



UNIVERSIDAD REY JUAN CARLOS

TESIS DOCTORAL

**Protocolo de encaminamiento adaptativo para la
optimización de comunicaciones de captura de
datos y diseminación dirigida en redes de
sensores inalámbricos**

Autor:

Francisco Javier Atero Gómez

Directores:

Dr. D. Mark R. Wilby

Dr. D. Juan José Vinagre Díaz

DEPARTAMENTO DE TEORÍA DE LA SEÑAL Y COMUNICACIONES

Junio de 2011

Yo, Mark Richard Wilby, con D.N.I. X5251521-T, como Director de la Tesis Doctoral realizada por Francisco Javier Atero Gómez y titulada *Protocolo de encaminamiento adaptativo para la optimización de comunicaciones de captura de datos y diseminación dirigida en redes de sensores inalámbricos*, certifico que ésta cumple con todos los requisitos necesarios, y por ello autorizo la defensa de la misma.

Fdo.

Fuenlabrada, a 6 de mayo de 2011

TESIS DOCTORAL

Protocolo de encaminamiento adaptativo para la optimización de comunicaciones de captura de datos y diseminación dirigida en redes de sensores inalámbricos

Autor: Francisco Javier Atero Gómez

Directores: Prof. Dr. D. Mark R. Wilby

Prof. Dr. D. Juan José Vinagre Díaz

Firma del Tribunal Calificador:

Firma

Presidente:

Vocal:

Vocal:

Vocal:

Secretario:

Calificación:

Fuenlabrada, de de

A Inma, por ser mi luz, por compartir esta travesía.

A Javier y Juan, por completar mi existencia.

A Paco y Maribel, por hacerlo todo posible,
por ser quien soy, por dárme todo.

Sé que, aunque allí arriba, estáis conmigo.

Resumen

Los avances recientes en comunicaciones inalámbricas y microelectrónica han motivado la aparición de dispositivos sensores inalámbricos de bajo coste y pequeño tamaño capaces de efectuar de forma autónoma tareas de monitorización del entorno, procesamiento de paquetes y comunicación con otros nodos a través de su interfaz radio. Estos dispositivos, cuyo despliegue lleva asociado un coste mucho menor que el de los sistemas cableados, se integran en el entorno a medir y son capaces de recolectar, procesar, agregar y transmitir información de variables recogidas tales como temperatura, humedad, aceleración, etc., así como de otros eventos detectados como actividad sísmica o detección de presencia o movimiento. Las Redes de Sensores Inalámbricos (WSN, *Wireless Sensor Network*) gozan de gran auge hoy en día ya que pueden utilizarse en multitud de aplicaciones que incluyen la monitorización ambiental, la agricultura de precisión, la gestión automatizada del hogar, la eficiencia energética de edificios, la detección de intrusos, la gestión de desastres y emergencias o la monitorización de señales corporales, entre muchas otras.

Una WSN puede estar formada por un gran número de sensores, los cuales tienen importantes restricciones en cuanto a la capacidad de procesamiento, de memoria, de potencia de transmisión y, sobre todo, energética. En este sentido, el proceso de comunicación del sensor inalámbrico con otros nodos debe ser capaz de minimizar el consumo energético del dispositivo con objeto de mantenerlo el máximo tiempo posible activo dentro de la red, en la cual los procesos de mantenimiento a menudo se limitan a desplegar nuevos nodos que sustituyan a los que ya han agotado totalmente su energía.

En esta Tesis se propone un nuevo protocolo de encaminamiento jerárquico para WSN denominado HARP (*Hierarchical Adaptive Routing Protocol*), cuyo diseño está principalmente enfocado al ahorro energético de los nodos y a la maximización del tiempo de vida útil de la red. La propuesta realizada de HARP para este tipo de redes se basa en la división de la red en *clusters* en los cuales el rol correspondiente al *Cluster-Head* es rotatorio con objeto de distribuir el consumo de la red de manera uniforme. HARP es capaz de construir estructuras jerárquicas

que minimizan el consumo energético y favorecen la escalabilidad de la red. Debido al diseño de sus tablas de encaminamiento y al proceso de operación implementado, HARP, a diferencia de la mayoría de protocolos actuales, permite a los nodos cursar cualquier tipo de tráfico introducido en la red, tanto del que parte del nodo sumidero con destino los nodos sensores (diseminación de datos) como el generado por las comunicaciones de captura de datos por parte de los nodos sensores con destino el nodo sumidero, así como el establecimiento de comunicaciones entre cualquier par de nodos de la red. Además, el protocolo propuesto proporciona mecanismos de tolerancia a fallos de red y de gestión de movilidad para adaptarse a las circunstancias cambiantes del entorno estudiado y a las necesidades de la aplicación, todo ello sin sacrificar la eficiencia energética de la red y minimizando la sobrecarga de paquetes de control generados. Para completar el proceso de diseño se propone s-HARP, extensión del protocolo propuesto que proporciona una formulación específica para la elección de los *Cluster-Head* y la formación de los *clusters* y que mejora de forma notable el rendimiento de HARP.

La estructura de rutas propuesta en HARP y su mecanismo de recuperación local de fallos son objeto de un estudio teórico que permite modelar el tiempo de vida de las rutas entre cualquier par de nodos de la red y compararlo con el correspondiente a otros tipos de protocolos de encaminamiento, demostrando que el protocolo propuesto proporciona estadísticamente mejores prestaciones en cuanto a la maximización de dicho tiempo.

La evaluación del rendimiento del protocolo se realiza en un nuevo simulador diseñado específicamente para esta Tesis y que permite comparar el comportamiento de HARP y s-HARP con respecto al de otros protocolos representativos basándose en la definición de los parámetros necesarios que permiten evaluar la eficiencia de cada uno de ellos. Los resultados obtenidos demuestran que tanto HARP como s-HARP permiten aumentar considerablemente el tiempo de vida útil de la red y maximizar el rendimiento energético del sistema. Por último, se evalúan los mecanismos de operación y tolerancia a fallos en otros escenarios distintos a las WSN y se mide el rendimiento del protocolo comparado con otros protocolos existentes en la actualidad.

Abstract

Recent advances in wireless communications and microelectronic systems have precipitated the development of extremely small and low-cost sensors. These sensors possess autonomous capacities and combine sensing, signal processing and wireless communication capabilities. Also, they can be deployed at much lower cost than using more traditional wired sensor systems. Additionally, they can be more easily embedded into the physical environment, facilitating the gathering, processing, aggregation and transmission of distributed and mobile information. Wireless Sensor Networks (WSN) are fast becoming an essential technology and they are fundamental to the development of a variety of applications, including augmented reality, habitat monitoring, precision agriculture, home automation, building energy efficiency, disaster recovery and emergency services, body signals monitoring and health-care.

A WSN is a system that consists of thousands of very small sensor nodes which have severe constraints in data processing, storage, communication and energy storage capabilities. For example, in the context of communications, data exchange between nodes leads to high energy use, which also depends on node separation. Hence, optimization in this area can substantially prolong the lifetime of the sensor, which improves network lifetime and minimizes maintenance requirements.

In this thesis a new hierarchical routing protocol for WSN, named HARP (*Hierarchical Adaptive Routing Protocol*), is proposed. HARP's design is primarily focused on energy savings and the maximization of useful network lifetime. The HARP proposal is based on clustering sensors into groups and cycling the energetically costly role of the Cluster-Head amongst all the nodes in the cluster. HARP also builds hierarchical structures which assist in saving energy and provides network scalability. The routing table design and the operation process in HARP, unlike other WSN routing protocols, allow the nodes to support bi-directional traffic flows, supporting optimized data acquisition of sensed data as well as dissemination of control data. Hence we have a maintained transmission route both to and from the sink node. The architecture also supports peer-to-peer communication among any pair of nodes. Furthermore, HARP provides failure

recovery mechanisms and an efficient mobility management process, which allows the system to accommodate environmental changes and node mobility within a limited network overhead. To complete the routing proposal, the formulation of s-HARP is provided in order to more intelligently select the best candidates to be Cluster-Heads further improving performance.

We provide a theoretical analysis of route structure and local recovery mechanism in HARP, which leads us to a model of the lifetime of the routes to a destination, which provides a basis for comparison to alternative protocols. Results show that HARP always provides a higher performance, maximizing the route lifetimes.

Simulations of HARP and s-HARP performance have been made using a new simulator. They provide a comparison with existing routing protocols in WSN and allow the analysis to be extended beyond the limits of the analytic models. The results demonstrate that both proposals, HARP and s-HARP, significantly increase the useful lifetime of the network and maximize the efficiency of the system. Finally, the simulator is used to analyze the general operation and recovery process in different scenarios.

Agradecimientos

Una vez finalizado el trabajo de esta Tesis Doctoral, debo enfrentarme a la parte más bonita de escribir que es la correspondiente a los agradecimientos. Sólo quiero brevemente incluir unas líneas para expresar mi sincera gratitud a las personas que de una forma u otra han contribuido a hacer posible este trabajo y que me han ayudado a afrontar este ilusionante reto.

De forma muy especial, quiero dejar constancia de mi agradecimiento a mis dos Directores de Tesis, Mark Wilby y Juan José Vinagre. Mark, gracias por haberme conducido, por tus reflexiones, por tus ideas, por tu dedicación, por tus viajes. Juanjo, amigo, gracias por tu confianza, tu cariño, tu disponibilidad, tus argumentos, tus noches en vela, tu preocupación, tu trabajo. Gracias a los dos por todo.

A Javier Ramos quiero agradecerle su confianza, su disponibilidad y la oportunidad que en su día me brindó de hacer esto posible. A mis compañeros del Departamento de Teoría de las Señal y Comunicaciones de la Universidad Rey Juan Carlos que me han acogido y ayudado en la adaptación a esta nueva etapa de mi vida. Gracias a todos vosotros. En especial, quería agradecer a mis compañeros del B102 Lorena, Carlos y Edu su inestimable ayuda, su tiempo, su trato, en definitiva, su cariño. Gracias de nuevo.

Por último, merece un apartado especial mi gratitud a todas las personas que forman parte de mi vida, mi familia, mis amigos. Muy especialmente mis padres, a quien todo les debo. Allí donde estáis sé que seguís cuidando de mí igual que antes. Y finalmente, a Inma, la otra mitad de mi equipo, por hacer también tuyo mi esfuerzo, mi sacrificio, por compartir los momentos buenos y también los malos a lo largo de estos años. Gracias por Javier y Juan, mis dos pensamientos felices. Gracias por todo.

*Cree a aquellos que buscan la verdad,
duda de los que la han encontrado.*

André Gide.

Índice general

Resumen	III
Abstract	V
Agradecimientos	VII
Prefacio	IX
Acrónimos	XV
1. Introducción	1
1.1. Motivación	2
1.2. Objetivos	7
1.3. Estructura	9
2. Protocolos de encaminamiento en WSN	11
2.1. Estado del arte	11
2.1.1. Clasificación en función del método de operación	13
2.1.2. Clasificación basada en la estructura de la red	14
2.2. Criterios de diseño	28
2.3. Conclusiones	36
3. Descripción de HARP	39
3.1. Arquitectura general del protocolo	40
3.1.1. Encaminamiento multi-salto	40

3.1.2.	Encaminamiento multi-ruta	41
3.1.3.	Encaminamiento salto a salto	44
3.1.4.	HARP como protocolo reactivo	44
3.1.5.	<i>Clustering</i> jerárquico	45
3.2.	Estructura de los mensajes de control	46
3.3.	Construcción del árbol jerárquico	63
3.3.1.	Estructura de las tablas de encaminamiento	63
3.3.2.	Inicialización del árbol jerárquico	64
3.3.3.	Optimización dinámica de la estructura	65
3.4.	Operación del protocolo	67
3.5.	Reconfiguración dinámica de la red	69
3.5.1.	Adición de nuevos nodos en la red	69
3.5.2.	Recuperación ante el fallo de un enlace	71
3.5.3.	Gestión de la movilidad	72
3.5.4.	Nodo sustituto	73
4.	Descripción del simulador	75
4.1.	Introducción	75
4.2.	Estado del arte en simuladores para WSN	79
4.3.	Descripción del simulador: WiNS	84
4.3.1.	Cola de eventos	87
4.3.2.	Estructura de clases	91
4.4.	Descripción de los experimentos	108
4.4.1.	Características generales	108
4.4.2.	Características específicas	109
5.	Análisis teórico de prestaciones	111
5.1.	Marco teórico	113
5.1.1.	Introducción	113
5.1.2.	Estudio del tiempo de vida de las rutas	115
5.1.3.	Algoritmo de cálculo de probabilidad de rutas disjuntas . . .	142
5.1.4.	Cálculo de la <i>pdf</i> del tiempo de vida del sistema	155

5.1.5.	Análisis del proceso de Recuperación Local	160
5.2.	Simulaciones	180
5.2.1.	Prestaciones de HARP	181
5.2.2.	Prestaciones de Recuperación Local	185
6.	HARP en WSN	189
6.1.	Introducción	189
6.2.	Arquitectura del protocolo	190
6.2.1.	Modelo energético	191
6.2.2.	Tipos de nodos	194
6.2.3.	Arquitectura propuesta	197
6.3.	Fases de operación del protocolo	200
6.3.1.	Fase de inicialización o <i>Set-up phase</i>	201
6.3.2.	Fase de transmisión de datos o <i>Steady phase</i>	215
6.4.	Mecanismos de recuperación ante fallos y gestión de movilidad	219
6.4.1.	<i>Slot</i> de recuperación	220
6.4.2.	Nodo sustituto	224
6.5.	Arquitectura de aplicación específica	228
6.6.	s-HARP	229
6.7.	Conclusiones	232
7.	Resultados y aplicaciones	235
7.1.	Descripción de los experimentos	235
7.2.	Resultados	237
7.2.1.	Prestaciones de HARP en redes homogéneas	238
7.2.2.	Prestaciones de HARP en redes heterogéneas	249
7.2.3.	Red homogénea frente a red heterogénea	255
7.2.4.	Otros escenarios de simulación	256
7.3.	Aplicaciones de HARP	260
7.3.1.	HARP en WSN	261
7.3.2.	HARP adaptado al proyecto GUIADE	276
7.4.	Conclusiones	286

8. Conclusiones, publicaciones y líneas futuras	287
8.1. Conclusiones y aportaciones	287
8.2. Publicaciones científicas	290
8.3. Líneas futuras	291
Apéndices	296
A. Ejemplo de aplicación de los algoritmos propuestos para el cálculo de probabilidad de rutas disjuntas	299
A.1. Primer nivel intermedio	300
A.2. Segundo nivel intermedio	303
A.3. Tercer nivel intermedio	312
A.4. Resultados	322
Bibliografía y referencias	326

Acrónimos y abreviaturas* **

BER	<i>Bit error rate</i>
CBR	<i>Constant Bit Rate</i>
cdf	<i>Cumulative density function</i>
CDMA	<i>Code Division Multiple Access</i>
CH	<i>Cluster-Head</i>
CSMA	<i>Carrier Sense Multiple Access</i>
CSMA/CA	<i>Carrier Sense Multiple Access with Collision Avoidance</i>
CST	<i>Child Strength List</i>
CT	<i>Child Table</i>
FIFO	<i>First In First Out</i>
FEL	<i>Future Events List</i>
GPS	<i>Global Positioning System</i>
GUI	<i>Graphical User Interface</i>
HARP	<i>Hierarchical Adaptive Routing Protocol</i>
ID	<i>Identifier</i>
iid	<i>Independent and identically distributed</i>
Kbps	<i>Kilobits per second</i>
LAN	<i>Local Area Network</i>

* A lo largo de esta Tesis se mantendrán en su forma original aquellos acrónimos derivados de una expresión inglesa cuyo uso se encuentre extendido en la literatura científica.

** De acuerdo con las recomendaciones de la Real Academia Española, en esta Tesis los acrónimos y siglas no se modifican para formar el plural.

MAC	<i>Media Access Control</i>
MANET	<i>Mobile Ad Hoc Network</i>
MAC	<i>Media Access Control</i>
nonCH	<i>nodo normal, que no ejerce el rol de CH</i>
OSI	<i>Open System Interconnection</i>
pdf	<i>Probability density function</i>
PT	<i>Parent Table</i>
QoS	<i>Quality of Service</i>
RF	<i>Radiofrecuencia</i>
RL	<i>Recuperación Local</i>
RS	<i>Recovery Slot</i>
RSSI	<i>Receive Signal Strength Indication</i>
SNR	<i>Signal to Noise Rate</i>
TDMA	<i>Time Division Multiple Access</i>
TMC	<i>Traffic Management Center</i>
TTL	<i>Time To Live</i>
UMTS	<i>Universal Mobile Telecommunication System</i>
WiNS	<i>Wireless Network Simulator</i>
WiMAX	<i>Worldwide Interoperability Microwave Access</i>
WSAN	<i>Wireless Sensor and Actor Network</i>
WSN	<i>Wireless Sensor Network</i>

Capítulo 1

Introducción

El objetivo principal de la presente Tesis Doctoral es la propuesta de un nuevo protocolo de encaminamiento en Redes de Sensores Inalámbricos (WSN, *Wireless Sensor Network*), denominado HARP (*Hierarchical Adaptative Routing Protocol*), que permite mejorar la eficiencia de los dos procesos principales que se producen en este tipo de redes: (i) *data gathering* o recolección de datos por parte de los nodos sensores y encaminamiento de los mismos hacia una o varias estaciones base o nodos sumidero encargados de procesarlos y que actúan además como pasarela con otras redes; (ii) *data dissemination* o envío de paquetes desde la estación base con destino los nodos sensores de la red. Con este objetivo se desarrolla un marco teórico que posibilita el análisis de las prestaciones de HARP y su comparación con otros protocolos existentes basándose en el tiempo de duración de las rutas disponibles hacia el nodo destino que proporciona el protocolo. Este análisis teórico se valida en un nuevo simulador desarrollado para WSN y que sirve a su vez para comparar empíricamente la eficiencia de HARP con los protocolos más representativos de este tipo de redes existentes en la actualidad. Por último, se proponen otras aplicaciones del protocolo y se evalúa su eficiencia en redes distintas a las WSN.

1.1. Motivación

Las WSN se han convertido en foco de atención de la comunidad científica durante los últimos años [48]. Este desarrollo en el plano de la investigación ha venido acompañado de unos primeros pero firmes pasos en el ámbito comercial que han reforzado la idea de la necesidad de este tipo de redes para innumerables aplicaciones que necesitan de una monitorización del entorno para desarrollar servicios en multitud de ámbitos dispares como la seguridad física, la sanidad o la agricultura. Este gran abanico de soluciones se encuadra dentro de la llamada inteligencia ambiental (AmI, *Ambient Intelligence*) [1], que promueve la creación de una interacción constante, directa y transparente del usuario con su medio circundante de forma que sea posible la oferta de servicios personalizados a las características particulares del mismo: hábitos, localización, preferencias, etc. Este modelo de computación ubicua (llamado también *context-aware computing* o *pervasive computing*) tiene como objetivo final el diseño de mecanismos e infraestructuras de comunicación que permitan a los nodos sensores integrarse en el medio ambiente de forma transparente e invisible al usuario [21].

Las WSN están formadas por pequeños dispositivos sensores que tienen capacidades limitadas de procesado, memoria y comunicación inalámbrica entre ellos y que además poseen importantes limitaciones en cuanto al consumo energético, ya que generalmente están alimentados mediante baterías. Típicamente una WSN puede contener cientos e incluso miles de estos dispositivos sensores, los cuales tienen la capacidad de comunicarse e intercambiar información entre ellos y con una o más estaciones base, que actúan como sumideros de toda la información generada y recogida por la red. Dada la estructura de la red formada por los nodos y la limitación de potencia de transmisión de los mismos, no todos los dispositivos podrán enviar o recibir directamente paquetes de la estación base, lo que hace necesario un mecanismo de retransmisión o encaminamiento de los diferentes paquetes que

circulan por la red. En estos últimos años parte de la investigación en WSN ha ido encaminada a estudiar el potencial de colaboración entre los diferentes nodos en los procesos de recolección de datos y procesado conjunto de los mismos (*data aggregation*), con el objetivo principal de minimizar el impacto que las transmisiones redundantes tienen en el rendimiento conjunto de la red y en el agotamiento de la batería de los nodos [23]. En la Figura 1.1 se muestra la estructura de una WSN con nodos sensores y la posibilidad de existencia de múltiples nodos sumidero.

