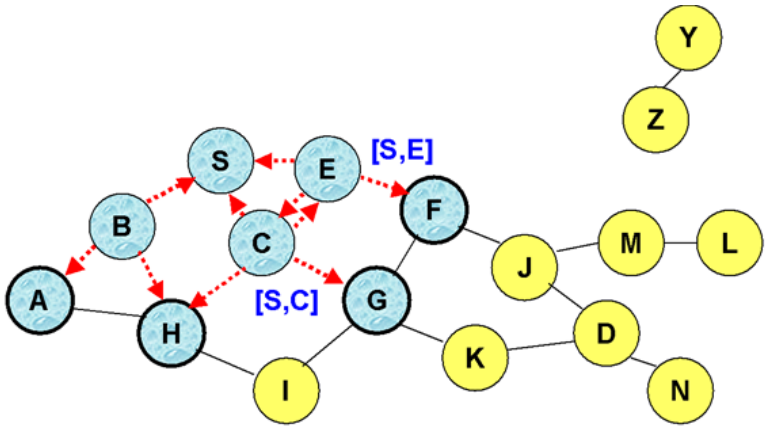
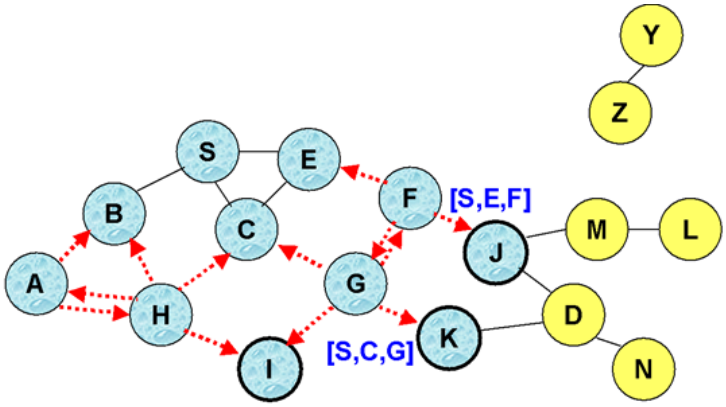
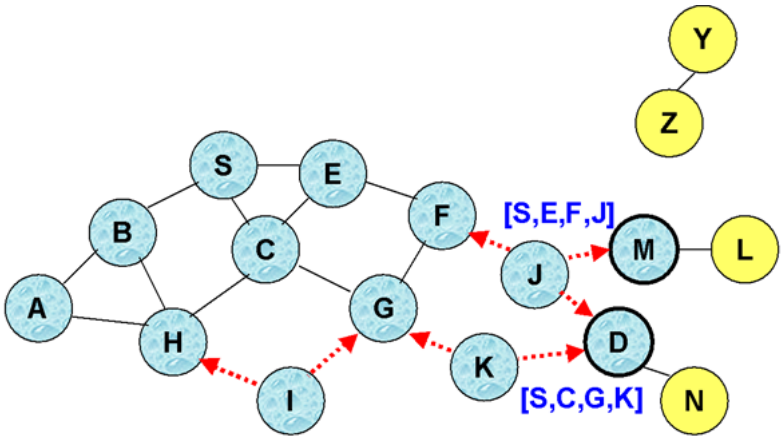



**Fig. 3.5.** Descubrimiento de Ruta (2).



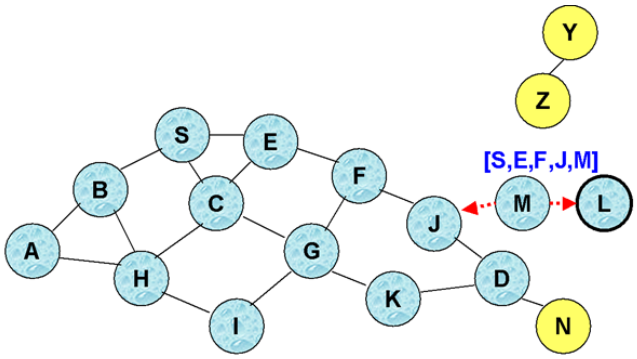

• El nodo C recibe los RREQ de G y H, pero no hace una retransmisión porque ya ha sido hecha una vez.

**Fig. 3.6.** Descubrimiento de Ruta (3).




•Los nodos J y K retransmiten el RREQ al nodo D

**Fig. 3.7.** Descubrimiento de Ruta (4).




• El nodo D no reenvía el RREQ, porque es el nodo destino en el proceso de Descubrimiento de Ruta

**Fig. 3.8.** Descubrimiento de Ruta (5).

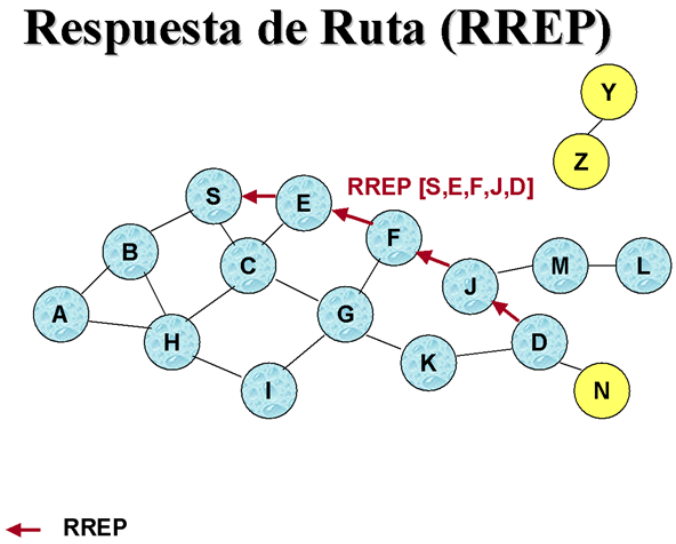



**Fig. 3.9.** Descubrimiento de Ruta (6).

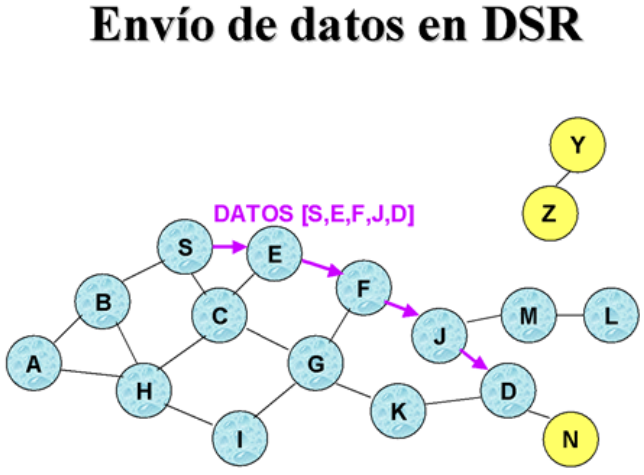



**Fig. 3.10.** Descubrimiento de Ruta (7).

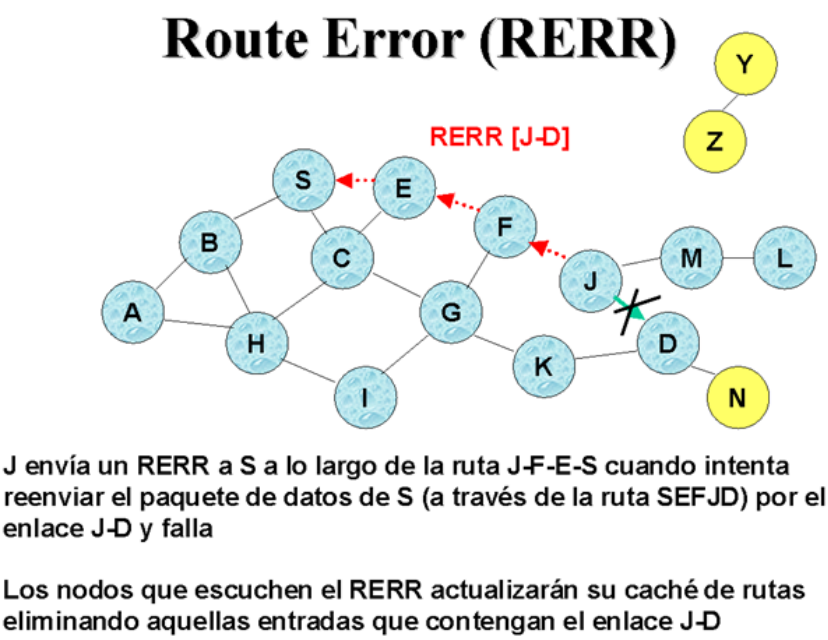



**Fig. 3.11.** Mantenimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento DSR para enviar datos desde un nodo origen S hasta un nodo destino D.

## *3.1.2.2 Ad hoc On-Demand Distance Vector (AODV)*

AODV [108] es un protocolo de encaminamiento reactivo salto a salto.

Es un protocolo de encaminamiento que establece rutas bajo demanda [56]. Esto significa que en AODV no se mantienen permanentemente actualizadas las rutas de cada nodo a cada nodo de la red, sino que se descubren y mantienen solamente cuando son necesarias.

Las rutas son descubiertas durante el proceso de Descubrimiento de Ruta [25], una fase en la cual un nodo fuente busca una ruta hacia un nodo destino para poder enviarle información. Este proceso termina cuando al nodo fuente se le retorna la ruta buscada.

AODV presenta las siguientes características:

- *Señalización de control baja:*

  No se realizan actualizaciones periódicas de información relacionada con el encaminamiento, dado su carácter reactivo.

- *Señalización de procesamiento mínima:*

  Los mensajes de AODV son sencillos y requieren poco cálculo.

- *Prevención de bucles*

  Existe un mecanismo para evitar la formación de bucles.

- *Funciona sólo con enlaces bidireccionales*

Con el fin de prevenir bucles cada nodo mantiene un número de secuencia (también llamado número de secuencia del destino) que sirve para evaluar la vigencia de la información de encaminamiento asociada y aumenta en uno cada vez que un nodo envía una nueva petición de ruta, RREQ (Route Request). Si un nodo recibe un RREQ destinado a él mismo antes de generar el mensaje de petición de respuesta, RREP (Route Reply) debe actualizar su número de secuencia $NumSeq_D$ al valor máximo entre su número de secuencia actual $NumSeq_{D\_actual}$ y el número de secuencia del destino que se halla contenido en el RREQ ( $RREQ.NumSeq$ ) más uno:

$$NumSeq_D = Max\left(NumSeq_{D\_actual}, RREQ.NumSeq + 1\right) \tag{3.1}$$

Los números de secuencia sirven para que siempre se seleccione la ruta más reciente hacia un destino. Si se da el caso de que un nodo fuente o nodo intermedio recibe dos rutas que contienen el mismo número de secuencia del destino, escogerá una ruta de las dos utilizando una métrica como puede ser aquella que contenga el menor número de saltos hacia el destino.

AODV utiliza para su correcto funcionamiento unas tablas de encaminamiento donde se almacena:

- *La dirección IP del destino*
- *La dirección IP del próximo salto*
- *El número de secuencia del destino*
- *Tiempo de vida (tiempo requerido para eliminar la ruta)*
- *Contador de saltos (número de saltos para alcanzar el destino)*

Cada entrada de la tabla de encaminamiento está asociada a un temporizador de tiempo de vida. Si una ruta no se usa, el temporizador expira. En cambio, si sí que es utilizada o se reciben mensajes de 'Hello', el temporizador asociado al tiempo de vida de la ruta se actualiza.

El funcionamiento del protocolo AODV se ilustra mediante unas imágenes que han sido cedidas amablemente por el profesor Nitin Vaidya.

El procedimiento de Descubrimiento de Ruta (Route Discovery) puede ser descrito de la forma siguiente *(Véase la Fig. 3.12):*

- *Cuando un nodo fuente desee enviar paquetes a otro nodo destino, lo primero que debe hacer es consultar su tabla de encaminamiento para comprobar si ya existe una ruta actualizada hacia dicho destino. En caso de que sí que exista, el nodo fuente la usará para enviar los paquetes al próximo salto en dirección hacia el destino. En caso de que no exista, el nodo iniciará un procedimiento*

*de Descubrimiento de Ruta enviando un paquete denominado petición de ruta, RREQ (Route Request) en modo broadcast.*

- *Cualquier nodo de la red que conozca una ruta actualizada hacia el destino (incluido el propio destino) puede enviar un paquete RREP al nodo origen.*
- *La información acerca de la ruta se mantiene en la tabla de encaminamiento de cada nodo.*
- *La información obtenida a través del envío de mensajes de RREQ y RREP es guardada junto con otra información en la tabla de encaminamiento.*
- *Los números de secuencia se usan para eliminar rutas viejas.*
- *Las rutas con números de secuencia antiguos son anuladas.*

Si un nodo inicia un procedimiento de Descubrimiento de Ruta, enviará un paquete de RREQ en modo broadcast con la información siguiente:

- *Dirección IP nodo origen*
- *Número de secuencia origen*
- *Dirección IP nodo destino*
- *Número de secuencia destino*
- *ID (identificador broadcast) del RREQ*
- *Contador del número de saltos*

El identificador broadcast del RREQ es un número que se incrementa cada vez que un nodo inicia el envío de un paquete de RREQ.

Cuando un nodo recibe un RREQ debe:

- *Mirar el campo ID broadcast del RREQ y la dirección IP del nodo origen para saber si ya lo recibió. Cada nodo mantiene un registro de la dirección IP del nodo origen y del ID broadcast del RREQ durante un tiempo para cada paquete de RREQ que recibe; este tiempo dependerá de la congestión, el tamaño y la topología de la red.*
- *Si el nodo comprueba que ya ha recibido el RREQ, rechaza el paquete (Véase la Fig. 3.14).*
- *Si no es así, registra esta información y procesa el paquete de RREQ.*
- *El procesamiento del paquete de RREQ se efectúa de la forma siguiente:*

  El nodo establece una entrada en la tabla de encaminamiento que contiene la ruta inversa *(Véase la Fig. 3.13).* En concreto, contiene, entre otros campos:

- *Dirección IP nodo origen*
- *Número de secuencia del nodo origen*
- *Número de saltos al nodo origen (se incrementa en uno el valor contenido para este campo en el RREQ)*
- *Dirección IP del nodo vecino que le envió el paquete RREQ*

La ruta inversa tiene un tiempo de vida determinado y cuando expira la información asociada es eliminada. La ruta inversa tiene su utilidad cuando el nodo recibe más tarde un RREP que debe ser entregado a la fuente a través de la ruta inversa creada.

Para poder responder al paquete de RREQ:

- *El nodo debe tener una tabla de encaminamiento con una entrada de ruta al destino que no haya expirado.*
- *Por otra parte, el número de secuencia del destino de la entrada de ruta en la tabla de encaminamiento ha de ser mayor o igual que el número de secuencia del destino del paquete de RREQ, es decir:*

$$NumSeq_{tabla} \geq NumSeq_{RREQ} \quad (3.2)$$

Si se cumplen estas dos condiciones, el nodo incrementa el contador de número de saltos del RREQ y genera un RREP.

Sino, el nodo incrementa el contador del número de saltos del RREQ y reenvía dicho paquete en modo broadcast a sus vecinos porque no posee una ruta actual hacia ese destino.

El paquete de RREP contiene la dirección IP origen y la dirección IP destino.

Además, si es el destino quien responde *(Véanse las Fig. 3.15, Fig. 3.16 y Fig. 3.17):*

- *Coloca su número de secuencia en el paquete (primero calcula este número de secuencia de acuerdo con la fórmula explicada en (3.1)).*
- *Inicializa el contador del número de saltos a cero.*
- *Coloca en el temporizador de tiempo de vida un valor de tiempo.*
- *Envía el paquete en dirección al nodo origen eligiendo como primer salto el mismo nodo a través del cual recibió el paquete de RREQ, aprovechando que este nodo tiene establecida la ruta inversa.*

Si responde un nodo intermedio:

- *Coloca el número de secuencia del destino en el paquete.*
- *Mete un determinado número de saltos en el contador, que coincide con el número de saltos de este nodo al destino.*

- *Coloca en el temporizador de tiempo de vida un valor de tiempo.*
- *Envía el paquete al nodo origen eligiendo como primer salto el mismo nodo a través del cual recibió el paquete de RREQ, aprovechando que este nodo tiene establecida la ruta inversa.*

Cuando un nodo intermedio recibe un RREP:

- *Incrementa en uno el contador del número de saltos del RREP.*
- *Establece un 'camino hacia delante' (forward path) (Véase la Fig. 3.18), que representa una entrada hacia el destino en su tabla de encaminamiento. El nodo intermedio usa el nodo del cual recibió el RREP como próximo salto hacia el destino. Así todos los nodos intermedios a lo largo de la ruta entre la fuente y el destino conocerán este camino de transmisión de datos si es seleccionado por la fuente.*
- *Esta entrada contiene:*
  - *Dirección IP nodo destino*
  - *Dirección IP próximo salto*
  - *Número de saltos al destino (suma uno al contador)*
  - *Temporizador del tiempo de vida*
  - *Número de secuencia del destino*
- *El nodo reenvía el paquete RREP al siguiente salto en dirección a la fuente.*

Si un nodo recibe un RREP para un destino por parte de más de un nodo vecino:

- *Reenvía el primer RREP.*
- *Reenvía otro RREP más tardío sólo si:*
  - *El número de secuencia del destino contenido en este último paquete es mayor.*

  *o bien*
  - *El número de saltos del contador contenido en este último paquete es menor.*
- *Sino, rechaza este último paquete de RREP.*

Cuando un nodo fuente recibe un RREP puede comenzar a usar la ruta contenida para encaminar paquetes de datos *(Véase la Fig. 3.19).* Si se da el caso de que un nodo fuente recibe múltiples RREPs seleccionará aquella ruta con el número de secuencia del destino mayor y el número de saltos hacia ese destino menor.

Resulta interesante observar que AODV no necesariamente proporciona la ruta más corta en términos de número de saltos de un nodo origen a un nodo destino. En AODV cada nodo acepta y procesa únicamente un RREQ, mientras que rechaza aquellos

RREQs que le llegan posteriormente y tienen asociados el mismo campo ID broadcast del RREQ y dirección IP del nodo origen que el que había llegado anteriormente. Por esta razón no es posible que AODV nos devuelva siempre la trayectoria más corta posible de entre todas las rutas, ya que algunas rutas que quizás serían más óptimas no llegan jamás a poder descubrirse debido a que cuando se propagan los RREQ para localizarlas, anteriormente a algún nodo intermedio concreto ya ha llegado otro RREQ procedente de alguna ruta diferente que motiva que el siguiente RREQ perteneciente a esta ruta tenga que ser rechazado. Ahora bien, entre las rutas calculadas, sí que termina escogiéndose aquella que presenta un número de saltos menor.

Cada nodo monitoriza el estado de los enlaces que le comunica con el próximo salto a través de una ruta que está siendo utilizada en ese momento (ruta activa). En el caso de que detecte la rotura del enlace, este nodo invalida en su tabla de encaminamiento todas aquellas entradas hacia destinos que no están en estos momentos disponibles debido a la rotura del enlace.

Además, el nodo enviará un paquete de error de ruta, RERR (Route Error) hacia el nodo fuente para informarle acerca de este suceso. Este proceso es conocido con el nombre de Mantenimiento de Ruta [99]. En el RERR se informa acerca de aquellos destinos que ya no pueden alcanzarse. Si existen nodos precursores (entre el nodo fuente y el nodo que ha detectado la rotura del enlace) que estaban utilizando el enlace, el paquete de RERR se propaga en modo broadcast y sino en modo unicast.

Cuando un nodo recibe un paquete de RERR comprueba que efectivamente el nodo que envió dicho mensaje sea su próximo salto hacia alguno de los destinos y si esa así el nodo invalida estas entradas de ruta en su tabla de encaminamiento y reenvía el paquete de RERR hacia la fuente. Cuando finalmente el paquete de RERR llega a la fuente, ésta puede decidir iniciar un nuevo proceso de Descubrimiento de Ruta si lo considera necesario.

Con el fin de comprobar si existe conectividad [109], cada nodo envía periódicamente a sus vecinos mensajes ‘Hello’ con la dirección IP del nodo, su número de secuencia actual y el tiempo de vida del enlace. Así cada nodo vecino puede aprovechar esta información para actualizar la entrada de la tabla de encaminamiento hacia ese vecino. Si durante un cierto intervalo de tiempo un nodo deja de recibir mensajes de ‘Hello’ de parte de un vecino concreto, entonces elimina la entrada de la tabla de encaminamiento asociada a dicho vecino. El intercambio de mensajes de encaminamiento no es necesario si existe otro mecanismo para averiguar si hay o no conectividad, como puede ser retroalimentación procedente de la capa de enlace de datos.

AODV [110] [111] presenta una serie de opciones de optimización, como la posibilidad de reparar a nivel local un enlace roto que forma parte de una ruta activa. Cuando se rompe un enlace, en lugar de enviar un paquete de RERR a la fuente, el nodo que ha detectado la rotura puede intentar repararlo localmente enviando un RREQ con el número de secuencia del destino incrementado en uno hacia ese destino. Los paquetes de datos se quedan almacenados en este nodo esperando recibir un RREP con una nueva ruta disponible hacia el destino. Si este nuevo procedimiento de Descubrimiento de Ruta no tiene éxito y el RREP no llega, entonces sí que será necesario informar a la fuente acerca de la rotura del enlace enviándole un paquete RERR.

Otra característica adicional consiste en el envío de RREP gratuitos desde un nodo intermedio hacia un nodo destino para informarle que ha sido el propio nodo intermedio quien ha respondido al RREQ generado por la fuente y facilitar de esta forma que exista una ruta del destino a la fuente por si debe realizarse un intercambio bidireccional de mensajes. Si no se hiciera de este modo, el destino no tendría conocimiento de que existe una ruta hacia la fuente (al no haber recibido el RREQ) y no podría contestar a la fuente si recibiera paquetes de datos y fuera preciso hacerlo.

Otra característica adicional distinta consiste en que si se cree que un enlace puede ser únicamente unidireccional, se enviará un RREP-ACK (RREP-Acknowledgement) para comprobar si el siguiente salto nos contesta enviando una confirmación conforme el RREP le ha llegado. Si la confirmación no llega, el nodo inserta el siguiente salto en su blacklist (lista negra), para indicar que los RREQs enviados por el siguiente salto deben ser ignorados debido a la unidireccionalidad del enlace.

En los últimos borradores (drafts) de Internet [112] [113] del protocolo AODV, dicho protocolo de encaminamiento ha sido modificado para permitir (a imitación de DSR) que sea posible almacenar la ruta en la cabecera de los paquetes durante el proceso de Descubrimiento de Ruta. Cuando los mensajes de RREQ y RREP son generados o reenviados por los nodos [114], éstos añaden previamente su dirección IP a la cabecera de los paquetes. Cada nodo actualiza también su tabla de encaminamiento con la información contenida en estos paquetes de señalización. Así, cada vez que un nodo recibe un mensaje de RREQ o RREP, aprende rutas no sólo hacia sus nodos vecinos, hacia la fuente y hacia el destino, sino también hacia el resto de nodos de la red, que figuran como nodos intermedios de la ruta que almacena el paquete. Si ya existe una entrada de ruta en la tabla de encaminamiento hacia un nodo intermedio, pero el contador de saltos hacia ese nodo intermedio de acuerdo con la información del paquete de señalización es menor que la información del contador de saltos

contenida en la tabla de encaminamiento, la entrada de ruta de la tabla de encaminamiento es actualizada para ese nodo.

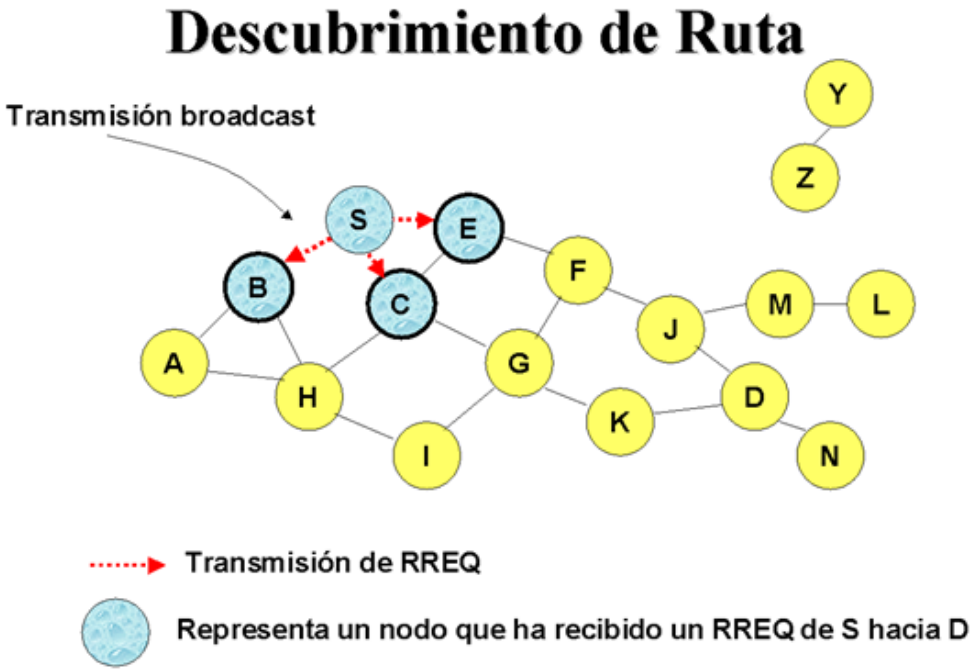



**Fig. 3.12.** Descubrimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento AODV para enviar datos desde un nodo origen S hasta un nodo destino D (1).

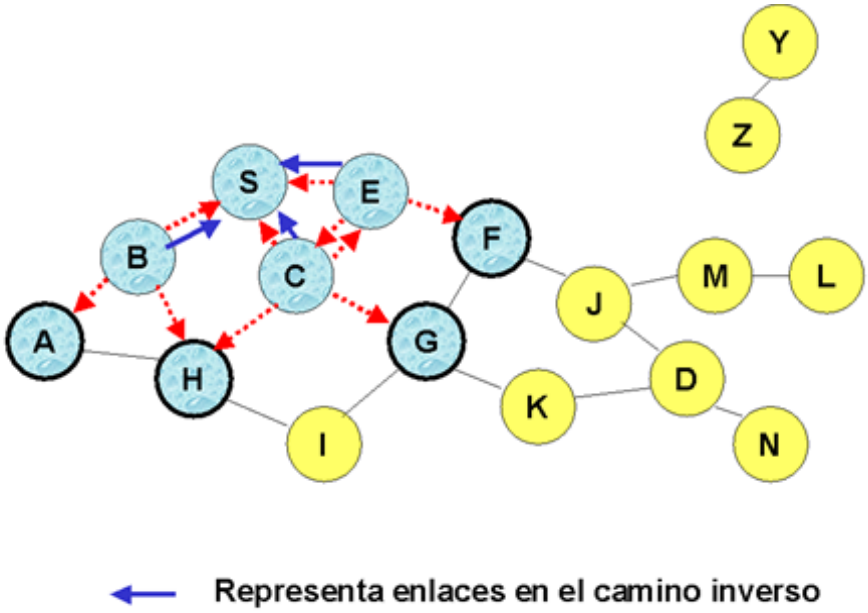
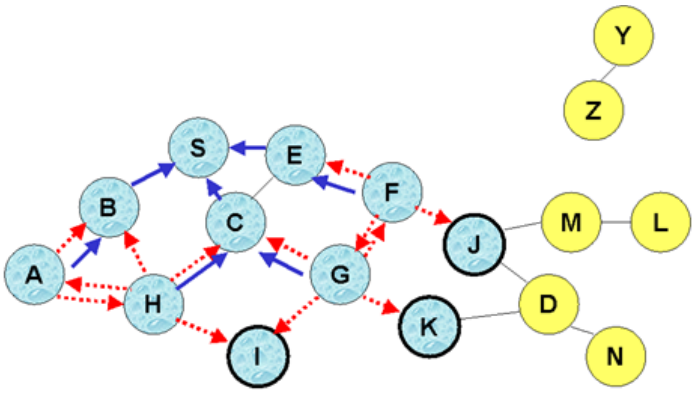



**Fig. 3.13.** Descubrimiento de Ruta (2).




• El nodo C recibe los RREQ de G y H, pero no hace una retransmisión porque ya ha sido hecha una vez.

**Fig. 3.14.** Descubrimiento de Ruta (3).

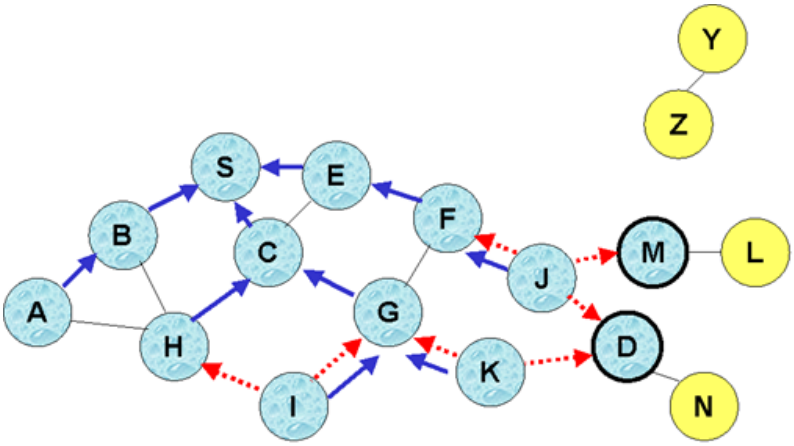
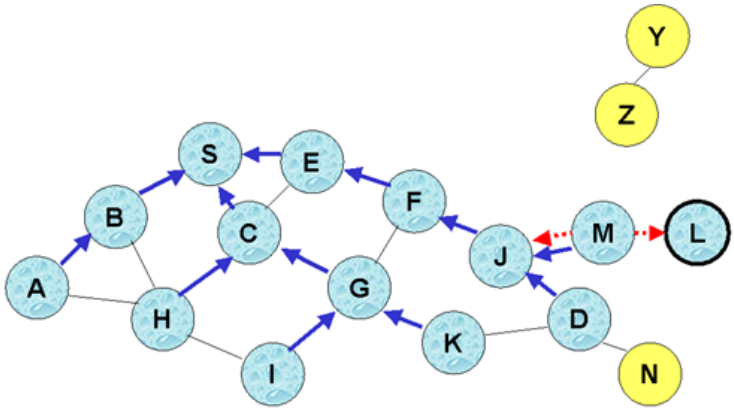



**Fig. 3.15.** Descubrimiento de Ruta (4).




• El nodo D no reenvía el RREQ, porque es el nodo destino en el proceso de Descubrimiento de Ruta

**Fig. 3.16.** Descubrimiento de Ruta (5).

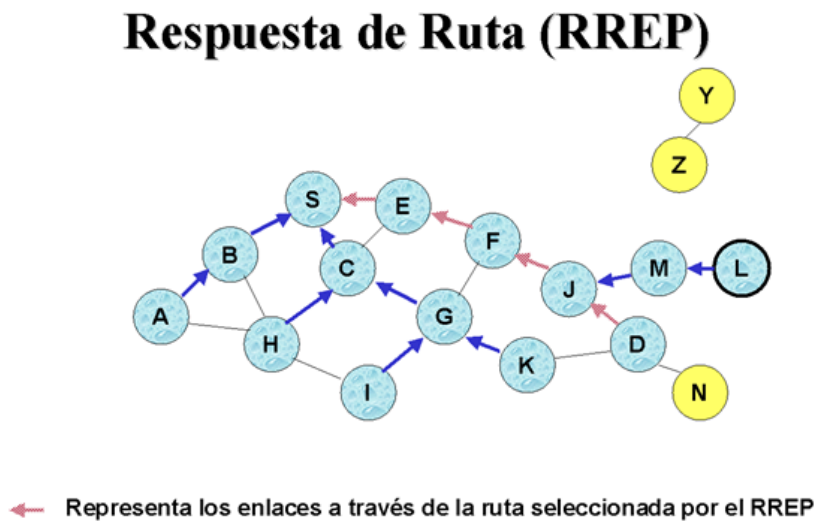



**Fig. 3.17.** Descubrimiento de Ruta (6).

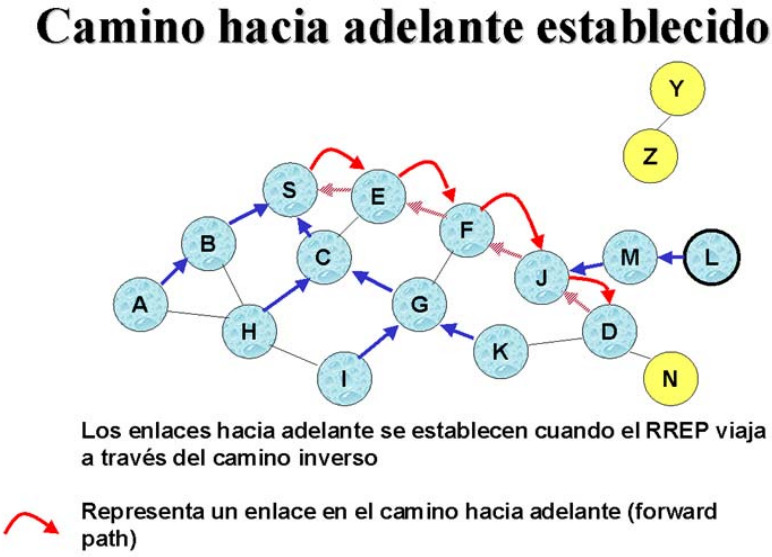



**Fig. 3.18.** Descubrimiento de Ruta (7).

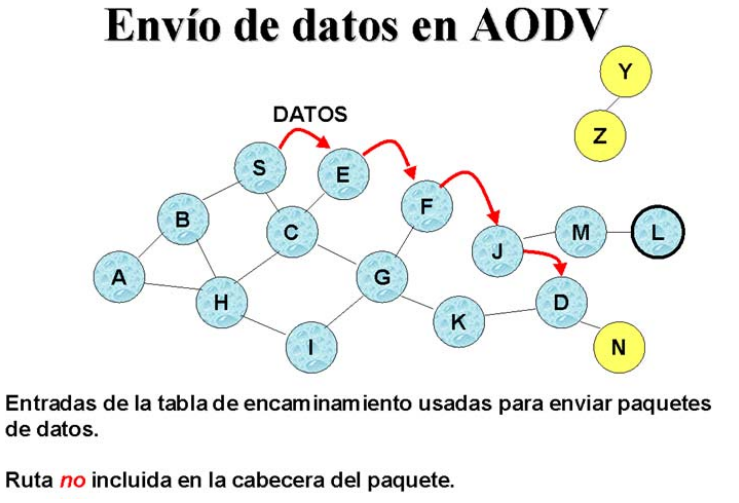



**Fig. 3.19.** Descubrimiento de Ruta (8).

## *3.1.3 Protocolos de encaminamiento híbridos*

Son aquellos protocolos que aprovechan a la vez las ventajas de los protocolos de encaminamiento proactivos y reactivos.

Estos protocolos suelen dividir la red en un determinado número de zonas, o bien árboles o clusters para crear diferentes grupos de nodos. En muchas ocasiones las rutas entre nodos próximos son halladas utilizando encaminamiento proactivo, mientras que las rutas entre nodos lejanos son descubiertas utilizando un protocolo de encaminamiento bajo demanda.

Como ejemplo de protocolo de encaminamiento híbrido describiremos el Zone Routing Protocol (ZRP).

### *3.1.3.1 Zone Routing Protocol (ZRP)*

Zone Routing Protocol (ZRP) [115] es un protocolo de encaminamiento híbrido.

Cada nodo especifica una zona de encaminamiento, la cual está formada por varios nodos móviles de la red ad hoc situados a una cierta distancia respecto de sí mismo dentro de un radio de zona (especificado en saltos radio) [99]. Las zonas pueden solaparse. Dentro de cada zona se usa un protocolo de encaminamiento proactivo para que cada nodo en el interior sepa cómo llegar a sus vecinos. En cambio, si se desea enviar paquetes a un nodo destino fuera de la zona, se usa un protocolo de encaminamiento bajo demanda [56].

ZRP [116] consta por tanto de tres protocolos de encaminamiento [25]:

- *El protocolo de encaminamiento proactivo Intrazone Routing Protocol (IARP)*
- *El protocolo de encaminamiento reactivo Interzone Routing Protocol (IERP)*
- *El protocolo de encaminamiento Bordercast Resolution Protocol (BRP)*

IARP es un protocolo de encaminamiento de estado enlace y se ocupa de que cada nodo tenga una tabla de encaminamiento con información actualizada para llegar a cada nodo dentro de la zona.

IERP usa los nodos frontera para poder encontrar una ruta hacia un nodo destino situado fuera de la zona. IERP usa el protocolo de encaminamiento BRP.

Cuando se pone en marcha un procedimiento de Descubrimiento de Ruta, el nodo fuente consulta su tabla de encaminamiento y si es necesario pone en marcha una búsqueda de ruta entre varias zonas para alcanzar un destino concreto. Si se rompe una ruta debido a la movilidad de un nodo dentro de la misma zona donde estaba ubicado, deben actualizarse las tablas de encaminamiento utilizadas por el protocolo

de encaminamiento proactivo. Si la movilidad de un nodo tiene lugar de una zona a otra, entonces será preciso ejecutar una consulta entre zonas.

La idea de utilizar un protocolo de encaminamiento bajo demanda para encontrar una ruta desde un nodo fuente hasta un nodo destino situado en otra zona permite reducir la señalización (en comparación con los protocolos proactivos) y los retardos en el Descubrimiento de Ruta (en comparación con los protocolos bajo demanda puros), pues dichas rutas son descubiertas con mucha mayor rapidez. La razón es que para encontrar una ruta hacia un nodo situado fuera de la zona de encaminamiento, la petición de ruta viajará únicamente hacia el router frontera dentro de la zona donde se halla ubicado el destino. Este router frontera podrá responder a la petición puesto que mantiene una tabla para hacer encaminamiento proactivo y sabrá cómo llegar al destino.

La desventaja de usar ZRP [94] es que se comportará como un protocolo puramente proactivo si las zonas de encaminamiento definidas son grandes y por el contrario reaccionará como un protocolo puramente reactivo si las zonas de encaminamiento definidas son pequeñas.

En la *Fig. 3.20* se ilustra un procedimiento de Descubrimiento de Ruta; el nodo S decide enviar información al nodo X y mediante el protocolo IARP llega a la conclusión de que X no pertenece a la misma zona que S. La búsqueda se propaga a través de los nodos frontera para tratar de localizar aquella zona donde se halla ubicado el nodo X. Al final, el nodo frontera G descubre que X está localizado dentro de su zona y envía una respuesta de ruta de vuelta hacia el nodo S.

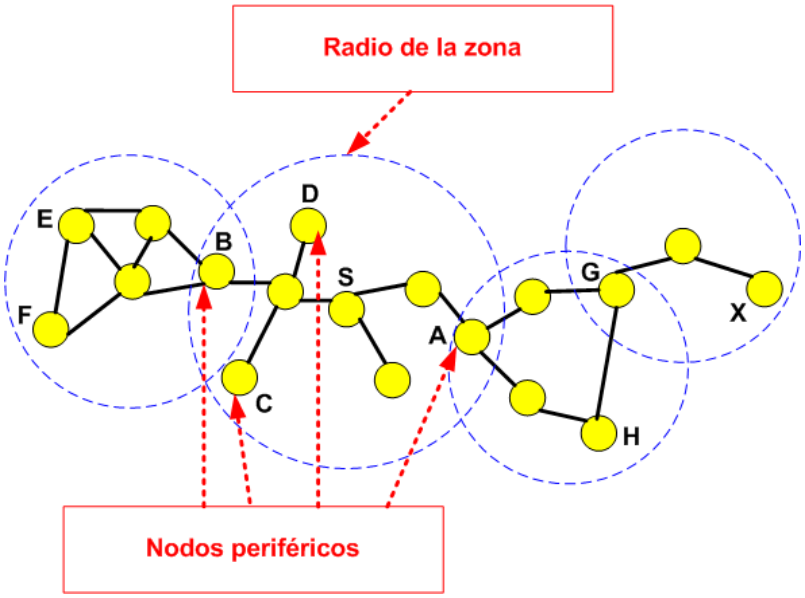



**Fig. 3.20.** Ejemplo de Descubrimiento de Ruta en una red ad hoc que utiliza el protocolo de encaminamiento ZRP.

## *3.2 Protocolos de encaminamiento best-effort y con calidad de servicio*

Una segunda posible clasificación de los protocolos de encaminamiento nos permite dividirlos en dos grandes grupos:

- *Protocolos de encaminamiento best-effort*
- *Protocolos de encaminamiento con calidad de servicio*

Veamos en qué consiste cada uno de estos grupos.

### *3.2.1 Protocolos de encaminamiento best-effort*

Muchos protocolos de encaminamiento [97], [117], [118], [119], [120], [121], [122], [123], [124], [108], [102], etc. tratan de encontrar la mejor trayectoria posible en un entorno altamente dinámico. Sin embargo, estos algoritmos sólo sirven para manejar el tráfico de datos best-effort y no son adecuados para aquellas aplicaciones con requisitos de calidad de servicio. Por este motivo ha sido necesario crear el denominado encaminamiento con QoS, el cual persigue dos propósitos [125]:

- *Seleccionar aquellas rutas que cuenten con recursos suficientes para satisfacer los parámetros de calidad de servicio.*
- *Conseguir mejorar globalmente la eficiencia de la red al utilizar mejor los recursos.*

Centremos nuestra atención en los protocolos de encaminamiento con calidad de servicio.

### *3.2.2 Protocolos de encaminamiento con calidad de servicio*

Los protocolos de encaminamiento con calidad de servicio [55] buscan aquellas rutas que disponen de recursos suficientes como para satisfacer las necesidades de un flujo.

Un módulo de gestión de recursos será el encargado de evaluar si existen suficientes recursos disponibles y se lo comunicará al protocolo de encaminamiento con calidad de servicio para ayudarle a encontrar las rutas que busca [55].