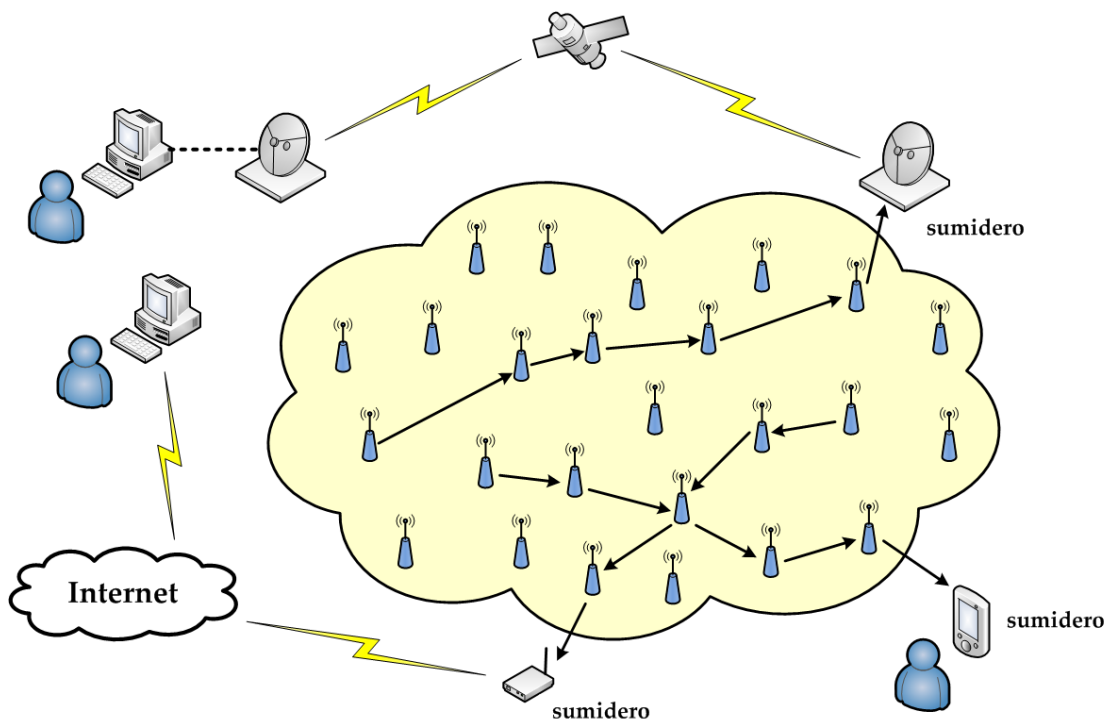


Figura 1.1: Ejemplo de estructura de una WSN con sensores para monitorización con tres nodos sumidero.

Los avances tecnológicos actuales permiten que la fabricación de estos dispositivos de pequeño tamaño y bajo coste sea técnica y económicamente viable. Cada uno de ellos se despliega en la zona objeto de estudio detectando eventos o tomando medidas de las condiciones ambientales y de los parámetros del entorno que los rodea. Estas mediciones son transformadas en señales eléctricas que pueden ser procesadas localmente por los nodos para revelar algunas características acerca del

fenómeno estudiado, proporcionando de esta forma información completa relativa a toda la zona de cobertura de la WSN.

Una WSN se estructura habitualmente en forma de estrella en la que el nodo que actúa de pasarela (*gateway*, nodo sumidero o estación base) entre la propia red y el servidor central o una red externa, se configura como el centro de la misma, y los nodos sensores ocupan posiciones en sus brazos siguiendo una estructura distribuida. Así, en el proceso de medición, recopilación y transmisión de datos en aplicaciones de *data gathering* desde los nodos sensores hacia la estación base, las comunicaciones en una WSN se efectúan mediante un esquema multi-salto en el que los dispositivos intermedios de la ruta hacen las veces de repetidores o re-encaminadores hasta que el mensaje alcanza el destino final. Teniendo en cuenta este esquema de transporte de información, la mayoría de las aplicaciones actuales de este tipo de redes tienen en cuenta únicamente el tráfico en este sentido ascendente desde los nodos sensores hacia la estación base, denominado comúnmente *upstream* [73], razón por la cual el encaminamiento diseñado para una WSN se centra precisamente en este sentido, construyendo algoritmos jerárquicos que optimizan por una parte la distancia de la ruta resultante y el consumo energético y, por otra, su duración y fiabilidad.

Sin embargo, el tráfico *upstream* de datos hacia la estación base no es el único que se transporta en una WSN. Existe también un tráfico denominado *downstream* en el que la estación base genera distintos flujos de paquetes con destino a los diferentes nodos de la red. Estas situaciones contemplan los casos en los que la estación base indica a los nodos el momento de realizar una medición (*query-drive mode*), les envía información específica necesaria para el control de la red o disemina un nuevo *firmware* para actualizar la configuración de los nodos (*data dissemination*). En este sentido, sólo un número limitado de protocolos de encaminamiento han sido propuestos por la comunidad científica para manejar de forma efectiva este

tipo de tráfico, clasificándose los mismos en función de su estructura de red y de su modelo de transmisión de datos, pero no de su funcionalidad [38].

Desde el punto de vista tecnológico, las WSN son, en definitiva, un subconjunto de la categoría superior formada por las Redes Móviles Ad Hoc Inalámbricas (MANET, *Mobile Ad Hoc Network*), capaces de proporcionar soluciones a las exigencias actuales relativas al establecimiento de redes autónomas que no presentan restricciones en cuanto a la necesidad de una infraestructura externa para su operación. Este grado de autonomía es conseguido mediante la capacidad de las MANET de autoconfigurarse y admitir actualizaciones dinámicas en tiempo y espacio.

Evidentemente, estas fuertes restricciones suponen un gran reto tecnológico que deberá superar los problemas subyacentes a las principales características intrínsecas de este tipo de redes: variabilidad en la topología y el canal, movimiento de los nodos y ausencia de infraestructura. Estas limitaciones inciden en la mayor parte de los aspectos involucrados en la operación de una red de comunicaciones. Entre ellos, uno de los más afectados es el encaminamiento, al mismo tiempo, crucial para el funcionamiento de una MANET y las prestaciones ofrecidas por ella. En particular, en las WSN el encaminamiento es responsable de variables fundamentales para la misma como el consumo energético de los nodos, el retardo de los paquetes o la tasa de error de bit (BER, *Bit Error Rate*). Estas variables inciden directamente en la calidad de servicio (QoS, *Quality of Service*) ofrecida e incluso en la vida útil de la red, por lo que deben ser tomadas en profunda consideración durante la investigación y desarrollo de esta nueva tecnología.

En este escenario, en el que el consumo energético es un parámetro crítico en el diseño de un protocolo de encaminamiento en WSN, las arquitecturas clásicas estratificadas en capas con niveles funcionales claramente diferenciados (como el modelo OSI) no resultan las más adecuadas al no estar permitida la cooperación entre los distintos niveles a pesar de que varios de ellos, como el nivel de acceso

al medio y el nivel de red, inciden directamente en el consumo energético de los nodos. Un modelo estructural inter-capa en el que los protocolos de los distintos niveles puedan cooperar entre sí para optimizar el rendimiento de la red es más eficiente en WSN, donde el diseño de los protocolos de las capas inferiores depende en gran medida de la aplicación específica, haciéndose necesarias nuevas técnicas que eliminen las ineficiencias energéticas de los protocolos actuales, que limitan por un lado el tiempo de vida de los nodos de la red y, por otro, el uso adecuado del ancho de banda de la misma.

En cuanto a las aplicaciones actuales de este tipo de redes, en su gran mayoría el diseño de las mismas se basa en la premisa de que los nodos sensores son estáticos o cuasi-estáticos, limitándose de esta forma la extensión de las WSN a otros ámbitos o escenarios de uso. La gestión de la movilidad de los nodos es otro requisito del que carecen muchos de los protocolos de encaminamiento actuales y que limita en gran medida su uso futuro. Esta gestión debe integrarse de forma efectiva en las funcionalidades del protocolo proporcionando mecanismos que conjuguen tolerancia a fallos y movilidad con el mantenimiento de la eficiencia energética en la red.

En este sentido, se propondrá en esta Tesis Doctoral un nuevo protocolo de encaminamiento para WSN que permita que tanto la operación, como el control y la gestión de la red se realicen de forma eficiente, mejorando los aspectos relativos a las prestaciones de los protocolos existentes para ambos tipos de tráfico, *upstream* y *downstream*. En particular, la propuesta del protocolo deberá tener en cuenta, además del consumo energético y las restricciones de los nodos en la red, la escalabilidad de la misma.

1.2. Objetivos

Esta Tesis Doctoral tiene como objetivo último la propuesta de un protocolo de encaminamiento para WSN que sea energéticamente eficiente, fiable y fácilmente escalable, además de proveer mecanismos ágiles de comunicación entre cualquier par de nodos de la red. De este objetivo general, motivados por las razones expuestas en la sección anterior, se desprenden los objetivos específicos, que se enumeran a continuación:

1. Describir los criterios de diseño, parámetros y características que debe tener un protocolo de encaminamiento en WSN para ser eficiente, así como las medidas objetivas mediante las cuales puede cuantificarse dicha eficiencia.
2. Proponer un nuevo protocolo de encaminamiento para WSN que defina los procesos necesarios para la creación y reconfiguración dinámica de rutas entre todos los nodos sensores de la red y la estación base o nodo sumidero, teniendo en cuenta los resultados del objetivo anterior. Para ello, atendiendo a las distintas clasificaciones existentes de protocolos de encaminamiento, se evaluarán las distintas alternativas de diseño, se describirán los módulos principales del protocolo y se propondrá una estructura específica para los mensajes de control y tablas de encaminamiento necesarios para su correcto funcionamiento.
3. Desarrollar un marco teórico que modele analíticamente la estructura de rutas que se crean en la red al aplicar el protocolo definido, basándose en el estudio del tiempo de vida de las rutas entre dos nodos cualesquiera de la red en función de la estructura definida y de los fallos que se producen en la misma como consecuencia de la naturaleza dinámica de las condiciones del entorno [9], [10], [12], [13]. El estudio de este parámetro es importante en WSN y MANET [98], [95], [115] ya que está directamente relacionado con el

número de veces que se inicia el proceso de descubrimiento de rutas, el cual introduce sobrecarga de paquetes en la red y penaliza la eficiencia del protocolo de encaminamiento utilizado. Estos resultados permitirán demostrar teóricamente que las prestaciones de HARP son superiores a las de otros protocolos existentes. Además, se propondrá un algoritmo que tendrá en cuenta la distribución de los nodos en la red para el cálculo probabilístico del tiempo de vida de las rutas.

4. Construir una formulación matemática que permita caracterizar los procesos de reconfiguración dinámica de rutas entre nodos ante los fallos producidos en la red definidos en el protocolo propuesto, los cuales permiten aumentar el tiempo de vida de las mismas y reducir de esta forma la sobrecarga de paquetes de control en la red.
5. Desarrollar un nuevo simulador para WSN y MANET basándose en el modelo de Simulación por Eventos Discretos en el que también se basan otros simuladores existentes como Omnet [134] y JavaSim [124]. Este simulador, que se implementará en un lenguaje de programación orientado a objetos como es Java [8], permitirá definir y modelar todas las variables y procesos asociados a la simulación, tales como movimiento de nodos, desvanecimientos y cambios de condiciones de propagación en enlaces, protocolos del nivel de enlace, colisiones entre paquetes o características de los nodos sensores, controlar en todo momento su funcionamiento y extraer los resultados necesarios para comparar las prestaciones del protocolo propuesto con respecto a otros existentes.
6. Integración del protocolo objeto de esta Tesis en el simulador anteriormente desarrollado, definiendo y programando todas sus funcionalidades a través de la creación de parámetros, clases y métodos Java específicos. Para comparar las prestaciones del protocolo definidos, se seleccionarán y programarán

igualmente varios protocolos existentes actualmente en WSN y MANET cuya aceptación por parte de la comunidad científica esté suficientemente contrastada.

7. Comparar mediante simulaciones las prestaciones del protocolo definido en relación a otros existentes en distintos escenarios y analizar si existen diferencias entre los resultados obtenidos y los teóricos calculados anteriormente. Estos resultados permitirán extraer consecuencias acerca de la eficiencia del protocolo objeto de esta Tesis y sus condiciones de funcionamiento.
8. Propuesta de utilización del protocolo de encaminamiento presentado en escenarios diferentes a las WSN y análisis de sus prestaciones comparadas con otros protocolos representativos.

1.3. Estructura

Se expone a continuación la estructura presente en esta Tesis, compuesta por ocho capítulos, cada uno de los cuales abordará los objetivos específicos indicados a continuación:

- En el **Capítulo 2** se presenta una revisión del estado del arte de los protocolos de encaminamiento existentes para WSN, se presentan las características que los diferencian de los protocolos para redes Ad-Hoc inalámbricas y se exponen los criterios de que deben ser tenidos en cuenta a priori en la etapa de diseño.
- El **Capítulo 3** describe detalladamente la arquitectura general del protocolo de encaminamiento objeto de este trabajo, HARP, compuesto por tres procesos principales: formación del árbol jerárquico, operación y recuperación local frente a fallos y auto-reconfiguración. En este capítulo se exponen las

características que permiten al protocolo propuesto mejorar la eficiencia en la capa de red de las WSN.

- En el **Capítulo 4** se incluye una breve recopilación de los principales simuladores existentes para WSN, se describen las fases principales seguidas en la elaboración del modelo de simulación de esta Tesis, se presentan las características del simulador desarrollado y se describe la estructura de clases Java creada.
- El **Capítulo 5** establece un marco teórico que analiza con detalle la estructura de rutas entre nodos construida por el protocolo, proponiendo expresiones explícitas para calcular el tiempo de vida de todas los caminos disponibles entre dos nodos. De la misma forma, se proponen expresiones analíticas para modelar el proceso de recuperación local de rutas y se realizan simulaciones que demuestran la mejora en las prestaciones ofrecidas por HARP con relación a otras filosofías existentes.
- El **Capítulo 6** describe la estructura específica del protocolo para WSN, particularizando la arquitectura del mismo y estableciendo las distintas fases de operación. Este apartado proporciona una definición completa del nuevo protocolo de encaminamiento propuesto.
- En el **Capítulo 7** se muestran los resultados de diferentes simulaciones en las que se comparan las prestaciones de HARP con respecto a otros protocolos existentes en distintos escenarios de WSN. A su vez, este apartado proporciona un análisis de la eficiencia del protocolo en otras aplicaciones distintas de las WSN.
- Por último, el **Capítulo 8** resume las conclusiones derivadas de los resultados obtenidos en este trabajo, presenta las publicaciones científicas asociadas a esta Tesis Doctoral y establece las líneas futuras de investigación con los nuevos retos que se pueden abordar como continuación de este trabajo.

Capítulo 2

Protocolos de encaminamiento en WSN

2.1. Estado del arte

El auge que actualmente experimentan las WSN tiene su explicación en el gran abanico de posibilidades de utilización de este tipo de redes debido principalmente al bajo coste, pequeño tamaño y extremada sencillez de los nodos sensores que las forman. Algunas de las principales aplicaciones de estas redes son la detección de eventos o sucesos, la monitorización de campos de cultivo, la detección de intrusos, la monitorización ambiental, la vigilancia y seguridad perimetral, el control de inventario, la gestión de desastres, catástrofes naturales y emergencias, la monitorización de señales corporales, la automatización del hogar y la eficiencia energética de edificios, entre otras [23].

Los nodos sensores son dispositivos sencillos pero que pueden hacerse tan complejos como se desee, ya que existe la posibilidad de añadir elementos externos o hardware adicional como es el caso de receptores GPS, pudiendo crear sensores

sofisticados con mayor versatilidad y con capacidad de ejecutar multitud de tareas diferentes. El bajo coste de los dispositivos se complementa con un mantenimiento de la red muy reducido, que a menudo se limita a repoblar el área de despliegue de los sensores con nuevos nodos que entren a formar parte de la red, bien para aumentar el tamaño o la densidad de la zona monitorizada, bien para sustituir a aquellos dispositivos que han dejado de funcionar, ya sea por agotamiento de su batería, sustracción, accidente o mal funcionamiento. En consecuencia, para reducir el coste asociado a la red es primordial gestionar correctamente el consumo energético de los nodos, maximizando el mayor tiempo posible su vida útil, lo que se consigue a través de la utilización de un protocolo eficiente de encaminamiento en la red.

A lo largo de los últimos años se han desarrollado numerosas propuestas de técnicas y protocolos de encaminamiento en este sentido, proporcionando diferentes alternativas en cuanto a arquitecturas de red y métodos de planificación de las comunicaciones. Los autores en [23] proponen una clasificación de los protocolos de encaminamiento en WSN en función de dos variables: la estructura de la red y la operación del protocolo, tal y como se muestra en la Figura 2.1.

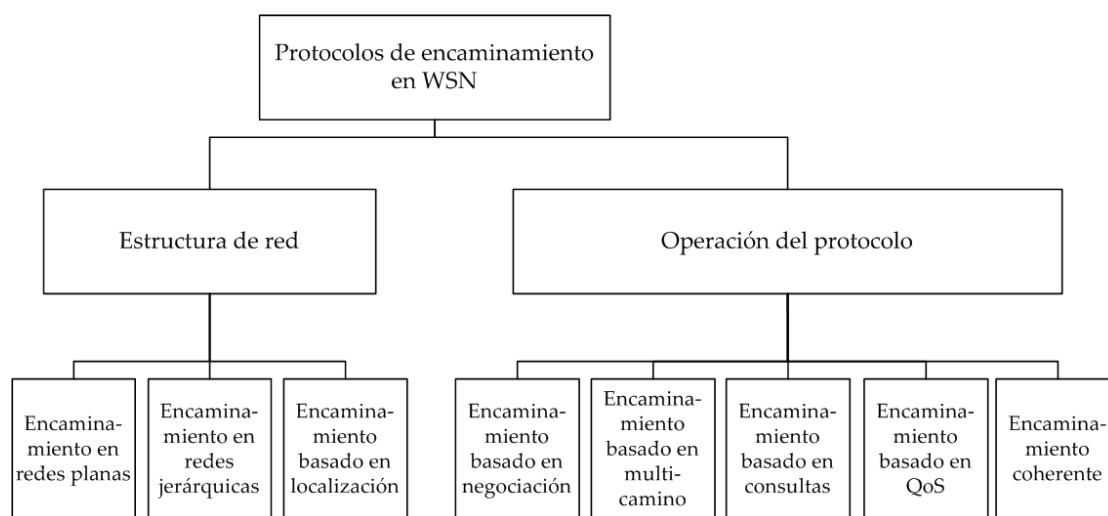


Figura 2.1: Taxonomía de protocolos de encaminamiento en WSN.

2.1.1. Clasificación en función del método de operación

En cuanto a la forma que tiene el protocolo de realizar las tareas de encaminamiento, pueden distinguirse cinco tipos distintos de protocolos en WSN:

1. **Encaminamiento basado en negociación** (*Negotiation based routing*): los nodos sensores intercambian diferentes mensajes de negociación con objeto de eliminar información duplicada y evitar de esta forma la transmisión de paquetes redundantes y la sobrecarga de la red producida por la recepción de varias copias duplicadas de los mismos datos.
2. **Encaminamiento basado en multi-camino** (*Multi-path routing*): con objeto de mejorar el rendimiento del protocolo, un nodo puede almacenar varias rutas o caminos diferentes hacia cualquier destino de la red, lo que ocasiona un aumento de la tolerancia ante fallos al existir rutas alternativas a una dada cuando ésta se rompe. Asimismo este tipo de encaminamiento proporciona mecanismos para el balanceo de la carga de la red. La existencia de diferentes rutas puede llevar implícito un aumento de la sobrecarga de la red con paquetes de control necesarios para el mantenimiento de las mismas.
3. **Encaminamiento basado en consultas** (*Query-based routing*): en este tipo de operación un nodo que desea conocer cierta información de la red inicia la propagación de una consulta a través de la misma, la cual, una vez que llega al nodo que contiene dicha información, es respondida enviando los datos solicitados al nodo origen.
4. **Encaminamiento basado en calidad de servicio** (*QoS-based routing*): este tipo de protocolos están obligados a mantener ciertas métricas de calidad de servicio en la red, como son el retardo extremo a extremo, el ancho de banda disponible o la energía consumida, a través del balanceo o reparto

uniforme de las transmisiones de los datos recogidos por la red y enviados al nodo sumidero.