Las métricas de QoS se pueden clasificar en tres grupos:

❖ *Métricas aditivas*

Una métrica aditiva $A_m$ se define como:

$$\sum_{i=1}^{h} L_i(m), \tag{3.3}$$

donde $L_i(m)$ es el valor de la métrica $m$ a lo largo del enlace $L_i(m) \in P$, siendo $P$ la ruta (path) y $h$ la longitud en número de saltos de $P$. El parámetro $i$ representa el número de enlace.

❖ *Métricas cóncavas*

Una métrica cóncava representa el valor mínimo a lo largo de un camino $P$ y queda definida como:

$$C_m = \min(L_i(m)), \tag{3.4}$$

donde $L_i(m) \in P$.

❖ *Métricas multiplicativas*

Una métrica multiplicativa representa el producto de valores de métricas de QoS y se define como:

$$M_m = \prod_{i=1}^{h} (L_i(m)), \tag{3.5}$$

donde $L_i(m) \in P$.

El ancho de banda se considera una métrica cóncava, mientras que el coste, retardo y jitter son considerados métricas aditivas. Como ejemplo de métrica multiplicativa podría en cambio proponerse la disponibilidad o fiabilidad de los enlaces a lo largo de una ruta, basándose en alguna especie de probabilidad de rotura de cada enlace.

Para que un nodo en la red ad hoc sea capaz de realizar encaminamiento con QoS, necesita almacenar información referente a la topología de red y actualizarla constantemente mediante el intercambio de mensajes de encaminamiento del estado de los enlaces con otros nodos, consumiéndose recursos de ancho de banda y energía.

En una red ad hoc donde la rotura de enlaces es frecuente (debido al fading y la movilidad), el protocolo de encaminamiento con calidad de servicio debe estar habilitado para actuar con rapidez y recalcular una ruta adecuada, sin que el nivel de calidad de servicio proporcionado pueda verse afectado.

Se han propuesto muchos protocolos de encaminamiento con calidad de servicio, de entre los cuales destacaremos el protocolo Ticket-based y el protocolo AODV con

calidad de servicio, los cuales son descritos en las secciones *3.2.2.1* y *3.2.2.2* respectivamente.

## *3.2.2.1 Protocolo de encaminamiento con calidad de servicio 'basado en ticket' (Ticket-based)*

El protocolo de encaminamiento con QoS 'basado en ticket' [125] es un protocolo de encaminamiento con calidad de servicio. Está específicamente diseñado para redes ad hoc y es distribuido. Demuestra un buen rendimiento incluso cuando durante el cálculo de las rutas con QoS la información obtenida para poder operar es imprecisa.

La idea básica consiste en que el nodo fuente emite un determinado número de tickets que son enviados en paquetes de prueba para encontrar una ruta con QoS que sea óptima. Cada paquete de prueba contiene uno o más tickets, cada uno de los cuales representa un caso de la prueba.

El número de tickets delimita también el número máximo de rutas buscado. Por ejemplo, si el nodo fuente edita tres tickets es porque pueden probarse en paralelo un máximo de tres caminos.

La información de estado se mantiene en cada nodo de la red ad hoc y se basa fundamentalmente en estimaciones tanto del retardo extremo a extremo como del ancho de banda del camino disponible hacia cada nodo de la red. Cuando un nodo intermedio recibe un paquete de prueba puede enviarlo a varios nodos vecinos para descubrir más de una ruta o bien reenviarlo a un único nodo vecino basándose en la información de estado disponible en él mismo.

La precisión del estado de la información disponible en el nodo fuente y los requisitos de calidad de servicio de la petición de conexión determinan el número de tickets que serán generados. Cuanto más estrictos sean los requisitos de calidad de servicio solicitados o más imprecisa sea la información, mayor será el número de tickets generados para que aumente la probabilidad de encontrar una ruta adecuada. En cambio, cuando no sea necesario, se generarán menos tickets, de forma que la señalización introducida disminuirá.

La *Fig. 3.21* muestra un ejemplo de una red ad hoc donde se utiliza el protocolo de encaminamiento 'basado en ticket' para encontrar una ruta del nodo S al nodo t. Se mandan dos paquetes de prueba $P_1$ y $P_2$ desde S. El número entre paréntesis que sigue indica el número de tickets contenido en el paquete de prueba. En el nodo j, $P_2$ se divide en $P_3$ y $P_4$, cada uno de los cuales tiene un ticket. Como máximo hay tres

paquetes de prueba viajando por la red. Se buscan tres rutas: s→ i→ t, s→ j→ t y s→ j→ k→ t.

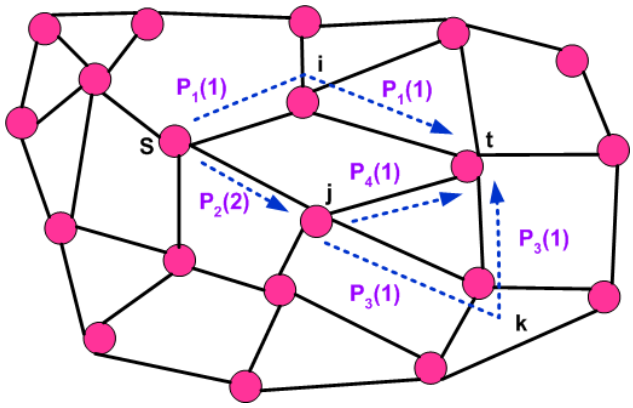



**Fig. 3.21.** Ejemplo de una red ad hoc que utiliza el protocolo de encaminamiento 'basado en ticket'.

Se proponen dos algoritmos heurísticos 'basados en ticket':

- *Encaminamiento orientado a QoS con retardo restringido (Delay-constrained QoS routing)*
- *Encaminamiento orientado a QoS con ancho de banda restringido (Bandwidth-constrained QoS routing)*

En el protocolo de encaminamiento orientado a QoS con retardo restringido, cada paquete de prueba contiene el retardo a lo largo de la ruta que hasta el momento ha efectuado ese paquete. Si, por ejemplo, un nodo intermedio A recibe un paquete de prueba PKT por parte de un nodo vecino B, el nodo A actualiza el campo de retardo en el paquete añadiendo el valor de retardo experimentado en el enlace de B a A al valor ya existente. El nodo A elabora una lista de nodos vecinos a quienes enviar paquetes de prueba. Entonces el nodo A distribuye los tickets presentes en el paquete de prueba PKT entre los nuevos paquetes de prueba que irán dirigidos cada uno a un nodo intermedio. Si al final terminan llegando muchos paquetes de prueba al destino (cada uno de los cuales contiene una lista de nodos intermedios), entonces éste seleccionará aquella ruta con el mínimo coste como ruta principal y el resto como rutas de reserva a usar cuando la primaria no esté disponible (debido, por ejemplo, a la movilidad de algún nodo intermedio).

Una característica del protocolo 'basado en ticket' consiste en que el nodo fuente emite dos tipos de tickets [126], amarillos y verdes, y los envía con paquetes de prueba.

Los tickets amarillos prefieren caminos que satisfagan las métricas de QoS (como, por ejemplo, el retardo, si se trata de un protocolo de encaminamiento con QoS y

retardo restringido). En cambio, los tickets verdes son expedidos para buscar aquellas rutas con QoS para las cuales el coste sea el mínimo.

La cantidad de tickets amarillos y verdes generada por el nodo fuente se basará en los requisitos de retardo y coste impuestos respectivamente por la petición de conexión.

Los tickets verdes se usan para encontrar una ruta de coste mínimo, mientras que los tickets amarillos quedarían como de reserva para aumentar la probabilidad de encontrar una ruta adecuada; por este motivo se utilizan dos tipos de tickets en vez de uno solo.

El protocolo de encaminamiento 'basado en ticket' trata de mejorar el ACAR (average call acceptance ratio). ACAR es el cociente del número de llamadas aceptadas con respecto al número de llamadas totales recibidas por la red, donde en este contexto 'llamada' es sinónimo de 'conexión'. Este protocolo se adapta dinámicamente a los requisitos de la aplicación y es capaz de funcionar correctamente, incluso conociendo una información de estado imprecisa.

Existe una especie de negociación entre el control de señalización adicional introducido y el coste de una ruta adecuada. La señalización adicional vendrá limitada por el número de tickets expedidos.

El protocolo asume que cada nodo mantiene información de estado global, pero para lograrlo se incrementa notablemente el consumo de ancho de banda.

Si la topología de la red ad hoc cambia muy rápidamente, puede ser que los algoritmos heurísticos no sean capaces de encontrar una ruta adecuada. En el protocolo de encaminamiento orientado a QoS con retardo restringido no se tiene en cuenta ni el tiempo de procesamiento ni el tiempo en de espera en cola cuando se calculan los retardos que los paquetes de prueba han experimentado. Por este motivo, algunos paquetes de datos sufren retardos excesivos cuando son enviados en realidad. Si es necesario volver a calcular las rutas y los paquetes de datos no pueden cumplir con los requisitos de tiempo real, entonces son transmitidos como paquetes best-effort, lo cual puede resultar perjudicial para aplicaciones de tiempo real con requisitos de calidad de servicio muy estrictos.

## *3.2.2.2 AODV con QoS*

Es un protocolo de encaminamiento con calidad de servicio.

En [127] los autores han modificado el protocolo de encaminamiento AODV [110] para que sea capaz de proporcionar calidad de servicio en redes ad hoc.

Los formatos de paquetes RREQ y RREP han sido alterados para que sean capaces de especificar los requisitos de calidad de servicio solicitados.

También ha cambiado la estructura de las tablas de encaminamiento, que tienen una entrada para cada nodo destino. Se les han añadido los campos siguientes:

- *Retardo máximo*
- *Ancho de banda disponible mínimo*
- *Lista de nodos fuente que piden garantías de retardo*
- *Lista de nodos fuente que piden garantías de ancho de banda*

Se ha añadido un campo denominado 'extensión de retardo máximo', cuyo significado es distinto para los paquetes de RREQ y RREP. Cuando el paquete RREQ contiene este campo indica el tiempo máximo que puede durar la transmisión desde ese nodo actual hasta el destino. En cambio, cuando el paquete RREP contiene este campo, indica el retardo estimado desde el nodo intermedio actual hasta el destino.

El nodo fuente utiliza este campo para encontrar una ruta hacia el destino con retardo máximo restringido.

Antes de enviar el paquete de RREQ, un nodo intermedio debe comparar el valor del NODE_TRAVERSAL_TIME (el tiempo que un nodo tarda en procesar el paquete) con el retardo sobrante indicado en el campo 'extensión de retardo máximo'. Si resulta que este retardo es menor que el NODE_TRAVERSAL_TIME, el nodo rechaza el paquete de RREQ. En caso contrario, el nodo resta del valor de retardo el NODE_TRAVERSAL_TIME y procesa el RREQ tal y como se especifica en el protocolo AODV.

El nodo destino devuelve un paquete RREP con el campo 'extensión de retardo máximo' conteniendo un valor de 0.

Cada nodo intermedio que reenvía el paquete RREP añade su propio NODE_TRAVERSAL_TIME al campo de retardo y reenvía el RREP hacia la fuente, aunque anteriormente cada nodo intermedio registra el valor del retardo en la entrada del destino correspondiente de la tabla de encaminamiento.

En la *Fig. 3.22* se muestra un ejemplo de una red ad hoc [128] que utiliza el protocolo de encaminamiento AODV con QoS con la extensión de retardo máximo para el encaminamiento. Como puede observarse, el $RREQ_1$ se propaga a través de los nodos intermedios (o routers del núcleo) a largo de la red durante el procedimiento de Descubrimiento de Ruta para encontrar una ruta que satisfaga el retardo máximo hasta que se devuelve un RREP al nodo fuente (o ingress router). En cambio, cuando se trata de localizar una nueva ruta usando el $RREQ_2$ que satisfaga el retardo máximo de 10 ms, se eliminará directamente el nuevo $RREQ_2$ en el nodo B porque el retardo

solicitado no puede ser cumplido, pues el retardo acumulado a lo largo de la ruta hacia el destino en el nodo B (almacenado en la caché) alcanza los 80 ms (cifra superior a 10 ms).

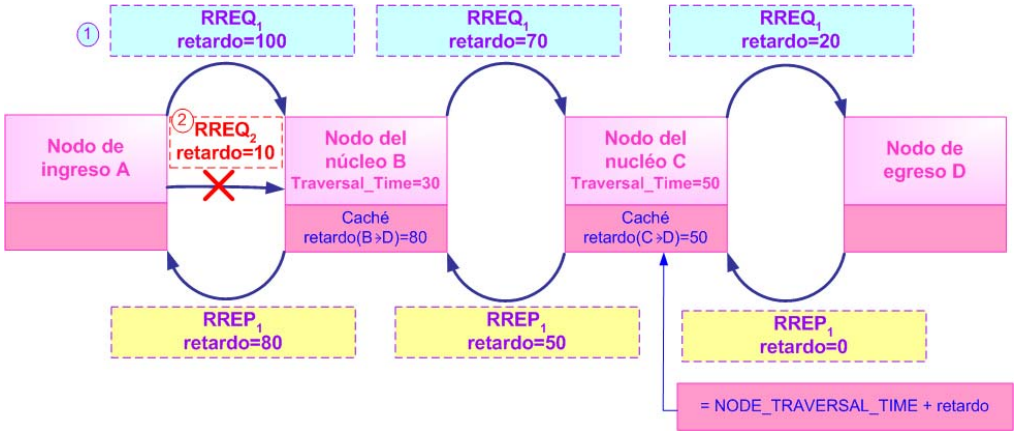



**Fig. 3.22.** Ejemplo de una red ad hoc que utiliza el protocolo de encaminamiento AODV con QoS con la extensión de retardo máximo.

De forma muy parecida, se propone también añadir el campo de 'extensión de ancho de banda mínimo' para encontrar una ruta (en caso de que exista) al nodo destino que satisfaga la restricción de ancho de banda mínimo. Antes de propagar el RREQ, un nodo intermedio debe de comparar su capacidad de enlace disponible con el valor del campo de extensión de ancho de banda. Si el ancho de banda solicitado no se encuentra disponible, el nodo debe descartar el RREQ, mientras que en caso contrario, el RREQ deberá ser procesado.

Cuando el destino genera un RREP en respuesta a un RREQ con una extensión de ancho de banda mínimo, el campo de ancho de banda en el RREP tiene un valor inicial de infinito (un valor muy elevado). Cada nodo que reenvía el RREP compara el campo de ancho de banda en el RREP con su propia capacidad del enlace y mantiene el mínimo de los dos valores en el campo de ancho de banda del RREP antes de reenviar dicho paquete. Este valor se introduce también en la entrada de la tabla de encaminamiento para el destino correspondiente e indica el ancho de banda mínimo disponible hacia ese destino. Así, es posible que un nodo intermedio responda a un RREQ posterior con una extensión de ancho de banda mínimo comparando el ancho de banda mínimo solicitado y el ancho de banda mínimo disponible almacenado en la entrada correspondiente de la tabla de encaminamiento.

En la *Fig. 3.23* se muestra un ejemplo de una red ad hoc que utiliza el protocolo AODV con QoS para el encaminamiento. El $RREQ_1$ se propaga a lo través de los nodos intermedios (o routers del núcleo) a largo de la red durante el procedimiento de Descubrimiento de Ruta para encontrar una ruta que satisfaga el ancho de banda mínimo hasta que se devuelve un RREP al nodo fuente (o ingress router). En cambio,

cuando se trata de localizar una nueva ruta usando el $RREQ_2$ que satisfaga el ancho de banda mínimo de 80 Kbps, se eliminará directamente el nuevo $RREQ_2$ en el nodo B porque el ancho de banda solicitado no puede ser satisfecho, pues el ancho de banda mínimo disponible almacenado en la caché del nodo B para el destino D es de 50 Kbps (cifra inferior a los 80 Kbps de ancho de banda mínimos necesarios).

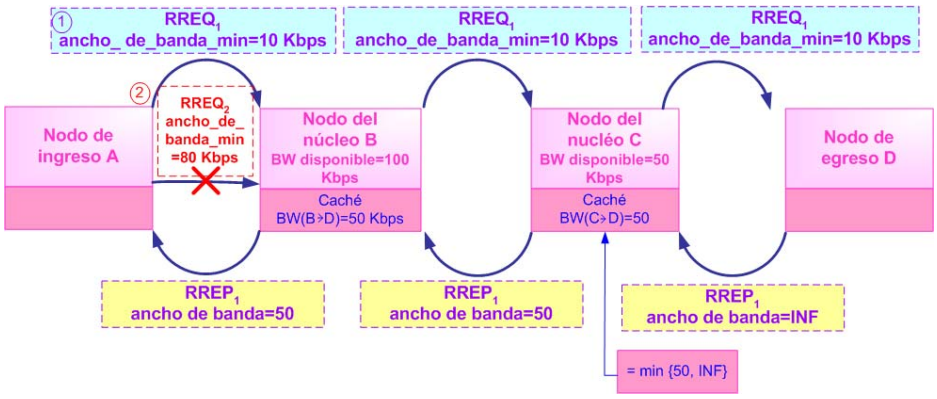



**Fig. 3.23.** Ejemplo de una red ad hoc que utiliza el protocolo de encaminamiento AODV con QoS con la extensión de ancho de banda mínimo.

Se genera un mensaje de 'QoS perdida' (QoS_LOST) [55] cuando un nodo intermedio advierte que ha aumentado el NODE_TRAVERSAL_TIME o bien se ha visto reducida la capacidad del enlace. Este mensaje QoS_LOST será enviado a todos aquellos nodos fuente que se considere que puedan verse afectados por el cambio de valor de este parámetro de QoS.

El protocolo AODV con QoS intenta proporcionar calidad de servicio modificando el ya conocido protocolo AODV para encontrar aquellas rutas que dispongan de recursos suficientes como para que las aplicaciones de tiempo real establecidas puedan funcionar adecuadamente. Sin embargo, este protocolo no es recomendable para aplicaciones que necesiten garantías estrictas de calidad de servicio, pues no se reservan recursos a lo largo de la ruta (aunque el obtener unas garantías de calidad de servicio estrictas con algún otro protocolo de encaminamiento resulta ser una utopía debido a las propiedades intrínsecas de las redes ad hoc).

Otra desventaja importante a destacar de este protocolo es que el tiempo NODE_TRAVERSAL_TIME usado para hacer los cálculos de retardo de los paquetes solamente contempla el tiempo de procesamiento, pero evita abordar otros retardos en la transmisión de los paquetes que van a ser muy considerables como el retardo a nivel de la capa MAC debido a la contienda por el acceso al medio. Por lo tanto, podemos concluir que los retardos de los paquetes serán bastante mayores que los estimados por el protocolo de encaminamiento, especialmente si la red ad hoc se encuentra congestionada. Para solucionar este problema se ha creado una nueva versión del protocolo que aparece explicada en [129] y en vez de usar el

NODE_TRAVERSAL_TIME define el FORWARDING_DELAY como el tiempo medio que un nodo intermedio tarda en procesar un paquete (incluyéndose en esta ocasión el tiempo de procesamiento, los tiempos medios de espera en cola y los retardos de propagación).

## *3.3 Encaminamiento y disponibilidad de energía*

En general, los nodos móviles de las redes ad hoc disponen de una batería de la que van consumiendo energía a medida que realizan diferentes operaciones hasta llegar a agotarla [99]. Por este motivo resulta de especial relevancia la conservación de la energía y la búsqueda de algoritmos de encaminamiento que a la hora de buscar la ruta óptima sean capaces de:

- *Maximizar el tiempo de vida de todos los nodos de la red.*
- *Minimizar la potencia total necesaria para encaminar paquetes.*

A continuación se presentan cuatro variaciones [130] de algoritmos de encaminamiento para redes ad hoc que consiguen alcanzar uno o los dos propósitos expuestos.

En el primer algoritmo, Minimum Total Transmission Power Routing (MTPR), para lograr obtener la potencia mínima total de una ruta entre los terminales $n_i$ y $n_j$ debería usarse como métrica la potencia de transmisión $P(n_i, n_j)$ que utiliza $n_i$ para transmitir a $n_j$.

Así, la potencia total de transmisión para la ruta $l$, $P_l$, sería:

$$P_l = \sum_{i=0}^{D-1} P(n_{i,} n_{i+1}), \tag{3.6}$$

donde los nodos intermedios $n_i \in ruta\ l\ (i = 0...D-1)$ y donde $n_0$ y $n_D$ son los nodos origen y destino, respectivamente.

La ruta deseada $k$ cumpliría:

$$P_k = \min_{l \in A} P_l, \tag{3.7}$$

siendo $A$ el conjunto de todas las rutas posibles.

Si modificamos el algoritmo de Dijkstra [139] para obtener la mínima potencia total de encaminamiento, como la potencia de transmisión depende directamente de la distancia, se seleccionan rutas con muchos saltos que además son inestables porque

la red varía dinámicamente su topología y aumenta considerablemente el retardo extremo a extremo de la red.

Para solucionar este problema aplicamos al algoritmo de Bellman-Ford [95] [96] distribuido una métrica de coste donde se incluye la potencia de recepción del transceptor (potencia consumida al recibir datos) que acostumbra a ser idéntica para todas las máquinas que utilizan el mismo transceptor.

Así, en el nodo $n_j$:

$$C_{i,j} = P_{transmit}(n_i, n_j) + P_{transceptor}(n_j) + f(n_j), \tag{3.8}$$

donde $n_i$ es el nodo vecino de $n_j$, $P_{transceptor}(n_j)$ representa la potencia del transceptor en el nodo $n_j$ y $f(n_j)$ es el coste de potencia total del nodo origen al nodo $n_j$.

El resultado se envía al nodo $n_i$, el cual calcula su coste en potencia como:

$$f(n_i) = \min_{j \in NH(i)} C_{i,j}, \tag{3.9}$$

donde $NH(i) = \{j;\ n_j\ es\ un\ nodo\ vecino\ de\ n_i\}$

Escogemos la ruta con mínimo coste desde el nodo origen a $n_i$ y repetimos este procedimiento hasta que se alcanza el nodo destino.

La desventaja de utilización de este tipo de encaminamiento es que una ruta es seleccionada para minimizar la potencia total de consumo de a lo largo de ese camino y por tanto de toda la red; sin embargo, no se tiene en cuenta cuál es el tiempo de vida de los nodos que pertenecen a esa ruta seleccionada; a dichos nodos se les exigirá un consumo de energía de sus baterías para poder hacer posible el encaminamiento de paquetes y si este consumo fuera excesivo para sus posibilidades, sus baterías podrían descargarse hasta llegar a agotarse. Por este motivo se ha definido un nuevo algoritmo: Minimum Battery Cost Routing (MBCR). En [132] se propone introducir características de la batería directamente en el protocolo de encaminamiento utilizando la capacidad de batería sobrante como métrica del tiempo de vida de cada terminal.

Definimos $f_i(c_i^t)$ como la función de coste de batería de la máquina $n_i$, donde $c_i^t$ representa la capacidad de la batería para el host $n_i$ en $t$ y toma un valor dentro de un rango entre 0 y 100. $f_i$ puede definirse como:

$$f_i(c_i^t) = \frac{1}{c_i^t}, \tag{3.10}$$

El coste de la batería $R_j$ para la ruta $j$ sería:

$$R_j = \sum_{i=0}^{D_j-1} f_i\left(c_i^t\right), \tag{3.11}$$

Entonces, para escoger la ruta con una capacidad de batería sobrante máxima, debería seleccionarse la ruta $i$ que tuviera el mínimo coste de batería que cumpliera:

$$R_i = \min\{R_j \mid j \in A\}, \tag{3.12}$$

donde $A$ contiene todas las posibles rutas.

La desventaja en la utilización de este algoritmo es que podría llegar a seleccionarse una ruta donde la energía que se consumiría en total sería la mínima, pero donde a la vez alguno de los nodos de dicha ruta se viera seriamente perjudicado, pues podría poseer una batería con muy poca capacidad y verse obligado a usarla hasta que se agotase.

Esto es lo que sucede en la *Fig. 3.24*, donde a la hora de seleccionar una de las dos posibles rutas entre una fuente y un destino se selecciona aquella ruta a través de la cual el coste total de batería consumido es menor. Por consiguiente, se escoge la ruta 1, aunque el nodo 3 de dicha ruta tenga una capacidad de batería muy baja y exista el peligro de que termine extinguiéndose.

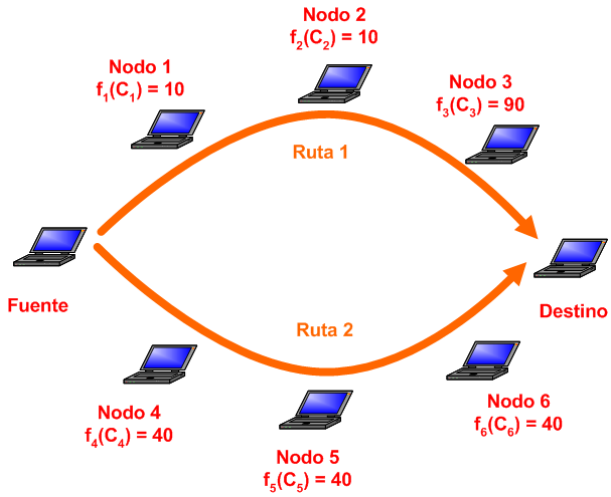



**Fig. 3.24.** Ejemplo de una red ad hoc que utiliza un protocolo de encaminamiento que tiene en cuenta la capacidad de batería.

Para evitar este problema se han mejorado las definiciones anteriores, creándose el algoritmo Min-Max Battery Cost Routing (MMBCR) [130].

En este caso, el coste de la batería $R_j$ para la ruta $j$ sería:

$$R_j = \max_{i \in ruta\_j} f_i\left(c_i^t\right) \tag{3.13}$$

La ruta $i$ deseada se calculará como:

$$R_i = \min\{R_j \mid j \in A\} \tag{3.14}$$

Utilizando este algoritmo se seleccionará ahora, en el ejemplo de la *Fig. 3.24,* la ruta 2 como ruta óptima para llegar de la fuente al destino.

Sin embargo, este algoritmo presenta la desventaja de que no siempre se puede garantizar que se escogerán los caminos de mínima potencia total de transmisión, con lo cual se reduciría también el tiempo de vida de los nodos de la red.

Con los algoritmos expuestos anteriormente no ha sido siempre posible maximizar el tiempo de vida de cada uno de los nodos de la red y a la vez conseguir que aquellos nodos con mayor capacidad de batería sean los que más consuman (haciendo un uso justo de la batería de cada nodo). Para lograr alcanzar estos dos objetivos definimos el algoritmo Conditional Max-Min Battery Capacity Routing (CMMBCR) [131]. La idea básica es que cuando todos los nodos de alguna de las rutas posibles entre un origen y un destino tienen suficiente capacidad de batería sobrante (por encima de un umbral), escogemos una ruta de entre estas que minimice la potencia total de transmisión. En cambio, si todas las rutas contienen nodos con menor capacidad de batería (por debajo de un umbral), las rutas que incluyen estos nodos son evitadas con objeto de extender el tiempo de vida de dichos nodos.

Se define la capacidad de la batería $R_j^c$ de la ruta $j$ en el tiempo $t$ como:

$$R_j^c = \min_{i \in ruta\_j} c_i^t, \tag{3.15}$$

$A$ es el conjunto que contiene las rutas posibles entre dos nodos en un tiempo $t$ que satisfacen la ecuación:

$$R_j^c \geq \gamma, \tag{3.16}$$

para cada $j \in A$. $\gamma$ representa un umbral entre 0 y 100.

Si $Q$ representa el conjunto que contiene todo los caminos posibles entre una fuente y un destino específicos en un tiempo $t$:

(a) Si $A \cap Q \neq \phi$, escoger una ruta en $A \cap Q$ aplicando el algoritmo MTPR.

(b) Sino, seleccionar la ruta $i$ con capacidad de batería máxima usando $R_i^c = \max\{R_j^c \mid j \in Q\}$.

Nótese que el rendimiento de este algoritmo (CMMBCR) dependerá del valor del umbral $\gamma$. Si $\gamma = 0$, la ecuación *(3.16)* se cumple siempre y es como si en todo momento se utilizara como protocolo de encaminamiento el algoritmo MTPR. Por otro lado, si $\gamma = 100$, la ecuación *(3.16)* no se cumple nunca y es como si siempre se utilizara como protocolo de encaminamiento el algoritmo MMBCR, evitando utilizar aquellas rutas que contienen nodos con menor capacidad de batería.

Se han realizado simulaciones [130] con 30 nodos distribuidos en un espacio de 100 m por 100 m. Cada nodo se mueve aleatoriamente a una v = 2 m/s. Se simula el tiempo de expiración de cada nodo en función de la secuencia (orden) de expiración de los nodos.

Se ha concluido a partir de las simulaciones que MBCR y MMBCR son algoritmos capaces de evitar la explotación o sobreconsumo de ciertos nodos cuando la mayoría de potencia consumida se debe a la transmisión y recepción. Así se retarda el tiempo de agotamiento de la batería del primer nodo. Sin embargo, como estos algoritmos no tienen en cuenta la minimización de la potencia de transmisión, se seleccionan proporcionalmente rutas más largas, con lo cual aumenta la carga media por nodo y así se reduce el tiempo de vida de muchos nodos de la red.

El algoritmo CMMBCR [131] representa, en cambio, un equilibrio entre el objetivo de minimizar la potencia de transmisión y maximizar el tiempo de vida de los nodos de la red.

Han surgido otras propuestas que relacionan el encaminamiento con la disponibilidad de energía. En [133] se estudian dos protocolos de encaminamiento que ajustan la potencia de transmisión de forma dinámica y tienen en cuenta para su correcto funcionamiento las tasas de error de la capa de enlace de datos y las consiguientes retransmisiones de paquetes. Estas consideraciones motivan que en [134] aparezca un protocolo de encaminamiento basado en una función de coste que tiene en cuenta tanto la tasa de error del enlace como la energía necesaria para realizar un único intento de transmisión a través del enlace.

En el estado ‘sleep’, un nodo no puede transmitir ni recibir datos, con lo cual consume mucha menos energía que en el estado activo; en este último estado se consume más energía aunque no se estén enviando ni recibiendo datos, porque se necesita mantener alimentados a los circuitos. Debido a esta razón, se han desarrollado diversos protocolos que tratan de mantener a los nodos una buena parte del tiempo en el estado ‘sleep’ y menos tiempo despiertos en el estado activo. Estos protocolos tratan de maximizar el ahorro de energía, minimizando al mismo tiempo el impacto introducido en el thoughput, la latencia y la latencia en el Descubrimiento de Ruta y trabajan tanto a nivel e la capa de red [135], [136], [137], [138] como a nivel de la capa de enlace de datos [139]. En [139] se presenta un protocolo que conserva la capacidad de batería de los nodos ‘apagando’ aquellos nodos que no se dedican a la transmisión y recepción activa de paquetes.

## *3.4 Comparación entre distintos protocolos de encaminamiento y elección de un protocolo de encaminamiento para redes ad hoc aisladas*

Numerosos trabajos de investigación [94], [56], [99] han tratado de clasificar y comparar los protocolos de encaminamiento definidos para redes ad hoc aisladas.

En [140], [141], [142] se han realizado diversos estudios comparando los protocolos de encaminamiento DSR y AODV con otros protocolos de encaminamiento tales como TORA o DSDV y se ha demostrado que en términos generales DSR y AODV superan en rendimiento al resto de protocolos. Por lo tanto, a la hora de realizar simulaciones en esta tesis doctoral se escogerá uno de estos dos protocolos dependiendo de cual sea el sistema que desee modelarse.

Además, se han realizado numerosos estudios comparando exclusivamente los protocolos de encaminamiento AODV y DSR [143], [144], [145]. Veamos cuáles son sus principales similitudes y diferencias.

*Ventajas de usar DSR como protocolo de encaminamiento:*

- *En DSR cada nodo intermedio a lo largo de una ruta puede aprovechar la opción 'promiscuous listening' y el hecho de que el encaminamiento esté basado en la fuente para aprender rutas. AODV (la versión clásica) necesitará en cambio realizar más procedimientos de Descubrimiento de Ruta para obtener la misma información.*
- *Con DSR la fuente aprende una ruta principal y varias alternativas después de haber realizado un procedimiento de Descubrimiento de Ruta. Esta propiedad resulta muy beneficiosa en una red de baja movilidad. Con AODV (la versión clásica) solamente se almacena en la fuente una ruta hacia el destino.*
- *DSR no requiere el intercambio periódico de mensajes de 'hello' entre vecinos para comprobar la conectividad, haciendo posible que los nodos de la red ad hoc puedan ponerse en el estado 'sleep' para conservar su energía, ahorrándose además un ancho de banda considerable.*
- *DSR puede ser usado tanto con enlaces unidireccionales como bidireccionales.*

*Ventajas de usar AODV como protocolo de encaminamiento:*

- *En AODV existen temporizadores para que las rutas no usadas antiguas terminen expirando si no son utilizadas. En DSR solamente existe esta característica en las últimas versiones.*
- *En los paquetes RREQ de DSR viaja toda la información referente a una ruta (encaminamiento basado en la fuente), mientras que en los paquetes AODV (la versión clásica) viaja la dirección destino del paquete, disminuyendo por lo tanto el tamaño de la señalización. Lo mismo sucede con los paquetes RREP.*
- *Cuando se produce un error en AODV se envía un paquete RERR a todos los nodos que utilicen el enlace que ha caído. En DSR solamente existe esta característica en las últimas versiones.*

*Resultados de la comparación entre AODV y DSR (las versiones clásicas)* [143], [144], [145]*:*

- *En redes con menos condiciones extremas (menor tamaño, carga y/o movilidad) se obtiene mejores resultados en cuanto al throughput y al retardo extremo a extremo para DSR. En cambio, en redes más cargadas y con mayor movilidad (condiciones más extremas) AODV supera en estas situaciones a DSR porque la información adicional que los paquetes de encaminamiento en DSR transportan en su cabecera acerca de toda la ruta (encaminamiento basado en la fuente) crecerá con el diámetro de la red y consumirá mucho ancho de banda. (Aunque DSR genera menor número de mensajes de señalización que AODV).*
- *DSR funciona correctamente a distintas velocidades de movimiento para los nodos, aunque al utilizar encaminamiento basado en la fuente aumenta el tamaño de la señalización. AODV funciona prácticamente igual de bien que DSR con distintas velocidades de movimiento para los nodos y no usa encaminamiento basado en la fuente pero debe poner en marcha a altas movilidades muchos procedimientos de Descubrimiento de Ruta y aumenta su señalización.*

## 3.5 Contribución: Desarrollo del protocolo de encaminamiento SEADSR para la mejora de la supervivencia en redes ad hoc aisladas

A continuación se presenta una contribución de esta tesis doctoral consistente en el diseño e implementación de un protocolo de encaminamiento denominado SEADSR para la mejora de la supervivencia en redes ad hoc aisladas.

### 3.5.1 Explicación teórica

Seguidamente se realiza una explicación teórica del protocolo SEADSR, que incluye una descripción formal así como un análisis detallado del algoritmo.

#### 3.5.1.1 Descripción de SEADSR

En este trabajo de investigación se asume que la potencia de transmisión es fija e igual para todos los nodos. No es aconsejable asumir que cada dispositivo móvil es capaz de ajustar dinámicamente sus rangos de transmisión (como hace el algoritmo MTPR y también el algoritmo CMMBCR en el caso de que se satisfaga una condición determinada [130]), porque la reducción de la potencia de transmisión en un nodo introduciría serios problemas en el protocolo a nivel de la capa MAC CSMA/CA del estándar IEEE 802.11 que hemos seleccionado. Si se asumiera que cada nodo era capaz de ajustar sus niveles de potencia de transmisión dinámicamente, el intercambio de paquetes RTS/CTS entre dos estaciones con el fin de reducir el problema del terminal escondido *(Véase la sección 2.1.1.1 Protocolo DCF del MAC IEEE 802.11, pág. 23)* no sería escuchado por parte de sus nodos vecinos, con lo cual existiría una mayor probabilidad de que se produjeran colisiones de paquetes.

Se ha seleccionado el protocolo DSR y no el AODV para ser mejorado porque DSR muestra mejor comportamiento con respecto al consumo de energía en comparación con AODV o el resto de protocolos [140].

El protocolo DSR [104] no puede en principio incorporar ninguno de los algoritmos mencionados en la sección *3.3 Encaminamiento y disponibilidad de energía* porque presenta dificultades a la hora de introducir una función de coste directamente en dicho protocolo, ya que DSR selecciona la ruta dependiendo de los tiempos de llegada a los nodos intermedios de los paquetes de RREQ que tienen el mismo identificador durante el proceso de Descubrimiento de Ruta. En DSR, un paquete de RREQ se

retransmite únicamente si no ha llegado anteriormente otro RREQ con una dirección del nodo fuente, dirección del nodo destino y un identificador idénticos al mismo nodo. Por esta razón, no es posible utilizar una función de coste que nos devuelva la trayectoria más corta posible de entre todas las rutas, ya que algunas rutas que quizás serían más óptimas no llegan jamás a poder descubrirse debido a que cuando se propagan los RREQ para localizarlas, anteriormente a algún nodo intermedio concreto ya ha llegado otro RREQ procedente de alguna ruta diferente que motiva que el siguiente RREQ perteneciente a esta ruta tenga que ser rechazado. Lo único que sí sería posible es establecer una función de coste para tratar de buscar la trayectoria más corta posible de entre todas las rutas que se descubren, aunque no se garantice que la ruta más óptima de todas las existentes haya llegado a poder ser descubierta.

A continuación se introduce un nuevo protocolo de encaminamiento que mejora la supervivencia de la red, manteniendo al mismo tiempo la sencillez de DSR: Simple Energy Aware Dynamic Source Routing (SEADSR).

La idea básica detrás de este algoritmo es la siguiente:

Cuando un nodo intermedio en una red ad hoc decide reenviar un mensaje de RREQ que ha recibido (siguiendo el procedimiento dictado por el protocolo de encaminamiento DSR), se introduce un retardo adicional antes de retransmitir dicho mensaje:

$$\tau = \frac{\left[C_{\max} - C(t)\right] \times \tau_{\max}}{C_{\max}}, \tag{3.17}$$

donde $C_{\max}$ es la capacidad máxima de la batería, $C(t)$ es el nivel de batería actual y $\tau_{\max}$ hace referencia a un parámetro de diseño que representa el máximo retardo introducido. Podemos apreciar que $\tau$ toma un valor entre 0 y $\tau_{\max}$ y es directamente proporcional a la energía consumida por el nodo.

Se asume que tanto en DSR como en SEADSR la ruta seleccionada será aquella que consiga llegar la primera al nodo fuente, transportada a través de un paquete de RREP.

En DSR, la selección de una ruta dependerá de los tiempos de llegada de los RREQ a los nodos intermedios y la circunstancia de que un RREQ llegue antes que otro a un nodo se relaciona con otros factores tales como el número de saltos a lo largo de una ruta, la congestión o el número de colisiones a nivel de la capa MAC para cada nodo intermedio a través de una ruta.

La selección de la ruta dependerá, en SEADSR, al igual que en DSR, de los factores previamente mencionados, pero este retardo adicional introducido establece además una interdependencia entre la selección de una ruta y los niveles de batería de los

nodos. Con la introducción del retardo adicional descrito en la ecuación *(3.17)*, aumenta la probabilidad de que los paquetes de RREQ transmitidos por nodos intermedios que tienen más nivel de batería sobrante $C(t)$ lleguen a otros nodos intermedios antes que aquellos paquetes que son transmitidos por nodos con un nivel de batería actual $C(t)$ bajo, favoreciéndose por tanto la selección de aquellas rutas que contienen nodos con un alto nivel de batería.

La *Fig. 3.25* ilustra la función $\tau(C)$ del retardo adicional introducido en *(3.17)* que se aplicará en cada nodo de la red ad hoc. El retardo adicional que experimente un paquete de RREQ que llegue a un nodo vendrá dado por la capacidad de batería de dicho nodo después de haber aplicado la función de la figura y este retardo oscilará entre 0 (cuando un nodo mantiene intacta su capacidad de batería) y $\tau_{\max}$ (cuando la capacidad de batería de un nodo se ha extinguido).

Es importante resaltar que el funcionamiento del algoritmo será diferente dependiendo de la función de retardo seleccionada. No es posible encontrar la función de retardo óptima debido a la variabilidad del tráfico y de la topología de red; no obstante, otras funciones de retardo alternativas podrían mejorar el rendimiento general del sistema [146].

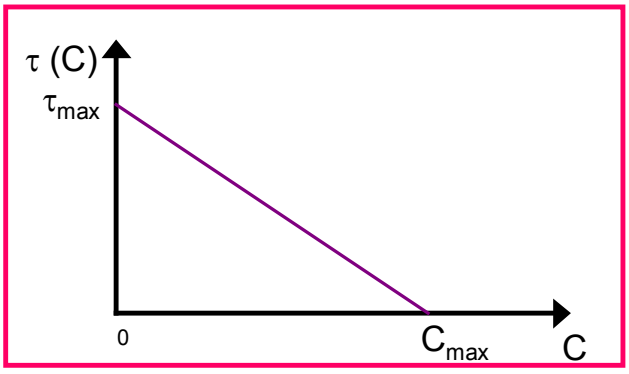



**Fig. 3.25.** Función de retardo $\tau(C)$ aplicada sobre un paquete de RREQ que llega a un nodo intermedio para determinar el retardo adicional que se introducirá sobre dicho paquete.

El parámetro $\tau_{\max}$ desempeña un papel importante en el proceso de selección de ruta. Cuanto mayor sea el parámetro $\tau_{\max}$, mayor será la influencia de los niveles de batería en comparación con otros factores. Si escogemos un valor de $\tau_{\max}$ grande, el algoritmo tenderá a seleccionar rutas que comprenden a nodos con altos niveles de batería, pero posiblemente tendrán más saltos (porque si un nodo tiene un nivel de batería alto es porque probablemente ha consumido niveles bajos de potencia de transmisión y la potencia de transmisión es directamente proporcional a la distancia). Por el contrario, si se escoge un valor pequeño para $\tau_{\max}$, los niveles de batería de los

nodos intermedios ya no serán un factor decisivo y el algoritmo seleccionará con una mayor probabilidad rutas con menos saltos y niveles de batería más bajos.

En [147] se presentaron simulaciones relacionadas con la supervivencia de red para redes con un número significativo de flujos de tráfico. Los resultados mostraron una mejora sobre DSR para los valores de escenario y parámetros seleccionados. El esquema en [147] se basa también en el mecanismo básico explicado en esta sección. Pero además, se ha llegado a la conclusión de que los valores de timeout de las cachés son decisivos en el proceso de selección de rutas y por consiguiente en el rendimiento del algoritmo. Cuando una ruta ha permanecido activa durante un tiempo, los nodos que forman parte de la ruta tendrán niveles de batería bajos y dicha ruta ya no será la óptima (desde el punto de vista energético). A pesar de todo, como esta ruta se halla contenida en la caché, será escogida preferiblemente antes que otras a la hora de establecer un nuevo flujo de datos hacia el mismo destino y no será posible iniciar un nuevo proceso de Descubrimiento de Ruta para localizar otras rutas con mejores propiedades desde el punto de vista energético. Por este motivo, se ha concluido que la caché de rutas perjudica al buen funcionamiento del algoritmo en cuanto a energía se refiere. Por consiguiente, el algoritmo de encaminamiento SEADSR finalmente propuesto no tiene caché de rutas. SEADSR es, por tanto, una modificación de DSR que suaviza los requerimientos de memoria.

En [146] se sugiere introducir también un retardo en el procedimiento de Descubrimiento de Ruta de DSR. No obstante, el algoritmo SEADSR es diferente debido a la inexistencia de cachés. Además, con SEADSR se han realizado simulaciones considerando redes mayores (escalabilidad) y mucho más móviles.

Las publicaciones [147], [148], [149] y [150] acreditan el buen funcionamiento del algoritmo SEADSR.

## *3.5.1.2 Análisis de SEADSR*

Considérese una topología de red como la ilustrada en la *Fig. 3.26*. En $t$ = 0 s un nodo fuente llamado S quiere enviar paquetes a un nodo destino D. Por lo tanto, el protocolo de encaminamiento debe encontrar el mejor camino hacia el destino. Se consideran dos posibles rutas llamadas SABD que atraviesa los nodos A y B, que tienen una capacidad de batería de 30 J, y la ruta SCD que atraviesa el nodo C con una capacidad de batería de 14 J. La capacidad de batería máxima para todos los nodos es $C_{\max}$ = 40 J y se asume que las fuentes tienen unas reservas de energía infinitas. Cuando un nodo actúa como nodo intermedio o router, se considera que

consume una potencia de transmisión de 5 W. Para facilitar el análisis se ha supuesto que las potencias de recepción y procesamiento son despreciables. Se ha asumido que en $t$ = 1 s otra fuente llamada S' desea enviar paquetes al mismo nodo destino, de forma que es necesario establecer una nueva ruta. Se ha supuesto que la red utiliza diferentes protocolos de encaminamiento para poder estudiarlos y compararlos. Los tres protocolos propuestos son MMBCR *(Véase la sección 3.3 Encaminamiento y disponibilidad de energía, pág. 129)*, DSR *(Véase la sección 3.1.2.1 Dynamic Source Routing (DSR), pág. 102)* y SEADSR.

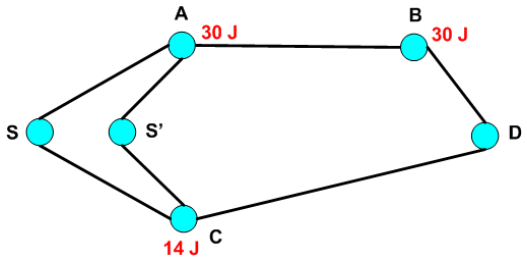



**Fig. 3.26.** Red ejemplo. Se muestran los valores de energía.

En la *Fig. 3.27* se ha comparado la energía consumida por los nodos intermedios en función del tiempo para los tres esquemas de encaminamiento.

Podemos apreciar que DSR consigue los peores resultados en términos de supervivencia de red, porque las dos fuentes siempre seleccionan la ruta a través de los nodos CD, de forma que en $t$ = 1,9 s el nodo C ha agotado su energía y únicamente permanece disponible la otra ruta. Por otro lado, el protocolo de encaminamiento MMBCR consigue en estos casos rutas con un tiempo de vida mayor en comparación con DSR porque las dos fuentes siempre usan la ruta a través de los nodos ABD que contiene nodos intermedios con mayor capacidad de batería. Por lo tanto, es cierto que con este protocolo de encaminamiento se consume más energía pero al mismo tiempo mejora la supervivencia de red, pues los nodos A y B agotan sus reservas de energía en $t$ = 3,5 s.

Finalmente, podemos apreciar que SEADSR logra un funcionamiento mejor. En este caso cada fuente selecciona una ruta diferente (SABD es escogida por la fuente S ya que los nodos A y B retardan el RREQ 0,25* $\tau_{max}$ cada uno y el nodo C retarda el RREQ 0,65* $\tau_{max}$ de acuerdo con *(3.17)*; S'CD es escogida por la fuente S' ya que los nodos A y B retardan el RREQ 0,375* $\tau_{max}$ cada uno y el nodo C retarda el RREQ 0,65* $\tau_{max}$ ). Así, la energía total consumida es mayor que en DSR y menor que en MMBCR, pero a la vez se extiende el tiempo de vida de todos los nodos de la red más que en los otros dos protocolos de encaminamiento. El nodo C agota su energía en $t$

= 3,8 s. Podemos considerar este caso como el típico, porque los niveles de energía en los diferentes nodos no son muy dispares.

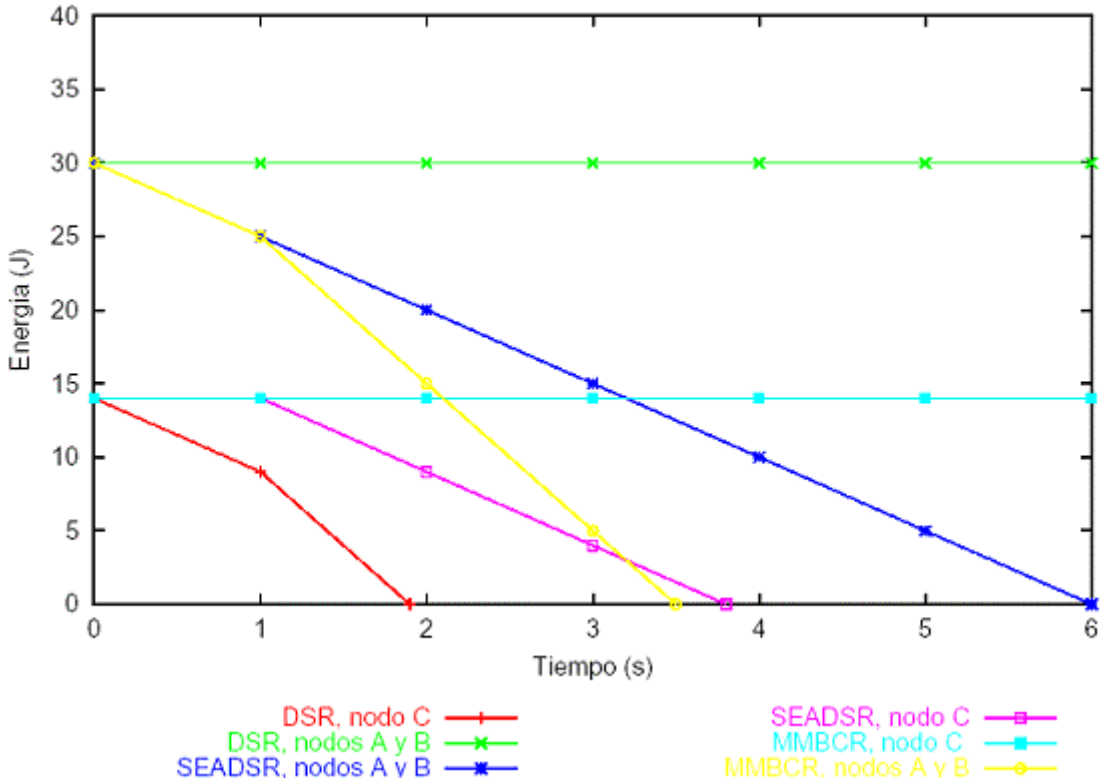



**Fig. 3.27.** Energía con DSR, SEADSR y MMBCR para la red de la *Fig. 3.26* ( $C_{\max}$ = 40 J).

Ahora se considerará el ejemplo de la *Fig. 3.28,* donde tenemos la misma topología y condiciones de red que en la *Fig. 3.26*, pero en este caso se asume que los nodos A y B tienen niveles de batería de 50 J y 10 J respectivamente. La capacidad de batería máxima para todos los nodos es $C_{\max}$ = 60 J.

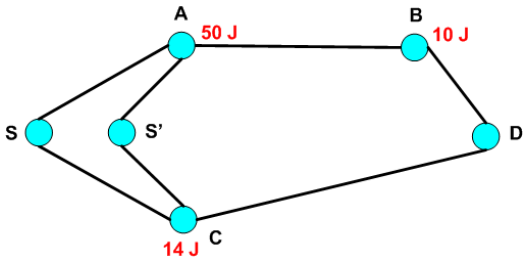



**Fig. 3.28.** Red ejemplo. Se muestran los valores de energía.

Podemos apreciar en la *Fig. 3.29* que DSR y SEADSR experimentan los peores resultados en términos de supervivencia de red, porque la ruta SCD es siempre seleccionada por las dos fuentes, de forma que en $t$ = 1,9 s el nodo C ha agotado su capacidad de batería y sólo la otra ruta permanece disponible. Este es el peor caso para el protocolo SEADSR: Únicamente puede funcionar igual que DSR en las peores condiciones y topología de red. Podemos apreciar que las condiciones de esta red

benefician al protocolo MMBCR, el cual selecciona la ruta SCD para la primera fuente y la ruta S'ABD para la segunda. A pesar de que la energía consumida es mayor, el consumo se distribuye a lo largo de las dos rutas de forma que los nodos B y C agotan sus reservas de energía en $t$ = 3 s y $t$ = 2,8 s respectivamente.

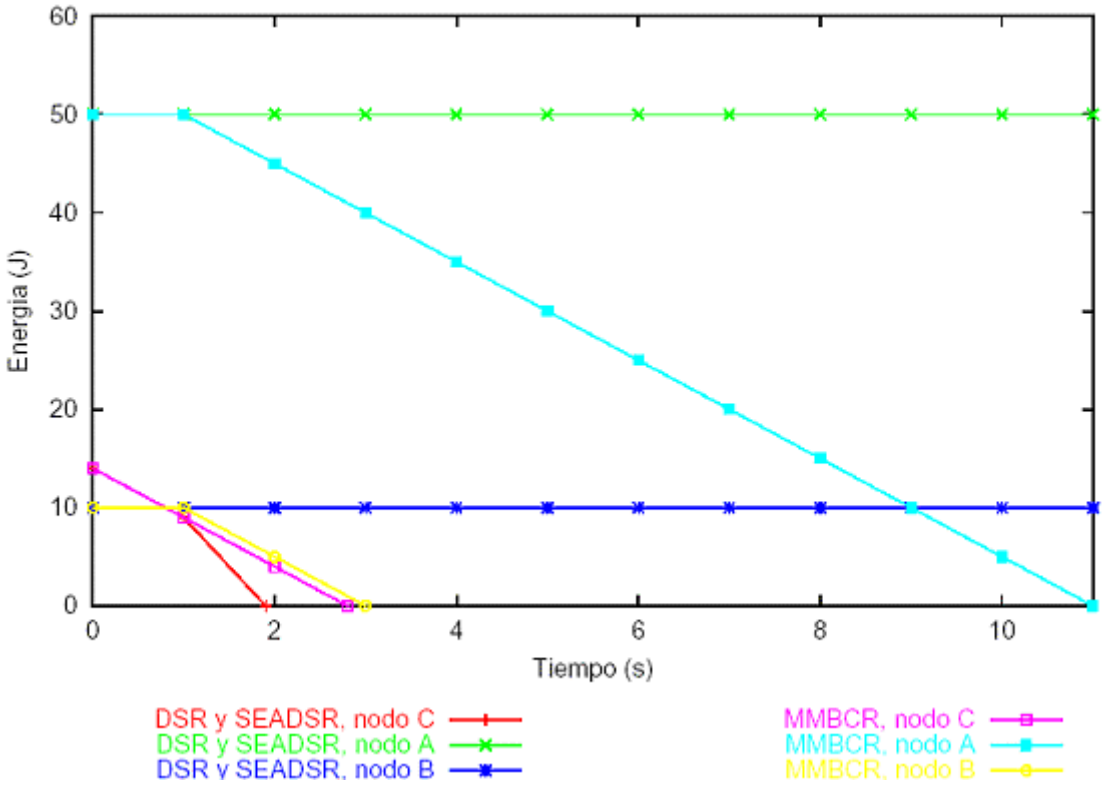



**Fig. 3.29.** Energía con DSR, SEADSR y MMBCR para la red de la *Fig. 3.28* ($C_{max}$ = 60 J).

## *3.5.2 Simulaciones*

A continuación se presenta el escenario de simulación así como un análisis detallado de las simulaciones que fueron llevadas a cabo en el contexto de esta tesis doctoral.

### *3.5.2.1 Escenario de simulación*

El simulador usado en este trabajo para la evaluación de los protocolos de encaminamiento es ns-2 [151].

La tecnología usada a nivel de las capas física y de enlace de datos es IEEE 802.11b. Cien terminales móviles se distribuyen aleatoriamente de acuerdo con una distribución uniforme en una región cuadrada de 600 m por 600 m. El tamaño del área y el número de nodos han sido seleccionados de forma que en media se necesitan varios saltos para llegar de la fuente al destino.

Cada nodo elige un punto de destino al azar dentro del área y se mueve hacia él a una velocidad $v$ uniformemente distribuida entre 0 y 3 m/s. Una vez el nodo ha alcanzado su destino, hace una pausa durante un periodo fijo de 20 segundos, escoge otro destino y repite el proceso.

Un nodo cualquiera puede ser seleccionado como nodo fuente con una probabilidad de 0,16 y tratará de encontrar una ruta hacia el destino durante un intervalo de tiempo entre 0 y 180 s. A diferencia de los resultados obtenidos en [146], la gran cantidad de flujos de tráfico que hemos estudiado nos permite determinar cómo se comportan las cachés. Se envían paquetes UDP (User Datagram Protocol) de 512 bytes, espaciados 100 ms y generados por fuentes de tráfico de tasa constante, CBR (Constant Bit Rate).

El parámetro de diseño $\tau_{\max}$ es 1 s para el SEADSR.

En las simulaciones realizadas se ha considerado que el nodo fuente siempre selecciona tanto en DSR como en SEADSR la primera ruta que le llega hacia ese destino concreto, pero podría en realidad haberse seleccionado la ruta más corta en términos de número de saltos o bien en relación a otros criterios.

Puesto que el objetivo es analizar el impacto de los protocolos de encaminamiento en la supervivencia de la red, se han considerado las pérdidas de energía de los nodos de la red ad hoc debidas únicamente a su funcionamiento como nodos intermedios, es decir, debido a la energía necesaria para reenviar paquetes originados por otros nodos. Por esta razón, se han modelado los nodos con dos baterías: una de estas baterías proporcionará energía permanentemente cuando el nodo funcione como nodo fuente; la otra capacidad de batería disminuirá cada vez que el nodo reenvíe un paquete, de forma similar a lo que ocurre en un nodo intermedio. La variable $C(t)$ en *(3.17)*, utilizada por el algoritmo de encaminamiento, representa el nivel de la segunda de estas baterías. Aunque este modelo pudiera considerarse poco realista, la existencia de una única batería produciría una rápida disminución del nivel energético de algunos nodos (las fuentes), impidiendo así el análisis del protocolo.

## *3.5.2.1.1 Análisis de las simulaciones*

La *Fig. 3.30* muestra las pérdidas de nodos debido al agotamiento de su energía en función del tiempo para el DSR estándar y el SEADSR. Se ha considerado que todos los nodos tienen una capacidad de batería de 0,8 J al principio de la simulación y se ha representado el tiempo medio de fallo del nodo enésimo usando DSR y SEADSR. Los intervalos de confianza son del 90% para 80 experimentos independientes. Las simulaciones tienen una duración de 600 s.

Se puede observar que en el estándar DSR hay más nodos que han agotado su capacidad de batería en función de tiempo que en SEADSR. Hasta el segundo 300 los intervalos de confianza son demasiado grandes para extraer conclusiones. No obstante, a partir de ese punto puede apreciarse que el número de nodos caídos aumenta en el estándar DSR más rápidamente que en SEADSR. Por consiguiente, SEADSR supera a DSR en cuanto a la supervivencia de la red.

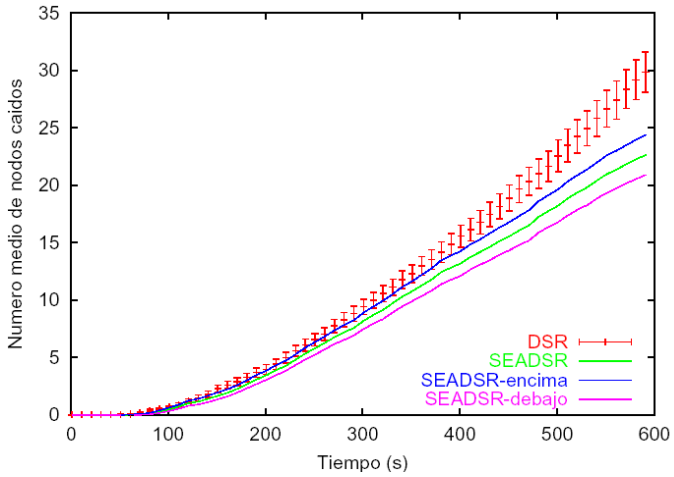



**Fig. 3.30.** Fallo de los nodos debido al agotamiento de sus reservas de energía en función del tiempo. Se muestran los intervalos de confianza del 90%.

Para comparar el protocolo de encaminamiento SEADSR con DSR en relación al throughput y al retardo de paquetes, se han hecho simulaciones de un sistema con suficiente capacidad de batería, de forma que durante el tiempo de simulación no fallaran los nodos debido al agotamiento de su energía. Después de algunas pruebas, se descubrió que todos los nodos de la red con una capacidad de batería de 5 J permanecen con vida al menos 400 s.

La *Fig. 3.31* muestra el throughput en función del tiempo para el estándar DSR y SEADSR, obtenido a partir de 80 experimentos independientes. Los primeros 180 s de la *Fig. 3.31* corresponden al periodo durante el cual las fuentes comienzan a enviar tráfico.

Cuando $t$ < 150 s, DSR es superior a SEADSR, puesto que selecciona las mejores rutas en relación al retardo extremo a extremo. Por el contrario, cuando el sistema está completamente cargado, el mejor balanceo de carga en SEADSR, que es una consecuencia de la ausencia de cachés en dicho protocolo, compensa la sobrecarga de señalización y retardos adicionales introducidos en los sucesivos procesos de Descubrimiento de Ruta.

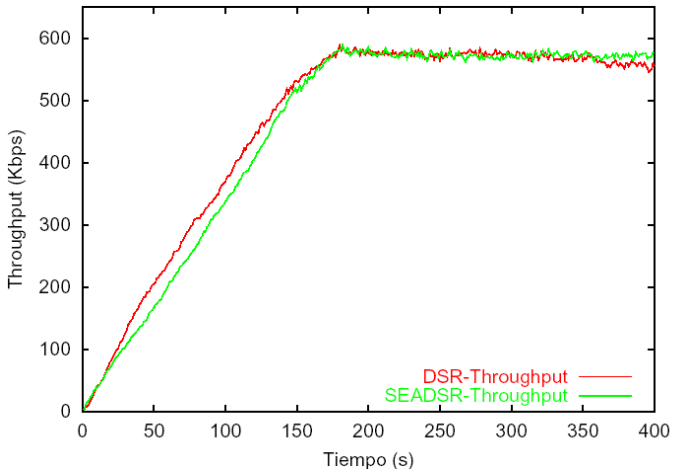



**Fig. 3.31.** Throughput en función del tiempo.

La *Fig. 3.32* representa el histograma de retardo de paquetes de datos utilizando parámetros con los mismos valores. Muestra el número de paquetes de datos en función del retardo que han experimentado en su camino desde la fuente hacia el destino a través de la red. Cada simulación tiene una duración de 400 s y los retardos de los paquetes son medidos desde el segundo 180 (cuando todas las fuentes están activas) hasta el segundo 400. Se han hecho 30 experimentos para DSR y para SEADSR. Cada punto de la abscisa en la *Fig. 3.32* corresponde a un intervalo de retardo de 10 ms. Por ejemplo, el valor de la abscisa 2 indica que aproximadamente 2000 paquetes de datos han sufrido un retardo entre 20 y 30 ms.

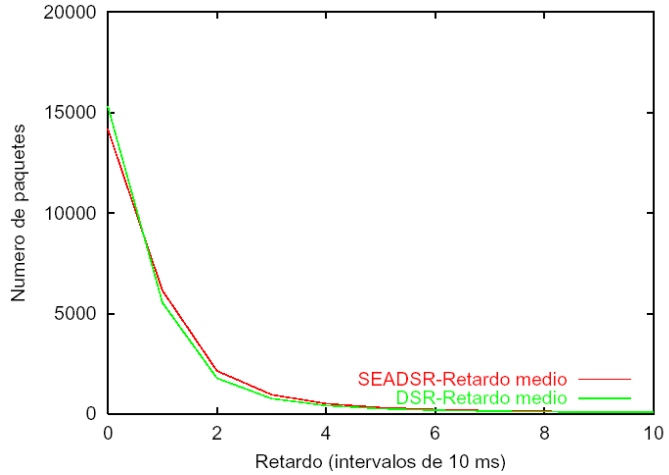



**Fig. 3.32.** Histograma del retardo de paquetes de datos

Los resultados muestran que el número de paquetes de datos con retardos inferiores a 10 ms es sólo un 5% más pequeño en SEADSR que en el estándar DSR. La situación cambia para retardos entre 10 ms y 50 ms. En este caso, encontramos más paquetes de datos cuando se usa SEADSR que con el estándar DSR. La mejora de

SEADSR en este intervalo de tiempo puede ser explicada por el hecho de que aunque SEADSR introduce un retardo adicional en el mecanismo de Descubrimiento de Ruta, esto influirá en un número relativamente pequeño de paquetes de datos en comparación con el número de paquetes favorecido mediante el balanceo de carga efectuado.

Por lo tanto, en las simulaciones de redes ad hoc efectuadas con condiciones exigentes para los protocolos de encaminamiento (número de nodos grande, alta movilidad y número medio de saltos elevado), se muestra que el protocolo propuesto SEADSR supera al estándar DSR en relación a la supervivencia de red sin que se vea reducida la capacidad del sistema.

En simulaciones de sistemas con poblaciones de nodos estables (ningún nodo muere debido a agotamiento de sus reservas de energía), los resultados indican que a pesar del retardo adicional de encaminamiento y del recargo de señalización introducidos por SEADSR, el número de paquetes de datos que llegan puntualmente al destino (retardos inferiores a 30 ms) sigue manteniéndose.

SEADSR alarga pues el tiempo de vida de los terminales inalámbricos en conjunto, lo cual repercute en el throughput total de la red, en el caso de que los recursos energéticos sean escasos.

## *3.6 Conclusiones*

En el Capítulo 3 se han presentado los protocolos de encaminamiento existentes en redes ad hoc aisladas.

Se han realizado dos clasificaciones de dichos protocolos atendiendo a sus diferentes propiedades. En una primera clasificación se ha distinguido entre protocolos de encaminamiento proactivos, reactivos e híbridos. En una segunda clasificación se ha diferenciado entre protocolos de encaminamiento best-effort y con QoS. Además, se han presentado ejemplos concretos de aquellos algoritmos de encaminamiento más destacados pertenecientes a alguna de las clases anteriormente establecidas.

Asimismo, se ha desarrollado una sección (la sección 3.3), que trata del encaminamiento y la disponibilidad de la energía y se ha pasado a explicar con detenimiento la primera contribución de esta tesis doctoral, “Desarrollo del protocolo de encaminamiento SEADSR para la mejora de la supervivencia en redes ad hoc aisladas”, procediéndose a realizar una explicación teórica y a presentar simulaciones detalladas de la misma. Se ha conseguido demostrar mediante simulaciones que SEADSR mejora la supervivencia de la red en comparación con el protocolo de

encaminamiento DSR, lo cual repercute en un aumento del throughput, en el caso de que los recursos energéticos sean escasos. También se ha podido comprobar que en sistemas con poblaciones de nodos estables (ningún nodo muere debido al agotamiento de sus reservas de energía), los resultados indican que a pesar del retardo y señalización adicional introducidas por el protocolo SEADSR, los paquetes consiguen llegar puntualmente a sus destinos.

Por otro lado, se ha realizado un análisis comparativo entre todos los protocolos de encaminamiento existentes con el fin de decidir qué protocolo de encaminamiento para redes ad hoc aisladas se adapta mejor para nuestros propósitos y utilizarlo con posterioridad en el análisis entre redes ad hoc interconectadas con redes IP fijas. DSR y AODV resultan más adecuados como protocolos de encaminamiento para la mejora de la eficiencia en una red ad hoc de acuerdo con las razones expuestas en la sección 3.4. *(Véase la sección 3.4 Comparación entre distintos protocolos de encaminamiento y elección de un protocolo de encaminamiento para redes ad hoc aisladas, pág. 134)*. La elección final de uno de los dos protocolos dependerá de las características y el tipo de tráfico de la red a considerar.

# 4 Modelos de calidad de servicio en redes ad hoc conectadas a redes fijas

Hasta ahora en este trabajo de investigación se ha analizado la literatura existente para intentar proporcionar calidad de servicio en redes ad hoc aisladas como un primer paso para alcanzar uno de los objetivos de la tesis doctoral:

- *Intentar proporcionar calidad de servicio extremo a extremo en la comunicación entre una red ad hoc y una red IP fija.*

En las siguientes secciones se indicará cuál es el estado actual de la investigación con respecto a este tema y se presentará una contribución relacionada para alcanzar el objetivo propuesto.

## 4.1 Estado actual de la investigación

Actualmente, la literatura existente acerca de la provisión de calidad de servicio entre redes ad hoc y redes IP fijas es escasísima. De acuerdo con nuestros conocimientos, solamente se ha presentado una publicación precursora que aborda este tema [152]. En dicho artículo, los autores presentan un hipotético modelo de calidad de servicio de alto nivel, fruto de la interacción entre una red ad hoc y un dominio de acceso. Este modelo plantea cuestiones fundamentales, pero no les da respuesta, sino que las deja abiertas, como cuál será el esquema de calidad de servicio que empleará la red ad hoc (FQMM, SWAN, INSIGNIA, etc.) o cuál será el esquema de calidad de servicio que empleará el dominio de acceso (DiffServ, IntServ). Sin embargo, esta publicación no deja de ser particularmente meritoria por varios motivos:

- *Expone claramente las restricciones de los modelos de calidad de servicio desarrollados para redes ad hoc aisladas: Éstos intentan garantizar la calidad de servicio solamente para el tráfico local (dentro de la red ad hoc), pero no van más allá.*
- *Defiende que como muy probablemente el tráfico de tiempo real viajará desde la red ad hoc hacia Internet, resulta crucial y esencial desarrollar un modelo de interacción para la provisión de calidad de servicio entre ambas redes.*

No obstante, desde nuestro punto de vista el tráfico que va desde la red IP fija hacia la red ad hoc también será importante y considerable.

Una de las contribuciones más significativas de este artículo es que identifica aquellas cuestiones pendientes de resolución con el fin de que lo investigadores centren en ellas su atención:

- *Subraya la necesidad de mapear las clases de servicio del modelo de calidad de servicio incluido en la red ad hoc a las clases de servicio del modelo de calidad de servicio incluido en la red IP fija.*
- *Indica que debe definirse un modelo de calidad de servicio que pueda ser usado por la red ad hoc para cooperar junto al gateway a la hora de mandar tráfico al exterior hacia la red IP fija.*
- *Propone que se establezca una negociación de los parámetros de calidad de servicio para que después dichos parámetros puedan ser garantizados y pueda ofrecerse calidad de servicio extremo a extremo entre la red ad hoc y la red basada en infraestructura.*

En esta tesis doctoral se ha pretendido solucionar todas estas cuestiones pendientes con las contribuciones que van a ser seguidamente introducidas.

## *4.2 Contribución: Desarrollo del modelo de calidad de servicio DS-SWAN para redes ad hoc conectadas a redes fijas*

A continuación se presenta una contribución de esta tesis doctoral consistente en el desarrollo de un modelo de calidad de servicio denominado DS-SWAN para redes ad hoc conectadas a redes fijas.

### *4.2.1 Explicación teórica*

Seguidamente se pasa a realizar una explicación teórica del protocolo DS-SWAN (Differentiated Services-SWAN), que incluye una descripción formal así como un análisis detallado del algoritmo. En la sección *4.2.1.1* se describe cómo funciona este modelo de calidad de servicio cuando el tráfico es enviado desde la red ad hoc hacia la red fija y en la sección *4.2.1.2* se hace una nueva descripción para el caso en que el tráfico fluya desde la red fija hacia la red ad hoc.

### *4.2.1.1 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija*

Se considera un escenario en el cual una red ad hoc está conectada a través de un gateway a una red IP fija (*Véase la*

*Fig.* 4.1). Se establecen una serie de conexiones de tráfico best-effort CBR y de tráfico de tiempo real VBR (Variable Bit Rate) para comunicar los nodos móviles de la red ad hoc con alguno de los hosts localizados en la red fija. Para el tráfico de tiempo real se ha seleccionado concretamente una aplicación de Voz sobre IP, VoIP (Voice over IP) VBR [153].

El objetivo va a ser tratar de mantener la calidad de servicio extremo a extremo para los flujos de tiempo real (VoIP) que se establecen desde la red ad hoc hacia la red IP fija. La red ad hoc utiliza el modelo SWAN [85] *(Véase la sección 2.5)* para diferenciar servicios y la red IP fija usa una arquitectura de QoS denominada DiffServ [67] *(Véase la sección 2.4)* para ofrecer también una diferenciación de servicios escalable en Internet; sin embargo, esto no es suficiente si lo que se pretende es proporcionar calidad de servicio extremo a extremo entre la red ad hoc y la red fija. Resulta imprescindible la cooperación entre ambos modelos para poder mantener una calidad de servicio extremo a extremo que sea efectiva.

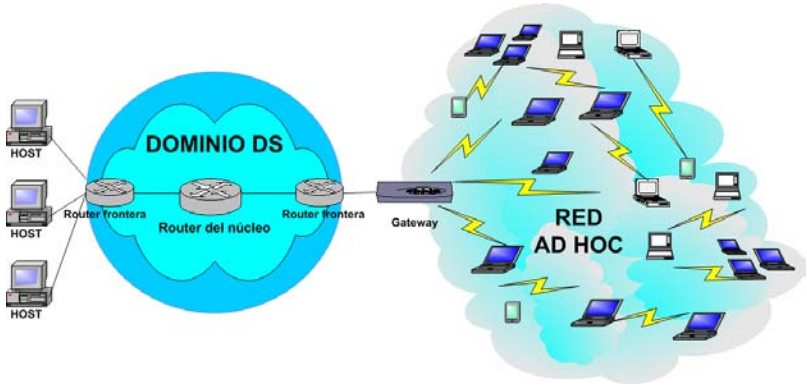



**Fig. 4.1.** Escenario propuesto.

Para el tráfico de tiempo real (VoIP), la clase de servicio DiffServ es la EF (Expedited Forwarding), que proporciona bajas pérdidas, baja latencia, bajo jitter y servicio de ancho de banda asegurado extremo a extremo. Los agregados EF son controlados con la ayuda de un medidor (token bucket) que se halla situado en el router frontera de ingreso. Se toleran algunas ráfagas, pero el tráfico de VoIP que excede el perfil es marcado con un codepoint diferente y entonces es descartado.

En este trabajo de investigación se ha observado que tanto el número de paquetes de tiempo real descartados en el router frontera de ingreso como el retardo extremo a extremo de las conexiones de tiempo real son parámetros de calidad de servicio para

los flujos de VoIP que guardan una estrecha relación con la selección de los parámetros del algoritmo AIMD del modelo SWAN en la red ad hoc. Este algoritmo de control de tasa se utiliza para retardar en cada nodo el acceso al medio únicamente de los paquetes best-effort, aplicándoles un conformador de tráfico denominado leaky bucket de acuerdo con una tasa previamente calculada. Si se reduce la tasa del conformador del tráfico best-effort (leaky bucket), entonces la tasa del tráfico best-effort puede ser controlada más eficientemente y el tráfico de tiempo real no se ve tan perjudicado por el tráfico best-effort (al intentar acceder al medio en la red ad hoc) y es capaz de mantener los parámetros de calidad de servicio requeridos. Por este motivo, resulta imprescindible que el modelo SWAN interactúe con el modelo DiffServ.

En esta tesis doctoral se propone un nuevo protocolo que permite la cooperación e interacción entre la arquitectura DiffServ en la red fija y el esquema SWAN en la red ad hoc para mejorar la provisión de calidad de servicio extremo a extremo.

El modelo de calidad de servicio propuesto, DS-SWAN (Differentiated Services-SWAN) es una mejora del protocolo SWAN y lo sustituye en la red ad hoc.

En el modelo de calidad de servicio propuesto, DS-SWAN, los nodos destino en la red IP fija monitorizan periódicamente los retardos extremo a extremo de los flujos de tiempo real que han sido establecidos. Para lograrlo, se introduce en la cabecera del protocolo de la aplicación de tiempo real (el protocolo RTP (Real-time Transport Protocol)) el 'timestamp' o tiempo de generación en cada paquete de datos y se calcula el retardo extremo a extremo en el destino como una diferencia de tiempos. La ITU-T (International Telecommunication Union) recomienda en su estándar G. 114 que el retardo extremo a extremo de VoIP debe ser inferior a 150 ms para mantener una calidad de conversación aceptable [154]. Por consiguiente, si el retardo extremo a extremo de uno o más flujos de VoIP VBR es mayor de 140 ms, entonces los nodos destino envían un paquete denominado QoS_PERDIDA al router frontera de egreso y éste lo reenvía al router frontera de ingreso (el más próximo al gateway) para prevenirle (*Véase la Fig. 4.2*).

Por otro lado, el router frontera de ingreso (el más próximo al gateway) monitoriza periódicamente el número de paquetes de tráfico EF (de tiempo real VoIP) que son descartados por el medidor (token bucket) porque están fuera del perfil establecido para este tipo de tráfico. Para la codificación PCM (Pulse Code Modulation) con el códec G.711, la tasa de pérdidas de paquetes de VoIP nunca debe ser superior al 5% de todos los paquetes generados para prevenir pérdidas significativas en cuanto a calidad [155]. Se ha observado a partir de las numerosas simulaciones ejecutadas en esta tesis doctoral que el número de paquetes de VoIP descartados en la red ad hoc se mantiene siempre como mucho en torno al 1%. Por lo tanto, se ha establecido que

si el número de paquetes de VoIP descartados por el router frontera es inferior al 4% (el número total de paquetes de VoIP descartados contando los perdidos en la red ad hoc será inferior al 5 %) y si además el router frontera de ingreso ha recibido un mensaje de QoS_PERDIDA, entonces este router frontera debe enviar mensajes de QoS_PERDIDA a los nodos de la red ad hoc para informarles de que el sistema está excesivamente congestionado para mantener el nivel de calidad de servicio deseado (debido a los excesivos retardos que sufren los flujos de VoIP en la red ad hoc). Si el número de paquetes perdidos en el router frontera es superior al 4%, la tasa total de pérdidas de paquetes de VoIP será superior al 5%, de tal forma que la calidad de la VoIP se verá degradada y no tendrá sentido enviar mensajes de QoS_PERDIDA para tratar de disminuir los retardos extremo a extremo de los flujos de VoIP porque la tasa de pérdidas de paquetes no va a disminuir y ya resulta ser excesiva para el buen funcionamiento de este tipo de aplicación.

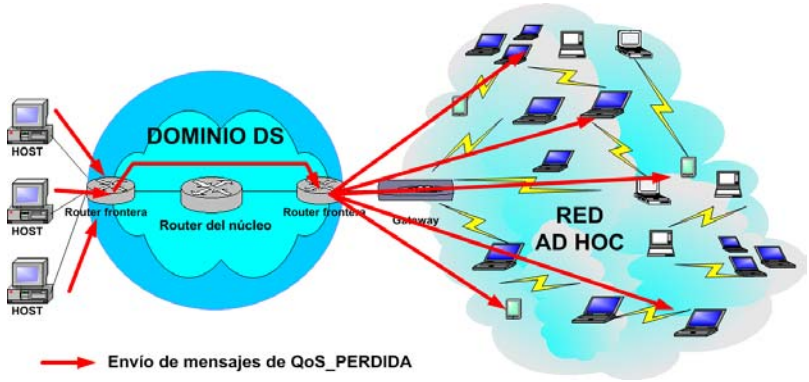



**Fig. 4.2.** Envío de mensajes de QoS_PERDIDA en el escenario propuesto.

Los nodos en la red ad hoc usan una cola para almacenar aquellos paquetes a nivel de la capa MAC que están esperando para poder acceder al medio. La cola utiliza una disciplina de servicio por prioridades (priority scheduling) para priorizar los paquetes de encaminamiento. Los paquetes de QoS_PERDIDA son tratados con la misma prioridad que los paquetes de encaminamiento porque son avisos y deben llegar a sus destinos lo antes posible.

Cuando un nodo móvil en la red ad hoc reciba un mensaje de QoS_PERDIDA deberá actuar en consecuencia estrangulando el tráfico best-effort con el fin de mantener los retardos extremo a extremo de los flujos de VoIP. Para ello, reaccionará modificando los valores de los parámetros del algoritmo de control de tasa AIMD del modelo SWAN *(Véase la sección 2.5 El modelo SWAN, pág. 77)*. Este algoritmo modifica la tasa del conformador de tráfico (leaky bucket) que será aplicada para retardar el tráfico best-effort. Cada nodo monitoriza los retardos a nivel de la capa MAC continuamente y esta información es usada por el controlador de tasa. Cada $T$ segundos, el algoritmo de control de tasa comprueba si uno o más paquetes sufren

retardos a nivel de la capa MAC superiores a un determinado umbral $D_{MAX}$. Si esto es así, el algoritmo reduce la tasa del conformador (y por lo tanto la tasa de transmisión del tráfico best-effort) aplicando una tasa de decremento (decremento multiplicativo del *r* %). En caso contrario, el dispositivo móvil incrementa su tasa de transmisión para el tráfico best-effort gradualmente (incremento aditivo con una tasa de incremento de *c* Kbit/s).

Con DS-SWAN estos parámetros se mantienen; pero además, cada vez que un nodo inalámbrico recibe un mensaje de QoS_PERDIDA, el nodo disminuye el valor de $c$ en $\Delta c^-$ Kbit/s, manteniéndose por encima de un cierto valor mínimo. Cuando no se recibe ningún mensaje de QoS_PERDIDA durante $T$ segundos, el nodo incrementa el valor de $c$ en $\Delta c^+$ bits/s hasta llegar al valor inicial. Así se previene la inanición del tráfico best-effort.

Cuando un nodo inalámbrico recibe un mensaje de QoS_PERDIDA incrementa el valor de $r$ en $\Delta r^+$ hasta un cierto valor máximo. Si no se recibe ningún mensaje de QoS_PERDIDA en el periodo $T$, el valor de $r$ disminuye un $\Delta r^-$ hasta el valor inicial.

SWAN tiene una tasa mínima $m$ para el conformador de tráfico best-effort (leaky bucket). En DS-SWAN, los nodos también pueden reducir $m$. Cuando un nodo recibe un mensaje de QoS_PERDIDA, reduce la tasa mínima en $\Delta m^-$ Kbit/s. No obstante, este valor de parámetro se mantiene por encima de un valor mínimo $m_0$ Kbit/s y es incrementado $\Delta m^+$ bits/s cada segundo hasta el valor inicial cuando los nodos móviles no han recibido un mensaje de aviso en $T$ segundos.

En este sentido, puede afirmarse que con DS-SWAN los parámetros del modelo SWAN cambian dinámicamente en concordancia con las condiciones de tráfico no sólo de la red ad hoc sino también de la red IP fija. El tráfico de tiempo real puede satisfacer de esta forma sus requisitos de ancho de banda y retardo, mientras que el tráfico best-effort puede usar el ancho de banda sobrante de forma eficiente.

La *Tabla 4.1* muestra los valores específicos de los parámetros que han sido seleccionados para realizar las simulaciones. No obstante, los operadores y usuarios pueden asignar valores a estos parámetros libremente de acuerdo con sus propias necesidades, basándose en las características de la red escogida.

| **Parámetros** | **Valor inicial de *c*** | **Δ*c*-** | **Δ*c*+** | **Valor mínimo de *c*** | **Valor inicial de *r*** | **Δ*r*+** | **Δ*r*-** | **Valor máximo de *r*** | **Tasa mínima inicial** | **Δ*m*-** | **Δ*m*+** | **$m_0$** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Valores en las simulaciones** | 41 Kbit/s | 10 Kbit/s | 50 bits/s | 11 Kbit/s | 50 % | 10% | 1% | 90% | 31 Kbit/s | 10 Kbit/s | 50 bits/s | 11 Kbit/s |

**Tabla 4.1.** Valores de parámetros para las simulaciones.

Hasta ahora se ha explicado que los nodos de la red ad hoc deben ser avisados cuando existen fuentes de VoIP que no pueden mantener sus retardos extremo a extremo por debajo de los 150 ms debido en muchos casos a un exceso de congestión en la red ad hoc. Surgirán diferentes diseños del modelo DS-SWAN dependiendo de a qué nodos de la red ad hoc se les envía un mensaje de QoS_PERDIDA para alertarles de la situación con el fin de que puedan actuar en consecuencia.