5. **Encaminamiento coherente** (*Coherent-based routing*): la información monitorizada por sensores es transmitida a los nodos encargados de tareas de agregación de la información sólo después de realizar un mínimo procesamiento de la misma. Este tipo de actuación es una estrategia típica utilizada por algoritmos que deseen optimizar la eficiencia energética en las comunicaciones de la red.

2.1.2. Clasificación basada en la estructura de la red

La estructura de la red creada juega un papel muy importante en la operación de los protocolos en WSN. Atendiendo a este criterio, pueden existir tres tipos distintos de protocolos de encaminamiento:

1. **Encaminamiento plano** (*Flat routing*): todos los nodos de la red con excepción de la estación base o nodo sumidero tienen el mismo rol y, en consecuencia, juegan el mismo papel en el funcionamiento de la red, colaborando unos con otros para llevar a cabo las tareas de monitorización encomendadas.
2. **Encaminamiento jerárquico** (*Hierarchical routing*): denominado también encaminamiento basado en *clusters* y originalmente propuesto para redes cableadas, sus ventajas relativas a la escalabilidad de la red y la eficiencia en las comunicaciones son ampliamente conocidas en este tipo de redes. Particularmente para WSN, el encaminamiento jerárquico puede utilizarse según una arquitectura en la cual los nodos con mayor energía se ocupan de las transmisiones de la información recopilada a través la red, mientras que los de menor energía se reservan únicamente para realizar tareas de monitorización del entorno y envío de la misma al nodo con mayor capacidad

energética. La creación de *clusters* y la asignación de diferentes roles en ellos puede contribuir, por un lado, a la eficiencia energética de la red y, por otro, a la escalabilidad de la misma. Esta estructura es particularmente útil en WSN, donde la información monitorizada por los nodos pertenecientes al mismo *cluster* tiene generalmente redundancia espacial y/o temporal, por lo que la agregación o fusión de la información dentro del mismo permite un ahorro considerable de energía al reducir el número de mensajes transmitidos, sin penalizar necesariamente la cantidad de información transmitida hacia el nodo sumidero.

3. **Encaminamiento basado en localización** (*Location-based routing*): por medio de diferentes técnicas de intercambio de información entre vecinos, los nodos de la red pueden conocer la posición del resto de sensores y utilizarla para encaminar los paquetes. Adicionalmente, los nodos pueden disponer de un sistema de posicionamiento como GPS para conocer su localización exacta en la red.

A continuación se presentan los protocolos más importantes atendiendo a esta clasificación basada en la estructura de la red creada.

2.1.2.1. Protocolos con encaminamiento plano

Dado que todos los nodos sensores de la red tienen las mismas funcionalidades, realizarán las mismas tareas de procesado de la información y de encaminamiento de paquetes, colaborando unos con otros en la monitorización y reenvío de los mismos a lo largo de la red. En general, las tablas iniciales de encaminamiento en este tipo de redes planas se construyen mediante *flooding* o inundación de la red. Debido al número elevado de nodos, en este tipo de redes a menudo no es posible la utilización de un único identificador global para cada uno de ellos. Esta consideración ha llevado a la implantación del encaminamiento centralizado, en el

cual el nodo sumidero envía consultas a determinadas regiones y espera los datos enviados por los sensores de las zonas seleccionadas. Los datos se solicitan a través de consultas o *queries*, por lo que se requiere un esquema de direccionamiento mediante atributos para especificar las propiedades de esos datos.

Dependiendo del papel desempeñado por el nodo sumidero, el encaminamiento plano puede clasificarse en tres tipos [137]:

- **Modelo de inundación** (*Traditional flooding model*): comprende tanto la inundación tradicional como otras técnicas básicas como *Gossiping* [63] y *Random Walks* [28]. En él los nodos sensores no tienen conocimiento de la topología de la red, por lo que transmiten a sus nodos vecinos (a todos o a uno elegido de forma aleatoria) el paquete recibido hasta que éste alcance el nodo destino (generalmente el nodo sumidero) o se llegue al límite máximo de saltos marcado por el valor del campo TTL (*Time To Live*) del paquete. Estas técnicas simples, si bien resultan efectivas en cuanto al encaminamiento, son poco utilizadas debido a la sobrecarga excesiva que introducen en la red.
- **Modelo basado en eventos** (*Event-driven mode*): los nodos sensores reaccionan ante los eventos producidos en el área de monitorización e inician la transmisión de paquetes con esa información con destino el nodo sumidero. Como protocolos representativos de este modelo se pueden destacar los siguientes:
 1. *Sensor Protocols for Information via Negotiation (SPIN)* [65]: es una familia de protocolos cuyo objetivo es la diseminación eficiente de la información entre sensores a través de la negociación e intercambio de descriptores previos a la transmisión de los datos, denominados *meta-data*, que evitan la transmisión de información redundante en la red ya que sólo se transmite un paquete a un nodo vecino si éste no tiene

conocimiento de su descriptor asociado. Esta técnica trata de limitar la sobrecarga y solapamiento de paquetes en el tiempo a través de la red, causadas por los protocolos de inundación tradicionales. SPIN utiliza además algoritmos adaptativos que tienen acceso a la lectura del nivel energético remanente del nodo y adapta el funcionamiento del mismo en función de su valor.

2. *Rumor Routing* [39]: en este protocolo cada nodo mantiene una tabla de eventos compuesta por la descripción del evento, el nodo destino y el siguiente salto hacia él. Cuando el nodo sumidero lanza a la red un mensaje que contiene la descripción de un evento, éste se propaga mediante la técnica de Difusión Directa (descrita más adelante) hasta que encuentra un nodo que contiene en su tabla el evento difundido, el cual redireccionará directamente el mensaje hacia su destino. El objetivo, por tanto, es crear caminos que lleven a cada evento para que, cuando se genere una consulta o se difunda un evento por la red, ésta se retransmita de forma aleatoria entre los nodos hasta que encuentre la ruta del evento, momento en el cual es encaminada directamente hacia el destino. Este proceso es mucho más eficiente energéticamente que el proceso de inundación de SPIN aunque su rendimiento desciende al incrementarse el número de eventos en la red y ante frecuentes cambios en la topología de la misma debido al coste asociado al mantenimiento de las tablas de eventos en los nodos. Este protocolo fundamentalmente se aplica en escenarios con un gran número de consultas pero con un número reducido de eventos definidos.
3. *Energy-aware Routing* [120]: esta técnica de encaminamiento considera que si todos los paquetes de datos se transmiten únicamente a través de una o más rutas óptimas (las de menor consumo energético total), la energía de los nodos que las forman se agotará mucho más deprisa

que la de otros nodos de la red, lo que puede provocar la generación de zonas desconectadas. Por tanto, en el proceso de establecimiento de rutas, se crea también y se mantiene un número determinado de rutas denominadas subóptimas, siendo seleccionada la ruta final de transmisión en función de un modelo probabilístico que determina un valor de probabilidad para cada una de las rutas existentes. De esta forma se consigue un gasto más uniforme de la energía en la red y un tiempo de vida total de la misma mayor, así como una mayor tolerancia ante los fallos producidos. El inconveniente principal de este método es el coste que supone el intercambio de información de control entre los nodos en el proceso de construcción de las rutas.

- **Modelo basado en consultas** (*Query-driven mode*): en este modelo, el nodo sumidero pide una determinada información a los nodos sensores. Se pueden destacar los siguientes protocolos:
 1. *Directed Diffusion* [26], [71]: la Difusión Directa es otra técnica de inundación controlada en la que toda la información generada por los sensores se define mediante un par atributo-valor. Cuando el nodo sumidero desea adquirir cierta información de la red inicia la diseminación de una tarea como *interés* de cierto dato, el cual describe la tarea a realizar en la red y es difundido a sus vecinos por parte de cada nodo que lo recibe. Esta difusión configura *gradientes* en la red, los cuales especifican el valor de un atributo y su dirección, definiendo un esquema de eventos, es decir, una concordancia interés-dato. El objetivo principal de esta técnica es combinar los datos procedentes de diferentes fuentes para eliminar la redundancia en las transmisiones y con ello disminuir el consumo energético de los nodos. Su utilización se descarta para aplicaciones que requieran una constante entrega de datos al nodo sumidero debido a la inundación constante a la que se vería sometida la red ni en

aquéllas con alta carga de consultas. Esta utilización del mecanismo de inundación convierten a este protocolo en muy poco escalable y práctico para aplicaciones que requieran QoS por la imposibilidad de controlar la latencia y la sobrecarga en la red.

2. *Gradient-based routing* (GBR) [118]: es una variante de la Difusión Directa en la que en el proceso de transmisión de los mensajes de interés el algoritmo almacena el número de saltos del mismo, lo que permite calcular a cada nodo el número mínimo de saltos necesarios para alcanzar un destino determinado (generalmente el sumidero), llamado *altura* del nodo. La diferencia entre la altura de un nodo y la de su vecino es lo que se denomina *gradiente* del enlace. Los nodos retransmitirán los paquetes recibidos por el enlace que tenga el mayor gradiente. GBR usa algunas técnicas complementarias como agregación de datos y balanceo de carga con objeto de incrementar el tiempo de vida de una WSN. Además, este protocolo puede utilizar otros métodos de disseminación como *Energy-aware routing*, en el cual un nodo incrementa el valor de su altura cuando su energía disminuye por debajo de cierto umbral, con lo que dificulta que los demás nodos lo escojan para retransmitirle la información. Por último, la actualización de la altura de un nodo tiene como consecuencia la propagación de esta información en la red con el objetivo de mantener la consistencia de los gradientes. Esta propagación puede añadir una considerable sobrecarga a la red y provocar problemas de latencia y congestión, lo que se convierte en el inconveniente principal de este método al incrementar el consumo energético.

2.1.2.2. Protocolos con encaminamiento jerárquico

Las técnicas de encaminamiento jerárquico son originarias de las redes cableadas aunque su particularización para WSN ha ido enfocada principalmente a la

mejora de las prestaciones del protocolo en dos parámetros: la escalabilidad de la red y, sobre todo, la eficiencia energética de las comunicaciones. Estas prestaciones ofrecidas convierten a este tipo de protocolos en los más utilizados en WSN para aplicaciones de monitorización del entorno y seguimiento de eventos.

Debido a la restricción energética de los nodos, en una arquitectura jerárquica los nodos con mayor capacidad energética pueden ser empleados para recopilar, procesar, agregar y transmitir la información de una determinada zona de la red, mientras que los nodos con bajo nivel de energía se utilizan únicamente para labores de monitorización del entorno y recogida de datos. En particular, la jerarquización de la red mediante la creación de *clusters* y la asignación de diferentes roles a los nodos de la red pueden favorecer en gran medida la eficiencia energética de la red y la escalabilidad del sistema, prolongando su tiempo de vida útil. A los nodos que realizan tareas especiales se les denomina *Cluster-Head* (CH) y son los encargados de procesar, agregar y fusionar la información recogida dentro del *cluster* para su posterior envío a otros CH o al nodo sumidero.

A continuación se presentan los protocolos más representativos de este tipo.

1. *Low Energy Adaptive Clustering Hierarchy (LEACH)* [64]: este protocolo es el más representativo en WSN durante la última década, siendo comparado y referenciado por la mayoría de los trabajos de investigación relativos a protocolos de encaminamiento en este tipo de redes. LEACH está basado en la división de los nodos de la red en *clusters* y la elección de un conjunto de CH en función de una probabilidad, los cuales ejercerán este rol durante un tiempo determinado. Esta idea de organización la red en *clusters* no es novedosa, aunque sí el hecho de que la elección de los CH no sea permanente. Periódicamente, tras una fase de operación normal y de recogida de información por parte del sumidero, la estructura de la red es inicializada y se completa un nuevo proceso de selección de CH. La operación de la red

se divide por tanto en rondas o *rounds*, en las cuales el consumo energético de los CH es mayor al del resto. Al finalizar esa ronda se elige un nuevo conjunto de nodos para ejercer las funciones de CH (fase de configuración o inicialización); de esta forma, la energía consumida se distribuye de forma más equitativa entre todos los nodos de la red. La probabilidad de elección de un nodo como CH está relacionada con un valor predefinido por la aplicación como es el número deseado de CH en cada ronda de operación, y con el número de rondas que hace que ese nodo no ejerce este rol, otorgando así una mayor probabilidad de elección cuanto mayor sea el número de rondas que transcurridas desde su última elección. Una vez que se han seleccionado los CH, el resto de nodos se asociarán a uno de ellos atendiendo a razones de eficiencia energética (generalmente, se unirán al CH más cercano). LEACH emplea las técnicas TDMA y CDMA para reducir las colisiones intra-*cluster* e inter-*cluster*, resultando muy eficiente energéticamente en aplicaciones de monitorización del entorno y envío de datos al nodo sumidero. El CH es responsable dentro de su *cluster* de planificar las transmisiones y de agregar al información recibida, asignando un *slot* temporal de la trama TDMA a cada uno de los nodos sensores asociados para el envío de la información recogida. Con esta forma de actuar se reduce el coste total de las comunicaciones y se extiende el tiempo de vida útil de la red. Sin embargo, LEACH posee algunos inconvenientes que hacen que durante los últimos años hayan surgido propuestas de mejora y nuevos protocolos basados en esta técnica de *clusters* rotativos, incluyendo el protocolo propuesto en esta Tesis. Uno de estos inconvenientes es que la elección de los CH puede no estar distribuida uniformemente por la red, existiendo la posibilidad de concentración de una gran parte de ellos en una zona determinada, lo que posibilita la aparición de nodos huérfanos incapaces de asociarse con ningún CH debido a la distancia que los separa. Otro inconveniente es la carga extra de paquetes de control necesaria en la fase de inicialización y elección de rol y en la pla-

nificación de los distintos *clusters*. La transmisión directa al nodo sumidero que los CH realizan tampoco favorece la reducción del consumo energético, siendo necesaria la creación de estructuras multi-salto entre los diferentes CH para alcanzar el nodo sumidero. Por último, un aspecto determinante que no favorece el uso de este protocolo es que no tiene en cuenta la energía remanente de los nodos a la hora de la elección de los CH, consideración que resulta fundamental para que el gasto energético de los nodos sea uniforme y se pueda extender de esta forma la vida útil de la red. Esta última desventaja ha dado lugar a numerosas propuestas de métodos de selección de CH, tales como utilizar técnicas *Fuzzy Logic* [107], *Based Energy Clustering* (BEC) [138], LEACH-TM [139], EAP [89], o MR-LEACH [50], entre otros muchos.

2. *Low Energy Adaptive Clustering Hierarchy-Centralized (LEACH-C)* [66]: los mismos autores de LEACH propusieron algunos años más tarde una mejora del protocolo original que utiliza un algoritmo distribuido para la formación de los *clusters* y que proporciona una distribución más uniforme de los CH en la red. Para ello el nodo sumidero tiene en cuenta en la fase de inicialización de cada ronda la posición de cada nodo sensor en red, la cual es recogida por un sistema de posicionamiento (tipo GPS o similar) y enviada por cada uno de ellos al comienzo de esta fase, añadiendo en este envío su nivel de energía remanente. Estos dos parámetros son utilizados por el sumidero para una elección mucho más eficiente de los CH con respecto a LEACH, impidiendo la asignación de este rol a aquellos nodos cuya capacidad energética esté por debajo de la media de los nodos de su entorno. Para ello LEACH-C hace uso de un algoritmo [29] que resuelve el problema NP-completo de encontrar los k *clusters* óptimos. Una vez elegidos los CH, el nodo sumidero difunde un mensaje en la red con el identificador de cada uno de ellos para que inicien la

asociación del resto de nodos a su *cluster* y la planificación de sus respectivas actividades.

3. *Power-Efficient Gathering in Sensor Information Systems (PEGASIS)* [86]: teniendo en cuenta la sobrecarga de paquetes necesaria para formar un *cluster* en LEACH, PEGASIS introduce una variación de este protocolo para que en lugar de *clusters* se forme una cadena entre los diferentes sensores de la red. Dicha cadena puede ser formada a instancias únicamente de los propios nodos usando algún algoritmo definido o, por el contrario, es el sumidero quien se encarga de calcularla y de difundir dicha información a los nodos. La información en la cadena se transferirá de nodo en nodo, agregándose en cada uno de ellos la información recibida junto con sus propios datos antes de transmitirse al siguiente, hasta llegar a un nodo determinado previamente que actuará como pasarela y se encargará de transmitirla hacia el sumidero. Este último nodo es rotativo con objeto de distribuir por igual el gasto energético entre todos los nodos de la red. A pesar de que PEGASIS disminuye la sobrecarga de paquetes producida en la formación de *clusters*, su principal desventaja radica en que el tiempo de transmisión de los datos hasta alcanzar el nodo sumidero puede prolongarse en exceso, lo que constituye un verdadero problema de retardo en redes con gran número de sensores, limitando también la escalabilidad. Para minimizar este problema, *Hierarchical-PEGASIS* mejora este algoritmo permitiendo la transmisión simultánea de varios nodos cuando su distancia sea lo suficientemente grande como para que no se produzcan colisiones ni interferencias entre ellos. Otro inconveniente que presenta esta estructura es que no se tiene en cuenta la energía del siguiente nodo de la cadena cuando se elige un camino de transmisión, por lo que este protocolo no se considera aconsejable en redes con mucha carga de información hacia el nodo sumidero.

4. *Threshold-sensitive Energy Efficient Protocols (TEEN [92] y APTEEN [93])*: estos protocolos han sido creados para su uso en aplicaciones en las cuales se precisa de una respuesta rápida frente a variaciones repentinas en los parámetros que están siendo monitorizados por los sensores. Basados en la filosofía de LEACH de formación de *clusters* aunque sin rotación de roles, una vez formados éstos en TEEN los nodos monitorizan el medio continuamente obteniendo las lecturas de ciertos parámetros, pero la transmisión de la información hacia el CH se realiza únicamente si estos valores medidos superan ciertos umbrales (inferior y superior) definidos previamente por los CH. De esta forma, se reduce el número de transmisiones al limitarlas a las necesarias para enviar el valor del parámetro medido cuando éste se encuentre fuera de un rango de interés. APTEEN (*Adaptive-TEEN*) varía periódicamente los umbrales utilizados en este protocolo de acuerdo con las necesidades de la aplicación para responder de forma más eficiente a eventos críticos.
5. *Self Organizing Protocol (SOP) [126]*: este protocolo hace hincapié en la distribución de diferentes roles en la red con objeto de que cada nodo cumpla una función específica en la misma. Esta distinción de roles permite separar los procesos de encaminamiento de los procesos de monitorización. Así, SOP define cuatro tipos de roles diferentes: (i) *Nodo sumidero*, con alta capacidad de computación, conexión con otras redes y sin limitación energética alguna; (ii) *Sensor especializado en monitorización*, capaz de recoger datos del entorno y comunicarlos a otros nodos, pudiendo moverse libremente por la red; (iii) *Nodo encaminador*, inmóvil, con capacidad de auto-organizarse con otros de su clase y formar estructuras jerárquicas de transmisión mediante un algoritmo que consta de 4 fases (descubrimiento, organización, mantenimiento y auto-reorganización); y (iv) *Nodo agregador*, que aporta funciones de fusión y agregación de la información que recibe, comprimiéndola antes

de su envío a otros nodos. En SOP se permite que los nodos desempeñen algún otro rol junto con el de encaminador para simplificar la complejidad de la red. Si bien esta estructura permite dotar al protocolo de mayor tolerancia a fallos y mejorar la eficiencia energética al distribuir las tareas entre los diferentes nodos, la sobrecarga extra debida al mantenimiento de dicha estructura es un lastre considerable para el consumo, ya que el protocolo actúa de forma proactiva para llevar a cabo esta tarea.