Se ha usado un diseño 'cross layer' para desarrollar este modelo de calidad de servicio. *(Véase la definición de diseño 'cross layer' en la sección 2 Modelos de calidad de servicio en redes ad hoc aisladas, pág. 15).* La capa LLC (Logical Link Control) consulta la cabecera de los paquetes provenientes de la capa de red en cada nodo de la red ad hoc para decidir si deben ser conformados por el leaky bucket o deben pasar a la capa MAC para poder ser directamente enviados al medio. Los retardos extremo a extremo de los paquetes pueden medirse como una diferencia de tiempos gracias al campo 'timestamp' de la cabecera del protocolo a nivel de la capa de aplicación RTP y esta información servirá para realizar mediciones y poder estrangular en mayor grado si es necesario el tráfico best-effort a nivel de la capa SWAN justo antes de enviarlo a la capa MAC. Además, las pérdidas de paquetes de VoIP en la red ad hoc se miden a nivel de las capas MAC y de red. Por consiguiente, puede observarse que todas las capas están interaccionando entre sí con objeto de hacer posible la diferenciación de servicios.

Se han diseñado dos implementaciones de DS-SWAN diferentes en esta tesis doctoral:

- *Cuando las fuentes best-effort CBR y los nodos intermedios a lo largo de las rutas en la red ad hoc son avisados de forma que estrangulan su tráfico best-effort. Caso 2: ("DS-SWAN - fuentes CBR").*
- *Cuando el router frontera envía un mensaje de QoS_PERDIDA sólo a las fuentes de VoIP que generan flujos con problemas para mantener sus retardos extremo a extremo por debajo de los 150 ms y a los nodos intermedios a lo largo de las rutas en la red ad hoc. Entonces estos nodos reenvían el mensaje de QoS_PERDIDA como un paquete broadcast a todos sus vecinos porque pueden estar compitiendo con ellos por el acceso al medio. Solamente cuando un nodo recibe un mensaje de QoS_PERDIDA como un paquete broadcast, estrangula su tráfico best-effort. Caso 3: ("DS-SWAN – fuentes de VoIP + vecinos").*

En la *Fig. 4.3* se puede observar el distinto comportamiento de las dos versiones del modelo DS-SWAN. Se muestra un ejemplo de una red ad hoc donde se han establecido un flujo de tiempo real de VoIP y tres flujos best-effort CBR, de forma que se envían paquetes a Internet a través del gateway. Primero se aplica la versión ("DS-SWAN – fuentes CBR"). Si se considera que el flujo de VoIP tiene problemas para mantener sus retardos extremo a extremo por debajo de los 150 ms, se enviarán mensajes de QoS_PERDIDA a las fuentes CBR y los nodos intermedios a lo largo de las rutas en la red ad hoc de tal manera que los nodos A, B, C, D, E, F, G, H, I y J estrangularán su tráfico best-effort. Aunque los nodos H, I y J no compiten por el acceso al medio con los nodos a lo largo de la ruta hacia la fuente de VoIP problemática, igualmente disminuyen su tasa de tráfico CBR porque han sido avisados en esta versión del modelo DS-SWAN.

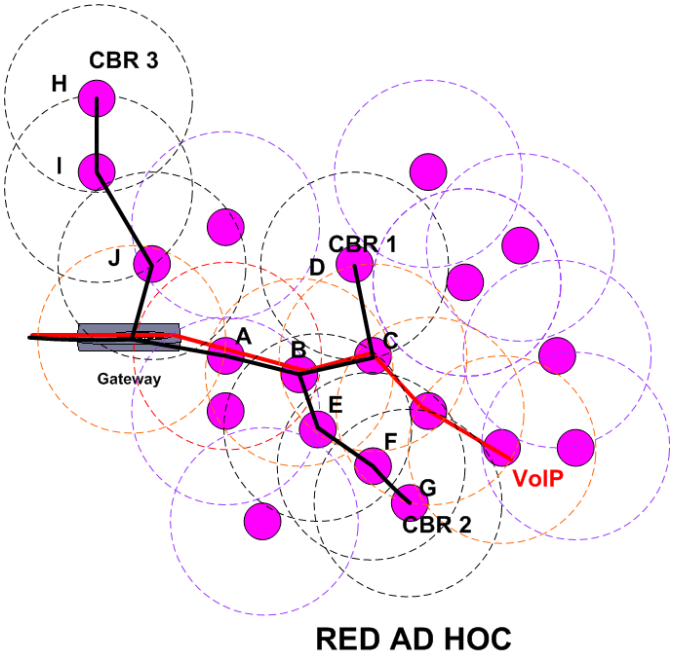



**Fig. 4.3.** Ejemplo de red.

Por otro lado, si se aplica la segunda versión ("DS-SWAN – fuentes de VoIP + vecinos"), entonces las fuentes CBR y los nodos intermedios que no son vecinos de la fuente de VoIP problemática y sus nodos intermedios a lo largo de la ruta no son avisados, de forma que en el ejemplo los nodos H, I y J no estrangularán su tráfico best-effort. Además, es importante observar, que con esta versión del DS-SWAN algunos nodos como el nodo E en el ejemplo recibirán el paquete de QoS_PERDIDA en modo broadcast más de una vez porque son vecinos de varios nodos (el nodo E tiene a los nodos B y C como vecinos), de forma que actuarán sobre los parámetros del leaky bucket para controlar la tasa del tráfico best-effort varias veces. Pensamos que está justificado estrangular más de una vez el tráfico best-effort de un nodo que

está compitiendo por el acceso al medio y dañando varios nodos con paquetes de tiempo real como la fuente de VoIP o los nodos a lo largo de la ruta. Los nodos a lo largo de la ruta hacia la fuente de VoIP problemática que tienen problemas para mantener sus retardos extremo a extremo estrangularán también más de una vez sus flujos CBR con esta versión del DS-SWAN y por este motivo este modo de funcionamiento beneficiará a los paquetes de VoIP significativamente porque comparten la misma cola en la capa MAC con el tráfico de fondo (background traffic) CBR.

En la sección *4.2.1.1.1* se presentan distintas versiones del modelo DS-SWAN cuando el tráfico viaja desde la red ad hoc hacia la red fija.

#### *4.2.1.1.1 Diversas versiones del modelo DS-SWAN para tráfico enviado desde la red ad hoc hacia la red fija*

Con el fin de estudiar en profundidad el protocolo DS-SWAN y conseguir cada vez mejores resultados se han introducido modificaciones en su funcionamiento, analizándose los resultados obtenidos en busca siempre de aquella solución que proporcione una mejor diferenciación de servicios y un mayor aprovechamiento de los recursos de la red.

En este trabajo de investigación se diseñaron dos implementaciones distintas del modelo DS-SWAN, que también fueron publicadas en [156]. Se realizaron simulaciones en una red ad hoc conectada a través de un gateway a una red IP fija que soporta la arquitectura DiffServ. Se compararon los resultados de las simulaciones en tres casos: Cuando se introducía únicamente el modelo SWAN en la red ad hoc o bien cuando se introducía una de las versiones del modelo DS-SWAN comentado en la sección 4.2.1.1 *(Véase la sección 4.2.1.1 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija, pág. 151)* para facilitar el mantenimiento de calidad de servicio extremo a extremo entre la red ad hoc y la red IP fija. Estas simulaciones realizadas ya demostraban claramente que el modelo DS-SWAN superaba al modelo SWAN en el escenario propuesto: Los retardos extremo a extremo de las fuentes de VoIP mejoraban considerablemente, mientras que el tráfico de fondo (background traffic) best-effort no sufría inanición. No obstante, las versiones del modelo DS-SWAN allí presentadas han sido superadas en rendimiento por la versión del modelo DS-SWAN (“DS-SWAN- fuentes de VoIP + vecinos”).

Por otro lado, en este trabajo de investigación se han ejecutado 40 simulaciones para evaluar y comparar la versión del modelo DS-SWAN ("DS-SWAN- fuentes de VoIP + vecinos") explicada en la sección *4.2.1.1* (pasa a denominarse *Caso 1: "DS-SWAN – fuentes de VoIP + vecinos")* con el *Caso 2: "DS-SWAN - (fuentes de VoIP + vecinos) con num_secuencia"*. En ambos casos el router frontera de ingreso envía un mensaje de QoS_PERDIDA solamente a las fuentes de VoIP generadoras de flujos con problemas para mantener sus retardos extremo a extremo por debajo de 150 ms y a los nodos intermedios a lo largo de las rutas. Entonces, estos nodos móviles envían el mensaje de QoS_PERDIDA como paquete en modo broadcast a todos sus vecinos porque pueden estar compitiendo con ellos por el acceso al medio. Sólo cuando un nodo móvil recibe un mensaje de QoS_PERDIDA como un paquete en modo broadcast, estrangula su tráfico best-effort. La diferencia entre ambos casos es que en el Caso 2 un mensaje de QoS_PERDIDA lleva el número de secuencia de forma que cuando un nodo recibe el mismo mensaje de QoS_PERDIDA en modo broadcast más de una vez, el nodo estrangula su tráfico best-effort solamente una vez, mientras que en el Caso 1 el nodo estrangula su tráfico best-effort cada vez que recibe un mensaje de QoS_PERDIDA (incluso si el mismo paquete es enviado en modo broadcast varias veces).

En [167] se muestran los resultados obtenidos al realizar las simulaciones. El Caso 1 muestra mejores resultados en comparación con el Caso 2 en términos de retardo medio extremo a extremo y pérdidas de paquetes para el tráfico de VoIP. Por lo tanto, se demuestra que con la versión "DS-SWAN – fuentes de VoIP + vecinos" se obtiene un mejor rendimiento y pensamos que queda demostrado y además está justificado estrangular más de una vez el tráfico best-effort de un nodo móvil que está compitiendo por el acceso al medio y perjudicando varios nodos con paquetes de tiempo real como una fuente de VoIP o sus nodos intermedios a lo largo de la ruta. No obstante, en ambas implementaciones del modelo DS-SWAN, los resultados han sido positivos y ha sido posible conseguir que los flujos de VoIP hayan sido transmitidos correctamente.

Por otro lado, en este trabajo de investigación se han ejecutado 40 simulaciones para evaluar y comparar la versión del modelo DS-SWAN que resulta ser la mejor y pasa a denominarse *Caso 1: "DS-SWAN – fuentes de VoIP + vecinos",* con dos implementaciones diferentes de DS-SWAN: *Caso 2: "DS-SWAN – fuentes de VoIP + vecinos – mod 1"* y *Caso 3: "DS-SWAN – fuentes de VoIP + vecinos – mod 2"*.

En todos los casos el router frontera de ingreso envía un mensaje de QoS_PERDIDA solamente a las fuentes de VoIP generadoras de flujos con problemas para mantener sus retardos extremo a extremo por debajo de 150 ms y a los nodos intermedios a lo

largo de las rutas. Entonces, estos nodos reenvían el mensaje de QoS_PERDIDA como un paquete en modo broadcast a todos sus vecinos porque pueden estar compitiendo con ellos por el acceso al medio. La diferencia entre todos los casos es la siguiente:

- *En el Caso 1 los nodos que reciben un mensaje de QoS_PERDIDA en modo broadcast deben estrangular su tráfico best-effort.*
- *En el Caso 2 los nodos que reciben un mensaje de QoS_PERDIDA como paquete broadcast deben estrangular su tráfico best-effort solamente en el caso de que dichos nodos estén congestionados. Se considera que un nodo padece problemas de congestión si sus retardos a nivel de la capa MAC durante el ciclo RTS-CTS-DATOS-ACK exceden un valor predefinido denominado $D_{MAX}$. En las simulaciones realizadas $D_{MAX}$ pasa a valer 20 ms. Cada nodo monitoriza y calcula independientemente sus retardos a nivel de la capa MAC. El retardo de los paquetes a nivel de la capa MAC para el modo de operación función de coordinación distribuida, DCF (Distributed Coordination Function) del IEEE 802.11 puede ser estimado como el tiempo total de espera antes de poder enviar un paquete, al que añadimos el tiempo transcurrido hasta que se recibe un ACK o reconocimiento positivo conforme el paquete de datos ha sido recibido correctamente si no ha habido colisión. Por ejemplo, si está activada la opción de envío RTS/CTS para reducir el problema del terminal escondido, el retardo $d$ de un paquete a nivel de la capa MAC se calcula como* [87][88]*:*

$$d = t_{espera} + t_{RTS} + t_{CTS} + t_{paquete} + t_{ACK} + 3t_{SIFS} + 3\tau, \quad (4.1)$$

*donde $\tau$ es el retardo máximo de propagación. $t_{espera}$ hace referencia al periodo de tiempo que el paquete ha estado esperando desde su llegada hasta que finalmente el paquete RTS ha podido ser transmitido (incluyéndose tanto el tiempo de backoff como posibles resoluciones de colisión).*

*Para evaluar el grado de congestión de un nodo, aplicamos la siguiente ecuación:*

$$T_{MAC} = a * t_{últimos\_retardos} + (1-a) * t_{retardos\_medios\_MAC}, \quad (4.2)$$

*donde $t_{últimos\_retardos}$ se refiere a los retardos medios medidos a nivel de la capa MAC para los últimos $N$ paquetes, $t_{retardos\_medios\_MAC}$ se refiere a los retardos*

*medios a nivel de la capa MAC desde el inicio de la simulación cuando el nodo empieza a enviar tráfico y $a$ es un parámetro de ajuste en el rango [0,1]. Este estimador añade una fracción de los últimos retardos medios a nivel de la capa MAC manteniendo una fracción $(1-a)$ de la historia pasada para eliminar fluctuaciones aleatorias. Cuanto más cercano sea el valor del parámetro $a$ a 0, mayor será el peso asignado a la historia pasada. Si los retardos medios a nivel de la capa MAC de los últimos $N$ paquetes varían rápidamente, el escoger un valor de $a$ pequeño permite al estimador ignorar la mayoría de las fluctuaciones aleatorias. En cambio, si los retardos medios a nivel de la capa MAC de los últimos $N$ paquetes varían lentamente, escoger un valor de $a$ grande permite al estimador 'seguirle la pista' a la entrada rápidamente. Por supuesto la selección del valor de $a$ es crítica. En las simulaciones $a$ toma el valor de 0,8 y $N$ toma el valor de 4. Si $T_{MAC} > D_{MAX}$ considero que los retardos son excesivos de forma que el nodo está congestionado.*

- *En el Caso 3 los nodos que reciben un mensaje de QoS_PERDIDA como paquete broadcast deben estrangular su tráfico best-effort solamente en el caso de que estos nodos se encuentren congestionados. Para evaluar el grado de congestión de un nodo, se establece que un nodo estará congestionado si los retardos medios $d$ de sus paquetes a nivel de la capa MAC son violados más de $N_{MAX}$ veces: (es decir, si los retardos medios de sus paquetes a nivel de la capa MAC exceden $D_{MAX}$ más de $N_{MAX}$ veces consecutivas). En las simulaciones $D_{MAX}$ toma el valor de 20 ms y $N_{MAX}$ vale 4.*

Es importante destacar que en las simulaciones realizadas se han escogido valores de parámetros concretos. Sin embargo, los operadores y usuarios pueden establecer libremente estos valores de acuerdo con sus propias necesidades, basándose en las características de la red.

En [168] se muestran los resultados obtenidos al realizar las simulaciones. El Caso 1 "DS-SWAN – fuentes de VoIP + vecinos" muestra los mejores resultados en comparación con las otras versiones en términos de retardo extremo a extremo y pérdidas de paquetes para el tráfico de VoIP. El throughput del tráfico CBR es muy similar en los tres casos. La razón es que si se decide no estrangular el tráfico best-effort de los nodos que están introduciendo congestión e incrementando el retardo extremo a extremo de las fuentes de VoIP problemáticas, entonces la reducción del

tráfico CBR no será suficiente y será necesario enviar más mensajes de QoS_PERDIDA para mantener los retardos extremo a extremo por debajo de los 150 ms. Además, debido a la movilidad, los paquetes CBR que en el pasado no han sido estrangulados porque no eran nodos congestionados, pueden perjudicar el buen funcionamiento de las aplicaciones de VoIP al cambiar las rutas de los flujos de VoIP, de forma que al final también resulta imprescindible una alta reducción de los paquetes CBR. No obstante, en todos los casos se obtiene unos resultados positivos y los flujos de VoIP son transmitidos correctamente.

### *4.2.1.2 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red fija hacia la red ad hoc*

Con el fin de generalizar los resultados obtenidos, se ha decido estudiar cómo se efectuaría la transmisión de tráfico en sentido contrario al analizado hasta ahora, es decir, desde la red fija hacia la red ad hoc.

El funcionamiento teórico del modelo DS-SWAN, tal y como ha sido explicado en la sección *4.2.1.1 (Véase la sección 4.2.1.1 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija, pág. 151),* continúa siendo el mismo y sirve por tanto como marco de referencia. Lo único que cambia es el papel que desempeñan ciertos elementos de la red. Se destacan las siguientes diferencias *(Véase la Fig. 4.4):*

- *Los tráficos best-effort y de tiempo real viajan desde la red fija hacia la red ad hoc.*
- *El router frontera de ingreso es ahora el más próximo a los hosts.*
- *Los nodos de la red ad hoc son los encargados de medir ahora los retardos extremo a extremo de los paquetes.*

Si dichos retardos son excesivos, los nodos de la red ad hoc enviarán un mensaje de QoS_PERDIDA al router frontera de egreso (el más próximo al gateway), el cual lo reenviará al router frontera de ingreso (el más próximo a los hosts) y éste comprobará si las pérdidas del tráfico de VoIP son inferiores al 5% *(Véase la Fig. 4.4)*; en caso afirmativo, el router frontera de ingreso informará mediante el envío de mensajes de QoS_PERDIDA a los nodos adecuados en la red ad hoc para que éstos estrangulen su trafico best-effort *(Véase la Fig. 4.5).* Los nodos de la red ad hoc que recibirán un mensaje de QoS_PERDIDA serán los mismos que en la versión “DS-SWAN – fuentes de VoIP + vecinos”, explicada en la sección *4.2.1.1*

*DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija, pág. 151).*

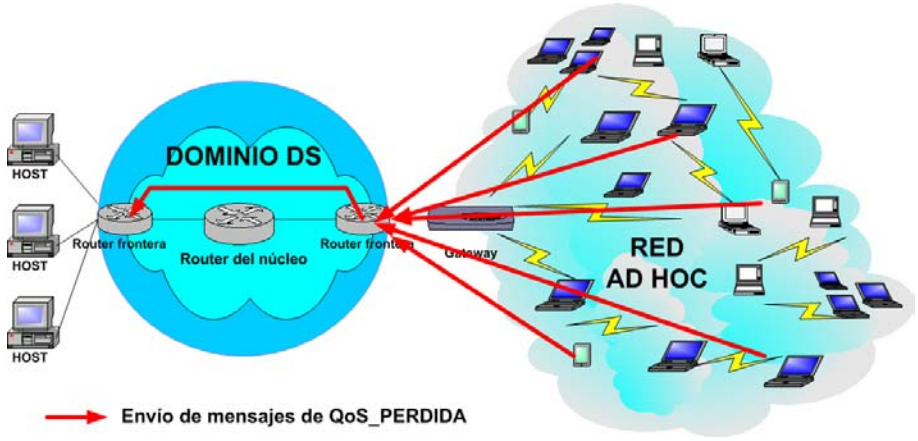



**Fig. 4.4.** Envío de mensajes de QoS_PERDIDA en el escenario propuesto.

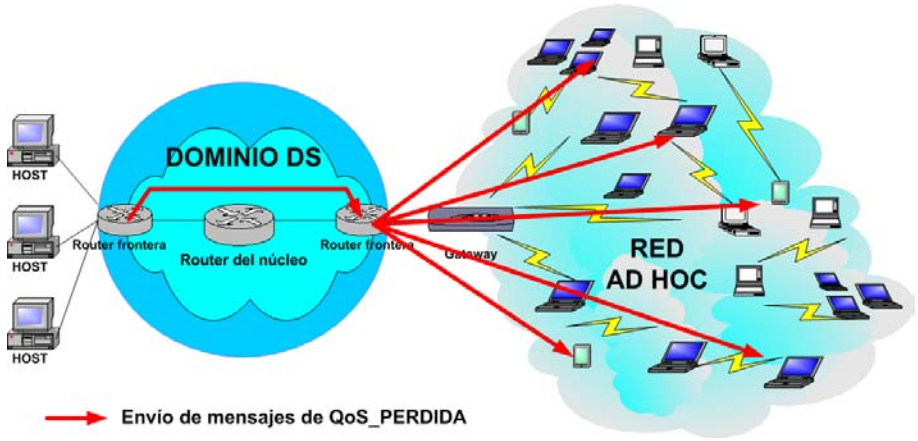



**Fig. 4.5.** Continuación del envío de mensajes de QoS_PERDIDA en el escenario propuesto.

El estudio del envío de tráfico en sentido inverso despierta mucho interés porque en este caso el gateway ha de competir con cualquier nodo de la red ad hoc como uno más por el acceso al medio y corre el peligro de convertirse en un 'cuello de botella' y estar siempre congestionado. El protocolo CSMA/CA debe de garantizar un acceso equitativo para todos los nodos de la red ad hoc sin distinciones, motivo por el cual este caso se convierte en uno muy interesante pero sin duda también más problemático.

Efectivamente, para que el modelo de calidad de servicio desarrollado en esta tesis doctoral pueda trabajar correctamente cuando el tráfico circula en el sentido explicado, ha sido preciso introducir algunos cambios.

Con el fin de que el envío de paquetes de QoS_PERDIDA no hiciera aumentar los retardos extremo a extremo del tráfico de VoIP (ahora los mensajes de QoS_PERDIDA se envían en ambos sentidos y continúan siendo prioritarios) se ha decidido que los mensajes de QoS_PERDIDA deben enviarse más espaciadamente

en el tiempo (cada 10 s y no cada 1 s como sucedía anteriormente). Además, ha sido preciso variar los parámetros del protocolo DS-SWAN con el fin de estrangular con cada mensaje de QoS_PERDIDA más efectivamente el tráfico best-effort.

La *Tabla 4.2* ilustra los nuevos parámetros. Se han modificado los valores de los parámetros $\Delta c^-$, $\Delta m^-$ y $\Delta r^+$ en comparación con los valores especificados en la *Tabla 4.1.*

| **Parámetros** | **Valor inicial de *c*** | $\Delta c^-$ | $\Delta c^+$ | **Valor mínimo de *c*** | **Valor inicial de *r*** | $\Delta r^+$ | $\Delta r^-$ | **Valor máximo de *r*** | **Tasa mínima inicial** | $\Delta m^-$ | $\Delta m^+$ | $m_0$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Valores en nuestras simulaciones** | 41 Kbit/s | 15 Kbit/s | 50 bits/s | 11 Kbit/s | 50 % | 20% | 1% | 90% | 31 Kbit/s | 20 Kbit/s | 50 bits/s | 11 Kbit/s |

**Tabla 4.2.** Valores de parámetros para las simulaciones.

## *4.2.2 Simulaciones*

En las secciones siguientes se presentarán los resultados obtenidos a la hora de simular las versiones más destacadas del protocolo DS-SWAN y compararlas con el modelo de calidad de servicio SWAN.

En la sección *4.2.1.1* se presenta el escenario de simulación básico. Este escenario es el utilizado en las simulaciones que se introducen en la sección *4.2.2.1.1.* En el resto de simulaciones efectuadas posteriormente este escenario sufre una serie de variaciones que son comentadas con detenimiento en cada sección correspondiente.

En la sección *4.2.2.1.1* se presentan las simulaciones efectuadas para tráfico enviado desde la red ad hoc hacia la red fija. En la sección *4.2.2.1.2* se presentan las simulaciones realizadas para analizar la escalabilidad cuando el tráfico es enviado desde la red ad hoc hacia la red fija. En la sección *4.2.2.1.3* se presentan las simulaciones efectuadas con tráfico TCP best-effort para tráfico enviado desde la red ad hoc hacia la red fija. En la sección *4.2.2.1.4* se muestran las simulaciones realizadas para tráfico desde la red fija hacia la red ad hoc. Finalmente, en la sección *4.3* se presentan las conclusiones.

## *4.2.2.1 Escenario de simulación básico*

En esta tesis doctoral se han realizado simulaciones utilizando NS-2 [151] con el fin de investigar el rendimiento del modelo DS-SWAN usando un modelo de capa física lo más realista posible.

El entorno de simulación se muestra en la *Fig. 4.6*. Se considera la presencia de un único dominio DiffServ (dominio DS) que cubre toda la red entre los hosts correspondientes y dos gateways inalámbricos. El escenario escogido está formado por 20 nodos móviles, 2 gateways, 3 routers fijos y 3 hosts. El borrador de Internet (Internet draft) "Global Connectivity for IPv6 Mobile Ad Hoc Networks" [157] describe cómo proporcionar acceso a Internet para las redes ad hoc modificando el protocolo de encaminamiento Ad Hoc On-Demand Distance Vector (AODV) [112] para poder descubrir gateways. Se ha seleccionado el protocolo AODV y no el DSR porque AODV es junto a DSR un protocolo de encaminamiento que muestra los mejores resultados en rendimiento de red; además este protocolo está libre de bucles, y si se rompe una ruta la incidencia es notificada inmediatamente; por otro lado, existen estudios que muestran un mejor comportamiento con respecto a los retardos extremo a extremo de los paquetes de datos CBR usando AODV en comparación con DSR u otros protocolos de encaminamiento [142] y esta propiedad va a resultar de especial importancia para las aplicaciones de tiempo real, que requieren retardos extremo a extremo acotados. En este escenario se considera que se ha utilizado un método de descubrimiento de gateway híbrido [158] para encontrar un gateway, de tal forma que se selecciona la ruta con el menor número de saltos hacia el gateway usando AODV.

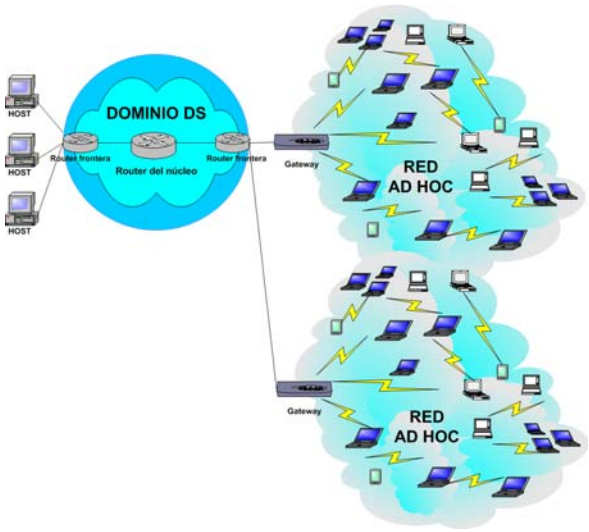



**Fig. 4.6.** Entorno de simulación.

Los nodos móviles se distribuyen uniformemente en una región rectangular de 700 m por 500 m. Los gateways están colocados en las coordenadas (100, 250) y (600, 250). Cada nodo móvil selecciona un destino aleatorio dentro del área y se mueve hacia él a una velocidad uniformemente distribuida entre 0 y 3 m/s. Una vez ha sido alcanzado el destino, el nodo hace una pausa de 20 s, selecciona otro destino y repite el proceso. Los enlaces inalámbricos están basados en la tecnología IEEE 802.11b.

Se asume que se transmite tráfico best-effort CBR y tráfico de tiempo real de VoIP VBR. En este trabajo de investigación se ha propuesto CBR como tráfico de fondo (background) en vez de TCP. La razón es que TCP se comporta deficientemente en una red ad hoc porque cuando hay pérdidas de paquetes debido a caídas de enlaces y cambios de rutas, los mecanismos de prevención de congestión del TCP se disparan [159]. Por el contrario, muchos autores [155], [160], [161] han utilizado CBR como el tráfico background exitosamente. No obstante, en la sección *4.2.2.1.3 (Véase la sección 4.2.2.1.3 Análisis de las simulaciones con tráfico best-effort TCP para tráfico enviado desde la red ad hoc hacia la red fija, pág. 180)* se demuestra mediante simulaciones que el protocolo DS-SWAN es igualmente capaz de obtener un buen rendimiento utilizando TCP como tráfico best-effort.

El tráfico de fondo CBR es generado por 13 de los hosts móviles, mientras que el tráfico de VoIP VBR es generado por 15 de los hosts móviles. Los destinos de los flujos best-effort y de VoIP se escogen aleatoriamente entre los tres hosts de la red fija.

El modo VBR [162] [163] es usado para el tráfico de VoIP. Se emplea una técnica de supresión de silencio en los códecs de voz de manera que no se generan paquetes durante el periodo de silencio. Para las llamadas de voz, se usa el códec ITU G.711 (ITU G.711 a-Law codec) [153]. El tráfico de VoIP se modela como una fuente on/off con periodos on y off de 1,004 s y 1,587 s respectivamente en media, cada uno distribuidos exponencialmente y dos tramas (cada trama contiene muestras de audio de 10 ms) son transportadas en cada paquete (80 + 80 bytes de carga). Las tramas son generadas durante los periodos de on cada 10 ms con un tamaño de 80 bytes y sin compresión alguna. La VoIP se establece sobre el protocolo RTP (Real-time Transport Protocol), utilizándose UDP/IP entre los protocolos RTP y de la capa de enlace de datos. La *Tabla 4.3* muestra los valores de los parámetros de tráfico usados en las simulaciones.

| ***Códec de voz*** | ***G711*** |
|---|---|
| ***Tiempo entre llegadas de paquetes*** | **10ms** |
| ***Tamaño del paquete de voz*** | **80 bytes** |
| ***Cabecera capa RTP*** | **12 bytes** |
| ***Cabecera capa UDP*** | **8 bytes** |
| ***Cabecera capa IP*** | **20 bytes** |
| ***Cabecera capa MAC*** | **34 bytes** |
| ***Cabecera capa física*** | **24 bytes** |

**Tabla 4.3.** Parámetros del tráfico de voz.

Los paquetes tienen un tamaño constante y son generados durante el periodo de on a una tasa entre llegadas constante. Las conexiones de VoIP se activan en un tiempo inicial escogido a partir de una distribución uniforme en [10 s, 15 s].

El tráfico de fondo es CBR con una tasa de 48 Kbit/s y un tamaño de paquete de 120 bytes. Para evitar la sincronización, las fuentes CBR inician su envío de paquetes en instantes escogidos aleatoriamente a partir del intervalo [15 s, 20 s] para la primera fuente, [20 s, 25 s] para la segunda fuente y así sucesivamente hasta llegar a [75 s, 80 s] para la última de las 13 fuentes.

En nuestro escenario la red fija usa DiffServ [67], [164] como mecanismo de QoS. Las funciones del router frontera de ingreso se muestran en la *Fig. 4.7*. Los paquetes entrantes son clasificados y marcados con un DSCP. Se usan los valores de DSCP recomendados de '46' para EF [73] y de '0' para BE [67]. Por lo tanto, el tráfico de VoIP se mapea al DSCP '46' asociado al PHB EF (Expedited Forwarding), mientras que el tráfico CBR se mapea al DSCP '0' asociado al PHB BE (Best Effort). La conformación del tráfico de EF (VoIP) y BE (Best-effort) (CBR) en el router frontera se hace en dos colas de descarte de tamaños 30 y 100 respectivamente. Los agregados EF y BE son conformados con un medidor de token bucket con CBS = 1000 bytes y CIR = 200 Kbit/s. CBS (Committed Burst Size) hace referencia al tamaño máximo del token bucket y es medido en bytes. CIR (Committed Information Rate) hace referencia a la tasa a la cual se generan tokens. Se toleran algunas ráfagas, pero el tráfico que excede el perfil es marcado con un codepoint diferente y después es descartado. Los paquetes con codepoint '46' pasan a ser etiquetados con el codepoint '51' si no cumplen y los paquetes con el codepoint '0' pasan a ser etiquetados con el codepoint '50' si no cumplen. Los paquetes aceptados son servidos usando un planificador round robin.

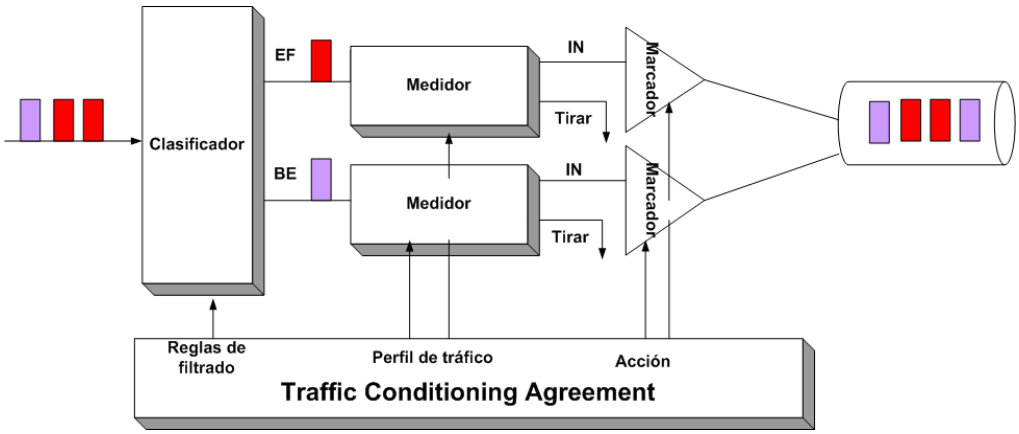



**Fig. 4.7.** Funciones del router frontera.

La arquitectura del router del núcleo está integrada por una cola para cada clase de tráfico. Se usa una disciplina de servicio round robin para servir los paquetes.

#### *4.2.2.1.1 Análisis de las simulaciones para tráfico enviado desde la red ad hoc hacia la red fija*

En este trabajo de investigación se han realizado 40 simulaciones para estudiar entre otros parámetros de calidad de servicio el retardo extremo a extremo así como las pérdidas y el jitter del tráfico de VoIP. También se ha analizado el throughput del tráfico CBR.

El análisis de las simulaciones se efectúa para tráfico enviado desde la red ad hoc hacia la red fija.

Se ha evaluado y comparado el rendimiento del modelo SWAN (Caso 1) con dos implementaciones distintas del modelo DS-SWAN descritas en la sección *4.2.1.1*: Caso 2: "DS-SWAN – fuentes CBR" y Caso 3 "DS-SWAN – fuentes de VoIP + vecinos".

Los resultados han sido publicados en [165], [166].

La *Fig. 4.8* muestra el retardo extremo a extremo para el tráfico de VoIP en todos los casos. Utilizando SWAN los retardos extremo a extremo aumentan progresivamente porque el sistema está congestionado con flujos de VoIP y tráfico de fondo. Desde el segundo 115 hasta el final de la simulación los retardos extremo a extremo son demasiado altos para una calidad de conversación aceptable [154]. En DS-SWAN los retardos extremo a extremo de los flujos de VoIP se reducen porque algunos nodos en la red ad hoc son avisados y actúan estrangulando su tráfico best-effort. Por este motivo, es posible que los flujos de VoIP sean capaces de mantener sus parámetros de QoS, obteniéndose una calidad en la conversación aceptable. En el Caso 3 los retardos medios extremo a extremo son menores que en el Caso 2 porque algunos nodos reciben más de una vez el mismo mensaje de QoS_PERDIDA en modo broadcast y porque cuando las rutas para el tráfico de VoIP cambian debido a la movilidad, los flujos de VoIP continuarán teniendo mayores problemas de retardo que en el Caso 2, pues en el Caso 2 el tráfico best-effort ha sido estrangulado en toda la red. Así, en el Caso 3 se hará necesario controlar más la tasa del tráfico best-effort.

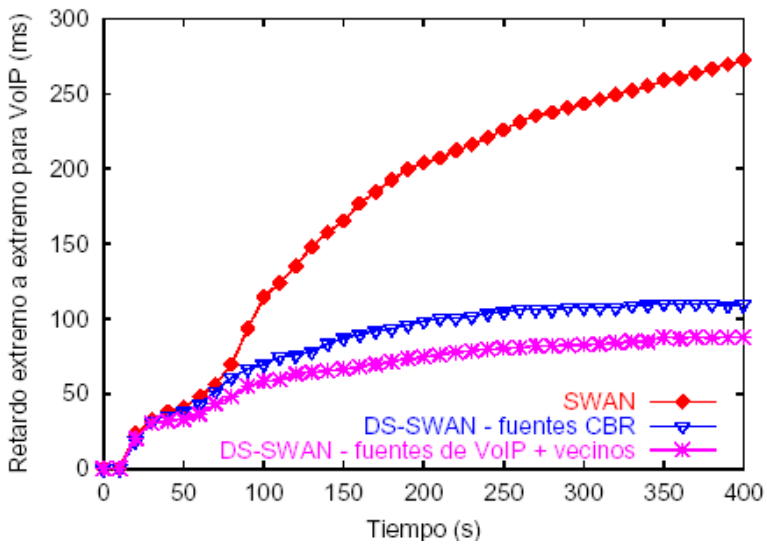



**Fig. 4.8.** Retardo extremo a extremo para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

La *Fig. 4.9* muestra el throughput para el tráfico de fondo. En DS-SWAN, el throughput medio para este tipo de tráfico es menor que en SWAN porque algunos nodos en la red ad hoc reaccionan disminuyendo la tasa del conformador del tráfico best-effort cuando reciben un aviso. En el Caso 2 el throughput medio es primero inferior que en el Caso 3 porque todos los nodos con tráfico best-effort controlan la tasa de sus flujos. Sin embargo, más tarde el throughput medio es mayor en el Caso 2 en comparación con el Caso 3 porque en el Caso 3 algunos nodos reciben el mismo mensaje de QoS_PERDIDA como un paquete broadcast más de una vez y porque en este caso cuando los flujos de VoIP cambian sus rutas debido a la movilidad hay más congestión por la presencia del tráfico CBR y tienen que estrangular más estos flujos. Por otra parte, en ningún caso se produce inanición del tráfico de fondo.

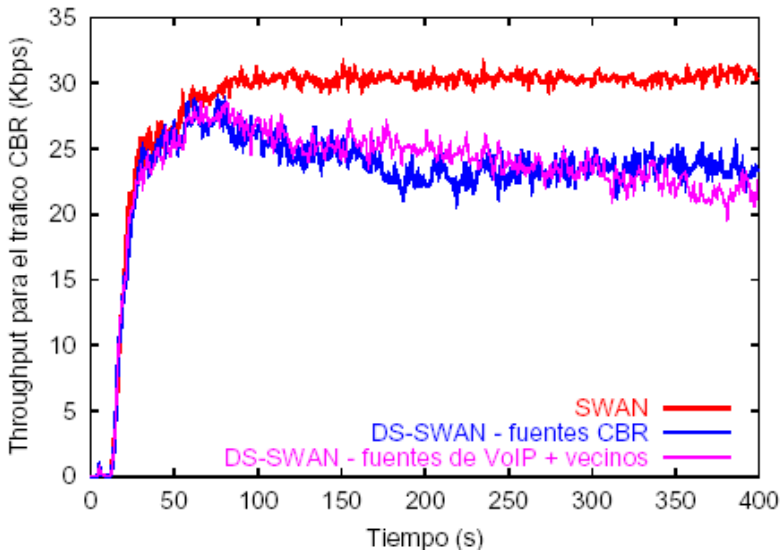



**Fig. 4.9.** Throughput para el tráfico best-effort: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

La *Fig. 4.10* muestra el retardo medio a nivel de la capa MAC para el tráfico de VoIP. Podemos apreciar que en todos los sistemas este retardo aumenta con un número creciente de fuentes CBR. En SWAN el retardo a nivel de la capa MAC aumenta de 2 ms a 15 ms cuando el número de fuentes CBR aumenta de 0 a 13 debido a la congestión y al considerable impacto de la carga de tráfico best-effort sobre las conexiones de VoIP. Por otro lado, en DS-SWAN se envían mensajes de QoS_PERDIDA para avisar a algunos nodos en la red ad hoc acerca de la congestión y reaccionan reduciendo la tasa del tráfico de fondo best-effort; la consecuencia es que el retardo medio a nivel de la capa MAC para el tráfico de VoIP mejora y las fuentes de VoIP sufren retardos solamente de aproximadamente 10 ms en el Caso 2 y 9 ms en el Caso 3 al final de la simulación. El Caso 3 muestra los mejores resultados porque los nodos en las zonas donde las fuentes de tráfico de VoIP tienen problemas para mantener sus retardos extremo a extremo estrangulan en algunos casos más de una vez su tráfico best-effort para el mismo mensaje de QoS_PERDIDA que reciben varias veces en modo broadcast, por tratarse de nodos vecinos de más de un nodo problemático. En consecuencia, la congestión se reduce más en estas zonas y el tráfico CBR no tiene tanto impacto sobre el tráfico de tiempo real.