6. *Energy-Efficient Distributed Unequal Clustering (EEDUC)* [82]: esta propuesta trata de optimizar el problema del *hotspot* o punto caliente en WSN. Este problema consiste en que, cuando el radio de transmisión de los nodos es fijo, la cantidad de paquetes retransmitidos por los nodos se incrementa notablemente a medida que la distancia hacia el sumidero es menor. En consecuencia, los nodos más cercanos al sumidero tienen que tratar con un tráfico mucho mayor de paquetes y consumen su energía de forma mucho más rápida. Anteriormente EEUC (*An Energy-Efficient Unequal Clustering*) [83] había tratado este problema y propuesto que los *clusters* más cercanos al sumidero fuesen más pequeños para que sus CH consumieran menos energía en tareas intra-*cluster* con objeto de compensar este mayor consumo debido a las retransmisiones inter-*cluster*. EEDUC, a diferencia de LEACH, selecciona los CH por competición local y no por probabilidad, y en la estructura inter-*cluster* se centra en transiciones multi-salto y en el equilibrio del tráfico a la hora de seleccionar el siguiente CH de la ruta, lo que mejora sensiblemente las prestaciones con respecto a LEACH.
7. *Energy-Balancing Unequal Clustering Protocol (EB-UCP)* [145]: este protocolo se basa en la premisa de *clusters* desiguales proporcionada EEUC en la que los *clusters* más cercanos al sumidero deberán tener un menor número de nodos asociados. Para ello EB-UCP divide a los nodos de la red en capas concéntricas cuyo centro es el sumidero y asigna a cada uno de ellos una

probabilidad de ser CH que es menor a medida que se encuentre en una capa más alejada del centro, de tal forma que una probabilidad baja en capas alejadas significa un menor número de CH en esa capa y por tanto que el CH deberá cubrir un área mayor, que normalmente contendrá más nodos sensores. Por el contrario, los nodos de las capas cercanas al sumidero tendrán una alta probabilidad de ser CH, con lo que en esas capas existirá un porcentaje mayor de CH que en otras y, en consecuencia, el número de nodos sensores asociados a cada *cluster* disminuirá. Con objeto de balancear el consumo energético de los CH, cada uno de ellos seleccionará en la ruta multi-salto hacia el sumidero al CH cuya energía remanente sea mayor de entre sus vecinos de la capa anterior. El objetivo final de esta técnica es que el consumo total en cada capa sea similar, teniendo en cuenta tanto las transmisiones inter-*cluster* como las intra-*cluster*.

8. *Energy-efficient Adaptative Hierarchical and Robust Architecture* (EDETA) [42]: diseñada para minimizar el consumo energético de la red, esta estructura de encaminamiento se basa en una jerarquía en dos niveles, uno inferior basado en *clusters* y otro superior formado por un árbol jerárquico dinámico entre todos los CH y el nodo sumidero. Esta estructura jerárquica creada sigue la filosofía de *clusters* desiguales en función del nivel de cada CH, de su energía remanente y del número de hijos asociados, con el objetivo principal de minimizar y balancear la carga energética en la red. EDETA es escalable, autoconfigurable y proporciona características de tolerancia a fallos y tiempos de latencia acotados.

2.1.2.3. Protocolos con encaminamiento basado en localización

Este tipo de algoritmos requiere el conocimiento de información relativa a la posición de los nodos de la red para poder llevar a cabo las labores de encamina-

miento. Los nodos de la red pueden obtener la información acerca de su localización tanto de forma absoluta, equipando a los sensores con un receptor GPS de bajo coste, como de forma relativa, calculando su posición con respecto a sus vecinos a través de la distancia que les separa de ellos (la cual se estima en función de la potencia de la señal recibida de cada uno).

Entre los principales protocolos de encaminamiento de este tipo en WSN, se pueden destacar los siguientes:

1. *Geographic and Energy Aware Routing (GEAR)* [27]: la técnica propuesta en este protocolo se basa en restringir el número de mensajes propagados por la red mediante Difusión Directa añadiendo información geográfica en el mensaje, de tal forma que se propague únicamente en la zona geográfica de interés predeterminada de antemano. A este aspecto se le une la selección del siguiente nodo vecino en función del resultado de comparar la energía remanente de todos ellos. GEAR introduce los conceptos de coste estimado y de coste aprendido, que son calculados y mantenidos en cada nodo con objeto de alcanzar el destino a través de la elección del siguiente salto entre sus vecinos en función de la diferencia entre ambos valores. El coste estimado es una combinación de energía residual y distancia hasta el destino, mientras que el coste aprendido es un refinamiento del coste estimado que tiene en cuenta el encaminamiento alrededor de agujeros en la red (cuando un nodo no tiene ningún vecino cercano a la región de destino).
2. *Minimum Energy Communication Network (MECN)* [111]: inicialmente diseñado para redes inalámbricas, MECN fue posteriormente aplicado a WSN debido a su capacidad para ahorrar energía. Su principio se basa en las comunicaciones multi-salto, en donde el coste de una comunicación directa entre dos nodos es mayor que el generado si se introducen nodos retransmisores entre ambos. En consecuencia, MECN identifica una región de retransmi-

sión para cada nodo de la red en función de su posición en la misma, construyéndose una estructura de nodos reencaminadores entre los sensores y el sumidero cuyas comunicaciones son energéticamente más eficientes que la transmisión directa. Cuando un nodo necesita transmitir un mensaje a un destino, el protocolo elige un camino de mínima energía de acuerdo con el algoritmo de Bellman-Ford de búsqueda del camino más corto. MECN es autoconfigurable y se ajusta dinámicamente a los fallos de red y al despliegue de nuevos sensores. Aunque puede utilizarse en redes con sensores móviles, el coste energético de calcular el camino óptimo en la región de retransmisión cuando hay movilidad se incrementa notablemente.

3. *Geographic Adaptive Fidelity (GAF)* [30]: al igual que MECN, es un protocolo definido inicialmente para redes ad hoc inalámbricas. En la fase de configuración la red se divide en una serie de zonas fijas, formando una malla virtual. Dentro de cada zona los nodos se distribuyen diferentes tareas en función del rol asignado, siendo uno de ellos el encargado de monitorizar el entorno y enviar los datos al sumidero como representante de esa zona, mientras que otros nodos pueden estar en estado de baja energía o *sleep* para ahorrar energía. Periódicamente este rol cambia para mantener equilibrada la carga energética de la zona. Cada nodo utiliza el sistema GPS para asociarse con los nodos de una zona. Los resultados de este protocolo muestran una alta eficiencia energética debido a este gasto uniforme de los nodos, lo que hace que se prolongue considerablemente el tiempo de vida de la red comparado con el de otros protocolos como AODV [103].

2.2. Criterios de diseño

Las características particulares de las WSN, cuyos nodos sensores están sometidos a importantes restricciones en cuanto a sus capacidades energéticas, de

procesado, de transmisión y de almacenamiento, hace que el diseño eficiente de protocolos de encaminamiento sea un factor crítico para prolongar la vida útil de este tipo de redes [106]. Además, la funcionalidad y el diseño de estos protocolos puede ser diferente en función de la aplicación o la arquitectura de la red, ya que no sólo se limita el alcance de este tipo de redes a la monitorización del entorno, tal y como se expone en [23], con lo que puede no ser eficiente aplicar un único algoritmo a todos los escenarios y aplicaciones posibles. En consecuencia, el encaminamiento en WSN se convierte en un reto que necesita de ideas innovadoras que despierten a la luz de los nuevos avances. A pesar de que los modelos de red existentes para MANET tienen ciertas similitudes con los de WSN, como pueden ser la existencia de una topología variable, las restricciones energéticas o enlaces inalámbricos con variabilidad, los protocolos y técnicas de encaminamiento desarrollado para este tipo más general de redes inalámbricas no pueden ser directamente aplicados a las WSN debido a las **propiedades intrínsecas de las mismas que las diferencian**, como son las siguientes [106], [125]:

- Las WSN están enfocadas principalmente a la monitorización del entorno, mientras que las MANET son diseñadas para computación distribuida más que para recogida de información.
- El número de nodos sensores en WSN puede ser varios órdenes de magnitud mayor que en las MANET.
- Generalmente las WSN son desplegadas por un único propietario, mientras que en las MANET este proceso puede incluir varios sujetos independientes. Además, teniendo en cuenta el punto anterior, en el primer caso los nodos se despliegan una única vez debido a su imposibilidad de mantenimiento, y su vida útil viene determinada por el tiempo de duración de su batería, mientras que los nodos de las MANET pueden insertarse o retirarse de la red tantas veces como sea necesario.

- Debido a la razón anterior, a diferencia de los nodos de las MANET, un nodo sensor puede no tener un único identificador o ID.
- Los nodos sensores son mucho más baratos que los nodos de las MANET y pueden llegar a desplegarse miles de ellos. Esto es debido, entre otros factores, a la menor capacidad de procesado y almacenamiento de los sensores.
- La capacidad energética de los nodos sensores suele ser muy limitada, mientras que los nodos de las MANET generalmente pueden ser recargados de alguna forma cuando su batería se agota.
- El flujo de paquetes en las MANET es *peer to peer*, es decir, que puede ser creado entre cualquier par de nodos de la red, mientras que las WSN generalmente cursan tráfico en dos direcciones, *upstream* y *downstream*.
- El requisito fundamental de una WSN es el consumo eficiente de energía para prolongar el tiempo de vida de la red, mientras que en una MANET generalmente las necesidades principales son la optimización de la QoS y la eficiencia en el uso del ancho de banda disponible.
- Por criterios de eficiencia energética, en WSN no siempre es posible alcanzar la misma redundancia de caminos que la disponible en redes MANET, lo que limita la tolerancia a fallos de las redes de sensores [96].

En los últimos años el diseño de protocolos de encaminamiento en WSN se ha convertido en un área importante de investigación, tratando de diseñar mecanismos energéticamente eficientes, de bajo coste y tolerantes a los fallos producidos en la red que tengan en cuenta las restricciones de los dispositivos que la forman. La definición de mecanismos que puedan gestionar las comunicaciones de la red evitando la degradación de la conectividad entre los nodos y aplicando técnicas de gestión energética se ve obligada a afrontar unos retos específicos inherentes a la propia estructura de las WSN. Basándose en las ideas presentadas en [137], [23]

y [38], se puede decir que estos retos definen los siguientes **criterios de diseño de un protocolo de encaminamiento en WSN**:

- *Restricciones hardware*: un nodo sensor necesita para funcionar, como mínimo, una unidad de monitorización con sensores, una unidad de procesado y memoria, una de transmisión y una fuente de alimentación. Opcionalmente, a esta configuración mínima pueden añadirse otros sensores y dispositivos tales como un sistema de localización para complementar las funcionalidades del nodo. Sin embargo, hay que tener en cuenta que estas nuevas funcionalidades llevan asociado un coste adicional que incrementa tanto el consumo energético como el tamaño físico y el coste final de los nodos. Es necesario, en consecuencia, proporcionar mecanismos que permitan equilibrar esta mejora de funcionalidad con el coste y los requisitos de bajo consumo de los nodos.
- *Flexibilidad*: el protocolo de encaminamiento debe ser capaz de adaptarse a los diferentes roles que pueden desempeñar los nodos en la red, así como a diferentes topologías y a los diferentes casos que puedan presentarse en cuanto a la existencia de uno, varios o ningún nodo sumidero.
- *Auto-configuración de la red*: el despliegue de WSN se efectúa en función de la aplicación, pudiendo ser manual (determinista) o aleatorio. En el primero de ellos, los paquetes pueden ser encaminados por rutas predeterminadas, mientras que en el despliegue aleatorio se crea una infraestructura ad hoc que necesita de mecanismos de auto-configuración y optimización que proporcionen conectividad a todos los sensores y una operación energéticamente eficiente. Este aspecto adquiere mayor importancia si se tiene en cuenta el bajo o nulo mantenimiento que puede hacerse en este tipo de redes, unido a la necesidad de poder efectuar nuevos despliegues de nodos mientras la red se encuentra inmersa en la fase de operación.

- *Consumo de energía optimizado*: como ya se ha indicado, el uso de mecanismos de ahorro de energía en la operación normal de los nodos es esencial para prolongar el tiempo de vida de la red. Este criterio cobra más fuerza si cabe al utilizar una estructura multi-salto en las rutas, en la que cada nodo juega un doble papel, como nodo sensor que monitoriza el entorno y como encaminador de otros paquetes. En estas circunstancias, el fallo de un nodo por el agotamiento de su batería puede implicar la rotura de rutas hacia el destino y con ello ocasionar cambios significativos en la topología de la red que requieran la iniciación de mecanismos de reconfiguración, introduciendo cierta sobrecarga.
- *Adaptación al método de captura de datos*: los métodos de recogida de información en una WSN son altamente dependientes de la aplicación y de la criticidad de los datos. Las distintas maneras de operación de una red en este sentido son: *time driven*, para monitorización periódica de parámetros; *event driven*, para cambios repentinos en las condiciones del entorno; *query driven* para responder a consultas generadas por el sumidero, o un híbrido de ellas. Esto hace que los nodos sensores deban adaptarse a la criticidad de los datos recogidos y al tipo de tráfico cursado, por lo que el protocolo debe tener en cuenta estos aspectos a la hora del cálculo de rutas y el ahorro de energía de los nodos.
- *Adaptación a la heterogeneidad de nodos y enlaces*: dependiendo de la aplicación, puede no siempre ser eficiente el despliegue de nodos con las mismas características y mismas capacidades energéticas y de computación, resultando en muchos casos beneficioso o necesario la introducción de nodos con capacidades mayores al resto y que jueguen papeles diferentes dentro de la red., como por ejemplo en WSN con actuadores (WSAN, *Wireless Sensor and Actor Network*). Es más, la propia muerte de algunos nodos por agotamiento de sus baterías puede llevar asociado el despliegue de otros nuevos

que ocupan su lugar en la red (en [123] los autores se refieren a esta situación como *re-energization of sensor networks*), lo que convierte este escenario en heterogéneo. La existencia de diferentes condiciones de propagación en los enlaces inalámbricos es otra propiedad que debe tenerse en cuenta a la hora de diseñar un protocolo de encaminamiento en WSN, y no sólo entre distintos enlaces, sino también por la propia asimetría en el comportamiento de cada uno de ellos.

- *Tolerancia a fallos de red*: los nodos sensores pueden fallar por varias causas, como agotamiento de su batería, mal funcionamiento, daños causados por agentes externos o simplemente no pueden ser utilizados en una ruta por estar en estado de baja energía (*sleep*). Además, debido a que las rutas multi-salto creadas están compuestas por enlaces inalámbricos, están presentes los problemas tradicionales asociados con este tipo de enlaces, como son las interferencias con otros enlaces, la existencia de obstáculos, el desvanecimiento o las pérdidas debidas al multi-trayecto. En consecuencia, el protocolo diseñado debe poder afrontar estas situaciones con eficacia proporcionando mecanismos de creación de rutas alternativas o de reparación de los caminos rotos, todo ello afectando lo menos posible al funcionamiento normal de la red y limitando la sobrecarga de paquetes y el consumo energético asociado a dichos procesos.
- *Escalabilidad*: este factor es de suma importancia debido al gran número de sensores que pueden desplegarse en una WSN. Cualquier protocolo que pretenda ser eficiente debe poder gestionar este gran número de dispositivos y adaptarse a situaciones de ampliación de la red o de conexión de la misma con otras WSN.
- *Movimiento de nodos sensores*: las primeras aplicaciones de las WSN contemplaban nodos fijos destinados a labores de monitorización del entorno.

Sin embargo, con el auge de este tipo de redes han ido surgiendo nuevas aplicaciones como pueden ser la detección o seguimiento de blancos (*target tracking*) en las que tanto el nodo sumidero como los nodos sensores pueden moverse libremente por la red. Este escenario implica un reto muy importante para los algoritmos de encaminamiento ya que necesitan implementar métodos de reconfiguración autónoma y automática para garantizar la estabilidad de la red y de las rutas creadas, siempre con la vista puesta en la eficiencia energética y la limitación de la sobrecarga en la red.

- *Adaptación al medio inalámbrico*: además de gestionar eficientemente la situación cuando se producen fallos en los enlaces o aumentos de la BER, y a pesar de que en general el ancho de banda requerido en WSN es bajo ya que suele encontrarse entre 30 y 250 Kbps para las principales aplicaciones, el protocolo de encaminamiento debe poder proporcionar técnicas de transmisión simultánea de los nodos de la red, teniendo en cuenta la existencia de un gran número de ellos y las limitaciones de retardo impuestas por la aplicación en la recogida de la información monitorizada. Esto implica que en ocasiones han de crearse estructuras tales como tramas TDMA que evitan colisiones entre nodos, permiten acotar en mayor medida la latencia de la red y son más eficaces en cuanto a la utilización del ancho de banda disponible.
- *Conectividad de la red*: a pesar de que inicialmente una red tenga una alta densidad de nodos sensores, con el paso del tiempo la muerte de algunos de ellos puede llevar asociada la existencia de nodos o zonas aisladas que pierden su conectividad con el resto de la red. Este hecho debe ser tenido en cuenta por el protocolo de encaminamiento para proporcionar mecanismos capaces de prevenir y tratar estas situaciones.
- *Cobertura*: el área cubierta por una WSN es un parámetro de diseño, dado que la densidad necesaria de nodos está limitada por la capacidad que tiene

cada uno de ellos de monitorizar una zona de radio restringido, obteniendo una vista muy limitada de la red.

- *Agregación de datos*: debido a factores tales como la correlación espacial o temporal de los datos monitorizados, en muchas ocasiones la información recogida por varios nodos cercanos entre sí puede contener cierta redundancia, por lo que debe ser procesada, agregada y fusionada con objeto de reducir el número de transmisiones en la red. Mediante estas operaciones de agregación y de supresión de duplicados un nodo puede ser capaz de retransmitir toda la información recibida de otros nodos vecinos en un único paquete de datos, optimizando de esta forma el consumo energético.
- *Calidad de servicio (QoS)*: existen aplicaciones de las WSN con restricciones en cuanto al retardo de transmisión de la información desde que es monitorizada hasta que alcanza el nodo sumidero. Estas aplicaciones imponen límites en la latencia de la red que a menudo se contraponen con la utilización de mecanismos de eficiencia energética en la operación de la misma. Un ejemplo de ello es la utilización de algoritmos de encaminamiento proactivos que realizan un mantenimiento continuo de la estructura de rutas y se adaptan automáticamente a los cambios producidos en la red, pero cuya operación implica un mayor gasto energético debido a la necesidad de intercambio de paquetes de control para llevar a cabo dichas tareas. En todo caso, son necesarios mecanismos dentro del diseño del protocolo que tengan en cuenta la energía de los nodos en cada momento (lo que conduce a los denominados protocolos *energy-aware*) con objeto de suavizar las restricciones de QoS impuestas si en un momento dado existiesen problemas serios de energía que impliquen una reducción en el tiempo de vida de la WSN.
- *Seguridad*: debido a la accesibilidad total al medio inalámbrico por parte de cualquier usuario, es importante proporcionar mecanismos que impidan que

los mensajes transmitidos sean espiados o manipulados por entidades ajenas a la WSN y se vea de esta forma comprometida la información obtenida. Las técnicas tradicionales de encriptación no son, en general, prácticas en WSN debido a las limitaciones de energía y capacidad de procesamiento de los sensores [116]. Aunque no es propiamente una función del protocolo de enrutamiento, debe tenerse en cuenta este aspecto a la hora de diseñar la estructura de mensajes y el intercambio de información entre los nodos. Más concretamente, los aspectos que tienen alguna relación con la seguridad en WSN son la confidencialidad de los datos, su autenticidad, su integridad y su frescura [49].