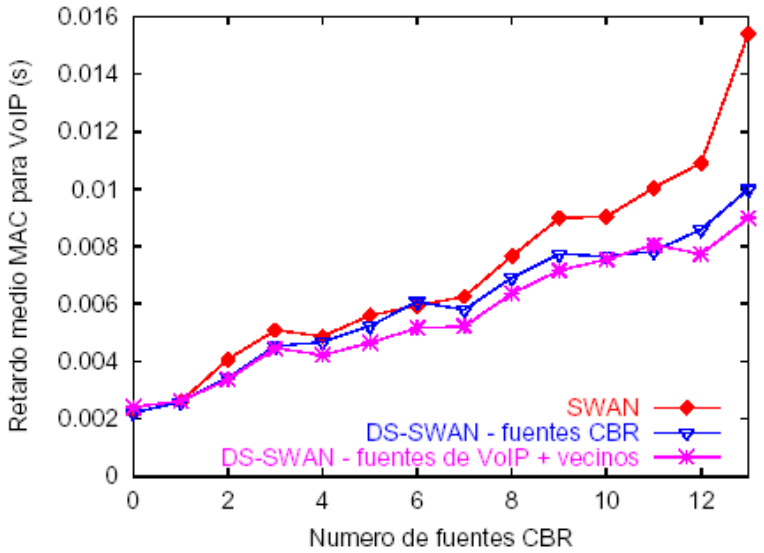



**Fig. 4.10.** Retardo a nivel de la capa MAC para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

El jitter para el tráfico de VoIP se ilustra en la *Fig. 4.11*. Podemos apreciar que el jitter crece en SWAN cuando el número de fuentes CBR y de conexiones de VoIP aumenta porque no se envían mensajes de QoS_PERDIDA para reducir los retardos extremo a extremo del tráfico de VoIP. Los paquetes de VoIP llegan cada vez más tarde a sus destinos debido a la congestión y como resultado el jitter aumenta progresivamente.

Por el contrario, en el modelo DS-SWAN se actúa sobre los parámetros del conformador de tráfico (leaky bucket) para controlar más la tasa del tráfico best-effort y así reducir los retardos extremo a extremo para los paquetes de tiempo real. Este mecanismo se aplica en el Caso 2 a todos los nodos con paquetes CBR en la red ad hoc de forma que se envían ráfagas de VoIP y por este motivo el jitter se reduce con respecto al modelo SWAN. En el Caso 3 el jitter es más pequeño que en el Caso 2 porque en el Caso 3 hay más nodos que reciben el mismo mensaje de QoS_PERDIDA en modo broadcast más de una vez y porque en este caso cuando los flujos de VoIP cambian sus rutas debido a la movilidad hay congestión debido a la presencia de tráfico CBR y tienen que estrangular más estos flujos. Como resultado, la congestión se ve reducida en estas regiones, de forma que los retardos extremo a extremo de los flujos de VoIP no aumentan tanto ni tampoco las variaciones del retardo.

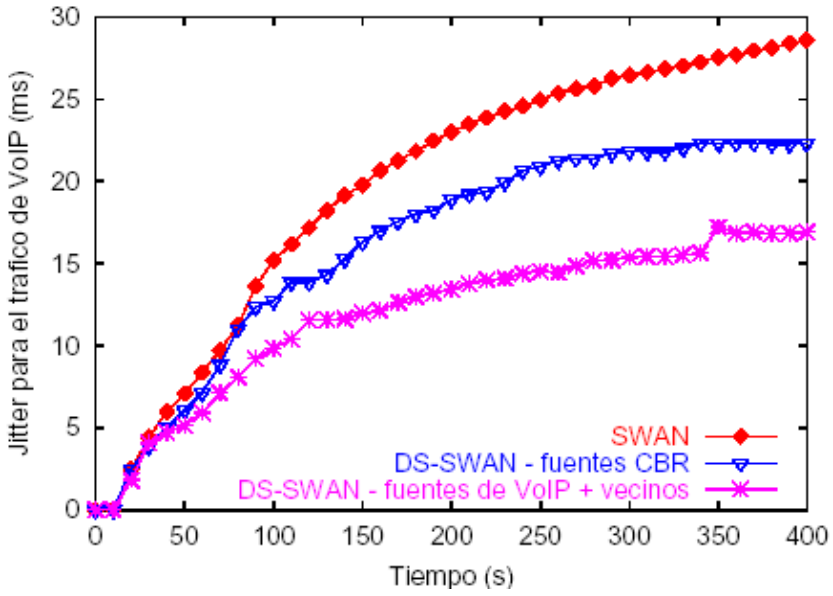



**Fig. 4.11.** Jitter para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

La *Fig. 4.12* muestra la tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP Se ha calculado mirando las pérdidas de paquetes en intervalos de diez segundos para cada una de las quince fuentes de VoIP de una simulación, tomando el valor máximo de pérdidas de entre todas las fuentes para cada intervalo y haciendo la media de los máximos para las cuarenta simulaciones efectuadas.

Hablando en términos generales, las pérdidas de paquetes se producen en muchos casos debido a la movilidad y a la congestión. Cuando un nodo fuente o nodo intermedio debe enviar un paquete, el protocolo de encaminamiento en la capa de red consulta si hay una ruta válida hacia el destino y la utiliza en caso de que exista. Ahora bien, si los nodos se han movido y no existe ninguna ruta disponible en este momento hacia el destino, el paquete es almacenado hasta que se descubra una ruta. Los

paquetes son descartados debido a la movilidad en la capa de red porque el buffer está lleno y no acepta más paquetes o bien el tiempo que un paquete ha estado almacenado ha sobrepasado el límite permitido. Por otro lado, si un paquete a nivel de la capa MAC consigue acceder al canal y la movilidad es alta, el próximo salto puede hallarse fuera de cobertura en ese momento y los paquetes en este caso se pierden a nivel de la capa MAC debido también a la movilidad. Las pérdidas de paquetes por congestión ocurren debido a varias razones. Cuando se excede el máximo tiempo permitido para el intervalo de backoff porque el canal inalámbrico está ocupado, el paquete es descartado a nivel de la capa MAC. Además, si la cola que almacena los paquetes que esperan para acceder al medio está llena debido a la congestión, los paquetes también serán eliminados a nivel de la capa MAC. Dos causas diferentes por las cuales se producen pérdidas de paquetes en la red ad hoc son 'ARP drops' y 'MAC callback'. Si un nodo desconoce la dirección hardware de un destino, envía una petición ARP (Address Resolution Protocol) en modo broadcast y almacena el paquete temporalmente en su caché. Para cada dirección hardware de destino desconocido hay un buffer con espacio para un único paquete. Cuando el nodo recibe otro paquete adicional hacia el mismo destino y todavía no ha recibido la respuesta ARP debido a la congestión, descarta el paquete almacenado anteriormente en el buffer ('ARP drop'). Por otro lado, 'MAC callback' significa que la capa MAC no es capaz de transmitir el paquete y entonces lo que hace es informar a la capa superior acerca del error que se ha producido en la transmisión. La causa de que se produzca un error en la transmisión es la caída de un enlace debido a la movilidad.

Tal y como se observa en la *Fig. 4.12*, las pérdidas de paquetes cuando las fuentes de VoIP empiezan a enviar tráfico son consecuencia tanto en SWAN como en DS-SWAN de la movilidad ('MAC callback') y de la congestión ('ARP drops'). En SWAN se descartan más paquetes de VoIP debido a la sobrecarga del buffer a nivel de la capa MAC por la presencia de muchos paquetes best-effort. Sin embargo, el número de paquetes perdidos es mayor en DS-SWAN que en SWAN, porque con DS-SWAN los paquetes de VoIP no compiten por el acceso al medio con tantos paquetes CBR pero compiten por el acceso al medio con los mensajes de QoS_PERDIDA enviados hacia la red ad hoc (que son encolados de forma prioritaria), produciéndose más colisiones e intervalos de backoff excedidos. El número de paquetes de VoIP perdidos al principio de la simulación es menor en el Caso 3 en comparación con el Caso 2 porque el tráfico best-effort ha sido estrangulado más y en mayor medida en ciertas zonas con fuentes de tráfico de VoIP que tienen problemas para mantener sus retardos extremo a extremo, de manera que se verá reducida la congestión. En cambio, en el Caso 2 se ha estrangulado el tráfico de best-effort en toda la red ad hoc. Sin embargo, más

adelante, el porcentaje de paquetes perdidos es mayor en el Caso 3 que en el Caso 2 porque el tráfico best-effort CBR sólo había sido estrangulado en aquellas zonas necesarias y cuando cambien las rutas de las sesiones de VoIP debido a la movilidad será preciso volver a estrangular aquel tráfico CBR que anteriormente no había sido estrangulado porque no molestaba y en esta ocasión sí que resulta perjudicial. Por este motivo se producen más pérdidas de paquetes de VoIP, que en el Caso 2, donde al cambiar las rutas de las sesiones de VoIP no hay tanto problema porque el tráfico best-effort ya había sido estrangulado en toda la red ad hoc.

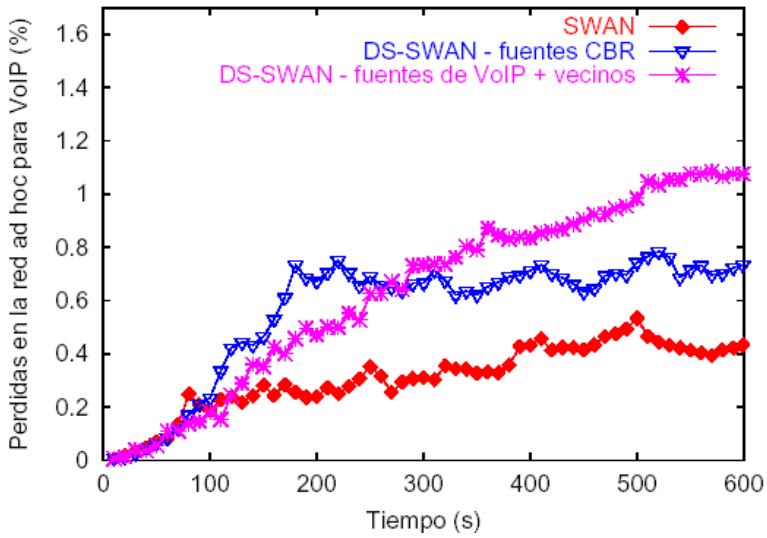



**Fig. 4.12.** Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

Las *Fig. 4.13 y Fig. 4.14* representan las pérdidas de paquetes en el router frontera de ingreso para el tráfico de VoIP y los paquetes de VoIP enviados que llegan a este router frontera respectivamente.

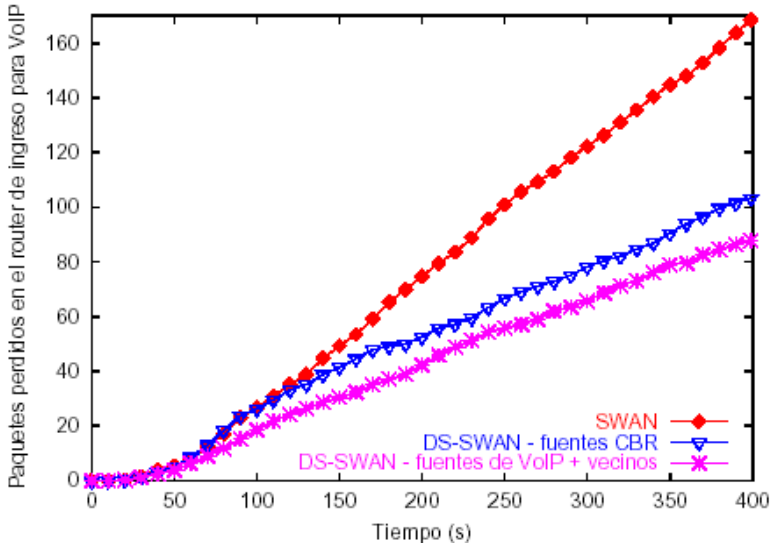



**Fig. 4.13.** Número de paquetes perdidos en el router frontera de ingreso para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

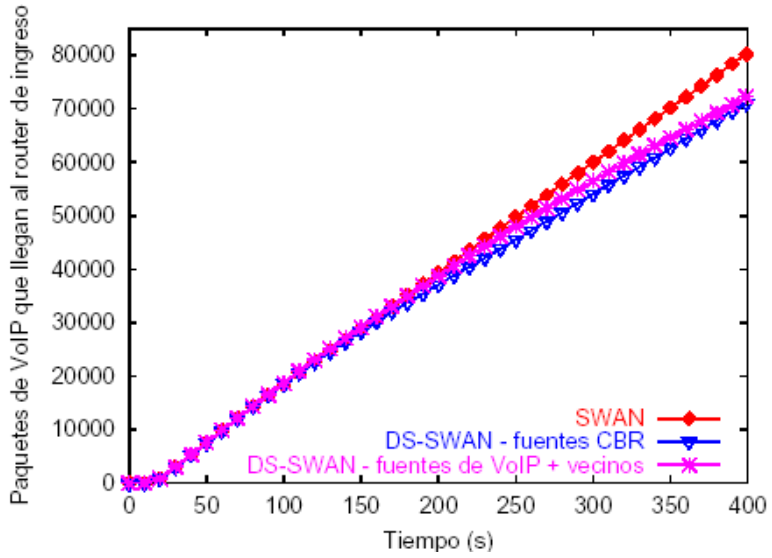



**Fig. 4.14.** Número de paquetes de VoIP enviados que llegan al router frontera de ingreso: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

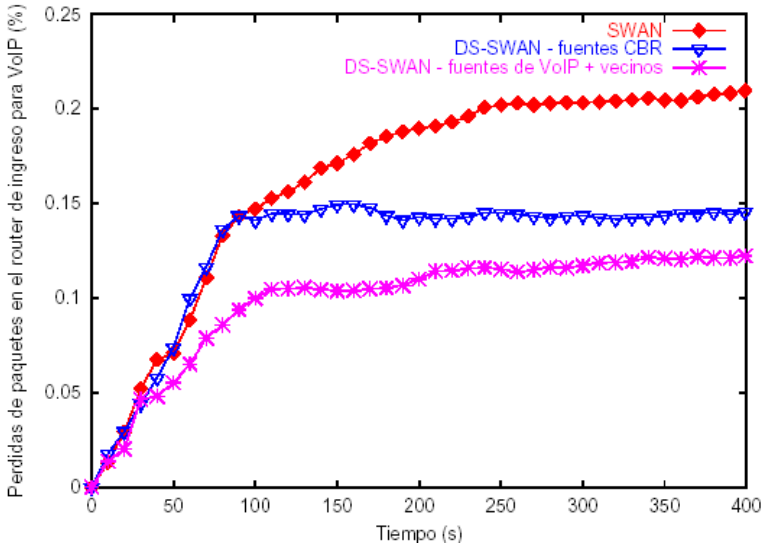



**Fig. 4.15.** Tasa de pérdida de paquetes en el router frontera de ingreso para el tráfico de VoIP: SWAN (Caso 1) vs. DS-SWAN (Casos 2 - 3).

La combinación de estas dos figuras se refleja en la *Fig. 4.15*, que representa la tasa de pérdida de paquetes en el router frontera para el tráfico de VoIP en %.

El número de paquetes de VoIP enviados que llegan al router frontera es mayor en SWAN que en las dos versiones de DS-SWAN porque con DS-SWAN se pierden más paquetes en la red ad hoc. El número de paquetes perdidos en el router frontera de ingreso aumenta en SWAN así como en DS-SWAN durante todo el tiempo a lo largo de la simulación porque los nodos envían ráfagas de paquetes de VoIP. Muchos paquetes de VoIP de estas ráfagas son descartados por el router frontera cuando son controlados por el medidor del token bucket porque están fuera del perfil. En SWAN se descartan más paquetes en comparación con DS-SWAN porque en la red ad hoc se

han perdido menos paquetes, y por lo tanto los nodos con paquetes de VoIP consiguen enviar más ráfagas de VoIP y algunos paquetes de estas ráfagas son descartados por el router frontera. Las pérdidas de paquetes para el tráfico de VoIP a la entrada del router frontera son mayores en el Caso 2 que en el Caso 3 porque aunque en el Caso 2 hay más pérdidas de paquetes de VoIP en la red ad hoc, en este caso se ha estrangulado el tráfico best-effort en toda la red y como consecuencia todavía se podrán enviar más ráfagas que en el Caso 3, si bien las diferencias de pérdidas entre ambos casos no son muy grandes.

Sin embargo, en todos los casos el porcentaje de paquetes perdidos en el router frontera (como máximo 0,22 % en el Caso 1) junto con el porcentaje de paquetes perdidos en la red ad hoc (como máximo 1,1 % en el Caso 3) no es superior al 5% de paquetes perdidos establecido para mantener una calidad en la conversación de VoIP aceptable [155]. Si añadimos a este resultado el hecho de que en SWAN el retardo medio extremo a extremo es mayor de 150 ms (degradación de la VoIP), podemos concluir que DS-SWAN supera a SWAN en el mantenimiento de los parámetros de calidad de servicio y consigue ofrecer a los flujos de tiempo real la calidad deseada para su correcto funcionamiento en un entorno congestionado, cosa que con el modelo SWAN no es posible.

De las dos versiones de DS-SWAN presentadas, la segunda versión (Caso 3: “DS-SWAN – fuentes de VoIP + vecinos”) supera a la primera en rendimiento, pues con esta versión es posible obtener unos parámetros de calidad de servicio para los flujos de tiempo real mejores en cuanto a retardos extremo a extremo, pérdidas de paquetes en el router frontera y jitter, sin que se produzca inanición del tráfico best-effort. Por este motivo se seleccionará esta versión del modelo DS-SWAN como la más adecuada para su implementación en una red ad hoc.

#### *4.2.2.1.2 Análisis de las simulaciones con respecto a la escalabilidad para tráfico enviado desde la red ad hoc hacia la red fija*

En vista de que de todas las versiones analizadas, aquella que presenta un mejor rendimiento es “DS-SWAN – fuentes de VoIP + vecinos”, se ha decidido estudiar en profundidad esta versión realizando diversas simulaciones.

Se han realizado simulaciones para estudiar la escalabilidad para tráfico enviado desde la red ad hoc hacia la red fija con respecto al número de fuentes de VoIP y a la

movilidad; también se ha analizado el impacto del tráfico de best-effort sobre las conexiones de VoIP.

Con el fin de analizar la escalabilidad del protocolo DS-SWAN con respecto al número de fuentes de VoIP, se han simulado dos tipos de redes:

- *Una red de tamaño pequeño (20 nodos) que dispone de 13 fuentes CBR (de tráfico best-effort) en la parte de la red ad hoc y un área donde se mueven los nodos inalámbricos de 700 m x 500 m. Los gateways están colocados en las coordenadas (100, 250) y (600, 250). El escenario de simulación y los valores de todos los parámetros es exactamente igual que en la sección 4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163). Caso 1: "DS-SWAN – 20 nodos".*
- *Una red de tamaño mayor (40 nodos) que dispone de 26 fuentes CBR (de tráfico best-effort) en la parte de la red ad hoc y un área donde se mueven los nodos inalámbricos de 990 m x 707 m. Los gateways están colocados en las coordenadas (141, 354) y (849, 354). El escenario de simulación y los valores del resto de parámetros es exactamente igual que en la sección 4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163). Caso 2: "DS-SWAN – 40 nodos".*

En las *Fig. 4.16 y Fig. 4.17* se ilustran el retardo máximo extremo a extremo del tráfico de VoIP y el throughput del tráfico CBR para estas dos redes con respecto al número de fuentes de VoIP. Se han medido estos parámetros a partir del segundo 80, porque es entonces cuando todas las fuentes CBR y de VoIP están ya activas (condiciones estacionarias).

Cuando el número de fuentes de VoIP es inferior a 4 no es necesario poner en marcha el protocolo DS-SWAN porque no hay congestión y se pueden mantener los retardos extremo a extremo del tráfico de VoIP perfectamente. Cuando el número de fuentes de VoIP es superior a 4, el protocolo DS-SWAN se pone en marcha para estrangular el tráfico best-effort y reducir de esta forma los retardos extremo a extremo de los flujos de VoIP, tal y como sucede en los Casos 1 y 2. En el Caso 2 resulta necesario estrangular en mayor medida las conexiones CBR para mantener los retardos extremo a extremo del tráfico de VoIP porque hay el doble (26) y están introduciendo mayor congestión, con lo cual disminuye en mayor grado el throughput del tráfico best-effort en este caso. En ambos casos, los retardos máximos extremo a extremo para el tráfico de VoIP son inferiores a 150 ms [154], manteniéndose una

calidad en la conversación aceptable y sin que llegue a producirse inanición del tráfico best-effort.

En todos los casos el número total de pérdidas del tráfico de VoIP como máximo alcanza el 1%.

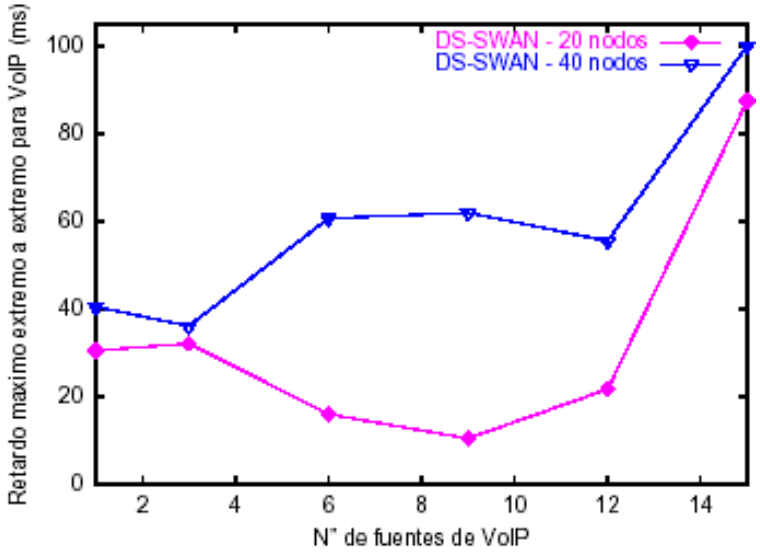



**Fig. 4.16.** Retardo máximo extremo a extremo del tráfico de VoIP: "DS-SWAN – 20 nodos" (Caso 1) vs. "DS-SWAN – 40 nodos" (Caso 2).

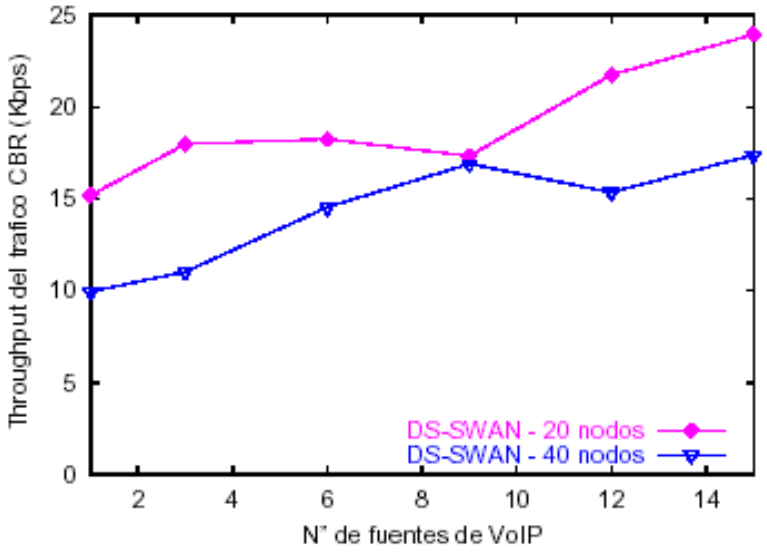



**Fig. 4.17.** Throughput para el tráfico CBR: "DS-SWAN – 20 nodos" (Caso 1) vs. "DS-SWAN – 40 nodos" (Caso 2).

Con el fin de analizar el impacto del tráfico best-effort en las conexiones de VoIP se han realizado simulaciones de una red donde:

- *No hay carga de tráfico best-effort (Caso 1).*
- *La carga del tráfico best-effort es de 32 Kbps (Hay 13 conexiones CBR) (Tamaño de los paquetes CBR: 80 bytes y tiempo entre paquetes: 0,02 s) (Caso 2).*

- *La carga del tráfico best-effort es de 48 Kbps (Hay 13 conexiones CBR) (Tamaño de los paquetes CBR: 120 bytes y tiempo entre paquetes: 0,02 s) (Caso 3).*

El escenario de simulación y los valores de resto de parámetros es exactamente igual que en la sección *4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163).*

Las *Fig. 4.18* y *Fig. 4.19* representan el retardo extremo a extremo y el jitter en el transcurso del tiempo de simulación para 15 conexiones de VoIP respectivamente. Resulta interesante observar como la presencia de tráfico best-effort resulta decisiva para que se produzca un aumento tanto del retardo extremo a extremo como del jitter de los flujos de VoIP, si bien todas las gráficas tienen tendencia a estabilizarse.

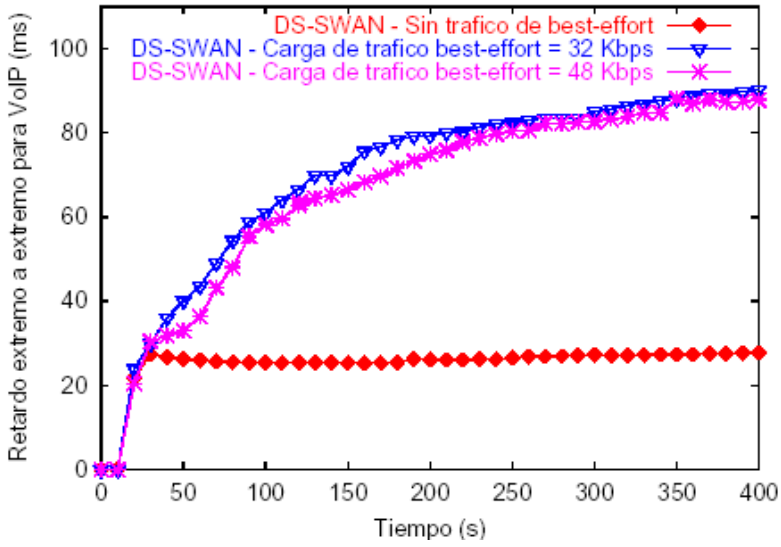



**Fig. 4.18.** Retardo extremo a extremo para el tráfico de VoIP.

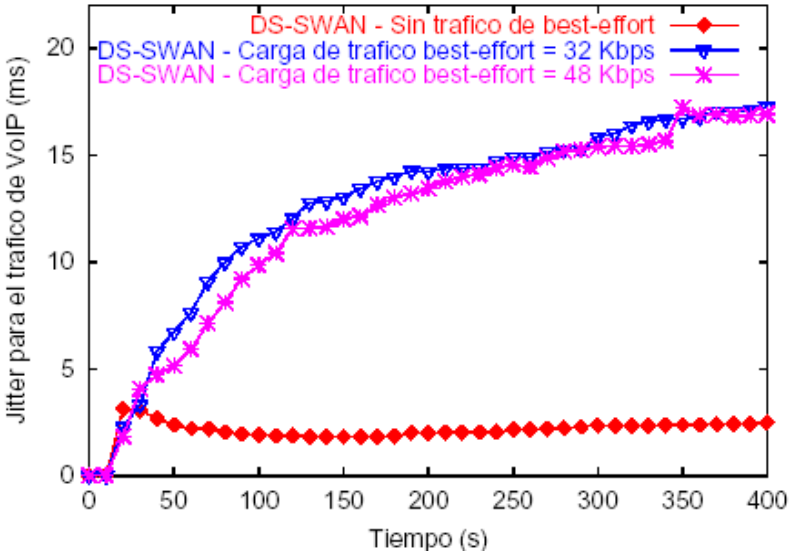



**Fig. 4.19.** Jitter para el tráfico de VoIP.

La *Fig. 4.20* representa el throughput para tráfico best-effort. Como puede observarse, cuando la tasa del tráfico CBR es menor (32 Kbps), el throughput disminuye en comparación con una red donde las conexiones CBR tienen una tasa de tráfico best-effort de 48 Kbps.

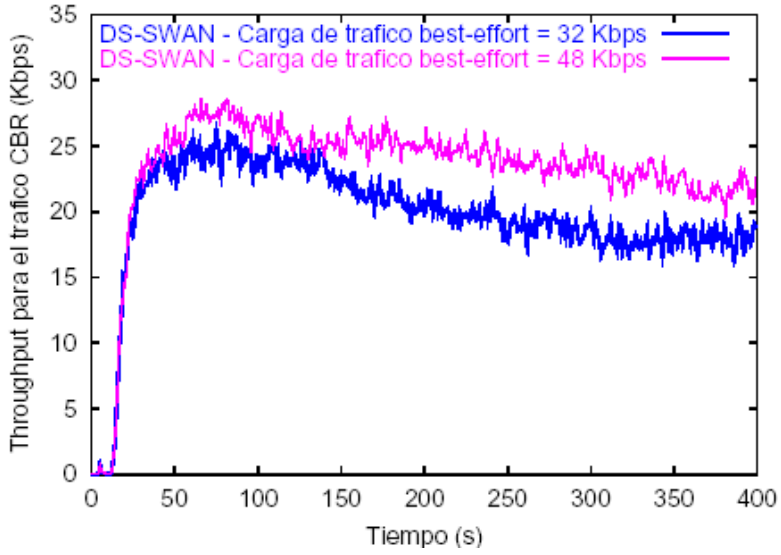



**Fig. 4.20.** Throughput para el tráfico CBR.

En todos los casos las pérdidas totales de paquetes de tiempo real se mantienen por debajo del 5% [155] necesario para mantener una buena calidad de VoIP.

Si además tenemos en cuenta que los retardos extremo a extremo de los flujos de VoIP son siempre inferiores a 150 ms [154] y el jitter está acotado, no se produce una degradación significativa de las aplicaciones de VoIP.

Con el fin de analizar el impacto de la movilidad y la escalabilidad del protocolo DS-SWAN con respecto a la movilidad, se han simulado dos tipos de redes:

- *Una red de tamaño pequeño (20 nodos) que dispone de 13 fuentes CBR (de tráfico best-effort) y 15 fuentes de VoIP (de tiempo real) en la parte de la red ad hoc y un área donde se mueven los nodos inalámbricos de 700 m x 500 m. Los gateways están colocados en las coordenadas (100, 250) y (600, 250). El escenario de simulación y los valores de todos los parámetros es exactamente igual que en la sección 4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163). Caso 1: "DS-SWAN – 20 nodos".*
- *Una red de tamaño mayor (40 nodos) que dispone de 26 fuentes CBR (de tráfico best-effort) y 30 fuentes de VoIP (de tiempo real) en la parte de la red ad hoc y un área donde se mueven los nodos inalámbricos de 990 m x 707 m. Los gateways están colocados en las coordenadas (141, 354) y (849, 354). El escenario de simulación y los valores del resto de parámetros es exactamente*

*igual que en la sección 4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163). Caso 2: "DS-SWAN – 40 nodos".*

En las *Fig. 4.21 y Fig. 4.22* se ilustran el retardo máximo extremo a extremo del tráfico de VoIP y el throughput del tráfico CBR para estas dos redes con respecto a la movilidad (tiempo de pausa).

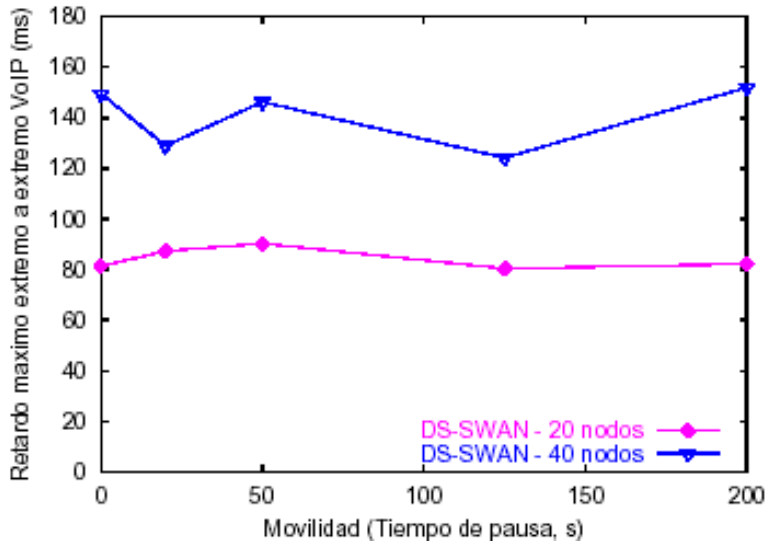



**Fig. 4.21.** Retardo máximo extremo a extremo del tráfico de VoIP: "DS-SWAN – 20 nodos" (Caso 1) vs. "DS-SWAN – 40 nodos" (Caso 2).

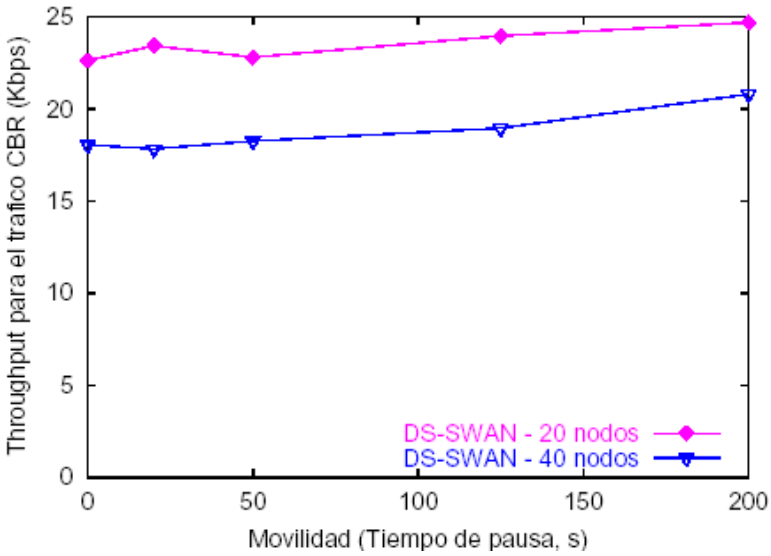



**Fig. 4.22.** Throughput para el tráfico CBR: "DS-SWAN – 20 nodos" (Caso 1) vs. "DS-SWAN – 40 nodos" (Caso 2).

Los retardos máximos extremo a extremo se mantienen muy estables y bajos en el Caso 1, presentando una mayor variabilidad en el Caso 2, donde dichos retardos aumentan cuando la movilidad es muy alta (hay mucha movilidad y el protocolo de encaminamiento está siempre descubriendo nuevas rutas con lo cual la latencia aumenta) o bien muy baja (si los nodos no se mueven pueden quedar algunos nodos

rodeados de muchos otros nodos intermedios y permanecer muy congestionados durante largo tiempo, con lo cual la latencia también aumenta).

El throughput del tráfico CBR disminuirá al disminuir el tiempo de pausa de los nodos de la red ad hoc debido fundamentalmente a que habrá más roturas de enlaces y Descubrimientos de Ruta debido a la movilidad.

Los retardos máximos extremo a extremo son más bajos y el throughput más alto en el Caso 1 en comparación con el Caso 2 porque en el Caso 2 tenemos un mayor número de fuentes de VoIP y de tráfico best-effort compitiendo por el acceso al medio, con lo cual aumenta la congestión.

En todos los casos el número total de pérdidas del tráfico de VoIP como máximo alcanza el 1%.

#### *4.2.2.1.3 Análisis de las simulaciones con tráfico best-effort TCP para tráfico enviado desde la red ad hoc hacia la red fija*

En esta ocasión se ha decido estudiar aquella versión del modelo DS-SWAN que presenta un mejor rendimiento, “DS-SWAN – fuentes de VoIP + vecinos”, pero con una particularidad: Esta vez el tráfico best-effort introducido será TCP en vez de UDP.

Se han realizado simulaciones para tráfico enviado desde la red ad hoc hacia la red fija.

El comportamiento del tráfico TCP es bastante diferente al UDP, pues la entrega de paquetes se realiza extremo a extremo de manera fiable y se garantiza a las aplicaciones la entrega de paquetes de datos de forma ordenada. Si se producen pérdidas de paquetes o estos llegan desordenados o excesivamente tarde, se puede solicitar a la fuente TCP el reenvío de aquellos paquetes que han tenido problemas para alcanzar sus destinos adecuadamente.

En las redes ad hoc se producen pérdidas o se desordenan paquetes debido a errores en el canal inalámbrico o cuando no existe una ruta debido a la movilidad; el TCP interpretará erróneamente que las pérdidas que se producen son debidas a la congestión y pondrá en marcha un procedimiento para controlar la congestión, reduciendo la ventana de control de congestión y produciéndose una degradación innecesaria del throughput de este tipo de tráfico.

Se considera un escenario en el cual hay 24 flujos de VoIP y 26 flujos TCP. Ha sido necesario aumentar la presencia tanto del número de conexiones de VoIP como de flujos TCP porque sino no podía observarse la influencia del tráfico best-effort sobre

las conexiones de tiempo real, ya que los flujos TCP enseguida ponen en marcha mecanismos para reducir la congestión y reducir su throughput si advierten que tienen pérdidas. De los 26 flujos TCP, 13 son aplicaciones FTP con un tamaño de paquete de 1024 bytes. Los otros 13 flujos son fuentes web modeladas como pequeñas transferencias de ficheros TCP con un tamaño de fichero aleatorio y un periodo de silencio entre transferencias aleatorio. El tamaño del fichero se obtiene a partir de una distribución Pareto con un tamaño de fichero medio de 10 Kbytes y un parámetro de configuración de 1,2. La longitud del periodo de silencio entre dos transferencias también sigue una distribución Pareto con el mismo parámetro de configuración y una media de 10 s. Así es posible crear un tráfico de fondo (background) best-effort a ráfagas.

Las fuentes TCP inician su envío de paquetes en instantes escogidos aleatoriamente a partir del intervalo [15 s, 20 s] para la primera y segunda fuente, [20 s, 25 s] para la tercera y cuarta fuente y así sucesivamente hasta llegar a [75 s, 80 s] para la penúltima y última fuente.

En el router frontera de ingreso se selecciona WRED como mecanismo para aplicar el control de policía porque es adecuado para evitar la congestión del tráfico TCP. WRED combina las capacidades del algoritmo RED *(Véase la sección 2.4.1.1* El algoritmo RED*, pág. 74)* con el nivel de prioridad asignado a cada paquete, concediéndose un tratamiento preferente a los paquetes más prioritarios.

WRED usa diferentes precedencias de descarte, dependiendo de la clase de prioridad del paquete. Este mecanismo puede ser configurado para descartar aquellos paquetes menos prioritarios cuando comienza a haber congestión, diferenciando servicios entre distintas clases de tráfico. Si se configurara WRED para ignorar el nivel de prioridad de cada paquete, entonces este mecanismo funcionaría igual que RED.

Cuando se empiezan a tirar paquetes porque comienza a haber congestión, la fuente lo advertirá y disminuirá su tasa de transmisión. Al tirarse algunos paquetes antes de esperar a que la cola esté totalmente llena, con WRED se evita descartar una gran cantidad de paquetes. La precedencia de descarte de cada paquete guardará una relación directa con el codepoint codificado en el campo DSCP. En las simulaciones realizadas, cuando la cola se llene con un número de paquetes entre 30 y 50, la probabilidad de descarte será de 0,1 para los paquetes de la aplicación FTP (codepoint ‘18’ asociado) y de 0,2 para los paquetes de la aplicación Web (codepoint ‘20’ asociado).

El escenario de simulación y los valores del resto de parámetros es exactamente igual que en la sección *4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163).*

La *Fig. 4.23* representa el retardo medio extremo a extremo para el tráfico de VoIP. Como puede observarse, con SWAN crecen excesivamente (por encima de 150 ms [154]) los retardos asociados a los paquetes de tiempo real, mientras que con DS-SWAN es posible controlar los retardos correspondientes a este tipo de tráfico y mantenerlos acotados en torno a los 120 ms.

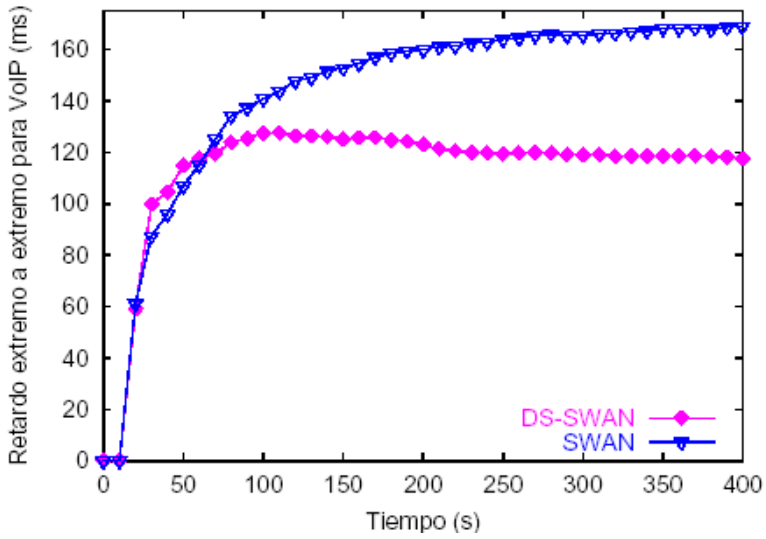



**Fig. 4.23.** Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).

La *Fig. 4.24* representa el jitter para el tráfico de VoIP. El jitter se mantiene acotado, pero es más pequeño cuando aplicamos el modelo DS-SWAN gracias al mecanismo introducido para que no aumenten tanto los retardos ni tampoco las variaciones de retardo de los flujos de tiempo real.

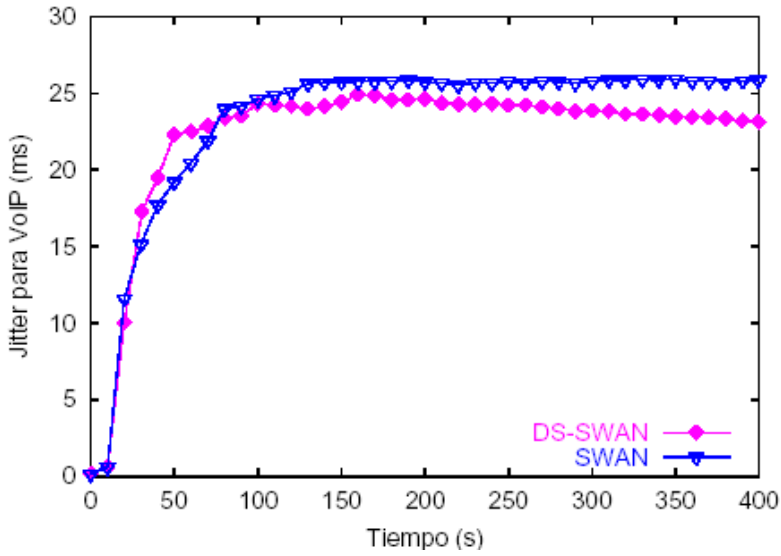



**Fig. 4.24.** Jitter para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).

La *Fig. 4.25* ilustra el throughput para el tráfico TCP en ambos sistemas. Con DS-SWAN es necesario estrangular el tráfico TCP para mantener los retardos extremo a

extremo del tráfico de VoIP, pero, a pesar del todo, ni tan siquiera en este caso llega a producirse una inanición del tráfico best-effort.

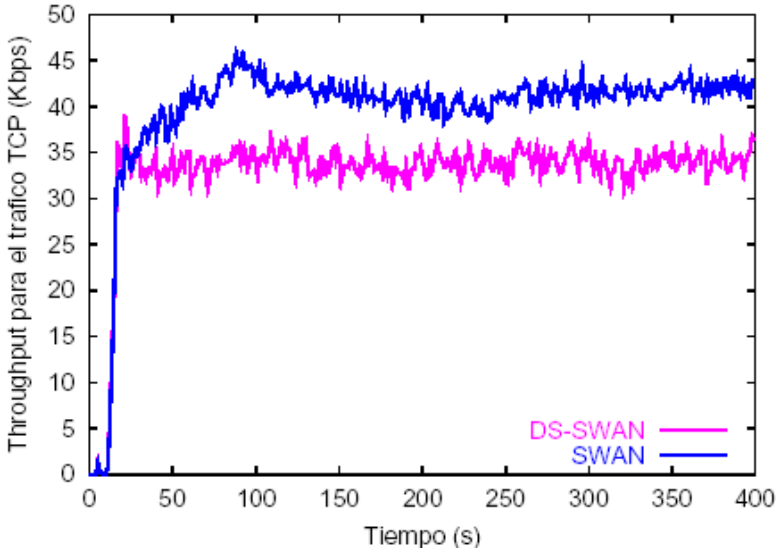



**Fig. 4.25.** Throughput para el tráfico best-effort: DS-SWAN (Caso 1) vs. SWAN (Caso 2).

El número total de pérdidas para el tráfico de VoIP contando las de la red ad hoc y las que se producen en el router frontera, no llega a superar en ningún caso el 3% de paquetes perdidos sobre los transmitidos, siendo inferiores al 5% estimado para que puedan enviarse con calidad los flujos de VoIP. Si añadimos que los jitters están acotados en ambos casos, pero solamente en el caso del modelo DS-SWAN se consiguen mantener los retardos extremo a extremo de los flujos de VoIP por debajo de los 150 ms, entonces queda demostrada la superioridad de este modelo para poder diferenciar servicios en redes ad hoc con respecto al modelo SWAN y su efectividad frente a cualquier tipo de tráfico best-effort (ya sea TCP o UDP).

## *4.2.2.1.4 Análisis de las simulaciones para tráfico enviado desde la red fija hacia la red ad hoc y estudio de la escalabilidad*

Se han realizado simulaciones para tráfico enviado desde la red fija hacia la red ad hoc y se ha analizado la escalabilidad del protocolo con respecto al número de fuentes de VoIP para tráfico enviado en el mismo sentido. La explicación teórica del modelo de calidad de servicio DS-SWAN cuando el tráfico se envía desde la red fija hacia la red ad hoc se halla recogido en la sección *(Véase la sección 4.2.1.2 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red fija hacia la red ad hoc, pág. 161).*

En vista de que de todas las versiones analizadas, aquella que presenta un mejor rendimiento es "DS-SWAN – fuentes de VoIP + vecinos", se ha evaluado y comparado el rendimiento del modelo "DS-SWAN-fuentes de VoIP + vecinos" (Caso 1) con el modelo SWAN (Caso 2).

Las 13 fuentes de tráfico de fondo CBR y 15 fuentes de tráfico de VoIP VBR se hallan situadas en la red basada en infraestructura y transmiten información hacia la red ad hoc en forma de paquetes. Los destinos de los flujos best-effort y de VoIP se escogen aleatoriamente entre los nodos móviles de la red ad hoc.

El resto de valores de los parámetros seleccionados para las simulaciones son los mismos que los explicados en la sección *4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163).*

La *Fig. 4.26* muestra el retardo medio extremo a extremo para el tráfico de VoIP. Como puede comprobarse, con el modelo SWAN sigue siendo imposible mantener la calidad de servicio para las fuentes de VoIP, pues sus retardos extremo a extremo superan los 150 ms [154], algo que no sucede con el modelo DS-SWAN. Como DS-SWAN permite la interacción de los mecanismos de calidad de servicio establecidos en la red ad hoc y en la red fija, se logra mantener la calidad de servicio de los flujos de tiempo real (retardos de los flujos de VoIP inferiores a 150 ms).

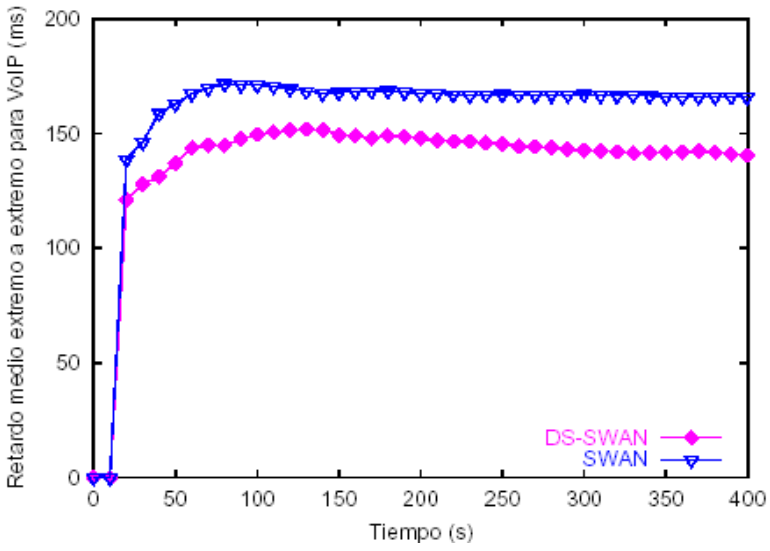



**Fig. 4.26.** Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).

La *Fig. 4.27* muestra el jitter para el tráfico de VoIP en ambos sistemas. En DS-SWAN el jitter es inferior que en SWAN gracias a los mecanismos introducidos para que no siga aumentando el retardo extremo a extremo de los flujos de VoIP ni tampoco la variación del retardo.

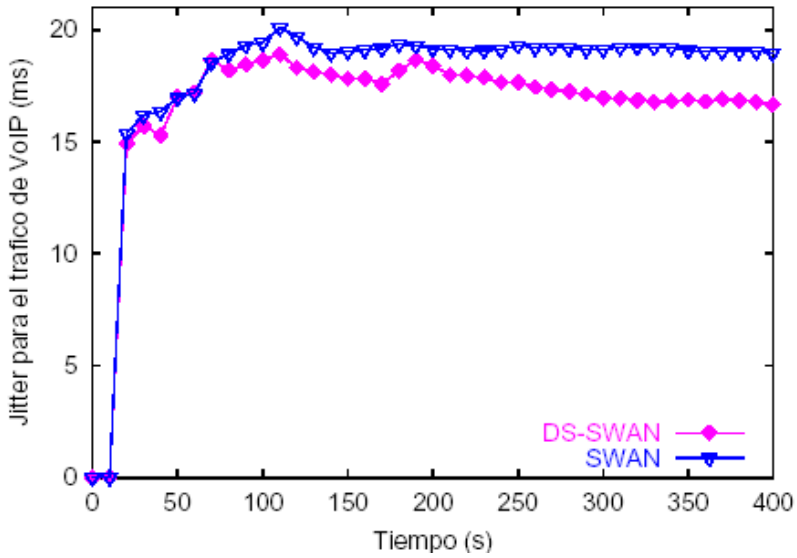



**Fig. 4.27.** Jitter para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).

La *Fig. 4.28* muestra el throughput para el tráfico CBR en ambos sistemas. Se observa que con DS-SWAN no se produce inanición del tráfico best-effort a pesar de la necesidad de estrangular dicho tráfico para mantener los retardos extremo a extremo de los flujos de tiempo real. Debe resaltarse que el sistema se recupera con una cierta rapidez de la disminución del throughput acontecida en ciertos momentos debido al estrangulamiento forzado de dicho tráfico.

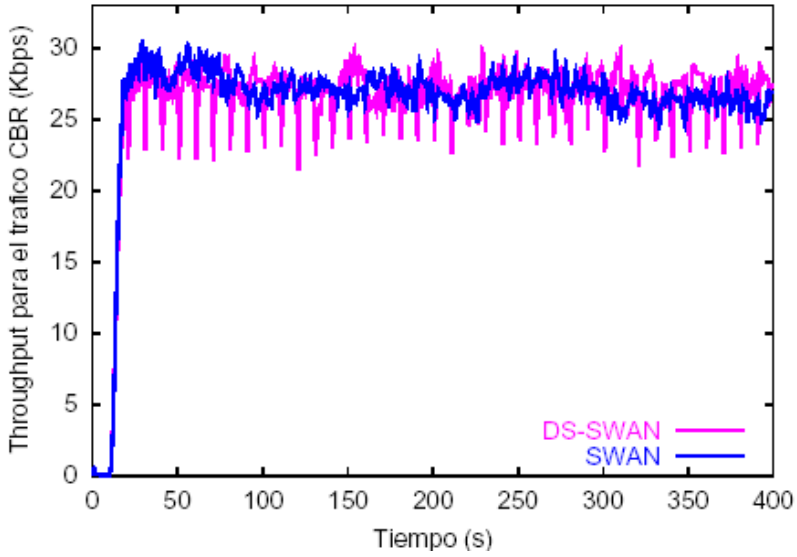



**Fig. 4.28.** Throughput para el tráfico best-effort: DS-SWAN (Caso 1) vs. SWAN (Caso 2).

La *Fig. 4.29* muestra la tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP aplicando ambos modelos. La tasa de pérdida de paquetes indica en cada instante temporal el número total de paquetes que hasta ese momento se han perdido con respecto al número total de paquetes enviados. Puede observarse que, si bien con DS-SWAN las pérdidas aumentan más en comparación con SWAN debido a la presencia de tráfico adicional de alta prioridad (los mensajes de QoS_PERDIDA), el

porcentaje de paquetes perdidos en la red ad hoc tiende a estabilizarse. Si además añadimos que la tasa de pérdidas de VoIP en el router frontera es nula, el porcentaje total de paquetes perdidos de VoIP con respecto a los enviados continúa siendo muy inferior a un 5 % [155], con lo cual de deduce que el mantenimiento de la calidad de servicio para este tipo de tráfico resulta del todo viable.

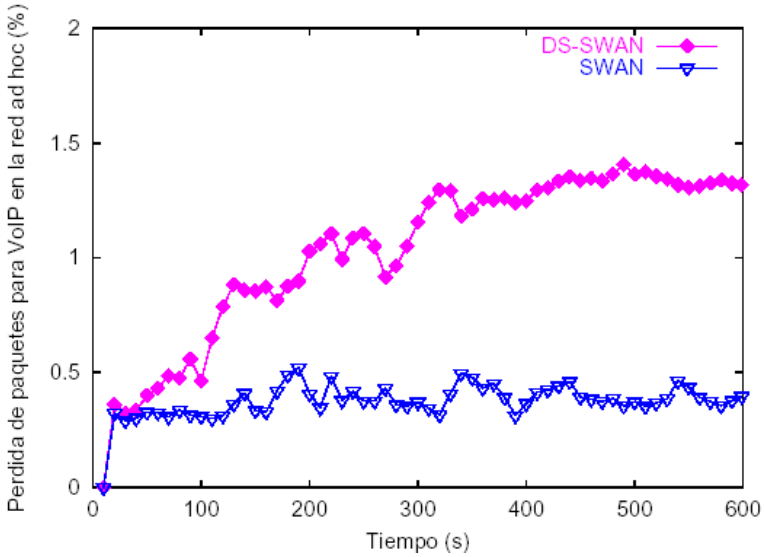



**Fig. 4.29.** Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: DS-SWAN (Caso 1) vs. SWAN (Caso 2).

Por lo tanto, después de haber analizado los parámetros de calidad de servicio para las conexiones de VoIP más destacables, puede concluirse que, aunque ha sido necesario introducir ciertas modificaciones en el sistema para favorecer la transmisión del tráfico de VoIP, esta transmisión continúa siendo realmente exitosa, sin llegar a producirse en ningún momento inanición del tráfico best-effort.

En vista de que de también se obtienen buenos resultados con la versión "DS-SWAN – fuentes de VoIP + vecinos" cuando el tráfico fluye hacia la red ad hoc, se ha decidido realizar en este trabajo de investigación más simulaciones enviando tráfico en este sentido.

Con el fin de analizar la escalabilidad del protocolo con respecto al número de fuentes de VoIP se han simulado dos tipos de redes:

- *Una red de tamaño pequeño (20 nodos) que dispone de 13 fuentes CBR (de tráfico best-effort) en la parte de la red fija y un área para la red ad hoc de 700 m x 500 m donde se mueven los nodos inalámbricos. Los gateways están colocados en las coordenadas (100, 250) y (600, 250). El escenario de simulación y los valores de todos los parámetros es exactamente igual que en la sección 4.2.2.1 Escenario de simulación (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163). Caso 1: "DS-SWAN – 20 nodos".*

- *Una red de tamaño mayor (40 nodos) que dispone de 26 fuentes CBR (de tráfico best-effort) en la parte de la red fija y un área para la red ad hoc de 990 m x 707 m donde se mueven los nodos inalámbricos. Los gateways están colocados en las coordenadas (141, 354) y (849, 354). El escenario de simulación y los valores del resto de parámetros es exactamente igual que en la sección 4.2.2.1 Escenario de simulación (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163). Caso 2: "DS-SWAN – 40 nodos".*

En las *Fig. 4.30* y *Fig. 4.31* se ilustran el retardo máximo extremo a extremo del tráfico de VoIP y el throughput del tráfico CBR para estas dos redes con respecto al número de fuentes de VoIP.

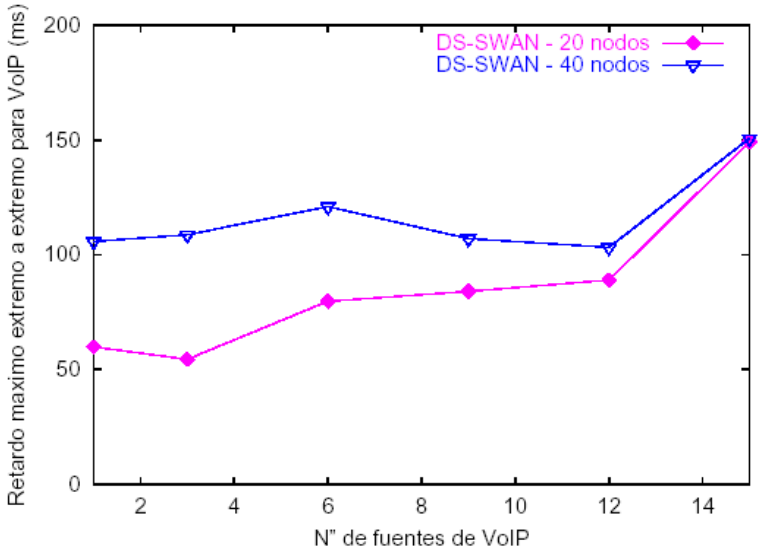



**Fig. 4.30.** Retardo máximo extremo a extremo del tráfico de VoIP: "DS-SWAN – 20 nodos" (Caso 1) vs. "DS-SWAN – 40 nodos" (Caso 2).

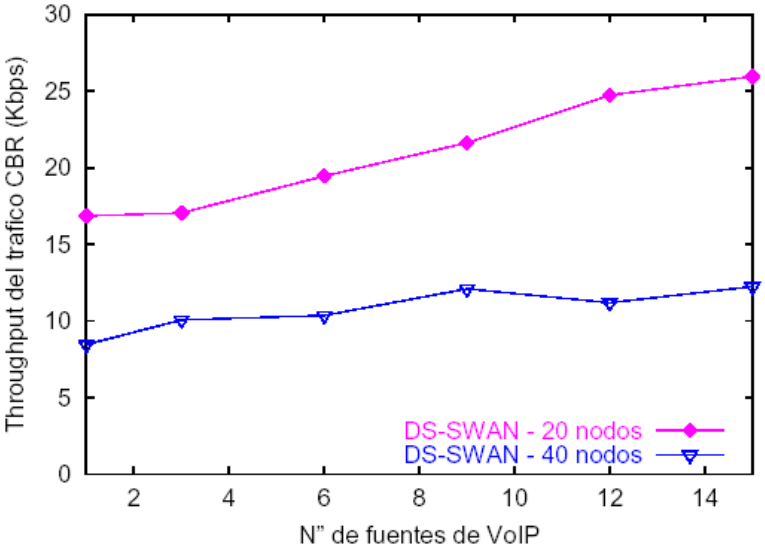



**Fig. 4.31.** Throughput para el tráfico CBR: "DS-SWAN – 20 nodos" (Caso 1) vs. "DS-SWAN – 40 nodos" (Caso 2).

Cuando el número de fuentes de VoIP es inferior a 4 no es necesario poner en marcha el protocolo DS-SWAN porque no hay congestión y se pueden mantener los retardos extremo a extremo del tráfico de VoIP perfectamente. Cuando el número de fuentes de VoIP es superior a 4 el protocolo DS-SWAN se pone en marcha para estrangular el tráfico best-effort y reducir de esta forma los retardos extremo a extremo de los flujos de VoIP, tal y como sucede en los Casos 1 y 2. En el Caso 2 resulta necesario estrangular en mayor medida las conexiones CBR para mantener los retardos extremo a extremo del tráfico de VoIP porque hay el doble (26) de fuentes CBR y están introduciendo mayor congestión, con lo cual disminuye en mayor grado el throughput medio del tráfico best-effort en este caso. En ambos casos, los retardos máximos extremo a extremo para el tráfico de VoIP son inferiores a 150 ms [154], manteniéndose una calidad en la conversación aceptable y sin que llegue a producirse inanición del tráfico best-effort. En todos los casos el número total de pérdidas del tráfico de VoIP como máximo alcanza el 1%, un porcentaje totalmente aceptable [155].

## *4.3 Conclusiones*

En el Capítulo 4 se ha analizado la necesidad de la existencia de modelos de calidad de servicio en redes ad hoc interconectadas con redes fijas. La literatura existente acerca de la provisión de calidad de servicio entre redes ad hoc y redes fijas es escasísima. Aunque una publicación precursora subraya la necesidad de desarrollar modelos de este tipo, hasta la fecha no ha aparecido ninguno.

Para cubrir este déficit se ha presentado el primer modelo de calidad de servicio de la investigación actual para la comunicación entre una red ad hoc y una red IP fija. Esta nueva contribución de esta tesis doctoral se denomina “Desarrollo del modelo de calidad de servicio DS-SWAN para redes ad hoc conectadas a redes fijas” y se ha procedido a realizar una explicación teórica y a presentar los resultados detalladamente mediante simulaciones de la misma.

Con el fin de estudiar el protocolo DS-SWAN y conseguir cada vez mayores mejoras, se han introducido modificaciones en su funcionamiento, analizándose los resultados obtenidos en busca siempre de aquella solución que proporcione mejor diferenciación de servicios y un mayor aprovechamiento de los recursos de red.

Se han presentado los resultados obtenidos a la hora de simular las versiones más destacadas del protocolo DS-SWAN y compararlas con el modelo de calidad de servicio SWAN. En las simulaciones realizadas, se ha considerado un escenario en el cual una red ad hoc estaba conectada a una red IP fija y se han establecido una serie

de conexiones de tráfico best-effort CBR y de tráfico de tiempo real VBR (VoIP) para comunicar los nodos móviles de la red ad hoc con alguno de los hosts localizados en la red fija. Como modelo de calidad de servicio en la red fija se ha usado la arquitectura DiffServ. Si se usa el protocolo SWAN como modelo de calidad de servicio en la red ad hoc, el retardo medio extremo a extremo de los flujos de VoIP (tráfico de tiempo real) es mayor de 150 ms; por lo tanto, estos flujos se ven degradados y queda demostrado que si se aplica separadamente un modelo de calidad de servicio en la red ad hoc y otro en la red IP fija, pero estos dos modelos no interaccionan entre sí para mantener la calidad de servicio extremo a extremo, las aplicaciones de tiempo real funcionarán incorrectamente.

En cambio, todas las versiones del modelo de calidad de servicio DS-SWAN presentadas cooperan con la arquitectura de Servicios Diferenciados en la red fija y sí que son capaces de mantener la calidad de servicio de las aplicaciones de tiempo real, sin que se produzca inanición del tráfico best-effort.

De entre todas las versiones presentadas del modelo de calidad de servicio desarrollado en esta tesis doctoral, la versión “DS-SWAN – fuentes de VoIP + vecinos” es la que supera en rendimiento al resto, mejorando los parámetros de calidad de servicio para los flujos de tiempo real en cuanto a retardo extremo a extremo, jitter y tasa de pérdida de paquetes, sin que se produzca inanición del tráfico best-effort. Por este motivo, se selecciona esta versión del modelo como la más adecuada para su implementación en una red ad hoc.

Se ha estudiado la escalabilidad de esta versión de DS-SWAN para redes ad hoc de distinto tamaño con respecto al número de fuentes de VoIP. También se ha estudiado la escalabilidad del protocolo con respecto a la movilidad, realizándose simulaciones para redes ad hoc de distinto tamaño. Además, se ha analizado el impacto del volumen de tráfico best-effort en las conexiones de VoIP para una red que utilice este modelo de calidad de servicio.

Asimismo, se ha analizado la influencia del tráfico best-effort sobre los flujos de tiempo real, introduciendo como tráfico best-effort fuentes CBR en el primer caso o bien fuentes TCP en el segundo. En todas las simulaciones realizadas los resultados obtenidos han sido siempre muy positivos y con la ayuda de DS-SWAN ha sido siempre posible transmitir los flujos de tiempo real correctamente.

En dichas simulaciones se ha analizado la transmisión de tráfico best-effort y de tiempo real desde la red ad hoc hacia la red fija. Posteriormente, se ha invertido el sentido del tráfico (desde la red fija hacia la red ad hoc) con el fin de comprobar si en este caso más problemático, donde el gateway corría el riesgo de convertirse en un nodo permanentemente congestionado, podía garantizarse el buen funcionamiento del

modelo desarrollado. Los resultados han demostrado que aunque ha sido necesario introducir ciertas modificaciones en el sistema para favorecer la transmisión del tráfico de VoIP, esta transmisión continúa siendo realmente exitosa, sin haya llegado a producirse en ningún momento inanición del tráfico best-effort. La escalabilidad del protocolo con respecto a un número creciente de fuentes de VoIP ha vuelto a ser corroborada de nuevo mediante simulaciones.

# 5 Protocolos de encaminamiento en redes ad hoc conectadas a redes fijas

Resulta de singular importancia el desarrollo de protocolos de encaminamiento capaces de encaminar tráfico desde la red ad hoc hacia la red fija. En la sección *5.1* se indicará cuál es el estado actual de la investigación con respecto a este tema. En la sección *5.2* se presentará una nueva contribución que consiste en el diseño e implementación de un protocolo de encaminamiento para una red ad hoc (SD-AODV) conectada a una red IP fija que basa su funcionamiento en criterios relacionados con la provisión de calidad de servicio.

## 5.1 Estado actual de la investigación

Los protocolos de encaminamiento específicamente diseñados para redes ad hoc aisladas son utilizados para enviar paquetes desde un nodo origen a un nodo destino dentro de la propia red; sin embargo, cuando la red ad hoc está conectada a Internet a través de un gateway, estos protocolos de encaminamiento no están capacitados para la transmisión de paquetes desde la red ad hoc hacia la red IP fija. Para conseguir comunicar una red ad hoc con una red IP fija es necesario modificar los protocolos de encaminamiento para redes ad hoc existentes con el fin de que sean capaces de descubrir gateways, tal y como se explica en la sección *5.1.1.*

### 5.1.1 Mecanismo básico para la conexión de redes ad hoc con Internet

El objetivo es poder comunicar una red fija IP con una red ad hoc. Para ello será necesario utilizar un protocolo de encaminamiento adecuado, que sea capaz de establecer y mantener las rutas requeridas. El borrador de Internet (Internet draft) “Global Connectivity for IPv6 Mobile Ad Hoc Networks” [157] describe como pueden acceder las redes ad hoc móviles a Internet gracias a la modificación del protocolo de encaminamiento AODV [112] para que sea capaz de descubrir gateways. IPv6 permite que una amplia variedad de dispositivos puedan estar conectados a Internet porque proporciona varias ventajas sobre IPv4 tales como un mayor espacio de

direccionamiento y soporte para la autoconfiguración de direcciones sin estados (stateless address autoconfiguration).

Para poder comunicar Internet con una red ad hoc los paquetes deben ser transmitidos a un gateway, tal y como se ilustra en la *Fig. 5.1.* Este dispositivo debe ser capaz de implementar tanto la pila de protocolos de la red ad hoc como los protocolos de la red fija, encaminando los paquetes de una red a otra.

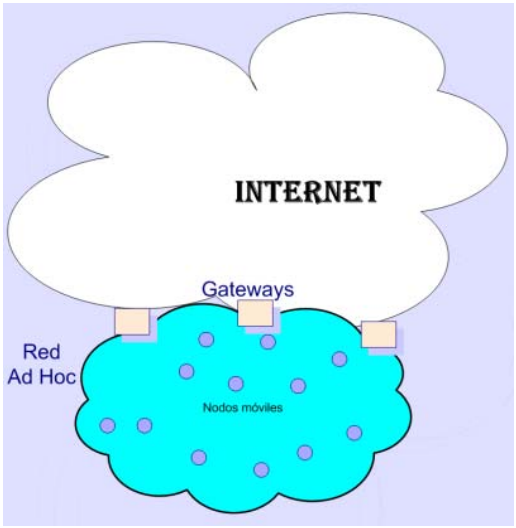



**Fig. 5.1.** Escenario de interconexión.

Las pilas de protocolos usadas por los nodos móviles, gateways y los nodos de Internet se muestran en la *Fig. 5.2.*

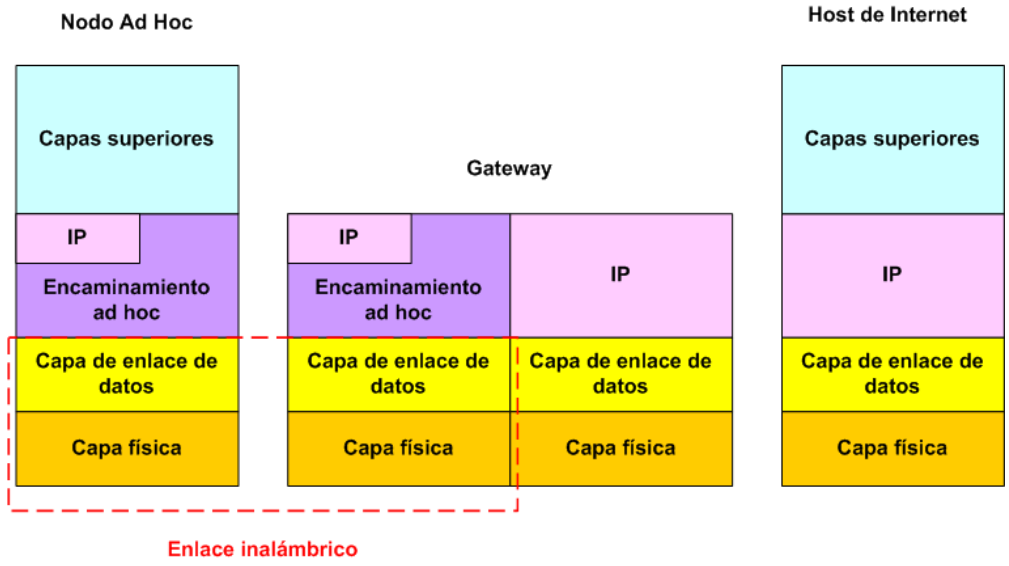



**Fig. 5.2.** Arquitectura de los protocolos.

Considérese que un nodo móvil desea enviar datos a Internet. Necesita una dirección global para enviar estos datos. Estas direcciones se denominan globales porque pueden ser usadas en redes públicas (como Internet). Para que un dispositivo pueda construir una dirección global IPv6 para su propio uso (autoconfiguración de

dirección), necesitará conocer cual es la longitud del prefijo de red, que es el valor decimal e indica cuantos bits contiguos de la parte izquierda de la dirección componen el prefijo. Además, precisa descubrir la ubicación y dirección de un gateway y usar la ruta hacia este dispositivo como ruta por defecto para enviar paquetes a Internet.

En este trabajo de investigación se ha seleccionado el método de descubrimiento de gateway híbrido [158], [169] para encontrar un gateway. Este método combina el descubrimiento de gateway proactivo [170] y reactivo [171], minimizando las desventajas de usar estos dos métodos separadamente. En el descubrimiento de gateway híbrido, el gateway envía periódicamente un mensaje RREP_I (un RREP extendido donde el flag I ( Internet-Global Address Resolution Flag) se usa para la resolución global de direcciones) a aquellos nodos móviles que se encuentran dentro de su rango de transmisión. El RREP-I contiene información acerca del gateway y se propaga dentro de una zona limitada (por un determinado número de saltos con respecto del gateway). Un nodo móvil que reciba un RREP_I debe usar la información acerca de la longitud del prefijo global y la dirección IPv6 del gateway que ha sido transportada en el mensaje para descubrir el prefijo global. Después, este nodo móvil autoconfigura una nueva dirección IPv6 enrutable y selecciona la dirección del gateway como ruta por defecto.

Si un nodo móvil fuera del rango de transmisión del gateway y de la zona de propagación del RREP_I quiere tener conectividad a Internet, envía en modo broadcast un mensaje RREQ_I (un Route Request extendido donde el flag I se usa para la resolución global de direcciones) al grupo multicast de gateways de Internet (Internet Gateway Multicast group), es decir, a la dirección IP del grupo de gateways en la red ad hoc. El nodo móvil puede usar cualquier dirección global disponible como dirección fuente (por ejemplo su dirección local (home address) de Mobile IPv6) o bien puede crear una nueva temporalmente usando el MANET_INITIAL_PREFIX, tal y como se describe claramente en el IP Address Autoconfiguration for Ad Hoc Networks [172]. Si otro nodo móvil recibe este RREQ_I, lo reenvía en modo broadcast hasta que el RREQ_I llega a un gateway, que responde devolviendo un RREP_I. Entonces el nodo fuente borra la dirección temporal y obtiene la dirección IPv6 globalmente enrutable del gateway.

Supóngase que un nodo fuente S en la red ad hoc desea enviar paquetes a un nodo destino D pero desconoce si este nodo está ubicado en la red ad hoc o bien en Internet [173]. En principio, el nodo S debe consultar su tabla de encaminamiento y usar una ruta hacia este destino, en el supuesto de que exista. En caso contrario, el nodo envía un RREQ y espera recibir un RREP. No obstante, el nodo S no recibirá ningún RREP si sucede que el nodo D es un nodo fijo. Si así fuera, el nodo fuente

debe enviar el paquete usando la ruta por defecto (si existe; de lo contrario dicho nodo debe de obtener esta ruta mediante el método ya explicado) y confiar en que el gateway entregará el paquete.

Por otro lado, si en vez de enviar paquetes desde la red ad hoc hacia la red fija, se envían en sentido contrario (desde Internet hacia la red ad hoc), los paquetes serán enviados a una dirección global de la red ad hoc, pero viajarán primero desde Internet hacia el gateway. El gateway será capaz de determinar cuál es el prefijo de red correspondiente. Entonces, utilizará el protocolo de encaminamiento AODV [112] clásico para poder encontrar una ruta hacia el destino, sin necesidad de que se realice ninguna modificación adicional de dicho protocolo.

## *5.2 Contribución: Desarrollo del protocolo de encaminamiento SD-AODV para la mejora de la supervivencia y la cooperación en el mantenimiento de la calidad de servicio en redes ad hoc conectadas a redes fijas*

A continuación se presenta una contribución de esta tesis doctoral consistente en el desarrollo de un protocolo de encaminamiento denominado SD-AODV para la mejora de la supervivencia y la cooperación en el mantenimiento de la calidad de servicio en redes ad hoc conectadas a redes fijas.

### *5.2.1 Explicación teórica*

Seguidamente se realiza una explicación teórica del protocolo de encaminamiento SD-AODV (Service Differentiation-AODV), que ha sido desarrollado en esta tesis doctoral; se incluye tanto una descripción formal como un análisis detallado del algoritmo.

En este trabajo de investigación se considera que tanto el protocolo de encaminamiento AODV [112] como el protocolo de encaminamiento que ha sido diseñado e implementado por nosotros en este trabajo de investigación (SD-AODV) pueden proporcionar acceso a Internet a la red ad hoc porque han sido hechas las modificaciones sugeridas en el borrador de Internet "Global Connectivity for IPv6 Mobile Ad-Hoc Networks" [157].

AODV es un protocolo de encaminamiento best-effort porque no proporciona calidad de servicio. Como se ha visto en la sección *3.1.2.2*, usa el mínimo número de saltos

hacia un destino como métrica primaria para seleccionar una ruta con independencia de la congestión de tráfico. Si el destino está ubicado en la red fija, se selecciona la ruta con menor número de saltos hasta el gateway, independientemente de si esa ruta atraviesa nodos muy congestionados. Si se establecen flujos de tiempo real entre una red ad hoc y una red IP fija, es necesario mantener unos ciertos requisitos de calidad de servicio como retardos extremo a extremo acotados. En consecuencia, hay una clara necesidad de que este protocolo sea modificado de forma que sea capaz de operar con un modelo de interacción de calidad de QoS para mantener los retardos extremo a extremo de los flujos de tiempo real.

En este trabajo de investigación se ha considerado que el modelo de QoS "DS-SWAN – fuentes de VoIP + vecinos" *(Véase la sección 4.2.1.1 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija, pág. 151)* debe ser seleccionado de entre todas las versiones del modelo DS-SWAN existentes para ser aplicado al sistema, porque es el que nos permite obtener un mejor rendimiento. En esta versión, un nodo recibe un mensaje de QoS_PERDIDA porque es una fuente de VoIP problemática o bien un nodo a lo largo de la ruta hacia la fuente que tiene problemas para mantener sus retardos extremo a extremo por debajo de 150 ms, o también porque es un vecino de estos nodos y está compitiendo con ellos por el acceso al medio, produciéndose congestión. Bajo estas condiciones de congestión, los nodos consumen más energía por las pérdidas de paquetes, al mismo tiempo que crece el retardo de dichos paquetes tanto a nivel de la capa MAC como extremo a extremo. Con la ayuda de DS-SWAN es posible mitigar los efectos nocivos de la congestión para mantener la calidad de servicio deseada para los flujos de VoIP retardando el acceso del tráfico best-effort a la capa MAC y en consecuencia al medio físico. Pero además, deben emprenderse nuevas acciones conjuntamente con el esquema DS-SWAN, no solamente para reducir la congestión existente en algunas regiones problemáticas, sino también para evitar que nuevas cargas de tráfico puedan aumentarla.

En esta tesis doctoral se presenta un protocolo de encaminamiento sencillo y escalable llamado SD-AODV (Service Differentiation-AODV) que aprovecha la cooperación entre DS-SWAN y DiffServ para evitar que el grado de congestión crezca en las zonas en las cuales las fuentes de VoIP tienen problemas para mantener sus retardos extremo a extremo.

El simple objetivo de SD-AODV consiste en redireccionar o desviar las rutas nuevas para que no atraviesen aquellos nodos que han recibido un mensaje de QoS_PERDIDA como paquete en modo broadcast y debido a ello han sido marcados como nodos congestionados. SD-AODV actúa de una manera completamente

distribuida suprimiendo nuevas peticiones de ruta (RREQs) en estos nodos congestionados para asegurarse de que el nuevo tráfico encaminado no aumente la congestión en los 'cuellos de botella' y poder así seguir manteniendo los parámetros de QoS deseados para el tráfico de tiempo real. Los nodos congestionados no propagan los RREQs, con lo cual las zonas congestionadas también reducen su tráfico gracias a este motivo.

El funcionamiento de SD-AODV se ilustra en la *Fig. 5.3*.

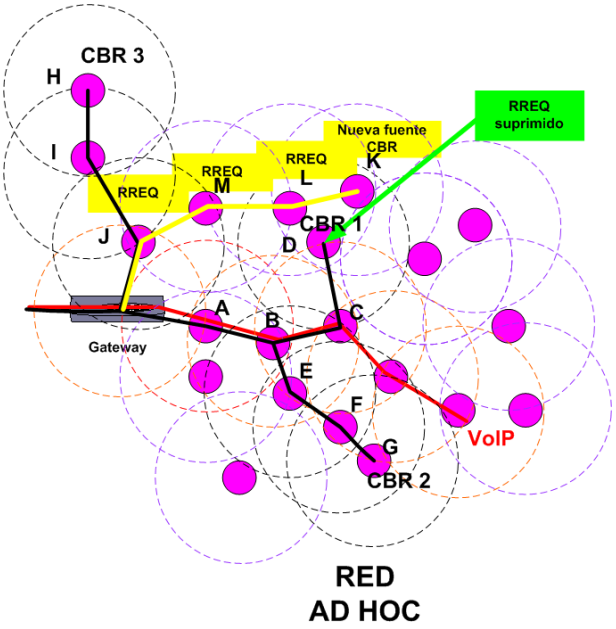



**Fig. 5.3.** Ejemplo de red.

Se muestra un ejemplo de una red ad hoc donde se han establecido un flujo de tiempo real de VoIP y tres flujos de tráfico best-effort CBR denominados CBR1, CBR2 y CBR3, de forma que los paquetes son enviados hacia Internet a través del gateway. Se aplica primero la versión ("DS-SWAN – fuentes de VoIP + vecinos") con SD-AODV. Se considera que si el flujo de VoIP tiene problemas para mantener sus retardos extremo a extremo por debajo de los 150 ms se enviarán mensajes de QoS_PERDIDA tanto a la fuente de VoIP como a los nodos intermedios a lo largo de la ruta en la red ad hoc. Entonces estos nodos reenviarán el mensaje de QoS_PERDIDA como un paquete broadcast de forma que finalmente los nodos A, B, C, D, E, F y G estrangularán su tráfico best-effort. Además, estos nodos son marcados como nodos congestionados. Más tarde, la nueva fuente CBR k quiere enviar paquetes hacia Internet y usa SD-AODV para encontrar una ruta. Se suprimirá la petición de ruta (RREQ) en la fuente CBR1 (nodo D) porque este nodo ha sido marcado como nodo congestionado, de forma que finalmente se establecerá una nueva ruta a través de los nodos K, L, M y J, los cuales no están compitiendo por el acceso al medio con la

fuente de VoIP que tiene problemas para mantener sus retardos ni con los nodos intermedios a lo largo de esa ruta.

La supresión 'ingenua' de la creación de rutas puede evitar el uso del único camino posible entre dos hosts; si esto sucede con alguna fuente de tráfico best-effort, no resulta tan importante; si esto sucede con alguna fuente de tráfico de VoIP tendría una importancia mayor, pero sería absurdo ofrecer a priori a un nuevo flujo de VoIP o best-effort una nueva ruta hacia el destino a expensas de flujos de tiempo real existentes.

Consideremos, por ejemplo, dos sesiones de tiempo real establecidas que usan dos rutas disjuntas; supongamos que aunque en el pasado hayan tenido problemas para mantener sus requisitos de calidad de servicio, gracias al funcionamiento del DS-SWAN, ahora ya no los tienen. Si una de ellas, debido a la movilidad, necesitara encontrar una nueva ruta para poder continuar enviando información y quisiera compartir nodos que pertenecen a la ruta de la otra fuente, ambas rutas terminarían congestionadas si se empleara el protocolo AODV tradicional. En cambio, con SD-AODV, la primera fuente tratará de encontrar una ruta disjunta a la de la segunda fuente para que ambas continúen pudiendo mantener sus requisitos de calidad de servicio sin problemas.

Aparecen zonas congestionadas en una red ad hoc cuando hay una excesiva competencia entre nodos móviles por el acceso al medio compartido; estas zonas suelen ser de naturaleza transitoria, porque los flujos establecidos en la red ad hoc son reencaminados a lo largo del tiempo debido a cambios en la topología de red y en la carga de tráfico. Cuando cae un enlace (por ejemplo debido a la movilidad) y se rompe una ruta hacia un destino, el nodo fuente debe de usar su protocolo de encaminamiento para encontrar una nueva ruta hacia ese destino (o bien puede utilizar una ruta almacenada en su tabla de encaminamiento en caso de que exista). Sin embargo, si algunos o todos los nodos a lo largo de la ruta antigua fueron marcados como nodos congestionados, ahora estos nodos y sus vecinos son desmarcados porque han cambiado las condiciones de tráfico y la topología y no es posible conocer a priori si estos nodos continuarán experimentando congestión o no.

La arquitectura de Servicios Diferenciados continuará cooperando con el modelo DS-SWAN de forma que DS-SWAN recibirá información actualizada dinámicamente sobre los valores de los parámetros de QoS en la red ad hoc y los nodos desmarcados podrían ser marcados como nodos congestionados otra vez en el futuro si fuera necesario.

Puede observarse que la selección de nodos congestionados depende de la interacción entre la arquitectura de Servicios Diferenciados y el modelo DS-SWAN, mientras que la cooperación entre los protocolos DS-SWAN y SD-AODV es

absolutamente necesaria para que SD-AODV pueda ser capaz de evitar que el tráfico quede concentrado en ciertos nodos.

Con SD-AODV es posible reducir el consumo de energía de los nodos congestionados, porque no se producirán tantas pérdidas de paquetes en estos nodos motivadas por la existencia de colisiones, sobrecarga de los buffers, etc. En consecuencia, no se consumirá tanta energía efectuando retransmisiones continuas de paquetes. Así, estos nodos no extinguirán sus recursos de energía antes de tiempo y el problema de que la red sufra una partición prematura se verá reducido.