2.3. Conclusiones

A la vista de la clasificación de la Sección 2.1, se hace necesaria una comparativa inicial de las ventajas e inconvenientes de cada uno de los tipos de enrutamiento principales que sirva de punto de partida para la elección de un modelo de enrutamiento eficiente en WSN y como justificación de la elección tomada relativa al protocolo propuesto en esta Tesis. Con este objetivo, en la Tabla 2.1 se resumen las principales características de los protocolos jerárquicos con respecto a los protocolos planos, obviándose la comparativa con los protocolos basados en localización por el coste energético excesivo en el que incurren durante su funcionamiento, tal y como se justifica en [42] y [23].

Para complementar este análisis puede consultarse la tabla comparativa incluida en [23], que incluye un estudio detallado de los protocolos más representativos de cada una de las estructuras de enrutamiento en función de diferentes parámetros, tales como eficiencia energética, escalabilidad, agregación de datos, movilidad, localización, técnicas de negociación, multi-camino, etc.

Encaminamiento jerárquico	Encaminamiento plano
Planificación basada en reserva	Planificación basada en contienda
Se evitan colisiones	Sobrecarga por colisiones
Ciclos de trabajo reducidos debido a modos de bajo consumo periódicos	Ciclos de trabajo variables
Agregación de datos realizada por el CH	Agregación de los datos procedentes del vecino en la ruta multi-salto
Encaminamiento simple pero no óptimo	Encaminamiento puede ser óptimo pero introduciendo complejidad añadida
Requiere sincronización entre nodos	Los enlaces se crean sin necesidad de sincronización
Sobrecarga inicial en la formación de la estructura jerárquica (<i>cluster</i>)	Sobrecarga cuando se construyen las rutas, únicamente en las regiones que requieren transmisión de datos
Baja latencia en la comunicación multi-salto a través de los CH	Latencia mayor mientras se configuran las rutas
Consumo de energía uniforme	Consumo de energía en función de los patrones de tráfico
Reparto equitativo del canal de transmisión	No se garantiza el reparto óptimo del canal

Tabla 2.1: Comparativa entre encaminamiento jerárquico y plano.

A la vista de estas tablas, puede concluirse que el encaminamiento jerárquico posee mayor capacidad para evitar colisiones y para la agregación de datos que las redes planas, aspectos ambos muy importantes a la hora de seleccionar un protocolo. Pero, sobre todo, las redes jerárquicas tienen mayor capacidad para minimizar el gasto de energía total en la red y para distribuir uniformemente este consumo entre todos los nodos, aspectos éstos fundamentales para prolongar la vida útil de la red. La escalabilidad de la red es otro factor decisivo que los protocolos jerárquicos favorecen en mayor medida frente a los planos. En consecuencia, puede afirmarse que, considerando términos de eficiencia energética, sobrecarga y escalabilidad, los protocolos jerárquicos obtienen mejores prestaciones que el resto de estructuras, especialmente acentuadas en las siguientes circunstancias:

- Existencia de diferentes roles en la red.
- Cambio periódico del rol de los diferentes nodos.
- Utilización de técnicas de acceso al medio que evitan las colisiones y adaptan las transmisiones al ancho de banda disponible.
- Posibilidad de períodos de bajo consumo (*sleep*) de los nodos.
- División de la operación de la red en varias fases.
- Construcción de estructuras multi-salto capaces de minimizar el consumo energético con respecto a las transmisiones directas.
- Existencia de procedimientos eficientes de agregación de datos.

Capítulo 3

Descripción de HARP

El objetivo principal de esta Tesis es proponer y analizar un nuevo protocolo de encaminamiento jerárquico para WSN. En este capítulo, se comenzará definiendo en la Sección 3.1 la arquitectura general propuesta por HARP y los objetivos perseguidos en el diseño de la estructura básica del protocolo. A continuación, en la Sección 3.2 se presentará detalladamente el conjunto de mensajes intercambiados por los nodos de la red necesarios para una correcta operación del protocolo. La Sección 3.3 se centra en la fase de inicialización de la red a través de la construcción del árbol jerárquico entre los nodos implicados, describiendo en detalle la estructura de tablas propuesta, que es la base para conseguir las prestaciones deseadas en HARP. Una vez finalizada la fase de inicialización de la red, la operación normal del protocolo se define en la Sección 3.4, haciendo hincapié en las capacidades que posee el protocolo para el encaminamiento del tráfico no sólo en sentido descendente (nodo raíz-nodos hojas) sino para cualquier tipo de aplicación de intercambio de mensajes en la red. Por último, la Sección 3.5 se centra en la descripción de los métodos y procedimientos definidos en HARP para la gestión de fallos en la red y la reconfiguración dinámica de rutas dentro de la misma.

3.1. Arquitectura general del protocolo

La definición de la arquitectura de este nuevo protocolo de encaminamiento está basada en las conclusiones presentadas en la Sección 2.3, las cuales hacen referencia al aumento de prestaciones que los protocolos jerárquicos proporcionan en las WSN. En definitiva, la estructura propuesta trata de maximizar algunos de los parámetros que miden las prestaciones de una red, tales como la eficiencia energética, la disponibilidad, la fiabilidad, la sobrecarga de paquetes, la escalabilidad y el retardo en la entrega de paquetes, entre otros. Las características principales relacionadas con la estructura del protocolo se desarrollan a continuación.

3.1.1. Encaminamiento multi-salto

El modelo de un salto es el modelo más simple de transmisión entre dos nodos de la red, representando la comunicación directa entre ambos. Al existir una dependencia de la energía consumida en la transmisión con respecto a la distancia al nodo destino, éste es un modelo caro en términos de consumo energético, así como inviable para grandes tamaños de red si se tiene en cuenta que los nodos tienen un rango de transmisión limitado y que no siempre son capaces de alcanzar el destino. Además, para los nodos de la red que transmitan al mismo tiempo y utilicen la misma técnica, se producirán irremediablemente colisiones e interferencias en dichas comunicaciones debidas a los altos radios de transmisión necesarios. Todo esto hace que la comunicación directa no sea, en general, una buena solución para redes inalámbricas y en particular para WSN. Además, en estas últimas suele existir correlación espacial y temporal en los eventos detectados, haciendo que varios nodos sensores cercanos entre sí deban transmitir simultáneamente los datos recogidos del entorno al nodo sumidero, pudiendo ocasionar colisiones con el consiguiente considerable gasto energético.

En contraposición a este método se encuentra el modelo de encaminamiento multi-salto o *multi-hop routing*. En esta técnica de comunicación entre dos nodos el emisor utiliza un nodo vecino más cercano al destino como nodo intermedio para el reenvío del paquete de comunicaciones y éste, a su vez, repite el mismo proceso hasta que dicho paquete alcance el nodo destino. Los paquetes, por tanto, se transmiten de nodo en nodo hasta llegar al destino y, en consecuencia, la energía total gastada se reparte entre todos los nodos intermedios, limitándose además la interferencia que una de estas transmisiones produce en el resto debido a su menor radio de transmisión con respecto al caso de una comunicación de un único salto. Estos dos factores, en suma, favorecen tanto el reparto y la optimización del consumo energético en la red como la reutilización del espectro de transmisión en la misma.

3.1.2. Encaminamiento multi-ruta

En el Capítulo 2, relativo al estado del arte en encaminamiento en WSN, se han referenciado trabajos de investigación que ponen de manifiesto las **ventajas de utilizar comunicaciones multi-ruta** o *multi-path routing* entre un origen y un destino a la hora de maximizar las prestaciones del protocolo. Existe una multitud de trabajos previos que describen estas ventajas, tanto para para redes inalámbricas como para redes cableadas, entre los cuales destacan [97], [45]), [54], [94], [99] y [133]. Entre estas ventajas se pueden resaltar las siguientes:

- Favorece una distribución más uniforme del tráfico en la red y un mayor balanceo de carga.
- Posibilita la agregación de ancho de banda en las comunicaciones: un flujo de datos puede ser dividido en varios flujos más pequeños, transmitiéndose

cada uno de ellos por una ruta diferente (por ejemplo, si existen flujos de paquetes de distintas aplicaciones).

- Proporciona una mayor tolerancia ante fallos (*route resilience*), lo que permite mejorar en gran medida el rendimiento del protocolo ya que evita la sobrecarga de paquetes de control que se produce al iniciar nuevos procesos de descubrimiento de rutas, favoreciendo además la escalabilidad de la red.

Mediante el encaminamiento multi-ruta es posible aliviar en mayor medida la congestión en la red, aumentar la disponibilidad y fiabilidad de las rutas a través del incremento de su tiempo de vida y minimizar los efectos de la aparición de cuellos de botella en algunos enlaces de la misma.

Debido a que en una red inalámbrica el ancho de banda disponible está limitado, el encaminamiento a través de una sola ruta puede no ser suficiente para cumplir los requisitos de capacidad establecidos para la aplicación. Sin embargo, la utilización simultánea de múltiples caminos hacia un destino concreto y el reparto del ancho de banda total en cada uno de ellos pueden satisfacer esos mismos requisitos de conexión. Además, cuanto mayor sea el ancho de banda disponible para la aplicación, menor será en general el retardo incurrido en las transmisiones extremo a extremo de la red.

A pesar de estas consideraciones, la transmisión a través de enlaces inalámbricos obliga a tener en cuenta las restricciones impuestas por la capa de enlace a la hora de evaluar el resultado de la agregación del ancho de banda en diferentes caminos. Las interferencias y colisiones entre paquetes que se producen en los medios inalámbricos impiden que esa agregación se realice de forma tan directa como la que se puede conseguir en un medio cableado. Las transmisiones por uno de los caminos pueden interferir con las transmisiones a lo largo de otra ruta alternativa, más si cabe si esas dos rutas tienen enlaces o nodos intermedios en común, limitando en consecuencia el ancho de banda disponible total. En cualquier caso, es

correcto pensar, y hay resultados que lo demuestran como [104], que el rendimiento de un protocolo de encaminamiento multi-ruta en redes inalámbricas densas es mejor que el conseguido mediante la utilización de algoritmos que utilicen una única ruta entre un origen y un destino. HARP ha sido construido con base en esta afirmación y los resultados de las simulaciones corroborarán en los Capítulos 5 y 7 dicha premisa.

En [108] se definen dos **tipos de redundancia de rutas** entre un origen y un destino, completándose esta clasificación con un tercer tipo introducido por [80], resultando las siguientes:

- *Link-disjoint multiple paths*: en las diferentes rutas alternativas no existen enlaces inalámbricos en común, aunque sí pueden contener nodos comunes.
- *Node-disjoint multiple paths*: en los caminos alternativos no existen nodos comunes, excepto el origen y el destino.
- *Interference disjoint routes*: todos los nodos intermedios de una ruta están lo suficientemente alejados de cualquier nodo intermedio de otra ruta y, por tanto, no existen colisiones ni interferencias en las transmisiones a través de ambas rutas.

Considerando esta clasificación, puede decirse que cuanto mayor sea el grado de independencia entre las rutas, mayores serán las ventajas de utilizar rutas disjuntas en las comunicaciones entre nodos, ya que mayor será la también independencia entre los fallos de red en ambas rutas y aumentará la posibilidad de balancear de la carga de la red en aplicaciones con altas restricciones de retardo extremo a extremo.

3.1.3. Encaminamiento salto a salto

Con el objetivo de favorecer la escalabilidad del protocolo y la toma de decisiones locales con relación al encaminamiento y a la recuperación ante fallos de red, HARP se concibe como un protocolo de encaminamiento distribuido que se incluye dentro de los denominados *salto a salto*. En este tipo de protocolos, cada nodo de la red tiene almacenadas en sus tablas de encaminamiento distintas rutas que contienen los diferentes destinos que conoce y el siguiente salto en el camino hacia el mismo, es decir, la decisión del siguiente salto en la ruta entre dos nodos se realiza salto a salto y es tomada de forma individual por cada uno de los nodos intermedios en el momento que le corresponda. Este tipo de actuación se contrapone a la forma de operar que tienen los llamados protocolos de encaminamiento *en origen*, en los cuales el nodo origen incluye en la cabecera del paquete de datos a transmitir la ruta completa que debe seguir éste hasta alcanzar el destino, limitándose la toma de decisiones del resto de nodos intermedios únicamente (y no siempre) al caso en que se produzcan errores en la transmisión del mismo, dentro de los procesos de mantenimiento de rutas.

3.1.4. HARP como protocolo reactivo

Una vez construido el árbol jerárquico y almacenada la estructura de rutas en las tablas de encaminamiento, HARP actúa como un protocolo reactivo a la hora de mantener las rutas establecidas. Al contrario que los proactivos, en los que esta estructura de rutas es mantenida y actualizada constantemente por los nodos de la red a través del intercambio de mensajes de control, los protocolos de encaminamiento reactivos sólo actúan o *reaccionan* cuando detectan la aparición de fallos de transmisión de los paquetes de datos a lo largo de la ruta. En función del proceso de recuperación específico que defina cada protocolo, es ese momento

se ejecutarán acciones concretas o se buscarán rutas alternativas para lograr el correcto encaminamiento del paquete de datos.

Esta forma de actuar en HARP viene impuesta por el objetivo primario del protocolo, que es el ahorro de energía en los nodos sensores, ya que en otro caso la proactividad obligada en el mantenimiento de rutas implicaría un aumento considerable del número de paquetes transmitidos en la red.

3.1.5. *Clustering* jerárquico

Con objeto de optimizar la eficiencia del modelo multi-salto e incrementar la escalabilidad de la red, en determinados tipos de redes, como pueden ser las WSN, se opta por un modelo de comunicación esquemático basado en *clusters* en el que los nodos de la red se organizan en grupos, en los cuales uno de los nodos es seleccionado como coordinador o CH. En este modelo de encaminamiento no uniforme, el CH es el responsable de la gestión de los paquetes desde el exterior del *cluster* hacia el interior y viceversa, actuando como nodo frontera del mismo en el intercambio de información con el exterior. A su vez, tal y como se detallará en el Capítulo 6, el CH puede asumir funciones de organización de las comunicaciones dentro de su *cluster*.

La comunicación tanto dentro de un *cluster* como entre diferentes CH y, eventualmente, hacia y desde el nodo sumidero, puede organizarse según los modelos de un solo salto y multi-salto indicados anteriormente, o bien como una combinación de ambos (un modelo se utiliza dentro de los *clusters* y otro modelo es usado entre los CH y entre éstos y el nodo sumidero).

En el caso particular de WSN, este modelo es especialmente interesante cuando se trabaja con aplicaciones que exigen tareas de agregación y compresión de la información antes de su envío hacia el nodo sumidero por la posibilidad de existencia

de los tipos ya mencionados de correlación entre los datos reocgidos por parte de varios nodos vecinos. La estructura de un modelo basado en *clusters* favorece la agregación de paquetes por parte del CH y en él es posible reducir la cantidad neta de datos a transmitir hacia el sumidero, con el consiguiente ahorro energético total.

3.2. Estructura de los mensajes de control

HARP define un conjunto de mensajes necesarios para la señalización y el intercambio de información de control entre los nodos de la red. La Tabla 3.1 muestra los distintos tipos de mensajes definidos y su significado. En esta tabla se incluyen la estructura de mensajes completa, si bien algunos de ellos se utilizan únicamente en la estructura del protocolo adaptada para WSN y no en la arquitectura general presentada en este capítulo. A lo largo de esta sección se realiza una descripción detallada de cada uno de ellos.

DATA: Paquete de datos

Los datos intercambiados por los nodos de la red se incluyen en este tipo de mensaje que, aunque no forma parte de la estructura de mensajes de control e HARP, se ha querido incluir en esta tarea de descripción. Además de los bytes de datos propiamente dichos, este mensaje incluirá una cabecera o *header* que alojará, entre otros campos, las direcciones de nivel de red del origen y del destino, sus niveles dentro de la estructura jerárquica, la necesidad o no de enviar un mensaje ACK de confirmación de recepción y un número identificativo (número de secuencia o relativo al instante temporal en que se inicia la transmisión del paquete). Opcionalmente, para aplicaciones que así lo requieran, puede utilizarse

Mensaje	Significado
CHADV	Notificación de elección como CH
CHCON	Notificación de cambio de condiciones
REMEM	Notificación de energía restante
JOINRQ	Petición de unión al <i>cluster</i>
ASSOCCH	Confirmación de asociación
JOINPAR	Petición de asociación con un nodo padre en el árbol
JOINREP	Respuesta a JOINPAR, confirmación de asociación con un padre
PAREQ	Petición o búsqueda de nuevo padre
CHREQ	Búsqueda o respuesta para asociar a un nuevo hijo
CHROUTE	Información de nuevas rutas creadas
CHDEL	Borrado de rutas
CHMISS	Descubrimiento de nuevas rutas
SCHE	Notificación de asignación de <i>slot</i> TDMA
CLADV	Notificación de información del <i>cluster</i>
DATAREQ	Petición de autorización para transmitir un paquete de datos
DATARES	Respuesta afirmativa al mensaje DATAREQ
ACK	Confirmación de recepción de un paquete

Tabla 3.1: Mensajes de control definidos en HARP: acrónimo y funcionalidad resumida.

la cabecera de un nuevo mensaje de datos para enviar ACK de otros mensajes recibidos (*piggybacking*).

En WSN los paquetes de datos contendrán las medidas realizadas por los nodos sensores (humedad, temperatura, aceleración, luminosidad, etc.), así como cualquier información relacionada con la detección de los eventos a los que esté destinada la monitorización del entorno.

CHADV (CH ADVertisement): Notificación de elección de CH

En el protocolo propuesto para WSN la estructura de los *clusters* cambia dinámicamente de manera cíclica con el tiempo para adaptarse a los distintos consumos energéticos que tienen los nodos en la red. Así, los nodos con rol de CH incurrirán en un mayor gasto de energía que el resto debido al proceso de recolección, agregación y envío de los datos del *cluster* (*intra-cluster*), así como a las transmisiones efectuadas dentro de la estructura formada por los distintos CH para alcanzar el nodo sumidero (*árbol inter-cluster*). Por esta razón, transcurrido un período determinado, la estructura de los *clusters* se renueva y, tras un proceso de selección, nuevos nodos pasan a ejercer el papel de CH. Cuando un nodo es elegido como CH debe difundir un paquete CHADV para notificar su nuevo rol al resto de nodos de su entorno y al nodo sumidero.

Este mensaje incluye la dirección de nivel 3 (ID de red) del nodo emisor (o un dato que le identifique unívocamente en la red, en caso de no tener asignado este identificador), así como la potencia de transmisión. Este parámetro servirá para realizar una comparación con la potencia recibida en el otro extremo del enlace y poder de esta forma estimar tanto la distancia que separa el emisor del receptor como las pérdidas de potencia asociadas al trayecto. Además, con objeto de evitar interferencias entre *clusters* vecinos, el CH escogerá un código CDMA de forma aleatoria entre los disponibles en una lista y lo incluirá dentro de este paquete para ser utilizado en las comunicaciones *intra-cluster* con los nodos que se asocien con él. En resumen, el paquete de control CHADV consta de los siguientes campos obligatorios:

- Código del mensaje CHADV.
- Identificador (ID) del nodo elegido como CH.

- Potencia de transmisión.
- Código CDMA escogido para las transmisiones intra-*cluster*.

CHCON (CHange of CONditions): Notificación de cambio de condiciones

Este mensaje sólo es utilizado por HARP en WSN y no tiene utilidad dentro del algoritmo genérico de operación del protocolo.

En HARP se permite que el nodo sumidero pueda cambiar las condiciones de selección de CH a lo largo de la operación de la red de sensores, con objeto de adecuar su número o su distribución espacial o de modificar la duración de las rondas de transmisión (lo que conlleva un cambio en la tasa de paquetes generados) para adaptarse dinámicamente a los requisitos de la aplicación. A su vez, debe poder ser capaz de modificar a su conveniencia parámetros tales como el código CDMA seleccionado para las comunicaciones inter-*cluster* (o en su defecto, el canal RF escogido) o la profundidad máxima de los árboles inter-*cluster* e intra-*cluster*, definida como la diferencia máxima de niveles entre el nodo raíz y el nodo hoja con nivel más bajo dentro del árbol. En consecuencia, en la fase de inicialización y selección de CH, el nodo sumidero puede difundir mediante *broadcast* un mensaje CHCON con los siguientes campos:

- Código del mensaje CHCON.
- Identificador (ID) del nodo sumidero.
- Potencia de transmisión (opcional).
- Código CDMA seleccionado.

- Valor de los parámetros asociados a la elección de CH: estos valores dependerán del algoritmo de selección utilizado. En caso de LEACH, este parámetro es un valor de probabilidad (en función del porcentaje deseado de CH en cada ronda), a diferencia de s-HARP (ver Sección 6.6), en el que existen más parámetros asociados a la selección de CH. Este campo es opcional: si está vacío, no hay variaciones en la fórmula de selección de CH.
- Valor máximo de la profundidad del árbol inter-*cluster* (opcional).
- Valor máximo de la profundidad de los árboles intra-*cluster* (opcional).

REMEN (REMAining ENergy): Notificación de energía restante

Este mensaje es opcional y va ligado al empleo de s-HARP como formulación para la elección de CH o de algún otro método dentro de los algoritmos llamados *energy-aware* (a los que también pertenece s-HARP) que tenga en cuenta la energía restante que poseen los nodos del entorno a la hora de seleccionar dichos CH.

Al inicio de cada ronda de transmisión, específicamente al principio de la fase de formación de los *clusters* en WSN, es necesario que cada nodo de la red difunda a los nodos de su entorno un paquete REMEN incluyendo en un campo la energía restante que tiene almacenada, con el objetivo de que los nodos de su alrededor puedan utilizar este valor en sus cálculos de selección de CH.

Existen tres campos en este mensaje de control:

- Código del mensaje REMEN.
- Identificador (ID) del nodo emisor.
- Energía restante.

JOINRQ (JOIN ReQuest): Petición de unión al *cluster*

Una vez difundidos los mensajes de notificación de CH, el resto de nodos de la red debe seleccionar uno de ellos como su CH asociado, el cual será finalmente quien recoja y agregue sus datos. Este proceso tiene como objetivo la distribución de los nodos restantes de la red, los que no han sido seleccionados como CH, en distintos *clusters* de tal forma que la selección entre uno u otro CH se realice atendiendo a razones energéticas. A estos nodos se les denominará a lo largo de esta Tesis nodos normales o *nonCH*. Según esta premisa, cada *nonCH* enviará un mensaje de asociación JOINRQ al CH cuyas pérdidas energéticas de transmisión estimadas sean menores. Esta elección normalmente coincide con el CH más cercano, aunque estas condiciones no son siempre equivalentes debido a la naturaleza dinámica de los canales inalámbricos. Para realizar este cálculo, el nodo receptor tiene en cuenta tanto el valor del RSSI (*Receive Signal Strength Indication*) del paquete CHADV recibido, que mide el nivel de potencia de señal en recepción, como el valor en él incluido de la potencia con la que fue transmitido. La diferencia entre ambos es la pérdida introducida por el canal de transmisión que, como se verá en la Sección 6.2, está relacionada con la distancia entre el emisor y el receptor.

Este mensaje incluye el *flag mandatory* que si está activado obliga al CH a aceptar la asociación del *nonCH* a pesar de que ya hubiese podido alcanzar el límite máximo de asociaciones.

Los campos incluidos en este mensaje de asociación son los siguientes:

- Código del mensaje JOINRQ.
- Identificador (ID) del nodo origen (*nonCH*).
- Identificador (ID) del nodo destino (CH).
- *Flag mandatory*.

- Potencia de transmisión.
- Potencia de recepción del paquete CHADV (necesaria para que el CH pueda ajustar su potencia de transmisión para cubrir la distancia que le separa del *nonCH*).
- Energía restante (opcional).

ASSOCCH (ASSOCIation to CH): Confirmación de asociación

Una vez recibido con éxito el mensaje JOINRQ de petición de unión al *cluster*, el CH confirma la asociación enviando un paquete ASSOCCH al nodo *nonCH*. Análogamente a lo definido en el mensaje anterior, en este paquete el CH incluirá la potencia de recepción del paquete JOINRQ con objeto de que el *nonCH* pueda calcular las pérdidas del enlace inalámbrico que les une en el sentido *nonCH* a CH y poder ajustar de esta forma su potencia de transmisión en futuras ocasiones. La razón por la que el enlace inalámbrico debe evaluarse en los dos sentidos es la asunción de que el mismo es asimétrico, presentando diferentes pérdidas de potencia en los dos sentidos, ampliamente aceptada por la comunidad científica [76], [90], [51], [121] y [55], entre otros.

Por otro lado, este mensaje se utiliza también en la creación de los árboles intra e inter-*cluster* para que el CH y el sumidero confirmen respectivamente la asociación a dicha estructura de los nodos que lo solicitaron y les informe de la potencia perdida en los enlaces con cada uno de ellos.

El mensaje de confirmación ASSOCCH incluye los siguientes campos:

- Código del mensaje ASSOCCH.
- Identificador (ID) del nodo origen (CH o *sink*).

- Listado de identificadores (ID) de los nodos destino (*nonCH* o CH).
- Potencia de transmisión (se utiliza para actualizar el cálculo del balance energético del enlace).
- Lista CST: listado de nodos CH ordenados de menor a mayor por el valor de la pérdida de potencia en el enlace con el sumidero o con el CH, según se esté usando en la construcción del árbol *inter-cluster* o *intra-cluster*, respectivamente.
- Listado con la potencia de recepción de los paquetes JOINRQ de los nodos de la lista CST (en el mismo orden).

JOINPAR (JOIN request to PARENT): Petición de asociación con un nodo padre en el árbol

Este mensaje se utiliza para enviar una petición de asociación como nodo hijo a un nodo perteneciente a la estructura jerárquica. Al igual que JOINRQ, contiene un *flag* que obliga al nodo padre a aceptar como hijo al nodo que lo envía. Además, este mensaje incluye el *flag main_parent* que sirve para informar al nodo padre que es el principal y por el que preferentemente se encaminarán las comunicaciones. En consecuencia, los campos que contendrá son los siguientes:

- Código del mensaje JOINPAR.
- Identificador (ID) del nodo origen.
- Identificador (ID) del nodo destino.
- *Flag mandatory*.
- *Flag main_parent*.

- Potencia de transmisión.
- Potencia de recepción del paquete JOINRQ emitido por el nodo padre para unirse al CH.

JOINREP (JOIN REPlY): Respuesta a JOINPAR, confirmación de asociación con nodo hijo

Cuando un nodo recibe un mensaje JOINPAR, debe confirmar la asociación con el nuevo nodo hijo a través del envío de un mensaje JOINREP. Los campos de JOINREP son los mismos que los de JOINPAR, añadiendo el nivel del nodo padre en el árbol, es decir:

- Código del mensaje JOINREP.
- Identificador (ID) del nodo origen.
- Nivel del nodo origen en el árbol jerárquico.
- Identificador (ID) del nodo destino.
- Potencia de transmisión.
- Potencia de recepción del paquete JOIPAR emitido por el nodo hijo.

CHROUTE (CHild ROUTE): Información de nuevas rutas creadas

Para informar a los nodos de niveles superiores (más cercanos al nodo raíz del árbol) de las nuevas asociaciones padre-hijo dentro del árbol se utilizan los paquetes CHROUTE, los cuales son generados por los nodos hijos tras la confirmación de

dichas uniones, conteniendo información relativa a sus propios hijos, es decir, de la estructura inferior del árbol. Este paquete se encaminará a lo largo del árbol jerárquico hasta llegar a su raíz (bien un CH en árboles intra-*cluster*, o bien el nodo sumidero en inter-*cluster*), actualizándose en cada nodo intermedio la información incluida en las tablas de encaminamiento en función del contenido de este paquete. Además de las nuevas asociaciones, en la propuesta de HARP para WSN este paquete puede contener un *flag* denominado *delete_old_route* e incluir información relativa a las asociaciones rotas con antiguos nodos padre (por ejemplo, cuando el nodo se ha movido) para, de esta forma, si el *flag* está activo, permitir a los nodos superiores del árbol la actualización de sus tablas de rutas a través de la eliminación de dicha entrada. Con este método se evita la utilización del paquete CHDEL descrito a continuación y se simplifica la operación en caso de fallo.

La estructura de campos es la siguiente:

- Código del mensaje CHROUTE.
- Identificador (ID) del nodo origen.
- Identificador (ID) del nodo destino.
- Potencia de transmisión (opcional).
- Listado de ID de nodos hijos y a los que se puede acceder a través de ellos.
- *Flag delete_old_route* (opcional).
- ID del antiguo nodo padre (junto con el *flag delete_old_route* activado).

CHDEL (CHild route DElete): Borrado de rutas

Cuando se producen fallos en la red y algunos de los enlaces entre los diferentes nodos se rompen, se dice que las rutas que los incluyen se vuelven fallidas o inuti-

lizables. En este caso, para que el protocolo de encaminamiento sea eficiente debe poder actualizar la información relacionada con el fallo, contenida en las tablas de rutas de los nodos y buscar rutas alternativas a las originales. Para la primera de las dos acciones se utiliza el mensaje CHDEL, mediante el cual un nodo informa a sus nodos padre del fallo de transmisión en alguno de los enlaces hacia sus hijos directos.

El proceso completo puede resumirse en las siguientes acciones:

1. Detección de fallo de transmisión a través del enlace con un hijo directo.
2. Búsqueda de rutas alternativas al destino a través de otro nodo hijo.
3. En caso negativo, selección de todos los nodos de la rama inferior a los que únicamente se accede por el enlace roto. Inclusión de dichos nodos en el paquete CHDEL con objeto de borrar sus rutas.
4. Inclusión de su ID en el campo nodo origen y envío del paquete CHDEL a todos sus nodos padre.
5. En recepción de un paquete CHDEL, actualización de las tablas de encaminamiento del nodo receptor y comprobación de rutas alternativas conocidas hacia los nodos incluidos en el paquete que no incluyan al nodo origen del CHDEL como siguiente salto.
6. Reenvío del CHDEL a sus padres incluyendo en sus campos los nodos para los cuales no se conoce otra ruta alternativa.

La estructura de campos de este mensaje es la siguiente:

- Código del mensaje CHDEL.
- Identificador (ID) del nodo origen.

- Identificador (ID) del nodo destino.
- Potencia de transmisión (opcional).
- Listado de ID de nodos hijos perdidos y de los nodos a los cuales se accede exclusivamente a través de ellos.

CHMISS (CH MISSed): Descubrimiento de nuevas rutas

Mensaje de inundación que se propaga a través de la red para descubrir nuevas rutas hacia un nodo determinado. El nodo que reciba este paquete y no sea el destino ni lo conozca, procederá a reenviarlo a sus nodos vecinos. El proceso llega a su fin cuando el CHMISS alcanza el nodo buscado o un nodo que conoce una ruta válida hacia el mismo. Una vez encontrada una forma de llegar al destino, se generará y propagará por la red un paquete CHROUTE con información acerca de la nueva ruta encontrada. Opcionalmente se permite que este paquete incluya el campo TTL con el número máximo de saltos que puede realizar el paquete para encontrar el nodo destino.

Los campos en CHMISS son los siguientes:

- Código del mensaje CHMISS.
- Identificador (ID) del nodo origen.
- Identificador (ID) del nodo buscado.
- Número de secuencia.
- Potencia de transmisión (opcional).
- TTL (opcional).

SCHE (SCHEdule): Notificación de asignación de *slot* TDMA

Este mensaje es enviado por el nodo raíz del árbol (CH o sumidero en WSN, según el caso) para la asignación de *slots* o franjas temporales de transmisión a cada uno de los nodos hijo, con objeto de crear la trama TDMA y evitar de esta forma que se produzcan colisiones entre los paquetes enviados por los nodos asociados al *cluster*. Este mensaje incluirá información relativa al primer *slot* y un listado de nodos hijo ordenado, en el que le será asignado el primer *slot* TDMA al primer nodo y el resto de nodos tomarán posesión del siguiente *slot* libre de forma secuencial.

Además, este mensaje se utiliza para actualizar la trama TDMA después de los procesos de reconfiguración del árbol tras fallos en las transmisiones o ante la movilidad de nodos. En estos casos, el nodo raíz enviará un paquete SCHE con los nuevos cambios únicamente con destino a los nodos involucrados.

Por último, este mensaje se utiliza para informar al nodo sustituto (descrito en la Sección 3.5) de algunos datos necesarios para su funcionamiento en caso de tener que activarse, como son el código CDMA de las transmisiones inter-*cluster* y el ID del padre principal dentro de esta estructura.

Los campos que utiliza SCHE se indican a continuación:

- Código del mensaje SCHE.
- Identificador (ID) del nodo origen.
- Listado de identificadores (ID) de los nodos destino.
- Potencia de transmisión (opcional).

- Orden del *slot* asignado dentro de la trama TDMA (opcional, para reconfiguraciones).
- Tiempo de inicio del primer *slot* (opcional).
- Duración del primer *slot* (opcional).
- Código CDMA inter-*cluster* (opcional).
- ID del padre principal inter-*cluster* (opcional).

CLADV (CL ADvertisement): Notificación de información de *cluster*

Cuando en WSN se inician los procesos de recuperación ante fallos, HARP hace uso del *slot* RS (definido en el Capítulo 6) para que los nodos puedan informar de los cambios de estructura producidos y detectados en la red. Para que estos procedimientos se lleven a cabo de forma fiable es necesario que el nodo raíz (sumidero o CH, dependiendo del árbol) difunda tanto al principio como al final del RS un mensaje CLADV con información relativa al árbol (inter o intra-*cluster*), con objeto tanto de incorporar nuevos nodos al mismo sin necesidad de esperar a una nueva ronda, como de incluir dichos cambios en la estructura de transmisión, adaptando la trama TDMA en curso. El mensaje CLADV incluye además el identificador del nodo sustituto, que toma el rol de CH en caso de fallo de este nodo.

Los campos de este mensaje son los incluidos también en el mensaje CHADV de notificación de CH, con alguno adicional:

- Código del mensaje CLADV.
- Identificador (ID) del nodo origen (sumidero o CH).

- Potencia de transmisión.
- Código CDMA escogido para las transmisiones intra-*cluster* (a incluir cuando CLADV se transmite al principio del RS).
- Identificador (ID) del nodo sustituto (sólo para comunicaciones intra-*cluster*).
- Instante temporal de inicio del siguiente RS, incluido en el caso de la transmisión de CLADV al final del RS. En caso de que el mensaje CLADV se envíe al principio del *slot*, este campo hace referencia al instante de finalización del RS.
- *Flag* RS_start: indica si el mensaje CLADV se envía al principio o al final del *slot*.

PAREQ (PARENT REQUEST): Petición o búsqueda de nuevo padre

Tras la adición de nuevos nodos en la red o si un nodo ha quedado huérfano porque todos los enlaces con sus posibles padres se han roto, debe iniciarse un proceso de asociación al árbol jerárquico a través de la búsqueda de nuevos nodos padre. Para ello, el nodo que desee asociarse difundirá un paquete PAREQ con los siguientes campos:

- Código del mensaje PAREQ.
- Identificador (ID) del nodo origen.
- Potencia de transmisión.

CHREQ (CHild REQuest): Asociación de nuevo hijo

El mensaje CHREQ tiene dos utilidades; el uso de una, otra o las dos dependerá de la aplicación específica para la que se utilice. En WSN, CHREQ es la respuesta de un nodo del árbol al mensaje PAREQ recibido para confirmar la nueva asociación al nodo huérfano. Sin embargo, en el algoritmo general de construcción del árbol jerárquico en HARP, este paquete puede utilizarse además para la búsqueda proactiva de nodos hijo por parte de un nodo que se acaba de unir a la estructura del árbol. Sus campos son:

- Código del mensaje CHREQ.
- Identificador (ID) del nodo origen.
- Nivel en el árbol del nodo origen.
- Identificador (ID) del nodo destino (sólo en caso de WSN).
- Potencia de transmisión.

DATAREQ (DATA authorization REQuest): Petición de asociación

Este paquete tiene como objeto la solicitud de autorización por parte de un nodo para transmitir un paquete de datos a un nodo padre. Se utiliza en WSN cuando se han producido reconfiguraciones en la red y los nodos sensores desean transmitir los datos recogidos en esa trama.

Este mensaje consta de los siguientes campos:

- Código del mensaje DATAREQ.
- Identificador (ID) del nodo origen.

- Identificador (ID) del nodo destino.
- Potencia de transmisión (opcional).

DATARES (DATA authorization RESponse): Respuesta a petición de asociación

Cuando el nodo padre recibe un paquete DATAREQ y el resultado de su procesamiento es positivo, contesta con un paquete DATARES para que el nodo hijo pueda iniciar la transmisión del paquete de datos. Los campos de este paquete son idénticos a los de DATAREQ:

- Código del mensaje DATARES.
- Identificador (ID) del nodo origen.
- Identificador (ID) del nodo destino.
- Potencia de transmisión (opcional).

ACK (ACKnowledge): Confirmación de recepción de paquete

Para confirmar que un paquete de datos o de control ha sido recibido correctamente, el receptor envía un paquete de confirmación ACK al nodo emisor. La no recepción de este paquete por parte del nodo emisor da lugar a los procesos de recuperación descritos en la Sección 3.5.

3.3. Construcción del árbol jerárquico

La arquitectura de encaminamiento propuesta en HARP está basada en una estructura particular de tablas de rutas y en la creación de un árbol jerárquico que permita la comunicación bidireccional entre cualquier par de nodos de la red. La formación de dicho árbol se basa en el paradigma *asociación padre-hijo* y se describe a continuación. Además, HARP proporciona un mecanismo de reestructuración dinámica del árbol que evita la formación de bucles y optimiza el encaminamiento a través del mismo.

3.3.1. Estructura de las tablas de encaminamiento

Cada nodo de la red almacenará en su memoria dos tablas de rutas o tablas de encaminamiento: la Tabla de Padres (PT, *Parent Table*) y la Tabla de Hijos (CT, *Child Table*). En esta segunda tabla se almacenarán no sólo los nodos hijo directos sino todos aquellos nodos a los que se pueda acceder por cualquiera de ellos (hijos de los hijos, etc.). Opcionalmente puede crearse también una tabla adicional para almacenar datos acerca de los nodos vecinos con el mismo nivel jerárquico. Cada una de estas tablas registrará información relativa a los nodos que vaya conociendo en la red y contendrá como mínimo los siguientes campos:

- *Destino*: ID del nodo conocido.
- *Nivel*: nivel del nodo conocido en la red dentro de su jerarquía.
- *Siguiente_Salto*: ID del siguiente nodo en la ruta hacia el nodo conocido, que según la estructura jerárquica será el ID de un padre o de un hijo directo.
- *Calidad* (opcional): potencia necesaria para transmitir un bit al nodo siguiente con un aceptable E_b/N_0 (SNR mínima requerida). Este parámetro se calcula en función del balance energético estimado para ese enlace. Ya

que un nodo puede tener varias alternativas de selección del siguiente salto, este parámetro le permitirá escoger aquélla que le comporte un menor gasto energético.

3.3.2. Inicialización del árbol jerárquico

Inicialmente es necesario asignar a uno de los nodos del árbol funciones de nodo raíz o *root*, el cual se inicializa con nivel igual a cero. La elección de este rol correrá a cargo de la aplicación. Una vez seleccionado, este nodo iniciará el proceso de formación mediante la difusión o *broadcast* de un paquete CHREQ con una potencia definida por la aplicación (que será función de parámetros tales como la densidad de la red, la profundidad máxima del árbol, el retardo máximo requerido, etc.). Con este paquete el nodo raíz descubrirá a sus nodos vecinos más cercanos quienes, tras la recepción de este mensaje, contestarán mediante un paquete JOINPAR para asociarse como hijos con un nivel en árbol igual a uno. Una vez recibido el JOINPAR, el nodo raíz incluirá en su CT a cada uno de ellos y les contestará con un mensaje JOINREP para confirmar dicha asociación. Los hijos que reciban este mensaje almacenarán en su PT los datos relativos al nodo raíz y a la calidad del enlace, momento en el cual puede considerarse completada la asociación.