SD-AODV es una modificación del protocolo de encaminamiento AODV [112], pero hubiera sido posible haber modificado cualquier protocolo de encaminamiento que hubiera trabajado en la red ad hoc.

SD-AODV no puede ser considerado un protocolo de encaminamiento con QoS porque no proporciona rutas de QoS, es decir, no trata de encontrar una ruta desde la fuente hacia el destino que satisfaga los requisitos de QoS extremo a extremo; sin embargo, contribuye junto a DS-SWAN en el mantenimiento de los parámetros de QoS para los flujos de tiempo real en la red ad hoc.

## *5.2.2 Simulaciones*

En las secciones siguientes se presentarán los resultados obtenidos a la hora de simular el protocolo SD-AODV y compararlo con el protocolo de encaminamiento AODV en una red ad hoc que está conectada a una red fija y donde se usa el modelo de calidad de servicio DS-SWAN y más concretamente la versión ("DS-SWAN - fuentes de VoIP + vecinos") *(Véase la sección 4.2.1.1 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija, pág. 151).*

En la sección *5.2.2.1.1* se van a realizar simulaciones para analizar tráfico que circula desde la red ad hoc hacia la red IP fija. En las simulaciones de la sección *5.2.2.1.2* se va a analizar el tráfico que circula en sentido inverso.

## *5.2.2.1 Escenario de simulación*

El escenario para todas las simulaciones realizadas resulta ser el mismo que el descrito en la sección *4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163).* Lo que sí hay que tener en cuenta es que en la sección *5.2.2.1.2* el envío de tráfico es diferente porque viaja en sentido contrario (desde la red fija hacia la red ad hoc).

### *5.2.2.1.1 Análisis de las simulaciones para tráfico enviado desde la red ad hoc hacia la red fija*

En este trabajo de investigación se ha evaluado y comparado el rendimiento de un sistema que usa ("DS-SWAN - fuentes de VoIP + vecinos") como esquema de QoS y AODV como protocolo de encaminamiento (Caso 1) con un sistema que usa el mismo esquema de QoS pero SD-AODV como protocolo de encaminamiento (Caso 2). El tráfico es enviado desde la red ad hoc hacia la red fija.

Los resultados han sido publicados en [165].

La *Fig. 5.4* representa el retardo medio extremo a extremo para el tráfico de VoIP. Hay una mejora significativa cuando se utiliza SD-AODV como protocolo de encaminamiento (Caso 2) en comparación con el Caso 1. La razón es que las fuentes de tráfico de VoIP con problemas para mantener sus retardos extremo a extremo y los nodos a lo largo de las rutas así como sus vecinos (con quienes compiten por el acceso al medio) no son sobrecargados con más flujos de tráfico CBR o VoIP una vez estos nodos han recibido el aviso correspondiente; por lo tanto, se suprimen las nuevas peticiones de ruta en estos nodos. Además, se reduce la probabilidad de que los nuevos flujos de VoIP experimenten mayor congestión porque las nuevas rutas para estos flujos evitan seleccionar como nodos intermedios aquellos que han sido previamente declarados como nodos congestionados. También se observa que cuando hay fuentes de VoIP que deben buscar una nueva ruta para sus flujos debido a la movilidad, no se quedan sin poderla encontrar.

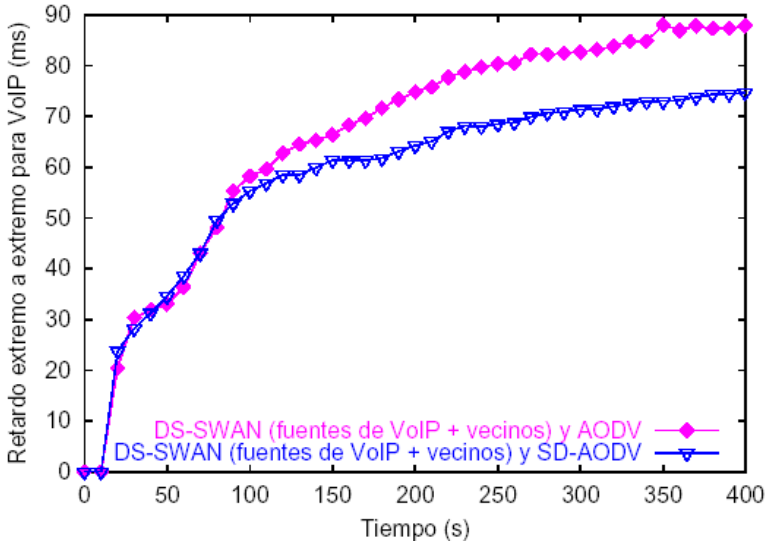



**Fig. 5.4.** Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

La *Fig. 5.5* representa el throughput medio para las fuentes de tráfico best-effort CBR. En ambos casos no hay inanición del tráfico best-effort. Sin embargo, usando SD-AODV como protocolo de encaminamiento las nuevas fuentes de tráfico CBR evitan seleccionar rutas con nodos marcados como nodos congestionados y como resultado aumenta el throughput medio con respecto al Caso 1 porque es menos probable que estas fuentes o sus nodos intermedios a lo largo de las rutas tengan que estrangular sus flujos.

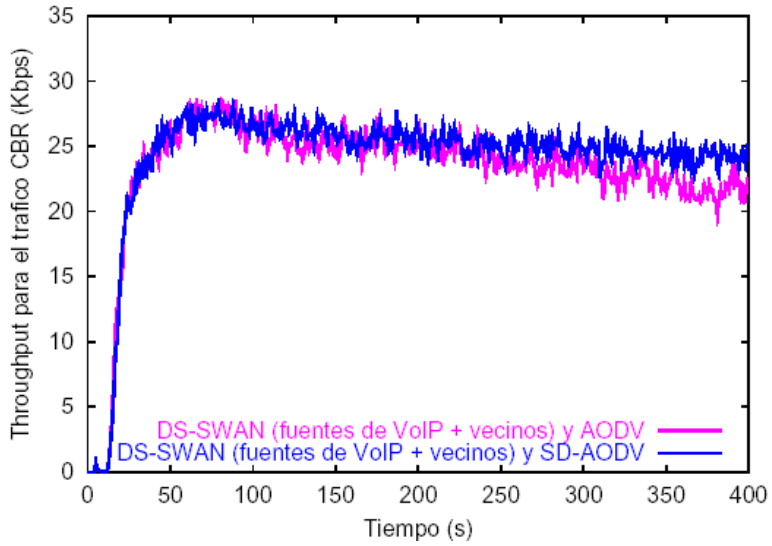



**Fig. 5.5.** Throughput para el tráfico best-effort: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

La *Fig. 5.6* representa la tasa máxima de pérdida de paquetes para el tráfico de VoIP en la red ad hoc. Se ha calculado mirando las pérdidas de paquetes en intervalos de diez segundos para cada una de las quince fuentes de VoIP de una simulación, tomando el valor máximo de pérdidas de entre todas las fuentes para cada intervalo y haciendo la media de los máximos para las cuarenta simulaciones efectuadas. El número de paquetes perdidos disminuye si se usa SD-AODV en vez de AODV como protocolo de encaminamiento porque SD-AODV balancea el tráfico de VoIP y CBR entre distintos nodos, reduciendo la congestión. Puede observarse como las dos gráficas alcanzan la convergencia; en el Caso 2 la curva alcanza la convergencia con mayor lentitud, pero como contrapartida la tasa de paquetes perdidos es siempre menor que en el Caso 1.

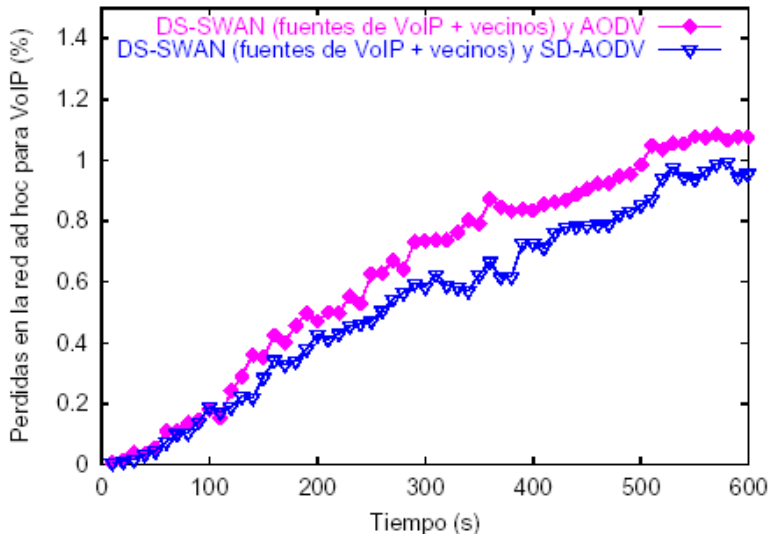



**Fig. 5.6.** Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

La *Fig. 5.7* representa la tasa de pérdida de paquetes en el router frontera de ingreso para el tráfico de VoIP en %. Se ha calculado el número de paquetes perdidos en el router frontera con respecto al número total de paquetes enviados durante intervalos de 10 segundos y después se ha hecho la media para las 40 simulaciones. El número de paquetes de VoIP enviados que llegan al router frontera es mayor en el Caso 2 que en el Caso 1 porque con AODV como protocolo de encaminamiento se pierden más paquetes de VoIP en la red ad hoc. El número de paquetes perdidos en el router frontera de ingreso aumenta en ambos casos durante todo el tiempo a lo largo de la simulación porque los nodos con paquetes de VoIP tienen problemas para acceder al medio debido a la congestión y cuando finalmente acceden al medio envían ráfagas de VoIP. Muchos paquetes de VoIP de estas ráfagas son descartados por el router frontera cuando son controlados por el medidor (token bucket) porque se hallan fuera del perfil. El número de paquetes descartados es muy similar en ambos casos.

Por otro lado, en ambos casos, el porcentaje de paquetes de VoIP perdidos en el router frontera (como máximo 0,13 % en el Caso 2) junto con el porcentaje de paquetes de VoIP perdidos en la red ad hoc (como máximo 1,1 % en el Caso 1) es inferior al 5 % de paquetes perdidos establecido para mantener una calidad de conversación para VoIP aceptable [155].

El número total de paquetes de VoIP perdidos (teniendo en cuenta las pérdidas de paquetes tanto en la red ad hoc como en el router frontera) es siempre menor si se usa SD-AODV como protocolo de encaminamiento.

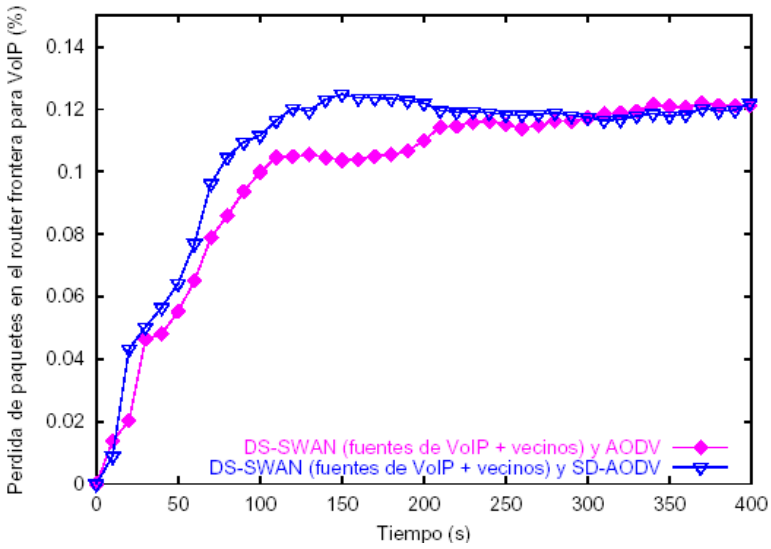



**Fig. 5.7.** Tasa de pérdida de paquetes en el router frontera de ingreso para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

El jitter para el tráfico de VoIP se ilustra en la *Fig. 5.8*. En el modelo DS-SWAN se actúa sobre los parámetros del conformador de tráfico (leaky bucket) para controlar más la tasa del tráfico best-effort y así reducir los retardos extremo a extremo de los paquetes de tiempo real. En el Caso 2 el jitter es menor que en el Caso 1 porque las fuentes CBR o de VoIP que buscan caminos para enviar el tráfico a sus destinos no seleccionan rutas con nodos intermedios en regiones donde los flujos de VoIP tienen problemas para mantener sus retardos extremo a extremo. Por consiguiente, la congestión no se incrementa tanto en estas regiones, de forma que los retardos extremo a extremo de los flujos de VoIP no aumentan tanto y las variaciones de los retardos tampoco.

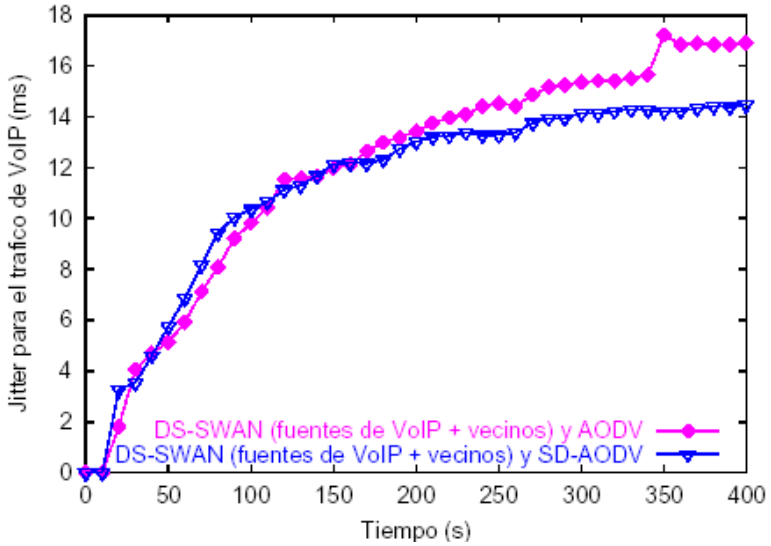



**Fig. 5.8.** Jitter para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

Ahora se desea analizar el impacto del protocolo de encaminamiento en la supervivencia de la red. Para lograrlo, se han ejecutado 40 simulaciones considerando el mismo escenario básico, pero en este caso se considera que los sistemas tienen recursos de energía escasos; concretamente se supone que todos los nodos tienen una capacidad de batería inicial de 70 J. La *Fig. 5.9* muestra las pérdidas de nodos debido al agotamiento de su energía en función del tiempo para un sistema que usa ("DS-SWAN – fuentes de VoIP + vecinos) como esquema de calidad de servicio y AODV como protocolo de encaminamiento con otro sistema que usa el mismo esquema de calidad de servicio pero SD-AODV como protocolo de encaminamiento (Caso 2). Puede observarse que cuando se usa SD-AODV en cada instante de tiempo hay menos nodos que han agotado sus recursos de energía porque este protocolo de encaminamiento favorece el balanceo de carga y como resultado no hay unos pocos nodos muy cargados de tráfico que enseguida agotan su capacidad de batería, sino que el tráfico se reparte entre los distintos nodos de la red ad hoc; en consecuencia, se alarga el tiempo de vida de los nodos móviles en la red ad hoc.

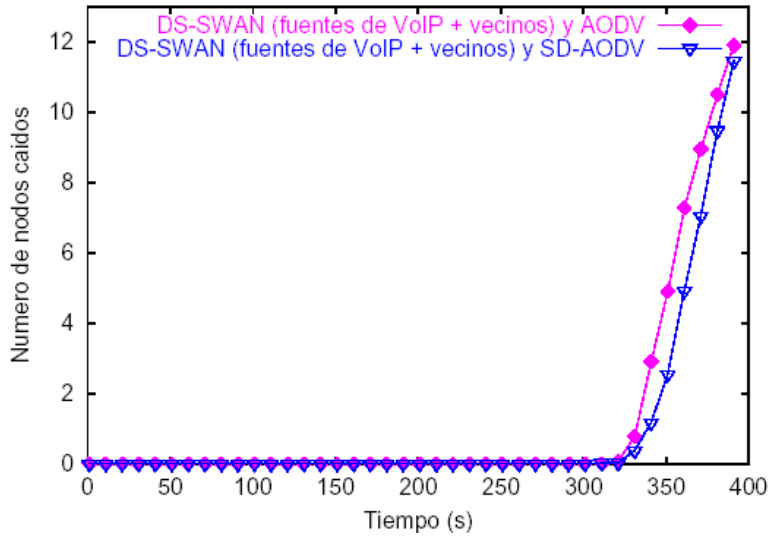



**Fig. 5.9.** Número de nodos caídos en la red ad hoc: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

De las dos versiones presentadas, la segunda versión (Caso 2: "DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV") supera a la primera en rendimiento, pues con esta versión es posible que SD-AODV colabore conjuntamente con DS-SWAN para mitigar las condiciones de congestión y reducir los problemas relacionados con la congestión, extendiendo el tiempo de vida de los nodos móviles en la red ad hoc. La combinación de ambos protocolos reduce no solamente los retardos extremo a extremo y el jitter de los flujos de VoIP, sino también las pérdidas de paquetes de VoIP y mejora el throughput medio del tráfico best-effort. Por este motivo se seleccionará el protocolo de encaminamiento SD-AODV para combinarlo con el modelo de interacción de

calidad de servicio DS-SWAN, pues ha demostrado ser el más adecuado para su implementación en una red ad hoc.

## *5.2.2.1.2 Análisis de las simulaciones para tráfico enviado desde la red fija hacia la red ad hoc*

En la sección *5.2.2.1.1* de este trabajo de investigación *(Véase la sección 5.2.2.1.1 Análisis de las simulaciones para tráfico enviado desde la red ad hoc hacia la red fija, pág. 199),* se ha evaluado y comparado el rendimiento de un sistema que usa ("DS-SWAN - fuentes de VoIP + vecinos") como esquema de QoS y AODV como protocolo de encaminamiento (Caso 1) con un sistema que usa el mismo esquema de QoS pero SD-AODV como protocolo de encaminamiento (Caso 2).

La comparación ha sido realizada cuando el tráfico de tiempo real y best-effort era enviado desde la red ad hoc hacia la red IP fija. Con el fin de generalizar los resultados y demostrar la viabilidad del protocolo SD-AODV, se ha decidido estudiar el funcionamiento de dicho protocolo combinado con el modelo de calidad de servicio ("DS-SWAN - fuentes de VoIP + vecinos") en un escenario con las mismas características que el explicado en la sección *4.2.2.1 (Véase la sección 4.2.2.1 Escenario de simulación, pág. 163)*, pero donde en este caso el tráfico fluye en sentido contrario. Esto significa que las 15 fuentes de VoIP y 13 fuentes CBR estarán situadas en la red basada en infraestructura y transmitirán información hacia la red ad hoc en forma de paquetes.

Como en este caso cada gateway compite como cualquier otro nodo de la red ad hoc por el acceso al medio, ha sido preciso modificar los valores de los parámetros de DS-SWAN resumidos en la *Tabla 4.1* y en su lugar se han tomado los valores de la *Tabla 4.2* descritos en la sección *4.2.1.2 (Véase la sección 4.2.1.2 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red fija hacia la red ad hoc, pág.161).*

En [174] se muestran los resultados obtenidos al realizar estas simulaciones.

La *Fig. 5.10* representa el retardo extremo a extremo para el tráfico de VoIP. Puede observarse que utilizando como protocolo de encaminamiento SD-AODV (Caso 2) es posible mantener más bajos los retardos extremo a extremo para el tráfico de tiempo real. De hecho, en este trabajo de investigación se han realizado simulaciones que demuestran que no hubiera sido preciso modificar los parámetros del modelo DS-SWAN de la *Tabla 4.1* para poder mantener los retardos extremo a extremo del tráfico de VoIP cuando se usa SD-AODV como protocolo de encaminamiento. De todas maneras, hemos preferido utilizar los nuevos valores para los parámetros del protocolo

DS-SWAN que figuran en la *Tabla 4.2* para comparar ambos sistemas en idénticas condiciones. Se observa que SD-AODV evita que se seleccionen aquellos nodos congestionados como nodos intermedios para realizar nuevas conexiones de VoIP o CBR, con lo cual se seleccionan nodos menos cargados de tráfico y así se favorece la calidad de las sesiones de tiempo real establecidas.

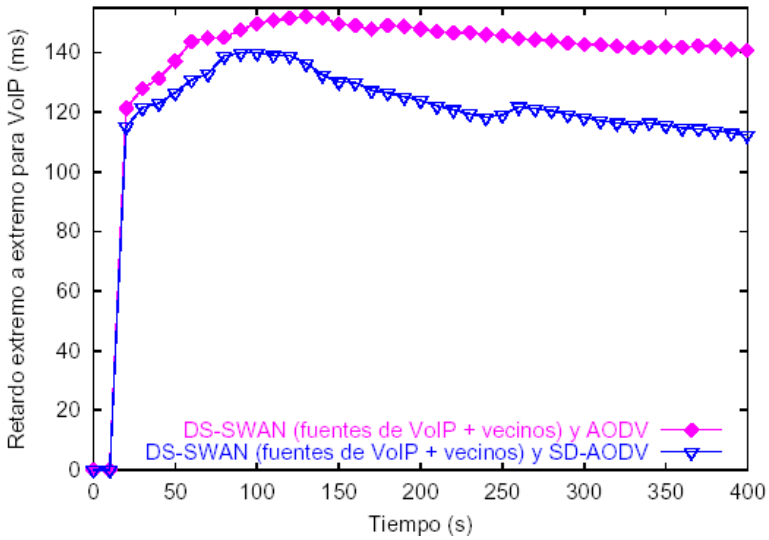



**Fig. 5.10.** Retardo extremo a extremo para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

La *Fig. 5.11* muestra el jitter para el tráfico de VoIP. El jitter para el tráfico de VoIP resulta claramente inferior cuando se usa SD-AODV como protocolo de encaminamiento (Caso 2). Si usando SD-AODV no aumenta tanto el retardo extremo a extremo para las conexiones tampoco lo hará ahora la variación del retardo o jitter porque con este protocolo de encaminamiento y gracias al balanceo de carga no aumenta tanto la congestión de la red.

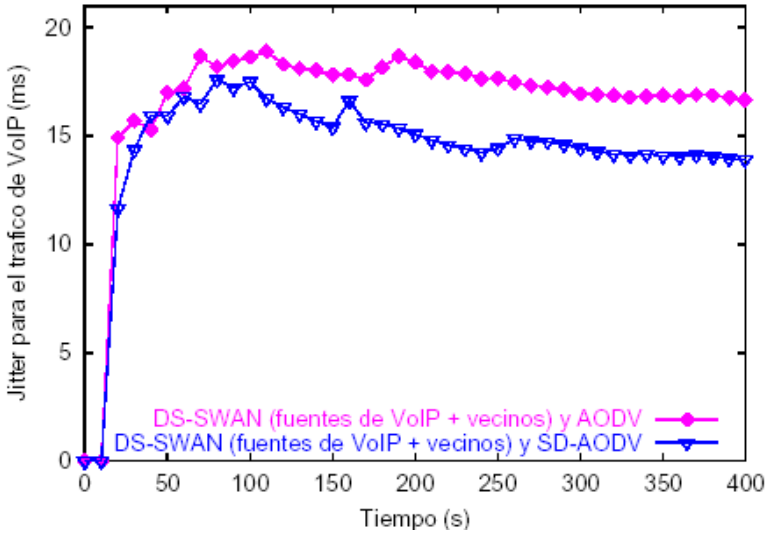



**Fig. 5.11.** Jitter para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

La *Fig. 5.12* muestra el throughput para el tráfico CBR. Tal y como puede apreciarse, el throughput del tráfico CBR es muy parecido en ambos casos y, a pesar de que en ciertos momentos resulta preciso estrangular el tráfico best-effort para mantener los retardos extremo a extremo del tráfico de tiempo real, los sistemas logran recuperarse rápidamente. Nótese que tiene particular mérito que en el Caso 2 se mantenga el throughput del tráfico best-effort a pesar de la fuerte reducción de los retardos extremo a extremo para los flujos de VoIP. Esto es debido a que cuando se buscan nuevas rutas con SD-AODV se seleccionan para las conexiones CBR nodos intermedios menos congestionados que no perjudicarán de forma tan nociva al tráfico de VoIP y por lo tanto el tráfico best-effort no será estrangulado tan frecuentemente.

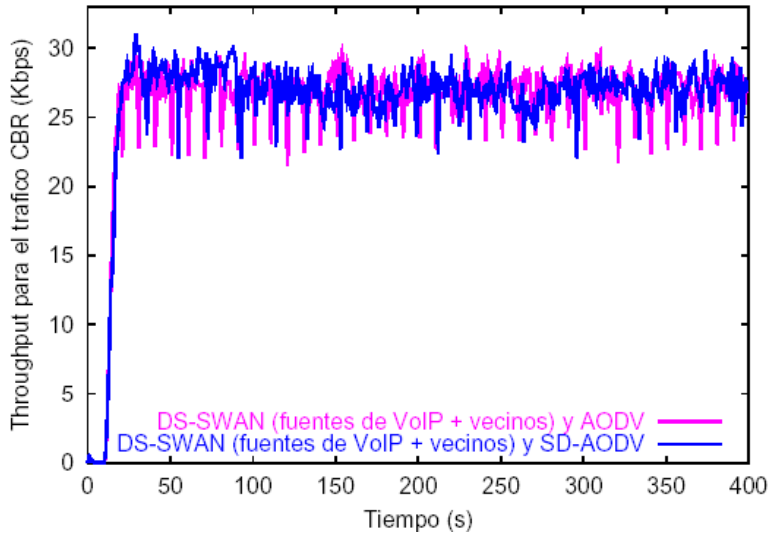



**Fig. 5.12.** Throughput para el tráfico best-effort: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

La *Fig. 5.13* representa las pérdidas de paquetes para el tráfico de VoIP en la red ad hoc. Se ha calculado mirando las pérdidas de paquetes en intervalos de diez segundos para cada una de las quince fuentes de VoIP de una simulación, tomando el valor máximo de pérdidas de entre todas las fuentes para cada intervalo y haciendo la media de los máximos para las cuarenta simulaciones efectuadas. El número de paquetes perdidos disminuye si se usa SD-AODV en vez de AODV como protocolo de encaminamiento porque SD-AODV balancea el tráfico de VoIP y CBR entre distintos nodos, reduciendo la congestión. Si comparamos el inicio de esta gráfica con la de la *Fig. 5.6*, que también representa la tasa de pérdida de paquetes en la red ad hoc para VoIP pero cuando de tráfico fluye en sentido contrario, se observa que en la *Fig. 5.13* la tasa de pérdidas es más abrupta desde el inicio de la simulación. La razón es que cuando el tráfico viaja desde la red ad hoc a la red IP fija el gateway debe competir como un nodo más de la red ad hoc junto a sus vecinos, por el acceso al medio,

motivo por el cual aumentan las pérdidas de paquetes. Cuando el tráfico fluye en sentido contrario esto no sucede.

Por otro lado, no se producen pérdidas de paquetes en el router frontera.

Por lo tanto, el porcentaje total de paquetes de VoIP perdidos (los que se pierden en la red ad hoc) es en ambos sistemas inferior al 5% de paquetes perdidos establecido para mantener una calidad de conversación para VoIP aceptable [155].

El número total de paquetes de VoIP perdidos (teniendo en cuenta las pérdidas de paquetes en la red ad hoc) es siempre menor si se usa SD-AODV como protocolo de encaminamiento.

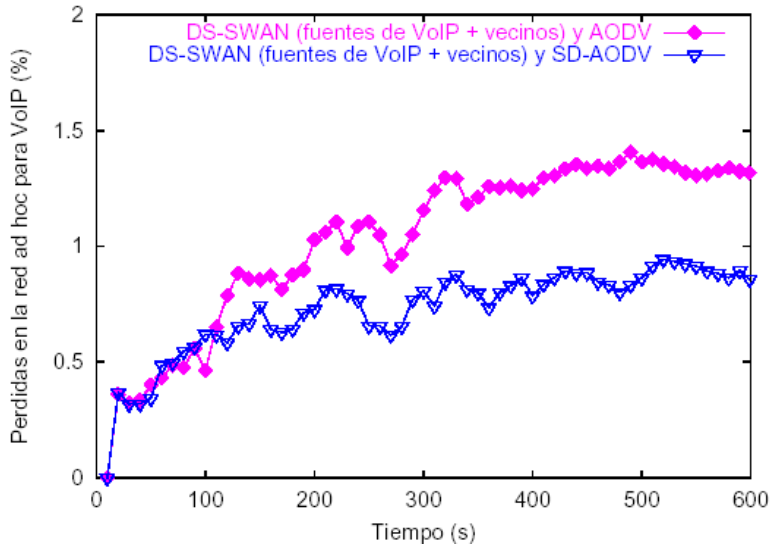



**Fig. 5.13.** Tasa de pérdida de paquetes en la red ad hoc para el tráfico de VoIP: DS-SWAN (fuentes de VoIP + vecinos) y AODV (Caso 1) vs. DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV (Caso 2).

De las dos versiones presentadas, la segunda versión (Caso 2: “DS-SWAN (fuentes de VoIP + vecinos) y SD-AODV”) supera a la primera en rendimiento, pues con esta versión es posible que SD-AODV colabore conjuntamente con DS-SWAN para mitigar las condiciones de congestión y reducir los problemas relacionados con la congestión. La combinación de ambos protocolos reduce no solamente los retardos extremo a extremo de los flujos de VoIP sino también el jitter y las pérdidas de paquetes para el tráfico de VoIP y mantiene el throughput del tráfico best-effort sin que sufra inanición. Por este motivo se reafirma la idea de que seleccionar el protocolo de encaminamiento SD-AODV para combinarlo con el modelo de interacción de calidad de servicio DS-SWAN es lo más adecuado para su implementación en una red ad hoc.

# *5.3 Conclusiones*

En el Capítulo 5 se han mostrado aquellos protocolos de encaminamiento que han sido desarrollados con el fin de poder conectar las redes ad hoc con redes fijas, introduciéndose un mecanismo básico para la interconexión de Internet y las redes ad hoc que basa su funcionamiento en el descubrimiento de gateways.

Seguidamente, se ha presentado una nueva contribución de esta tesis doctoral, "Desarrollo del protocolo de encaminamiento SD-AODV para la mejora de la supervivencia y la cooperación en el mantenimiento de la calidad de servicio en redes ad hoc conectadas a redes fijas", procediéndose a realizar una explicación teórica y a presentar los resultados detalladamente mediante simulaciones de la misma.

Se han presentado los resultados obtenidos a la hora de simular el protocolo de encaminamiento SD-AODV y compararlo con el protocolo AODV. En las simulaciones realizadas, se ha considerado un escenario en el cual una red ad hoc está conectada a una red IP fija y se han establecido una serie de conexiones de tráfico best-effort CBR y de tráfico de tiempo real VoIP (VBR) entre alguno de los nodos móviles de la red ad hoc y alguno de los hosts localizados en la red fija. Como modelo de calidad de servicio en la red fija se ha usado la arquitectura DiffServ. Como modelo de calidad de servicio en la red ad hoc se ha usado DS-SWAN, que es un modelo desarrollado en el marco de esta tesis doctoral, que interacciona con la arquitectura DiffServ para mejorar la diferenciación de servicios extremo a extremo *(Véase la sección 4.2.1.1 DS-SWAN (Differentiated Services-SWAN) para tráfico enviado desde la red ad hoc hacia la red fija, pág. 151)*.

En las simulaciones realizadas, se ha analizado la transmisión de tráfico best-effort y de tiempo real desde la red ad hoc hacia la red fija, y, posteriormente, se ha invertido el sentido del tráfico (desde la red fija hacia la red ad hoc) con el fin de comprobar si, en este caso más problemático, donde el gateway corría el riesgo de convertirse en un nodo permanentemente congestionado, podía garantizarse el buen funcionamiento del protocolo de encaminamiento desarrollado.

Los resultados para tráfico enviado en cualquiera de los dos sentidos han demostrado que la transmisión del tráfico de VoIP es realmente exitosa.

SD-AODV colabora conjuntamente con el modelo DS-SWAN para mitigar las condiciones de congestión y reducir los problemas relacionados con la congestión, beneficiándose con ello las aplicaciones de tiempo real, que son capaces de mantener sus requisitos de calidad de servicio. La combinación de ambos protocolos reduce no solamente los retardos extremo a extremo de los flujos de VoIP sino también el jitter y las pérdidas de paquetes para el tráfico de VoIP y mantiene el throughput del tráfico

best-effort sin que sufra inanición. Por tanto, SD-AODV ha demostrado ser el protocolo de encaminamiento más adecuado para ser combinado con el modelo DS-SWAN en el mantenimiento de los parámetros de calidad de servicio para los flujos de tiempo real entre una red ad hoc y una red IP fija.

Asimismo, ha quedado demostrado que con SD-AODV se reduce el consumo de energía de los nodos congestionados; así estos nodos no extinguirán sus recursos de energía antes de tiempo y el problema de que la red sufra una partición se verá reducido.

En todas las simulaciones efectuadas se ha observado que cuando se usa el protocolo de encaminamiento SD-AODV y hay fuentes de VoIP que deben buscar una nueva ruta para sus flujos debido a la movilidad, no se quedan sin poderla encontrar. Sin embargo, esto no siempre tiene porqué ser así, pues este factor está relacionado con el número de nodos de la red ad hoc (cuanto mayor sea este número, mayor será la probabilidad de que con SD-AODV las fuentes de VoIP puedan encontrar una nueva ruta), el número de fuentes de VoIP, la congestión, etc.

# 6 Conclusiones

En esta tesis doctoral se ha pretendido hacer factible una comunicación en las mejores condiciones posibles entre una red ad hoc y una red IP fija. Para lograr este propósito, se han abordado dos objetivos distintos: Proporcionar calidad de servicio entre una red ad hoc y una red IP fija y mejorar la supervivencia de una red ad hoc conectada a una red IP fija.

Para poder alcanzar estos objetivos se ha realizado un estudio exhaustivo tanto de los modelos de calidad de servicio como de los protocolos de encaminamiento existentes en redes ad hoc aisladas. Fruto de dicho estudio ha surgido una primera contribución consistente en el diseño e implementación del protocolo de encaminamiento SEADSR para la mejora de la supervivencia en redes ad hoc aisladas. Se ha conseguido demostrar mediante simulaciones que SEADSR supera al estándar DSR en relación a la supervivencia de red sin que se vea reducida la capacidad del sistema.

A partir de esta base se ha podido analizar la diferenciación de servicios en redes ad hoc conectadas a redes fijas; como consecuencia de esta investigación ha sido posible desarrollar el modelo de calidad de servicio DS-SWAN para la provisión de calidad de servicio entre redes ad hoc y redes IP fijas. Se ha demostrado que con DS-SWAN es posible mantener la diferenciación de servicios entre una red IP que use la arquitectura DiffServ y una red ad hoc, algo que no sucede cuando se usa el anteriormente propuesto modelo SWAN para la red ad hoc. Las simulaciones realizadas han demostrado la eficiencia y rendimiento del protocolo DS-SWAN bajo una variedad representativa de cargas de tráfico best-effort y usando diferentes modelos de tráfico best-effort (TCP o bien CBR). Se ha probado su escalabilidad para redes ad hoc de distinto tamaño con respecto al número de fuentes de VoIP y la movilidad. El estudio de tráfico en sentido inverso (hacia la red ad hoc) ha resultado ser muy interesante porque el protocolo CSMA/CA debe garantizar un acceso sin prioridades al medio inalámbrico y este funcionamiento parecía que podía causar problemas a la hora de aplicar nuestro modelo para la diferenciación de servicios; las simulaciones realizadas han demostrado el buen funcionamiento de DS-SWAN incluso en los casos más problemáticos.

Por último, se ha logrado mejorar la supervivencia de una red ad hoc conectada a una red basada en infraestructura mediante el diseño e implementación del protocolo de encaminamiento SD-AODV. SD-AODV colabora conjuntamente con el modelo DS-SWAN para mitigar las condiciones de congestión y reducir los problemas

relacionados con la congestión, beneficiándose con ello las aplicaciones de tiempo real, que son capaces de mantener sus requisitos de calidad de servicio. La combinación de ambos protocolos reduce no solamente los retardos extremo a extremo de los flujos de VoIP sino también el jitter y las pérdidas de paquetes para el tráfico de VoIP y mantiene el throughput del tráfico best-effort sin que sufra inanición. Se ha demostrado que con este protocolo de encaminamiento es posible no sólo alargar la supervivencia de la red ad hoc, sino también disminuir la congestión y mejorar la diferenciación de servicios entre ambas redes. Las simulaciones realizadas han demostrado la efectividad de dicho protocolo cuando el tráfico fluía desde la red ad hoc hacia la red basada en infraestructura y también en sentido inverso.

Las contribuciones presentadas en esta tesis doctoral tienen una singular importancia, pues hasta la fecha no se ha desarrollado ningún modelo de calidad de servicio que permita la interacción y favorezca la cooperación entre una red ad hoc y una red IP fija con el fin de proporcionar calidad de servicio extremo a extremo.

A partir del trabajo de investigación realizado, se deducen las siguientes conclusiones específicas:

- Los modelos de calidad de servicio desarrollados para redes ad hoc aisladas no sirven para proporcionar calidad de servicio entre una red ad hoc y una red IP fija, porque tratan de diferenciar servicios para el tráfico dentro de cada uno de sus dominios, pero no van más allá. En esta tesis doctoral se ha conseguido demostrar mediante la utilización de simulaciones que si existe un modelo de calidad de servicio en la red ad hoc (SWAN) y otro modelo de calidad de servicio en la red IP fija (DiffServ) y estos modelos no colaboran entre sí, no será posible el mantenimiento de los parámetros de calidad de servicio que una aplicación de tiempo real requiera.
- Para poder diferenciar servicios entre una red ad hoc y una red IP fija resulta indispensable contar con un modelo de calidad de servicio que permita la cooperación entre ambas redes. En esta tesis doctoral se ha diseñado y evaluado un modelo de calidad de servicio (DS-SWAN) que sí que funciona correctamente porque se basa en la cooperación para el mantenimiento de la calidad de servicio entre las redes ad hoc y las redes IP fijas.
- Para poder diferenciar servicios entre una red ad hoc y una red IP fija resulta imprescindible mapear las clases de servicio del modelo de calidad de servicio incluido en la red ad hoc a las clases de servicio del modelo de calidad de servicio incluido en la red IP fija. En esta tesis doctoral ha quedado justificada la necesidad de que las clases de servicio del modelo SWAN (VoIP

y best-effort CBR) sean mapeadas a las clases de servicio del modelo DiffServ (PHB EF y best-effort).

- Debe establecerse una negociación de los parámetros de calidad de servicio para que después dichos parámetros puedan ser garantizados y pueda ofrecerse calidad de servicio entre la red ad hoc y la red basada en infraestructura. En las simulaciones realizadas, el modelo SWAN comprueba mediante el control de admisión si tiene recursos de ancho de banda suficientes para poder establecer una sesión de tiempo real y también si estos recursos se mantienen mientras dura la sesión de tiempo real. En DiffServ también se comprueba si los flujos de paquetes pertenecientes a las distintas clases cumplen o no con el perfil acordado para que puedan contar con aquellos recursos que han demandado. También el protocolo DS-SWAN realiza una negociación de los parámetros de calidad de servicio comprobando la tasa de pérdidas de paquetes de tiempo real y el retardo de dichos paquetes durante toda la simulación y mantiene en todo momento unos valores adecuados para estos parámetros de calidad de servicio de manera adaptativa.
- Para poder diferenciar servicios en una red ad hoc aislada o bien entre una red ad hoc y una red IP fija resulta indispensable contar con un modelo de calidad de servicio que esté basado en un diseño cross layer. El diseño cross layer promueve la interacción entre capas; se trata de que el mayor número de capas posible unan sus esfuerzos con un objetivo común: La diferenciación de servicios. En esta tesis doctoral ha quedado demostrado que resulta absolutamente necesario apoyarse en este planteamiento para lograr el objetivo expuesto.
- Los protocolos de encaminamiento guardan una estrecha relación con la provisión de calidad de servicio. El rendimiento de la red depende estrechamente de la velocidad a la cual los protocolos de encaminamiento pueden recalcular nuevas rutas entre pares fuente-destino después de haberse producido cambios en la topología de red (si no existe ninguna ruta alternativa almacenada en la caché del nodo fuente). El retardo en el cálculo de nuevas rutas tendrá un cierto impacto en la calidad de servicio ofrecida a las sesiones de tiempo real establecidas. En esta tesis doctoral ha quedado demostrado que cuando el protocolo de encaminamiento de la red ad hoc (SD-AODV) conectada a la red IP fija basa su funcionamiento en criterios

relacionados con la provisión de calidad de servicio, mejora la diferenciación de servicios entre ambas redes.

- Los protocolos de encaminamiento guardan una estrecha relación con el tiempo de vida de las baterías. En esta tesis doctoral se ha puesto de manifiesto que se puede alargar la supervivencia de la red ad hoc si se emplea en una red ad hoc aislada un protocolo de encaminamiento (SEADSR) que tenga en cuenta a la hora de seleccionar sus rutas la capacidad de las baterías de los nodos móviles. También se ha comprobado lo mismo usando en una ad hoc conectada a una red IP fija un protocolo de encaminamiento (SD-AODV) que tenga en cuenta a la hora de seleccionar sus rutas la congestión existente.

# *7 Líneas futuras*

Los resultados obtenidos en este trabajo de investigación han demostrado lo beneficioso que resulta aplicar modelos de calidad de servicio que favorezcan la cooperación entre redes ad hoc y redes fijas para mejorar la diferenciación de servicios extremo a extremo. También han demostrado que con la ayuda de los protocolos de encaminamiento es posible mejorar la supervivencia en redes ad hoc aisladas o conectadas a redes fijas.