Este proceso de envío de CHREQ es ahora repetido por cada uno de los nodos del nivel 1 para asociar a sus propios nodos hijo sucesivamente hasta llegar a los nodos más alejados del nodo raíz, que serán los nodos *hoja* en la estructura del árbol jerárquico. En caso de que algún nodo quede sin asociarse, el sistema se configurará para que después de un tiempo determinado inicie la transmisión de mensajes PAREQ con potencia incremental hasta que sea capaz de alcanzar un nodo ya asociado a la red y reciba como contestación un paquete CHREQ de asociación. En este proceso se permite que los nodos se asocien a varios padres

simultáneamente y puedan tener además nodos hijo en común con otros nodos vecinos. Ésta es la base para crear estructuras multi-camino en las que una comunicación puede ser encaminada por rutas alternativas hacia un mismo destino.

Una vez asociados todos los nodos de la red y actualizadas todas las tablas de encaminamiento, los nodos hoja procederán a la creación y envío a sus nodos padre de un paquete CHROUTE con su ID en el campo origen y el campo de nodos hijo vacío. Cada nodo padre que reciba este paquete actualizará su CT, trasladará el ID del nodo hoja al campo destinado a la inclusión de los ID de los hijos en el paquete CHROUTE y se incluirá él mismo en el campo origen, reenviando por último el citado paquete CHROUTE a sus padres. De esta forma, cada nodo que reciba este paquete actualizará su CT e incluirá su ID en el mismo, hasta llegar al nodo raíz. En esta situación, un nodo en el nivel i almacenará en su PT los enlaces directos a sus padres situados en el nivel $i - 1$, y en su CT los enlaces directos a sus hijos (nivel $i + 1$) y a todos los nodos a los que se accede a través de ellos (niveles $i + 2$, $i + 3$, etc.). En definitiva, cuanto más cerca está un nodo del nodo raíz, es decir, más arriba se encuentra dentro del árbol jerárquico, mayor número de rutas de nodos hijo almacenará en su CT.

El cronograma de paquetes de control en este proceso de creación del árbol jerárquico puede visualizarse en la Figura 3.1.

3.3.3. Optimización dinámica de la estructura

El estructura del árbol descrita anteriormente se puede actualizar dinámicamente para adaptarse a los nuevos cambios en la red, cumpliendo dos objetivos principales: la optimización de las cadenas de las rutas y la prevención ante la formación de bucles entre nodos.

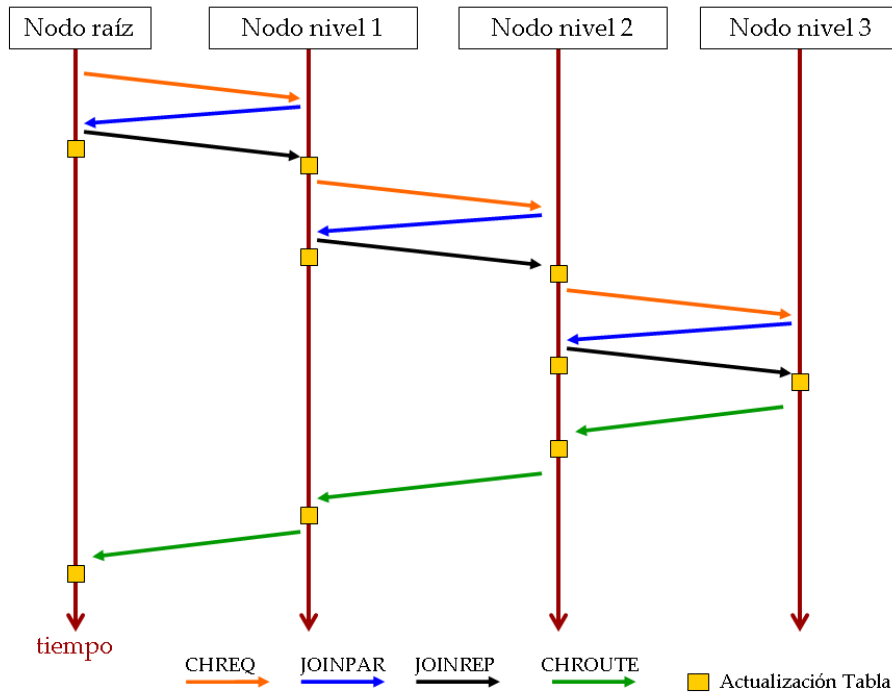


Figura 3.1: Cronograma de construcción de un árbol jerárquico de cuatro niveles en HARP.

HARP hace uso del mensaje CHREQ, que se genera en la red para nuevas asociaciones, para optimizar además su estructura. Cuando un nodo recibe un mensaje CHREQ debe comprobar el campo *Nivel* del mismo, pudiendo darse alguna de las situaciones siguientes en las que el nodo puede actuar proactivamente:

- Si el nivel es el mismo que el suyo, comprueba que el nodo emisor no esté en sus tablas PT o CT. En caso de estar contenido lo elimina y, si se trata de un nodo incluido en CT, genera los respectivos paquetes CHDEL.
- Si el nivel incluido en el paquete es mayor que el suyo en una unidad comprueba que el nodo se encuentre en la tabla de hijos. Si no es así le envía un nuevo paquete CHREQ para asociarlo.
- Si el nivel incluido en el paquete es mayor que el suyo en más de una unidad le enviará siempre un paquete CHREQ para asociarlo. Como resultado de dicha

asociación, el nivel del nodo que originó el primer CHREQ se actualizará, y con él toda su rama inferior.

- Si el nivel incluido en el paquete es menor, comprueba si el nodo se encuentra en la tabla PT como padre. De no ser así, le envía un paquete JOINPAR de asociación.

3.4. Operación del protocolo

La estructura de tablas propuesta en HARP permite el envío de paquetes de datos entre cualquier origen y cualquier destino de la red, es decir, la gestión de los tres tipos de tráfico que pueden cursarse:

1. *Downstream* o *data dissemination*: tráfico descendente en el que el origen del paquete es el nodo raíz y el destino cualquier otro nodo de la red.
2. *Upstream* o *data gathering*: sentido ascendente, con origen en cualquier nodo y destino el nodo raíz.
3. *Peer-to-peer*: tráfico existente entre dos nodos cualesquiera de la red a excepción del raíz.

Cuando un nodo de la red necesita enviar un paquete de datos a cualquier otro nodo de la misma, seguirá el siguiente proceso: en primer lugar comprobará si el nodo destino está almacenado en su PT; si lo está (es un padre), le enviará el paquete directamente; de lo contrario, consultará si en su CT tiene almacenado uno o más registros pertenecientes a ese nodo destino. En caso de que el destino esté almacenado, identificará si es o no un hijo directo comparando los ID de los campos *Destino* y *Siguiente_Salto*. En caso de coincidir, enviará el paquete a su hijo directo; en caso contrario, seleccionará (suponiendo que existe más de

una entrada) el siguiente salto de la ruta basándose en alguna premisa. En el caso más general de HARP, inicialmente estará configurado para seleccionar de forma aleatoria el siguiente salto de entre varias alternativas. Para su utilización en WSN, HARP utiliza el campo *Calidad* de la tabla para seleccionar el siguiente salto en función del que tenga mínima energía de transmisión. Si el nodo destino no está contenido en las tablas PT ni CT, entonces el nodo seleccionará uno de sus padres (siguiendo los mismos criterios de selección de una ruta hacia alguno de sus hijos) y le enviará el paquete de datos. El nodo que reciba el paquete (ya sea padre o hijo directo), comprobará si es el destinatario final del paquete de datos analizando el campo *Destino* del mismo. Si no lo es, aplicará el mismo procedimiento descrito hasta que el nodo alcance por fin el nodo destino.

En caso de que el paquete de datos llegue al nodo raíz (situación que ocurre cuando el nodo destino se encuentra en una rama distinta a la del origen entre las que parten del nodo raíz), será este nodo quien encaminará el paquete por la rama adecuada, ya que es él quien almacena la información relativa a todos los nodos de la red.

Como consecuencia de los fallos producidos en la red, puede ocurrir que la transmisión del paquete al siguiente nodo de la ruta escogida sea fallida y en consecuencia no se reciba el paquete ACK de confirmación de recepción. En este caso, cuando el nodo detecta que el enlace se ha roto y no conoce más rutas hacia el destino, debe iniciar el proceso de recuperación de rutas propuesto en HARP para poder hacer frente a estos fallos y encaminar con éxito el paquete de datos hacia su destino final. Este mecanismo se describe con detalle en la Sección 3.5.

3.5. Reconfiguración dinámica de la red

En este apartado se describen los distintos mecanismos que HARP posee para hacer frente a los errores producidos en los enlaces y nodos de la red, así como los procesos de reconfiguración de la estructura del árbol jerárquico que se adaptan dinámicamente a situaciones de movilidad de nodos y de inserción de nuevos sensores en la red. Estos mecanismos se engloban dentro de uno más general que se puede denominar *Local Recovery* y que hace referencia a la esencia del funcionamiento de los mismos: la reestructuración local de la red. Este aspecto es fundamental a la hora de evaluar prestaciones tales como la escalabilidad o el retardo extremo a extremo de un protocolo de encaminamiento, según se ha descrito en el Capítulo 2.

En primer lugar, se describe en la Sección 3.5.1 el mecanismo de asociación ante el despliegue de nuevos nodos en la red; a continuación, en la Sección 3.5.2 se detallará el proceso de recuperación definido en HARP para adaptarse a cambios en la estructura de la misma; más adelante, la Sección 3.5.3 hace referencia explícita a cómo se gestiona la movilidad de los nodos en la red preservando la estructura jerárquica del árbol; para finalizar, en la Sección 3.5.4 se describe otro proceso definido en HARP para dotar de consistencia a la estructura de red ante posibles fallos del nodo raíz: el nodo sustituto.

3.5.1. Adición de nuevos nodos en la red

A menudo los protocolos de encaminamiento en redes inalámbricas deben tratar con situaciones de inserción o posicionamiento de nuevos nodos dentro de la estructura de la red para hacer frente por un lado a la disminución del número de sensores debido al agotamiento de sus baterías y, por otro, a extensiones en la cobertura o la densidad de la red. En la mayoría de los casos previstos se requiere

que la red global sea capaz de reconfigurarse de forma automática y autónoma incluyendo estos nuevos nodos dentro de su estructura sin necesidad de actuaciones externas. Este requisito hace que sea necesario un protocolo de red capaz de auto-organizarse y reestructurarse autónomamente para hacer frente a estas situaciones.

En consecuencia, HARP debe implementar un mecanismo que permita la adición de nuevos nodos en la red, integrándolos dentro de su estructura jerárquica de forma que reduzca al mínimo el impacto sobre el rendimiento del protocolo. Este mecanismo denominado *joining mechanism* permite hacer frente a situaciones de este tipo.

Cuando un nodo es añadido a la red y se programa para monitorizar el entorno y enviar un paquete de datos, el primer paso en HARP es su asociación a la estructura jerárquica de la red. Para ello generará y difundirá en su entorno un paquete PAREQ solicitando la asociación con algún nodo padre e inicializará el parámetro *Nivel* almacenado en memoria al valor -1 . Cualquier miembro de la red que escuche este mensaje responderá al nuevo nodo a través de un mensaje CHREQ. Tras un período de tiempo que dependerá de la aplicación, el nuevo nodo seleccionará de entre todos los CHREQ recibidos aquél que permita la asociación con el nodo de menor nivel dentro del árbol jerárquico (más cercano al nodo raíz), enviándole un mensaje JOINPAR de asociación. Si existiesen varios nodos con el mismo nivel se completarían las asociaciones como nodo hijo con todos ellos con objeto de favorecer la redundancia de caminos e implementar de esta forma una estructura multi-ruta.

Una vez asociado con uno o más nodos padre, el nodo que ha sido desplegado difundirá un paquete CHREQ con objeto de optimizar la estructura de la red y de asociar otros nodos que hayan podido quedar aislados debido a su excesiva distancia con los nodos que forman la estructura jerárquica de la misma. Las

consecuencias que ese paquete tiene en relación con la actualización del árbol jerárquico han sido descritas en la Sección 3.3.3.

3.5.2. Recuperación ante el fallo de un enlace

Debido a los fallos producidos en la red y causados por elementos, fenómenos o situaciones tales como obstáculos, desvanecimientos, movilidad, malfuncionamiento de los nodos, agotamiento de baterías, etc., puede ocurrir que la transmisión de un paquete al siguiente nodo de la ruta escogida sea fallida y en consecuencia no se reciba el paquete ACK de confirmación de recepción. En este caso, cuando el nodo detecte que el enlace se ha roto, guardará el paquete de datos en un buffer e iniciará el proceso de *Recuperación Local de Fallos* definido a continuación.

Ante un fallo de red, en primer lugar el nodo debe borrar de sus tablas (bien PT o CT, según sea el nodo almacenado) la entrada correspondiente a dicho enlace. Una vez completado este paso, el nodo enviará un paquete CHDEL a sus padres para que los nodos superiores del árbol borren también esta ruta. Este paquete se propagará por el árbol eliminando esa ruta de las tablas de los nodos que la contengan. Más tarde, el nodo aplicará el mismo algoritmo de operación descrito anteriormente para continuar con la transmisión del paquete hacia su destino. Si al aplicarlo encuentra una o más entradas en las tablas para ese nodo, encaminará el paquete; de lo contrario, al no conocer al nodo destino, procederá al envío de dicho paquete a uno de sus padres según lo explicado previamente.

En caso de que finalmente el nodo no pueda enviar el paquete a ninguno de sus nodos padre se iniciará el proceso de *joining mechanism* descrito en el apartado anterior.

En ocasiones, sobre todo ante situaciones de movilidad o fallos múltiples de la red, puede ocurrir que el paquete de datos llegue al nodo raíz y, como consecuencia

de haber actualizado sus tablas borrando las rutas inválidas al recibir los paquetes CHDEL, dicho nodo no encuentre en ellas ninguna entrada para el destino del paquete. En ese caso, deberá iniciar el descubrimiento de nuevas rutas hacia el nodo destino a través de la difusión de un paquete CHMISS. Este procedimiento inundará la red con paquetes CHMISS hasta que el nodo destino sea encontrado, introduciendo una sobrecarga de paquetes de control en la red muy elevada ya que en el caso peor se pueden llegar a transmitir tantos paquetes CHMISS como nodos haya en la red.

3.5.3. Gestión de la movilidad

La movilidad de los nodos es una característica fundamental que debe tenerse en cuenta a la hora de diseñar un protocolo de encaminamiento en redes inalámbricas, tanto ad hoc como WSN, como ya se ha mencionado en el Capítulo 2. En consecuencia, es necesario incorporar mecanismos que permitan la gestión de la movilidad de los nodos de forma que se integren de forma óptima dentro del funcionamiento normal del protocolo, reduciendo su impacto en el rendimiento del mismo.

HARP hace uso del mecanismo de unión de nuevos nodos a la red (*joining mechanism*) para gestionar la movilidad dentro de la misma. Cuando un nodo se desplaza de un punto de la red a otro lo suficientemente alejado, los enlaces establecidos con sus nodos vecinos dejan de ser útiles, por lo que cuando éstos quieran comunicarse con él ejecutarán los procesos de recuperación descritos en este apartado para la actualización de la estructura de red. A su vez, el nodo que se ha movido iniciará el proceso de unión a la red pero sólo en el momento en el que desee enviar un paquete a cualquier destino (no ha de olvidarse que HARP es un protocolo reactivo en este sentido), siguiendo el cronograma de intercambio de mensajes que se presenta en la Figura 3.2. Esta situación puede dar lugar al

inicio de procesos de descubrimiento de rutas hacia el nodo en movimiento cuando existan paquetes pendientes de entrega, lo que conlleva la sobrecarga de paquetes de control en la red. El análisis de esta situación se incluye en la Sección 5.1.5.

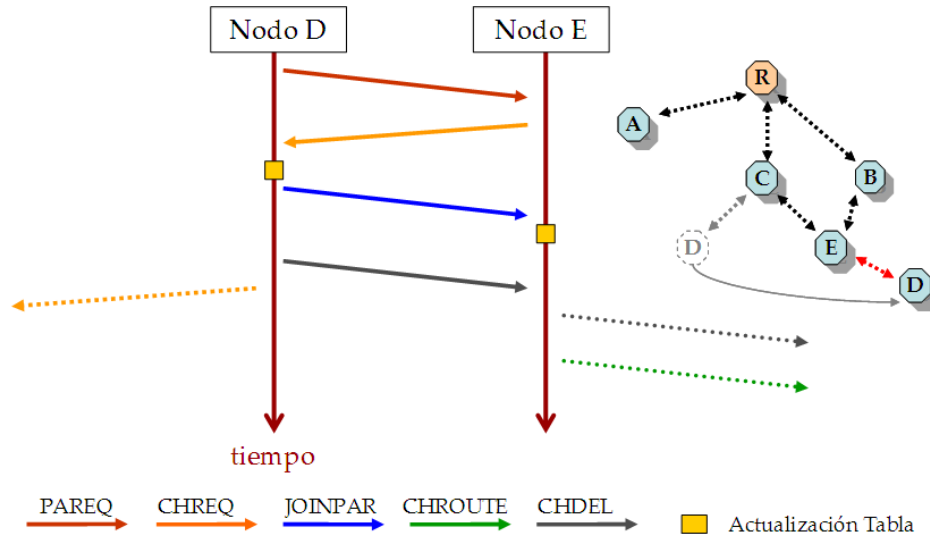


Figura 3.2: Cronograma de mensajes de control para gestionar el movimiento del nodo D en la red a través de su asociación con el nodo E.

Opcionalmente, se contempla en HARP una alternativa híbrida en la que un nodo comprobará proactivamente el estado de los enlaces con nodos vecinos a través de mensajes balizas o *beacon*, pudiendo así iniciar los procesos de recuperación local de fallos sin necesidad de esperar a tener un paquete de datos que enviar. Esta opción conlleva un mayor gasto energético para mantener información del estado de los enlaces con los nodos vecinos pero puede llevar asociado numerosos beneficios en cuanto al rendimiento del protocolo en aplicaciones con alta tasa de fallos y alta movilidad.

3.5.4. Nodo sustituto

Según la descripción del protocolo efectuada hasta el momento, el nodo raíz es el responsable último de encontrar el destino de cualquier paquete que se desee

encaminar a través de la red. Esta situación conlleva que un hipotético fallo de este nodo pueda traducirse en una disminución considerable de la eficiencia del protocolo.

Para minimizar esta situación, se define en la red el rol de *nodo sustituto*, que tiene como misión desempeñar las funciones de nodo raíz ante un fallo en el funcionamiento del mismo. La elección de este nodo sustituto dependerá de la aplicación, pero como norma general será el nodo raíz quien seleccione e informe a la red que uno de los nodos ha sido designado como sustituto en función de variables como distancia, energía restante, capacidad de procesado, memoria, etc. Así, los nodos vecinos, que minimizan el alcance de la reestructuración de la red en caso de tener que producirse, son los que más posibilidades cuentan a priori para serlo.

Este rol es de suma importancia en la propuesta de HARP como protocolo para WSN, que será descrita en el Capítulo 6.

Capítulo 4

Descripción del simulador

4.1. Introducción

En un acercamiento inicial al mundo real de los sensores inalámbricos, se puede decir que este tipo de redes de comunicación son difíciles de programar en una primera aproximación, debido principalmente a la naturaleza distribuida de la red y a la interacción entre los propios dispositivos, que dificulta el proceso de depuración de los errores producidos en la programación. La simulación de redes de comunicaciones, además de una herramienta valiosa para la investigación y el desarrollo de protocolos y sistemas, es un instrumento igualmente útil cuando se utiliza previa al despliegue de redes reales. A pesar de sus limitaciones, entre las cuales se incluye el compromiso entre coste, complejidad y fidelidad, una simulación permite estimar el comportamiento y las prestaciones de una red antes de su despliegue y puesta en marcha.