Sin embargo, quedan por estudiar ciertos aspectos, que mantienen o bien abren nuevas líneas de investigación:

- Se sugiere para ello continuar la labor de búsqueda de nuevos modelos de calidad de servicio que promuevan la interacción entre redes ad hoc y redes fijas para la mejora de la diferenciación de servicios. La cantidad de posibilidades y combinaciones que puedan surgir es enorme y resultará muy provechosa.
- Además resultaría de gran utilidad efectuar más simulaciones del protocolo de encaminamiento SD-AODV para el caso donde hubiera fuentes de VoIP que al buscar una nueva ruta debido a la movilidad se quedaran sin poderla encontrar. En las nuevas simulaciones debería de comprobarse la escalabilidad del protocolo con respecto al del número de nodos, de fuentes de VoIP, la congestión, etc, y comprobar cómo influyen estos factores en la búsqueda de rutas.
- También resultaría de gran valor estudiar en mayor profundidad el tema de la selección de un gateway de entre los existentes para la conexión de una red ad hoc y una red IP fija y seleccionar siempre el mejor de acuerdo con alguna condición predefinida.
- Otro aspecto muy interesante puede consistir en continuar desarrollando nuevos protocolos de encaminamiento que alarguen la supervivencia de la red ad hoc y en concreto estudiar la aplicación del algoritmo SEADSR como protocolo de encaminamiento para alargar la supervivencia de una red ad hoc conectada a una red IP fija.

# *Bibliografía*

[1] D. Remondo and I. G. Niemegeers, “Ad hoc networking in future wireless communications”, *Computer Communications*, vol 26, no. 1, Jan. 2003, pp. 36-40.

[2] K. Shohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie, "Protocols for Self-Organization of a Wireless Sensor Network", IEEE Personal Communications, Oct. 2000, pp.16-27.

[3] QoS:http://www.cisco.com/warp/public/732/net_enabled/end-to-end.html.

[4] QoS:http://www.cdt.luth.se/utbildning.direkt/multimedia/slides/fluckiger/1998/part_III/chapter17/all.html.

[5] R. Braden, D. Clark, and S. Shenker, “Integrated services in the internet architecture: an overview”, Request for Comments (Informational) 1633, Internet Engineering Task Force, June 1994.

[6] D. D. Clark, S. Shenker, and L. Zhang, ”Supporting real-time applications in an integrated services packet network: architecture and mechanism”, In SIGCOMM Symposium on Communications Architectures and Protocols, pages 14–26, Baltimore, Maryland, August 1992. ACM. Computer Communication Review, Volume 22, Number 4.

[7] M. C. Domingo and D. Remondo, “State of Art in Multi-Hop Ad Hoc Networks”, EUNICE Summer School on Next Generation Networks, Eunice’2003, Budapest, Hungary, Sept. 2003.

[8] B. Li, “QoS-aware adaptive services in mobile ad-hoc networks”, in: Proceedings of the Ninth IEEE International Workshop on Quality of Service (IWQoS 2001), Karlsruhe, Germany, Lecture Notes in Computer Science, vol. 2092, Springer, Berlin, 2001, pp. 251–268.

[9] Bluetooth, http://www.bluetooth.com.

[10] IEEE P802.11, The Working Group for Wireless LANs, http://grouper.ieee.org/groups/802/11/.

[11] HiperLAN/1, http://portal.etsi.org/bran/kta/Hiperlan/hiperlan1tech.asp.

[12] HiperLAN/2, http://www.hiperlan2.com/.

[13] HiperLAN/3, http://www.mpirical.com/companion/Multi_Tech/HiperAccess.htm.

[14] HiperlLAN/4, http://www.mpirical.com/companion/Multi_Tech/HiperLink.htm.

[15] A. Grado-Caffaro y M. Grado-Caffaro, “Comunicaciones Inalámbricas de Banda Ultra Ancha”, en Comunicaciones World, Noviembre 2001, p. 114-115.

[16] Ultra-wideband Tutorial, http://www.ieee802.org/802_tutorials/.

[17] IEEE Standard 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications (1999).

[18] C. S. Choi and C. W. Choi, “DSR Based Bluetooth Scatternet”, ITC-CSCC 2002, Phuket, Thailand, July 2002.

[19] FCC News: http://www.hiperlan2.com/newsdocs/member/n111303e.pdf.

[20] FCC News: http://www.uwb.org/news/articles/FCCRelease021303.pdf.

[21] A. León-García e I. Widjaja, “Redes de Comunicación: Conceptos fundamentales y arquitecturas básicas”, ed. Mc Graw-Hill, 2002.

[22] D. Remondo, "Tutorial on Wireless Ad Hoc Networks," in Proc. of the 2nd International Working Conf. on Performance Modelling and Evaluation of Heterogeneous Networks (HET-NETs '04), Ilkley, West Yorkshire, U.K., July 26-28, 2004.

[23] J. H. Schiller, “Mobile Communications”, *ed. Addison Wesley Professional*, 2000.

[24] K. Xu, M. Gerla, S. Bae, “Effectiveness of RTS/CTS Handshake in IEEE 802.11 Based Ad Hoc Networks”, Ad Hoc Networks Journal, Volume 1, Issue 1, July 2003, pp. 107 – 123.

[25] C-K Toh, “Ad Hoc Mobile Wireless Networks”, Prentice Hall, 2002.

[26] Q. Ni, L. Romdhani and T. Turletti, "A Survey of QoS Enhancements for IEEE 802.11 Wireless LAN", in Journal of Wireless Communication and Mobile Computing (JWCMC), 2004; 4: 1-20.

[27] A. Köpsel and A. Wolisz, “Voice transmission in an IEEE 802.11 WLAN based access network”, Proceedings of the 4th ACM international workshop on Wireless mobile multimedia, Rome, Italy, 2001.

[28] W. Pattara-Atikom, P. Krishnamurthy and S. Banerjee, “Distributed Mechanisms for Quality of Service in Wireless LANs”, IEEE Wireless Communications Magazine, vol. 10, No.3, pp. 26-34, June 2003.

[29] I. Aad and C. Castelluccia, “Remarks on per-flow differentiation In IEEE 802.11”, European Wireless 2002, Florence, Italy, February 25th-28th, 2002.

[30] I. Aad and C. Castelluccia, “Introducing service differentiation into IEEE 802.11” , in Proceedings of ISCC2000, Antibes, France, July 2000.

[31] I. Aad and C. Castelluccia, “Differentiation mechanisms for IEEE 802.11”, in Proceedings of IEEE Infocom 2001, Anchorage – Alaska, April 2001.

[32] I. Aad and C. Castelluccia, “Priorities in WLANs", Computer Networks, 41/4, February, 2003, p. 505-526.

[33] D-J. Deng and R-S. Chang, “A priority scheme for IEEE 802.11 DCF access method”, *IEICE Transactions on Communications*, E82-B(1), January 1999.

[34] A. Lindgren, A. Almquist, and Olov Schelén, "Evaluation of Quality of Service schemes for IEEE 802.11 wireless LANs", In Proceedings of the 26th Annual IEEE Conference on Local Computer Networks (LCN 2001), Tampa, Florida, USA, November 2001.

[35] A. Lindgren, A. Almquist and O. Schelén, “Quality of Service Schemes for IEEE 802.11 - A Simulation Study”, In Proceedings of the Ninth International Workshop on Quality of Service (IWQoS 2001), Karlsruhe, Germany, June 6-8, 2001.

[36] M. Benveniste, G. Chesson, M. Hoeben, A. Singla, H. Teunissen, and M. Wentink, “EDCF proposed draft text. IEEE working document 802.11-01/131r1”, March 2001.

[37] S. Choi, J. del Prado, S. Shankar N, and S. Mangold, “IEEE 802.11e Contention-Based Channel Access (EDCF) Performance Evaluation,” in Proc. IEEE ICC’03, Anchorage, Alaska, USA, May 2003.

[38] D. Gu and J. Zhang, "Evaluation of EDCF Mechanism for QoS in IEEE 802.11 Wireless Networks", World Wireless Congress (WWC), San Francisco, USA, May 2003 (WWC 2003).

[39] S. Mangold, S. Choi, P. May O. Klein G. Hiertz and L. Stibor, “IEEE 802.11e Wireless LAN for Quality of Service”, In: European Wireless, Florence, Italy, 26-28 February 2002.

[40] A. Lindgren, A. Almquist and O. Schelén, “Quality of Service Schemes for IEEE 802.11 Wireless LANs - An Evaluation”, In the Special Issue of the Journal of Special Topics in Mobile Networking and Applications (MONET) on Performance Evaluation of QoS Architectures in Mobile Networks, Volume 8, Number 3, June 2003.

[41] M. Benveniste, “TCMA Proposed Draft Text,” Tech. rep., IEEE wkg. doc. 802.11-01/117r2, 2001.

[42] L. Romdhani, Q. Ni, and T. Turletti, "Adaptive EDCF: Enhanced Service Differentiation for IEEE 802.11 Wireless Ad Hoc Networks", IEEE WCNC'03 (Wireless Communications and Networking Conference), New Orleans, Louisiana, March 16-20, 2003.

[43] L. Romdhani, Q. Ni, and T. Turletti, “AEDCF: enhanced service differentiation for IEEE 802.11 wireless ad-hoc networks”, *INRIA Research Report No. 4544*, 2002.

[44] J. L. Sobrinho and A. S. Krishnakumar, “Quality-of-Service in ad hoc carrier sense multiple access networks”, IEEE Journal on Selected Areas in Communications, 17(8):1353–1368, August 1999.

[45] J. L. Sobrinho and A. S. Krishnakumar, "Real-time traffic over the IEEE 802.11 MAC", Bell Labs Technical Journal, vol. 1, no 2, pp. 172-187, Autumn 1996.

[46] W. Pattara-atikom, S. Banerjee and P. Krishnamurthy, “Starvation Prevention and Quality of Service in Wireless LANs”, Proceedings of the IEEE 5th Intl. Symposium on Wireless Personal Multimedia Communications (WPMC), Honolulu, Hawaii, October 2002.

[47] M. Shreedhar and G. Varghese, “Efficient Fair Queuing Using Deficit Round-robin”, IEEE/ACM Trans. Net., vol. 4, no. 3, 1996, pp. 375–85.

[48] W. Pattara-Atikom, P. Krishnamurthy, S. Banerjee, "Comparison of distributed fair QoS mechanisms in wireless LANs", GLOBECOM 2003 - IEEE Global Telecommunications Conference, vol. 22, no. 1, San Francisco, USA, Dec 2003, pp. 553-557.

[49] N. H. Vaidya, P. Bahl, and S. Gupta, “Distributed fair scheduling in a wireless LAN”, *In Sixth Annual International Conference on Mobile Computing and Networking*, Boston, Massachusetts, August 2000.

[50] S. R. Golestani, "A self-clocked fair queueing scheme for broadband applications," in Proc. IEEE INFOCOM, Toronto, Ontario, Canada, Jun. 1994, pp. 636-646.

[51] A. Banchs and X. Pérez, “Providing Throughout Guarantees in IEEE 802.11 Wireless LAN”, in Proceedings of IEEE Wireless Communications and Networking Conference (WCNC 2002), Orlando, Florida, March 2002.

[52] A. Banchs, M. Radimirsch and X. Pérez, “Assured and expedited forwarding extensions for IEEE 802.11 Wireless LAN”, Quality of Service, 2002. Tenth IEEE International Workshop on , Miami Beach, USA, 2002.

[53] A. Banchs and X. Perez, “Distributed weighted fair queuing in 802.11 Wireless LAN”, in Proc. IEEE ICC ’02, pp. 3121–3127, College Park, USA, May 2002.

[54] S-B. Lee, G-S. Ahn, X. Zhang and A.T. Campbell, "INSIGNIA: An IP-Based Quality of Service Framework for Mobile Ad Hoc Networks", Journal of Parallel and Distributed Computing (Academic Press), Special issue on Wireless and Mobile Computing and Communications, Vol. 60 No. 4 pg. 374-406, April 2000.

[55] T. Bheemarjuna Reddy, I. Karthigeyan, B.S. Manoj and C. Siva Ram Murthy, “Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions”, In Press, 2 July 2004, Elsevier Ad Hoc Networks.

[56] I. Chlamtac, M. Conti and J. Liu, "Mobile Ad hoc Networking: Imperatives and Challenges", Ad Hoc Network Journal, Vol.1 N.1, January-February-March, 2003.

[57] S-B. Lee and A. T. Campbell, "INSIGNIA: In-band Signaling Support for QOS Mobile Ad Hoc Networks", in 5th International Workshop on Mobile Multimedia Communications (MoMuC, 98), Berlin, Germany, October, 1998.

[58] S-B. Lee, G-S Ahn, X. Zhang and A. T. Campbell, "Evaluation of the INSIGNIA Signaling System", 8th IFIP International Conference on High Performance Networking (Networking 2000), Paris, France, May, 2000.

[59] S-B. Lee, G-S. Ahn and A.T. Campbell, "Improving UDP and TCP Performance in Mobile Ad Hoc Networks with INSIGNIA", June 2001, IEEE Communication Magazine.

[60] S-B. Lee, G-S. Ahn, X. Zhang and A. T. Campbell, "INSIGNIA", Internet Draft, draft-ietf-manet-insignia-01.txt, Work in Progress, October 1999.

[61] H. Xiao, K. C. Chua and W. K. G. Seah, “Quality of Service Models for Ad Hoc Wireless Networks”, The handbook of ad hoc wireless networks, 2003, pp. 467-482.

[62] H. Xiao, W. K. G. Seah, A. Lo, and K. C. Chua, “A Flexible Quality of Service Model for Mobile Ad-Hoc Networks”, In Proc. of the Vehicular Technology Conference 2000 -- Spring, Tokyo, Japan, 2000, pp. 445–449.

[63] H. Xiao, K. C. Chua, K. G. Seah, and A. Lo, “On service prioritization in mobile ad-hoc networks”, In IEEE ICC 2001, Helsinki, Finland, June 2001.

[64] H. Xiao, “A Flexible Quality of Service Model for Mobile Ad Hoc Networks”, Ph.D thesis, National University of Singapore, Mar. 2002.

[65] H. Xiao, K. G. Seah, A. Lo, and K. C. Chua, “On service differentiation in multihop wireless networks”, In ITC Specialist Seminar on Mobile Systems and Mobility, pages 1-12, Lillehammer, Norway, March 2000.

[66] K.C. Chua, H. Xiao and K.G. Seah, “Relative service differentiation for mobile ad hoc networks”, IEEE Wireless Communications and Networks Conference, New Orleans, USA, 2003.

[67] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, “An architecture for differentiated service”, *Request for Comments (Informational) 2475*, Internet Engineering Task Force, December 1998.

[68] K. Nichols, S. Blake, F. Baker and D. Black, “Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers”, *Request for Comments (Proposed Standard) 2474, Internet Engineering Task Force*, December 1998.

[69] K. Kilkki, "Differentiated Services for the Internet", New Riders Publishing, June 1999, 384 pages.

[70] K. Nichols, V. Jacobson and L. Zhang, “A two-bit Differentiated Services Architecture for the Internet”, *RFC-2638*, July-1999.

[71] Y. Bernet, “Networking Quality of Service and Windows Operating Systems”, QUE Publishing, November 2000, 702 pages.

[72] V. Jacobson, K. Nichols and K. Poduri, “An expedited forwarding PHB”, *Request for Comments (Proposed Standard) 2598, Internet Engineering Task Force*, June 1999.

[73] Davie, B. et al., “An Expedited Forwarding PHB”, *RFC 3246*, (2002).

[74] J. Heinanen, F. Baker, W. Weiss and J. Wroclawski, “Assured Forwarding PHB Group”, *RFC 2597*, June 1999.

[75] G. Armitage, "Quality of Service in IP Networks", Pearson Higher Education, April 2000, 309 pages.

[76] D. D. Clark and W. Fang, “Explicit allocation of best-effort packet delivery service”, *IEEE/ACM Transactions on Networking, 6(4):362–373*, August 1998.

[77] J. Ibanez and K. Nichols, “Preliminary Simulation Evaluation of an Assured Service”, *Internet Draft, draft-ibanez-diffserv-assured-eval-00.txt*, August 1998.

[78] Cisco Systems, “DiffServ-The Scalable End-to-End QoS Model”, *White Paper*, March 2001.

[79] K. Wu and J. Harms, "QoS Support in Mobile Ad-hoc Networks," *Crossing Boundaries- the GSA Journal of University of Alberta*, Vol. 1, No. 1, Nov. 2001, pp.92- 106.

[80] H. Arora and H. Sethu, “A Simulation Study of the Feasibility of Differentiated Services Framework for QoS in Mobile Ad-hoc Networks”, *Applied Telecommunications Symposium*, San Diego, California, USA, Apr. 2002.

[81] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance", *IEEE/ACM Transactions on Networking*, no. 4, August 1993 pp. 397-413.

[82] H. Arora, "Towards achieving QoS guarantees in mobile ad hoc networks", Masters Thesis, Drexel University, Department of Computer Science, Philadelphia, PA, November 2003.

[83] D. D. Clark and W. Fang, "Explicit Allocation of Best Effort Packet Delivery Service", IEEE/ACM Transactions on Networking, August 1998, Vol 6. No. 4, pp. 362-373.

[84] W. Zhao, D. Olshefski and H. Schulzrinne, “Internet quality of service: An overview”, Technical Report CUCS-003-00, Columbia Univ., Computer Science Dept., Feb. 2000.

[85] G.-S. Ahn, A. T. Campbell, A. Veres and L.-H. Sun, "SWAN", draft-ahn-swan-manet-00.txt, *Work in Progress*, February 2003.

[86] G.-S. Ahn, A. T. Campbell, A. Veres and L.-H. Sun, "Supporting Service Differentiation for Real-Time and Best Effort Traffic in Stateless Wireless Ad Hoc Networks (SWAN)", *IEEE Transactions on Mobile Computing*, September 2002.

[87] A. Veres, A. T. Campbell, M. Barry and L.-H. Sun, "Supporting Service Differentiation in Wireless Packet Networks Using Distributed Control", *IEEE Journal of Selected Areas in Communications (JSAC)*, *Special Issue on Mobility and Resource Manegement in Next-Generation Wireless Systems*, Vol. 19, No. 10, pp. 2094-2104, October 2001.

[88] M. Barry, A. T. Campbell, and A. Veres , "Distributed Control Algorithms for Service Differentiation in Wireless Packet Networks", *Proc. IEEE INFOCOM'2001*, Anchorage, Alaska, April 2001.

[89] M. C. Domingo y D. Remondo, "Diferenciación de servicios por clase en redes inalámbricas", URSI 2003, XVIII Simposium Nacional de la Unión Científica Internacional de Radio, Coruña, Sept. 2003.

[90] M. C. Domingo and D. Remondo, "Per-Flow Service Differentiation via Virtual MAC", WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, INRIA Sophia-Antipolis, France, March 2003.

[91] G.-S. Ahn, A. T. Campbell, A. Veres and L.-H. Sun, "SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks", *Proc. IEEE INFOCOM'2002*, New York, USA, June 2002.

[92] H. Arora, Ll. Greenwald, U. Rao and J. Novatnack, "Performance comparison and analysis of two QoS schemes: SWAN and Diffserv", Drexel Research Day Honorable Mention, April 2003.

[93] S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", RFC2501, Request for Comments, January 1999.

[94] M.Abolhasan, T.A.Wysocki, and E.Dutkiewicz, "A Review of Routing Protocols for Mobile Ad hoc Networks", In Elsevier Journal of Ad hoc Networks, 2 (2004), 1-22.

[95] R. E. Bellman, "Dynamic Programming", Princeton University Press, Princeton, NJ 1957.

[96] L. R. Ford, D. R Fulkerson, "Flows in Networks", Princeton University Press, Princeton, NJ, 1962.

[97] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance vector routing (DSDV) for mobile computers", in Proc. ACM SIGCOMM'94, London, U. K., pp. 234–244.

[98] C. E. Perkins, "Ad Hoc Networking", Addison-Wesley, Boston, 2001.

[99] S. Basagni, M. Conti, S. Giordando, and I. Stojmenovic, "Mobile Ad Hoc Networking", IEEE Press & Wiley Inter-Science, 2004.

[100] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, L. Viennot, “Optimized link state routing protocol for ad hoc networks”, IEEE INMIC, Pakistan, 2001.

[101] J. Moy, “Link-state routing”, In Martha E. Steenstrup, editor, Routing in Communications Networks, pages 135 -- 157. Prentice Hall, 1995.

[102] P. Jacquet, P. Muhlethaler, A. Qayyum, “Optimized Link State Routing Protocol”, Internet Draft, draft-ietf-manetolsr-00.txt, November 1998.

[103] S. Keshav, An Engineering Approach to Computer Networking, Addison-Wesley Pub Co, 1997.

[104] D.B. Johnson and D.A. Maltz, “DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks,” in Ad Hoc Networking, C.E. Perkins, ed. Addison Wesley, 2001.

[105] D.B. Johnson, D.A. Maltz, Y. Hu and J.G. Jetcheva, “The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR),” http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt , 19 July 2004, IETF Internet Draft.

[106] Y.-C. Hu and D. B. Johnson, “Caching Strategies in On-Demand Routing Protocols for Wireless Ad Hoc Networks”, Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (MobiCom 2000), ACM, Boston, MA, August 2000.

[107] Y.-C. Hu and D. B. Johnson, “Implicit Source Routes for On-Demand Ad Hoc Network Routing”, Proceedings of the 2001 ACM International Symposium on Mobile Ad Hoc Networking & Computing (MobiHoc 2001), pp. 1-10, ACM, Long Beach, CA, October, 2001.

[108] C. E. Perkins and E. M. Royer, “Ad-Hoc On-Demand Distance Vector Protocol”, in C. E. Perkins (Ed.), Ad Hoc Networking, pp. 173-179, Addison-Wesley, 2000.

[109] E. M. Belding-Royer and C. E. Perkins, "Evolution and Future Directions of the Ad hoc On-Demand Distance Vector Routing Protocol", Ad hoc Networks Journal, 1(1), pp. 125-150, July 2003.

[110] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, "Ad Hoc On Demand Distance Vector (AODV) Routing", IETF Internet draft, draft-ietf-manet-aodv-13.txt, February 2003 (Work in Progress).

[111] C. E. Perkins, E. M. Belding-Royer, and S. R. Das. "Ad Hoc On Demand Distance Vector (AODV) Routing." IETF RFC 3561.

[112] C. E. Perkins, E. M. Belding-Royer, and I. Chakeres, "Ad Hoc On Demand Distance Vector (AODV) Routing", IETF Internet draft, draft-perkins-manet-aodvbis-00.txt, Oct 2003 (Work in Progress).

[113] C. E. Perkins, E. M. Belding-Royer and I. Chakeres, "Ad Hoc On Demand Distance Vector (AODV) Routing", IETF Internet draft, draft-perkins-manet-rfc3561bis-01.txt, July 2004 (Work in Progress).

[114] S. Gwalani, E. M. Belding-Royer, C. E. Perkins, “AODV-PA: AODV with Path Accumulation”, Next Generation Internet Symposium, held in conjunction with ICC, Anchorage, Alaska, May 2003.

[115] Z.J. Hass, R. Pearlman and P. Samar, “Zone routing protocol for ad-hoc Networks”, Internet Draft, draft-ietf-manet-zrp-04.txt, work in progress, 2002.

[116] M. R. Pearlman and Z.J. Haas, "Determining the Optimal Configuration for the Zone Routing Protocol", IEEE Journal on Selected Areas in Communication, 17 (8). pp. 1395-1414, 1999.

[117] S. Murthy and J. J. Garcia-Luna-Aceves, “An efficient routing protocol for wireless networks”, ACM Mobile Networks Applicat. J., Special Issue on Routing in Mobile Communication Networks, 1996.

[118] E. Gafni and D. D. Bertsekas, “Distributed algorithms for generating loop-free routes in networks with frequently changing topology”, IEEE Trans. Commun., vol. COMM-29, Jan. 1981.

[119] M. S. Corson and A. Ephremides, “A distributed routing algorithm for mobile wireless networks”, Wireless Networks, vol. 1, pp. 61–81, 1995.

[120] V. D. Park and M. S. Corson, “A highly adaptive distributed routing algorithm for mobile wireless networks”, in Proc. IEEE INFOCOM’97, Kobe, Japan, pp. 1405–1413.

[121] D. Johnson and D. Maltz, “Dynamic source routing in ad hoc wireless networks,” Mobile Computing, E. Imielinski and H. Korth, Eds. Norwell, MA: Kluwer, 1996.

[122] C.-K. Toh, “Associativity-based routing for ad-hoc mobile networks,” Wireless Personal Commun., vol. 4, pp. 103–139, 1997.

[123] R. Sivakumar, B. Das, and V. Bharghavan, “An improved spine-based infrastructure for routing in ad hoc networks,” in Proc. IEEE Symp. Computers and Communications, 1998.

[124] Z. J. Haas and M. R. Pearlman, “The performance of query control schemes for the zone routing protocol,” in Proc. ACM SIGCOMM’98, Vancouver, British Columbia, Canada.

[125] S. Chen and K. Nahrstedt, “Distributed quality-of-service routing in ad hoc networks”, IEEE Journal on Selected Areas in Communications 17 (8) (1999) 1488–1504.

[126] S. Chen and K. Nahrstedt, “Distributed qos routing with imprecise state information”, in Proceedings of 7th IEEE International Conference on Computer, Communications and Networks, Lafayette, LA, Oct. 1998, pp. 614–621.

[127] C.E. Perkins, E.M. Royer, S.R. Das, “Quality of Service for Ad Hoc On-Demand Distance Vector Routing” (work in progress), IETF Internet Draft, draft-ietf-manet-aodvqos-00.txt, July 2000.

[128] Z.-Y. Demetrios, “A Glance at Quality of Services in Mobile Ad-Hoc Networks”, Final Research Report for CS260 – Seminar in Mobile Ad Hoc Networks, Fall 2001.

[129] C.E. Perkins and E.M. Royer, “Quality of Service for Ad Hoc On-Demand Distance Vector Routing” (work in progress), IETF Internet Draft, draft-perkins-manet-aodvqos-02.txt, October 2003.

[130] C-K. Toh, “Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks”, IEEE Communications Magazine, Vol. 39, No. 6, June 2001.

[131] C-K. Toh, H. Cobb and D. A Scott, “Performance Evaluation of Battery-Life-Aware Routing Schemes for Wireless Ad Hoc Networks”, Proceedings of IEEE International Conference on Communications (IEEE ICC), Helsinki, Finland, June 2001.

[132] S. Singh and C.S. Raghavendra, “Power-Aware Routing in Mobile Ad Hoc Networks,” Proc. of MobiCom’98, Dallas, Texas, U.S.A., Oct. 1998.

[133] A. Misra and S. Banerjee, “MRPC: Maximizing Network Lifetime for Reliable Routing in Wireless Environments”, IEEE Wireless Commun. and Networking Conf. (WCNC), Orlando, Florida, U.S.A., March 2002.

[134] S. Banerjee and A. Misra, “Minimum Energy Paths for Reliable Communication in Multi-hop Wireless Networks,” Mobihoc’02, Lausanne, Switzerland, June 2002.

[135] B. Chen, K. Jamieson, H. Balakrishnan and R. Morris, “Span: An energy-effcient coordination algorithm for topology maintenance in ad hoc wireless networks”, ACM Wireless Networks Journal, 8(5):481-494, September 2002.

[136] J. Wu, F. Dai, M. Gao and I. Stojmenovic, “On calculating power-aware connected dominating sets for efficient routing in ad hoc wireless networks”, IEEE/KICS Journal of Communications and Networks, 4(1):59-70, March 2002.

[137] Y. Xu, J. Heidemann and D. Estrin, “Adaptive Energy-Conserving Routing for Multihop Ad Hoc Networks”, Research Report 527, USC/Information Sciences Institute, October 2000.

[138] Y. Xu, J. Heidemann and D. Estrin, “Geography-informed energy conservation for ad hoc routing”, In Proc. of 7th Annual International Conference on Mobile Computing and Networking, Rome, Italy, pages 70-84, July 2001.

[139] S. Singh. and C.S. Raghavendra, “PAMAS-Power Aware Multi-Access protocol with Signalling for Ad hoc Networks”, ACM Comm. Review, Jul. 1998.

[140] J. C. Cano and P. Manzoni, “A performance comparison of energy consumption for Mobile Ad Hoc Network routing protocols”, in: Proc. 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, San Francisco, USA, Aug. 2000, pp. 57 –64.

[141] J. Broch, D. A. Maltz, D. B. Johnson, Y-C. Hu, and J. Jetcheva. “A performance comparison of multi-hop wireless ad hoc networks.”, in: Proc. of the 4th Int. Conference on Mobile Computing and Networking (ACM MOBICOM’98), Dallas, Texas, USA, October 1998, pp. 85-97.

[142] A. Boukerche, “Performance comparison and analysis of ad hoc routing algorithms”, IEEE International Conference on Performance, Computing, and Communications, 4-6 April 2001, pp.171 – 178.

[143] C. Perkins, E. Royer, S. Das and M. Marina, “Performance of two on-demand Routing Protocols for Ad-hoc Networks”, IEEE Personal Communications, February 2001, pp. 16-28.

[144] S. R. Das, C. E. Perkins and E. M. Royer, "Performance comparison of two on-demand routing protocols for ad hoc networks", in Proc. INFOCOM 2000 Conf., vol. 1, Tel-Aviv, Israel,Mar. 2000, pp. 3-12.

[145] D. Sun, H. Man, “TCP flow-based performance analysis of two on-demand routing protocols for mobile ad hoc networks”, Vehicular Technology Conference, Rhodes, Greece, vol.1, pp. 272 – 275, 2001.

[146] W. Yu and J. Lee, “DSR-based Energy-aware Routing Protocols in Ad Hoc Networks,” Int. Conf. on Wireless Networks (ICWN), Las Vegas, Nevada, USA, 2002.

[147] M. C. Domingo, O. León and D. Remondo, “On the Extensión of Battery Life with Dynamic Source Routing”, IFIP Workshop and EUNICE Summer School on Adaptable Networks and Teleservices, Eunice' 2002, Trondheim, Norway, Sept. 2002.

[148] M. C. Domingo and D. Remondo, “A New Energy-Aware Routing Protocol for Mobile Ad Hoc Networks”, Med-Hoc-Net 2003, Mahdia, Tunisia, June 2003.

[149] M. C. Domingo, D. Remondo y O. León, “Nuevo protocolo de encaminamiento para la mejora de la supervivencia en redes ad hoc”, Jitel 2003, Las Palmas de Gran Canaria, Sept. 2003.

[150] M. C. Domingo, D. Remondo and O. León, “A Simple Routing Scheme for Improving Ad Hoc Network Survivability”, in Proc. of IEEE Global Telecommunications Conference (IEEE GLOBECOM 2003), San Francisco, USA, Dec. 2003.

[151] Ns-2: Network Simulator, http://www.isi.edu/ nsnam/ns/.

[152] Y. Morgan and T. Kunz, "PYLON: An architectural framework for ad-hoc QoS interconnectivity with access domains", Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS-36), Hawaii, USA, January 2003, IEEE Computer Society Press 2003.

[153] D. Chen, S. Garg, M. Kappes and K.S. Trivedi, “Supporting VBR Traffic in IEEE 802.11 WLAN in PCF Mode,” in Proc. OPNETWORK’02, Washington D.C., Aug. 2002.

[154] ITU-T Recommendation G.114, “One way transmission time", May 2000.

[155] P.B. Velloso, M. G. Rubinstein and M. B. Duarte, “Analyzing Voice Transmission Capacity on Ad Hoc Networks”, International Conference on Communications Technology - ICCT 2003, Beijing, China, April 2003.

[156] M.C. Domingo and D. Remondo, “Quality of Service Support in Wireless Ad Hoc Networks Connected to Fixed DiffServ Domains”, Conference on Personal Wireless Communications (PWC 2004), Delft, The Netherlands, Lecture Notes in Computer Science, Berlin, 2004, Springer Verlag.

[157] R. Wakikawa, J. T. Malinen, C. E. Perkins, A. Nilsson, and A. J. Tuominen, “Global connectivity for IPv6 mobile ad-hoc networks”, Internet Engineering Task Force, Internet Draft (Work in Progress), July 2002.

[158] P. Ratanchandani and R. Kravets, “A hybrid approach to internet connectivity for mobile ad hoc networks”, Proceedings of WCNC 2003, Volume 3, New Orleans, Louisiana, USA, March 2003, pp. 1522-1527.

[159] A. Jain, A. Pruthi, R.C. Thakur, and M.P.S. Bhatia, “TCP analysis over wireless mobile ad hoc networks”, Personal Wireless Communications, 2002 IEEE International Conference on , New Delhi, India, 15-17 Dec. 2002, pp. 95 – 99.

[160] D. Chen, S. Garg, M. Kappes and K. S. Trivedi, “Supporting VoIP traffic in IEEE 802.11 WLAN with enhanced medium access control (MAC) for quality of service”, www.research.avayalabs.com/techreport/ALR-2002-025-paper.pdf.

[161] M. C. Domingo and D. Remondo, “An Improved Resource Allocation Scheme for VBR VoIP Support in Ad Hoc Networks Connected to Fixed IP Networks”, Accepted for publication in the Special Issue on “Wireless Ad Hoc and Sensor Networks” of the Journal of Internet Technology (JIT).

[162] M. C. Domingo and D. Remondo, “Analysis of VBR VoIP Traffic for Ad Hoc Connectivity with a Fixed IP Network”, Proceedings of IEEE Vehicular Technology Conference (IEEE VTC 2004-Fall), Los Angeles, USA, Sept. 2004.

[163] M. C. Domingo and D. Remondo, “Analyzing Voice Transmission between Ad Hoc Networks and fixed IP Networks providing end-to-end Quality of Service”, Accepted for publication in “Wireless Networks and Mobile Computing" (edited by Ding-Zhu Du and Guoliang Xue), in Book Series "Network Theory and Applications", Springer Verlag.

[164] M. C. Domingo and D. Remondo, “An Improved Service Differentiation Scheme for VBR VoIP in Ad-Hoc Networks Connected to Wired Networks”, Service Assurance with Partial and Intermittent Resources (SAPIR 2004), Fortaleza, Brazil, vol. 3126 of Lecture Notes in Computer Science, pages 301-310, Berlin, 2004, Springer Verlag.

[165] M C. Domingo and D. Remondo, “An Interaction Model and Routing Scheme for QoS Support in Ad Hoc Networks Connected to Fixed Networks”, International Workshop on Quality of Future Internet Services (QofIS 2004), Barcelona, Spain, vol. 3266 of Lecture Notes in Computer Science, Berlin, 2004, Springer Verlag.

[166] M. C. Domingo and D. Remondo, “A Cooperation Model between Ad Hoc Networks and Fixed Networks for Service Differentiation”, Proceedings of IEEE Wireless Local Networks (IEEE WLN 2004), held in conjunction with LCN, Tampa, Florida, USA, Nov. 2004.

[167] M. C. Domingo and D. Remondo, “An Interaction Model for QoS Support in Ad Hoc Networks Connected to Fixed IP Networks”, Accepted for publication in the

Special Issue on “Mobile Systems, E-commerce and Agent Technology” of the International Journal of Wireless and Mobile Computing (IJWMC).

[168] M.C. Domingo and D. Remondo, “An Interaction Model between Ad-hoc Networks and Fixed IP Networks for QoS Support”, ACM/IEEE MSWIM 2004, Venice, Italy, October 2004.

[169] J. Xi and C. Bettstetter, “Wireless Multi-Hop Internet Access: Gateway Discovery, Routing, and Addressing”, In Proc. International Conference on Third Generation Wireless and Beyond (3Gwireless'02), San Francisco, California, USA, May 28-21, 2002.

[170] M. Ghassemian, P. Hofmann, H. Aghvami, C. Prehofer, “Analyses of Addressing and QoS Approaches for Ad Hoc Connectivity with the Internet”, in Proc. of PIMRC 2003, Beijing, China, September 7-10, 2003.

[171] M. Ghassemian, P. Hofmann, C. Prehofer, V. Friderikos, H. Aghvami, “Performance Analysis of Internet Gateway Discovery Protocols in Ad Hoc Networks”, IEEE WCNC 2004, Atlanta, Georgia, USA.

[172] J. P. Jeong , J.-S. Park , K. Mase, Y.-H. Han, B. Hakim, J.-M. Orset, “Requirements for Ad-hoc IP Address Autoconfiguration”, IETF Internet draft, draft-jeong-manet-addr-autoconf-reqts-00.txt, August 2003 (Work in Progress).

[173] A. Nilsson, C. E. Perkins, A. Tuominen, R. Wakikawa and J. T. Malinen: “AODV and IPv6 Internet Access for Ad-hoc networks”, ACM Mobile Computing and Communications Review, July 2002, Vol. 6.

[174] M. C. Domingo and D. Remondo, “A Cooperation Model and Routing Protocol for QoS Support in Ad Hoc Networks Connected to Fixed IP Networks”, submitted for publication in Service Assurance with Partial and Intermittent Resources (SAPIR 2005), Lisbon, Portugal, Lecture Notes in Computer Science, Berlin, 2005, Springer Verlag.

# *Glosario de acrónimos*

Esta lista resume los acrónimos usados en esta tesis doctoral:

| | |
|---|---|
| ABR | Associativity-Based Routing |
| AC | Access Category |
| ACK | Acknowledgement |
| AEDCF | Adaptative Enhanced DCF |
| AF | Assured Forwarding |
| AIFS | Arbitration Inter Frame Space |
| AIFSN | Arbitration Inter Frame Space Number |
| AIMD | Additive Increase Multiplicative Decrease |
| AODV | Ad-hoc On-Demand Distance Vector |
| ARME | Assured Rate MAC Extensión |
| ARP | Address Resolution Protocol |
| AS | Assured Service |
| BE | Best Effort |
| BQ | Base QoS |
| BRP | Bordercast Resolution Protocol |
| CBR | Constant Bit Rate |
| CBS | Committed Burst Size |
| CE | Congestion Experienced |
| CIR | Committed Information Rate |
| CMMBCR | Conditional Max-Min Battery Capacity Routing |
| CSMA/CA | Carrier Sense Multiple Access with Collision Avoidance |
| CTS | Clear to Send |
| CW | Contention Window |
| DCF | Distributed Coordination Function |
| DDRR | Distributed Deficit Round Robin |
| DFS | Distributed Fair Scheduling |
| DiffServ | Differentiated Services |
| DIFS | DCF Inter-Frame Space |
| DIME | DiffServ MAC Extension |
| DRR | Deficit Round Robin |
| DS | Differentiated Services |
| DSCP | Differentiated Services Codepoint |

| | |
|---|---|
| DSDV | Destination-sequenced Distance Vector |
| DSR | Dynamic Source Routing |
| DS-SWAN | Differentiated Services-Stateless Wireless Ad-hoc Networks |
| DWFQ | Distributed Weighted Fair Queuing |
| ECN | Explicit Congestion Notification |
| ECT | ECN-Capable Transport |
| EDCF | Enhanced DCF |
| EF | Expedited Forwarding |
| EQ | Enhanced QoS |
| ETSI | European Telecommunications Standards Institute |
| EWMA | Exponentially Weighted Moving Average |
| FCC | Federal Communications Commission |
| FIFO | First In First Out |
| FQMM | Flexible QoS Model for MANETs |
| FTP | File Transfer Protocol |
| HiperLAN | High-Performance Radio Local Area Network |
| IAPP | Inter-Access Point Protocol |
| IARP | Intrazone Routing Protocol |
| IEEE | Institute of Electrical and Electronics Engineers |
| IERP | Interzone Routing Protocol |
| IETF | Internet Engineering Task Force |
| IFS | Inter Frame Space |
| IntServ | Integrated Services |
| IP | Internet Protocol |
| ISP | Internet Service Provider |
| ITU | International Telecommunication Union |
| LAN | Local Area Network |
| LLC | Logical Link Control |
| MAC | Medium Access Control |
| MANET | Mobile Ad-hoc Network |
| MBCR | Minimum Battery Cost Routing |
| MF | Multiplicative Factor |
| MMBCR | Min-Max Battery Cost Routing |
| MPR | Multipoint Replaying |
| MTPR | Minimum Total Transmission Power Routing |
| NP | Non-deterministic polynomial-time |
| OLSR | Optimized Link State Routing |

| | |
|---|---|
| PAN | Personal Area Network |
| PCF | Point Coordination Function |
| PCM | Pulse Code Modulation |
| PDA | Personal Digital Assistant |
| PF | Persistence Factor |
| PHB | Per-Hop Behavior |
| PIFS | PCF Inter Frame Space |
| QoS | Quality of Service |
| RED | Random Early Detection |
| RERR | Route Error |
| RES | Reservation |
| RIO-C | Random Early Detection with In/Out Coupled |
| RREP | Route Reply |
| RREP-ACK | Route Reply-Acknowledgement |
| RREQ | Route Request |
| RTP | Real-time Transport Protocol |
| RTS | Request To Send |
| SCFQ | Self-Clocked Fair Queueing |
| SD-AODV | Service Differentiation-Ad-hoc On Demand Distance Vector |
| SEADSR | Simple Energy Aware Dynamic Source Routing |
| SIFS | Short Inter Frame Space |
| SLA | Service Level Agreement |
| SWAN | Stateless Wireless Ad Hoc Networks |
| TCA | Traffic Conditioning Agreement |
| TCP | Transmission Control Protocol |
| TORA | Temporally-Ordered Routing Algorithm |
| TOS | Type of Service |
| TXOP | Transmission Opportunity |
| UDP | User Datagram Protocol |
| UP | User Priority |
| UWB | Ultra Wide Band |
| VA | Virtual Application |
| VBR | Variable Bit Rate |
| VMAC | Virtual MAC |
| VoIP | Voice over IP |
| VS | Virtual Source |
| WLAN | Wireless Local Area Network |

| | |
|---|---|
| WRP | Wireless Routing Protocol |
| ZRP | Zone Routing Protocol |