En consecuencia, antes de su implementación en dispositivos reales, es una práctica extendida el desarrollo de estos algoritmos en entornos de simulación que permitan evaluar las funcionalidades implementadas en un entorno virtual pero que

ha sido diseñado adaptando diferentes procesos o situaciones reales, mientras que también proporciona acceso a todas las herramientas de depuración de un entorno de compilación actual. Las simulaciones permiten a su vez la evaluación de las prestaciones de un algoritmo en multitud de entornos cambiantes, en diferentes circunstancias y bajo determinadas condiciones, algo que casi con toda seguridad no estaría al alcance del diseñador si no dispusiese de esta herramienta. Además, un factor de suma importancia en WSN es que, mediante simulación, se puede reducir el tiempo de observación de la red, que en estos entornos se extiende hasta que todos los nodos de la misma agoten su batería, lo que puede llevar meses o incluso años.

Según estas premisas, una vez realizada la definición completa de los mecanismos y procedimientos contenidos en HARP, el siguiente paso de esta Tesis consiste en la evaluación de sus prestaciones mediante simulaciones y su comparación con otros protocolos para WSN. Esto conlleva en primer lugar la elección de un simulador que proporcione todas las herramientas necesarias para llevar a cabo la evaluación del rendimiento del protocolo.

En la Sección 4.2 se ha incluido una revisión del estado del arte de los simuladores existentes en la actualidad para WSN. Algunos de estas herramientas de simulación son bastante complejas de entender y utilizar, por lo que la implementación de HARP en uno de ellas conllevaría perder el control sobre los mecanismos de ejecución y los diferentes procedimientos en ellos establecidos. Para construir esta implementación sería necesaria la creación de un módulo nuevo con todos los procesos y mecanismos definidos en HARP y su integración con los módulos existentes a través de la compilación del código fuente en el simulador elegido. Debido al gran número de módulos que pueden incluir estos simuladores y a que su definición e implementación puede haber sido desarrollada por autores distintos, a menudo esta programación suele ser compleja y costosa, además de no garantizar que las funcionalidades implementadas sean ejecutadas tal y como se propone en

su descripción. Esta razón es la escogida como principal para no utilizar uno de estos simuladores en esta Tesis Doctoral. Además, la mayoría de los simuladores presentados han sido diseñados para el estudio de redes inalámbricas genéricas, sin tener en cuenta las características específicas de las WSN, mientras que por otra parte los simuladores creados explícitamente para WSN no poseen en general las funcionalidades necesarias para simular todos los mecanismos propuestos en HARP, tales como la posibilidad de encaminar distintos tipos de tráfico, la recuperación local o la movilidad de los nodos. Ésta es precisamente una de las grandes ventajas de crear un nuevo simulador: se tienen específicamente en cuenta las funcionalidades, procedimientos y parámetros que van a ser evaluados a la hora de proceder a su diseño.

Si bien es cierto que alguno de estos simuladores son de amplia utilización por la comunidad científica, el objetivo principal de esta Tesis, centrada en proponer un protocolo eficiente de encaminamiento para WSN, se alcanza de forma más efectiva a través de la propuesta y desarrollo de un simulador propio que permita controlar todos los mecanismos propuestos en HARP. Esta opción conlleva un mayor esfuerzo en términos generales, ya que en este escenario es necesaria no sólo la programación del protocolo en sí, sino también la definición y programación de un simulador completo que contenga todos los aspectos de funcionamiento necesarios, así como la implementación adicional de otros protocolos ya existentes para la comparación de prestaciones con HARP.

En consecuencia, tal y como se indicaba en los objetivos propuestos de esta Tesis, se ha desarrollado un nuevo simulador para WSN, que puede generalizarse para su utilización con cualquier red ad-hoc inalámbrica móvil, basándose en el modelo de simulación por eventos discretos [69], [79], [110], [117] en el que también se basan otros simuladores existentes incluidos en la sección siguiente. Este simulador, que ha sido desarrollado en un lenguaje de programación orientado a objetos como es Java [8], permite definir y modelar de forma precisa todos los as-

pectos relacionados con las WSN, tales como características de los nodos sensores, movimiento, desvanecimientos en enlaces, protocolos del nivel de enlace, colisiones entre paquetes, etc., así como controlar en todo momento las variables y los parámetros asociados a cada simulación.

A continuación se indica un resumen de las **etapas principales seguidas en la elaboración del modelo de simulación**:

1. *Fase de especificación*: formulación del problema
 - objetivos del simulador
 - asunciones básicas
 - parámetros que intervienen en la simulación
 - variables de entrada y salida del sistema
2. *Fase de definición de modelos*: análisis y definición de los modelos que se utilizarán en el simulador: propagación, consumo energético en nodos, etc.
3. *Programación del simulador*:
 - diagrama de clases Java
 - métodos a implementar para cada clase
 - diagramas de flujo
 - diagramas de estado y eventos
4. *Depuración de código*
5. *Validación del modelo*: comprobación de la validez de los modelos diseñados a través de la comparación del modelo simulado con resultados teóricos
6. *Diseño de experimentos*:
 - definición de las estrategias utilizadas

- definición de los casos y entornos objeto de simulación
- definición de los parámetros y variables a evaluar
- especificación del tipo y número de simulaciones

7. *Ejecución de simulaciones y extracción de resultados*

8. *Análisis y recopilación de resultados*

9. *Conclusiones*

Una vez definido el simulador, el siguiente paso es la programación del protocolo propuesto y su integración modular dentro de la estructura de programación del mismo, así como la inclusión de manera análoga de otros protocolos de red que se utilizarán para realizar la comparación de prestaciones con respecto a HARP. La descripción del simulador desarrollado y de todo sus módulos se incluye en la Sección 4.3 de este capítulo.

4.2. Estado del arte en simuladores para WSN

En este apartado se citan algunos de los simuladores existentes en la actualidad para WSN. A pesar de que algunos de ellos no son específicos para redes de sensores sino que su utilización está generalizada para redes ad hoc inalámbricas, el uso que de ellos se hace en algunos de los artículos de investigación incluidos en la sección de bibliografía aconseja su inclusión dentro de este apartado.

A continuación se enumeran y describen estos simuladores:

- **ns-2** [136]: simulador de eventos discretos de código abierto dirigido a la creación de redes de investigación. Fue desarrollado en sus orígenes por la Universidad de Berkeley y actualmente cuenta con el apoyo de la Agencia

de Investigación de Proyectos Avanzados de Defensa de los Estados Unidos (DARPA). Utiliza C++ y OTcl (*Object Tool Command Language*). Esta herramienta proporciona un apoyo sustancial para la simulación de TCP, encaminamiento y los protocolos *multicast* en redes cableadas e inalámbricas (locales y por satélite). Es un simulador ampliamente utilizado por la comunidad científica aunque la integración de un módulo nuevo puede ser costosa debido a la heterogeneidad en la programación de los ya existentes.

- **ns-3** [100]: al igual que su predecesor, es un simulador de eventos discretos y de código abierto diseñado principalmente para uso educativo e investigador. Pensado para sustituir a ns-2, este simulador no es una versión actualizada del mismo sino que es una herramienta totalmente nueva y que no es compatible con su predecesora. Actualmente se encuentra en el proceso de lanzamiento y estabilización de una primera versión *ns* – 3.10 que data de enero de 2011.
- **OPNet** [9]: herramienta comercial que se utiliza principalmente para aplicaciones de análisis, diseño, modelado y simulación de redes de comunicaciones, dispositivos, protocolos y aplicaciones de red. Proporciona facilidades para la gestión del rendimiento, el análisis de configuración de red y planificación de la misma. OPNET dispone de un amplio abanico de protocolos y tecnologías implementadas, como son PPP, DSL, ATM, CSMA/CA, CSMA/CD, Ethernet hasta 10Gps, Token Ring 802.5, Frame Relay, IPv4, IGRP, OSPF, RIP, EIGRP, BGP, TCP, UDP, etc. Además incluye un entorno de desarrollo que permite modelar los últimos protocolos diseñados como IPv6, IPv6 Móviles, OSPFv3, etc. OPNet dispone además de versiones académicas para la enseñanza en universidades y centros de docencia.
- **OMNeT++** [134]: software de código libre programado en C++, basado en el diseño modular de componentes y con un entorno de simulación de

arquitectura abierta. Proporciona una interfaz gráfica robusta y un núcleo de simulación embebido. Su principal aplicación es la simulación de redes de comunicaciones debido a su arquitectura flexible y genérica, lo que lo convierte en una plataforma de simulación versátil cada vez con mayor éxito dentro de la comunidad científica.

- **GloMoSim** [150]: entorno de simulación escalable para redes inalámbricas con licencia académica (no es software libre) programado en Parsec (*Parallel Simulation Environment for Complex Systems*), un lenguaje basado en C y especialmente diseñado para la simulación de eventos discretos en redes de comunicación de gran escala. Su desarrollo está detenido desde hace algunos años, principalmente debido a que su desarrollador original (Rajive Bagrodia, UCLA) se ha volcado en la versión propietaria de GloMoSim (QualNet) a través de la empresa SNT (*Scalable Network Technologies* [10]), programa del que sí se lanzan nuevas versiones periódicamente. Actualmente GloMoSim permite simular protocolos de la capa de enlace como son CSMA y 802.11, así como protocolos de red (como IP), transporte (como TCP y UDP) y del nivel de aplicación (tales como FTP, HTTP y Telnet). La última versión disponible es la v.2.0 que data de Diciembre de 2000.
- **SensorSim** [101]: herramienta de simulación construida sobre el simulador ns-2 que proporciona facilidades adicionales para la simulación de redes de sensores, como son: modelos nuevos para el nodo sensor y el canal inalámbrico, modelo energético para las baterías, escenario de generación nuevo, etc.
- **TOSSIM** [11]: simulador para WSN que utiliza TinyOS [68]. TinyOS es un *kernel* (parte central o núcleo) de un sistema operativo de código libre basado en eventos, el cual proporciona modularidad y concurrencia en las operaciones típicas de las WSN. Para desarrollar los algoritmos y protocolos en este simulador se utiliza nes-C [56], que proporciona un lenguaje para de-

sarrollar sistemas embebidos como los dispositivos sensores, en el que cada aplicación se crea ensamblando diferentes componentes. NesC es un lenguaje de programación que se utiliza para crear aplicaciones para dispositivos sensores programados en el sistema operativo TinyOS. Ambas propuestas permiten integrar estos dispositivos en un entorno basado en eventos, ante los cuales reaccionan y se comunican entre sí. TinyOS proporciona librerías de protocolos de red (*dissemination* para la distribución de paquetes a los nodos de la red, y *collection* para la recogida de datos desde los nodos de la red hacia el nodo sumidero), servicios distribuidos o módulos y librerías para dispositivos sensores específicos (como el chip transmisor de RF Chipcon CC2420 [3]), entre otras facilidades.

- **JavaSim o J-Sim** [124]: JavaSim es un paquete de simulación orientado a objetos escrito en Java y que utiliza el lenguaje de *scripts* TCL para la definición de escenarios y simulaciones. J-Sim es de libre distribución desde 1997. Su principal ventaja radica en la sencillez con la que diferentes módulos pueden ser añadidos y eliminados de forma *plug-and-play*. Se utiliza fundamentalmente para la simulación y la emulación de WSN mediante la incorporación de uno o más dispositivos sensores reales. J-Sim contiene modelos reales para los nodos sensor y sumidero, canales de sensores inalámbricos y canales de comunicación y medios físicos, así como modelos para el consumo de energía. A pesar de contar con un buen diseño, flexible y extensible, J-Sim no ofrece demasiadas implementaciones de protocolos inalámbricos. Su última versión data de febrero de 2004.
- **Castalia** [2]: Castalia es un simulador para WSN basado en la plataforma OMNeT++ y en el cual pueden desarrollarse y evaluarse diferentes algoritmos y protocolos utilizando un modelo de canal y de radio real basado en medidas empíricas. Este simulador permite ajustar multitud de parámetros de funcionamiento, proporciona diferentes niveles de potencia de transmisión

y permite gestionar recursos tales como la CPU y la memoria del dispositivo sensor.

- **SWANS** [88], [7]: SWANS (*Scalable Wireless Ad-hoc Network Simulator*), es un simulador construido sobre JiST (*Java in Simulation Time*) [34] especialmente diseñado para evaluar el comportamiento de redes ad-hoc con un número elevado de nodos. SWANS está compuesto por módulos software independientes que proporcionan funcionalidades a cada uno de los elementos incluidos en la simulación. La utilización conjunta de JiST y SWANS permite alcanzar un alto grado de rendimiento en la simulación, permitiendo simular redes uno y dos órdenes de magnitud mayores en cuanto al número de nodos que lo que se puede alcanzar con GloMoSim y ns-2, respectivamente, utilizando la misma cantidad de memoria y de tiempo para llevarla a cabo.
- **WSNSim** [43]: WSNSim es un simulador desarrollado expresamente para WSN que permite la evaluación simultánea sobre una plataforma común de varios protocolos de encaminamiento específicos para este tipo de redes, como son LEACH, HEED y SNIPER. En la actualidad su uso no está generalizado.
- **VisualSense** [12]: es un entorno de simulación específicamente diseñado para WSN. Está integrado dentro del entorno Ptolemy (U.C. Berkeley) y utiliza un método de programación gráfica basada en Java y en C. VisualSense es de código abierto y existe en la actualidad un grupo que colabora en su mantenimiento y actualización. La última versión data de abril de 2008.
- **WSNet** [13]: WSNet es un simulador para redes inalámbricas basado en eventos discretos similar a otros como ns-2, JiST, GloMoSim y OMNeT++ pero con funcionalidades añadidas y modelado de los siguientes elementos: amplio catálogo de modelos radio diseñados y simulación precisa de cada uno de ellos, definición de estructuras complejas tales como sistemas MIMO

y soporte para múltiples interfaces radio, simulación del consumo de energía, soporte para adición y muerte de nodos en la red, implementación de medidas físicas por parte de los nodos sensores, disponibilidad de diversas técnicas de modulación, modelos diferentes de retardo de propagación e interferencias, soporte para técnicas de acceso múltiple como CDMA y FDMA, etc.

4.3. Descripción del simulador: WiNS

El simulador desarrollado en esta Tesis, denominado WiNS (*Wireless Network Simulator*), ha sido programado utilizando el lenguaje Java con objeto de proponer un modelo de desarrollo orientado a objetos capaz de servir como escenario natural de aplicación para las redes inalámbricas desplegadas en el mundo real. Algunas de las **características de Java** que lo hacen idóneo para este tipo de tarea son:

- Flexibilidad de modelado.
- Facilidad para el desarrollo y depuración de código.
- Utilización en múltiples plataformas y sistemas operativos.
- Ejecución rápida del modelo.
- Simulación continua y discreta.

WiNS es un simulador basado en eventos discretos que tiene como objetivo imitar el funcionamiento de los procesos reales en redes de comunicaciones inalámbricas y, específicamente, en WSN. Para ello genera una historia artificial de sucesos en el sistema y sus repercusiones, ordenando éstos cronológicamente desde el principio de la simulación. En contraposición a la simulación en tiempo continuo, en el que el avance de la simulación se produce por incrementos temporales fijos, esta técnica toma el control de la variable *tiempo* y permite avanzar en intervalos o

saltos de duración variable en función de la planificación de ocurrencia de tales eventos en un tiempo futuro. Un requisito para aplicar esta técnica es que las variables que definen el sistema no cambien su comportamiento durante el intervalo simulado.

Este método de simulación se basa en la definición de entidades (*entities*) que poseen ciertos atributos (*attributes*) o propiedades y que pueden realizar diferentes acciones (*activities*). En Java esta implementación es inmediata, ya que las entidades serán los objetos de las clases definidas, los atributos serán sus variables y las acciones serán equivalentes a los métodos definidos para cada clase.

En este tipo de simulación el estado del sistema, definido como el conjunto de variables o parámetros que lo describen en un momento dado en relación con los objetos existentes, cambia cuando se produce un evento, que no es más que un suceso que ocurre en un determinado instante de tiempo y que cambia algunas variables del sistema. En resumen, la aparición de estos sucesos hace que el estado del sistema cambie en un conjunto discreto de puntos en el tiempo, llevando al sistema completo a un nuevo estado con variables de simulación diferentes. Para adaptar esta técnica a la simulación de redes de comunicación, WiNS trata los eventos como *paquetes* (instancias de la clase **Packet**, descrita en uno de los apartados siguientes), los cuales vendrán determinados por dos tipos de sucesos distintos:

- Generación, procesado, transmisión y recepción de un paquete de datos o de control por parte de un nodo.
- Cambio en las condiciones de simulación: funcionamiento de los nodos, canales inalámbricos, movilidad y despliegue de nuevos nodos en la red.

Cuando se desee introducir un evento en el simulador, se creará un objeto *paquete* y se insertarán como atributos todos los datos necesarios para su identi-

ficación, comenzando por el tipo de evento según la clasificación anterior (*paquete de transmisión* o *paquete dinámico*, respectivamente) y su instante temporal de aparición.

Los nodos de la red se modelan como objetos independientes pertenecientes a la clase **TreeNode** (descrita a continuación) que tienen capacidad de ejecutar sus propias tareas y están caracterizados por parámetros tales como memoria, velocidad de procesamiento, tamaño de buffer, etc. Cada uno de ellos tiene capacidad para actuar de forma autónoma según la información almacenada y mantenida en sus tablas de encaminamiento, adaptándose a los cambios de su entorno tales como fallos en enlaces o movilidad. Estos enlaces entre nodos también son objetos (en este caso instancias de la clase **Link**) y WiNS permite configurar el nivel físico adaptando diferentes modelos de propagación. En consecuencia, es posible caracterizar un enlace inalámbrico mediante su retardo de propagación, las pérdidas de potencia que introduce y su probabilidad de error, en un modelo equivalente a una cola FIFO con tiempo de servicio igual al tiempo de propagación.

WiNS permite definir funcionalidades de capa 2 (acceso al medio) y capa 3 (encaminamiento) según el modelo OSI, estableciendo el *paquete* como unidad mínima de transmisión y enmascarando los procesos destinados a las transmisiones de cada uno de los bits que lo conforman. Al tener un planteamiento modular, este simulador es totalmente configurable para adaptarse a las circunstancias y parámetros de cada simulación. En las secciones siguientes se describirá la estructura del simulador y la integración de las diferentes clases Java dentro del mismo, constituyendo todo ello una herramienta de simulación con más de 12.600 líneas de código Java según la Tabla 4.1.

En las Figuras 4.1 y 4.2 se muestran respectivamente una ventana de Netbeans [5], que es el entorno de desarrollo Java utilizado en esta Tesis para desarrollar el simulador, y dos fragmentos de la traza de salida con los resultados obtenidos

clase Java	nº de métodos implementados	nº de líneas de código
Main.java	1	100
TreeD.java	30	1000
StartSimulation.java	27	1200
Packet.java	22	350
Dynamic.java	5	300
TreeNode.java	62	5500
Routing_Table.java	38	500
*_sub.java	20	400
Link.java	8	50
Queue.java	36	520
Medium_Acess.java	12	1000
Failure_Manager.java	12	200
PostProcess.java	39	1500

Tabla 4.1: Clases Java definidas en WiNS, número de métodos definidos e implementados y líneas de código aproximadas de cada una de ellas (la información de las clases que heredan se ha incluido en los datos de la clase principal).

para AODV durante una simulación comparativa con el protocolo HARP propuesto. En la primera de ellas el entorno está dividido en dos ventanas: en la ventana superior se accede al código de las clases Java implementadas, mientras que en la parte inferior se muestra la ventana de salida del sistema con los mensajes proporcionados por el simulador relativos a las acciones ejecutadas en la simulación de AODV. Las trazas de salida del sistema proporcionan información acerca de la simulación y sirven de entrada a la herramienta MATLAB [6] utilizada para la representación gráfica de los resultados obtenidos.

4.3.1. Cola de eventos

Un simulador por eventos discretos está constituido por un elemento principal denominado *Future Events List* (FEL) o cola de eventos del sistema (en esta Tesis también denominada *eventqueue*) y que contiene todos los eventos generados o

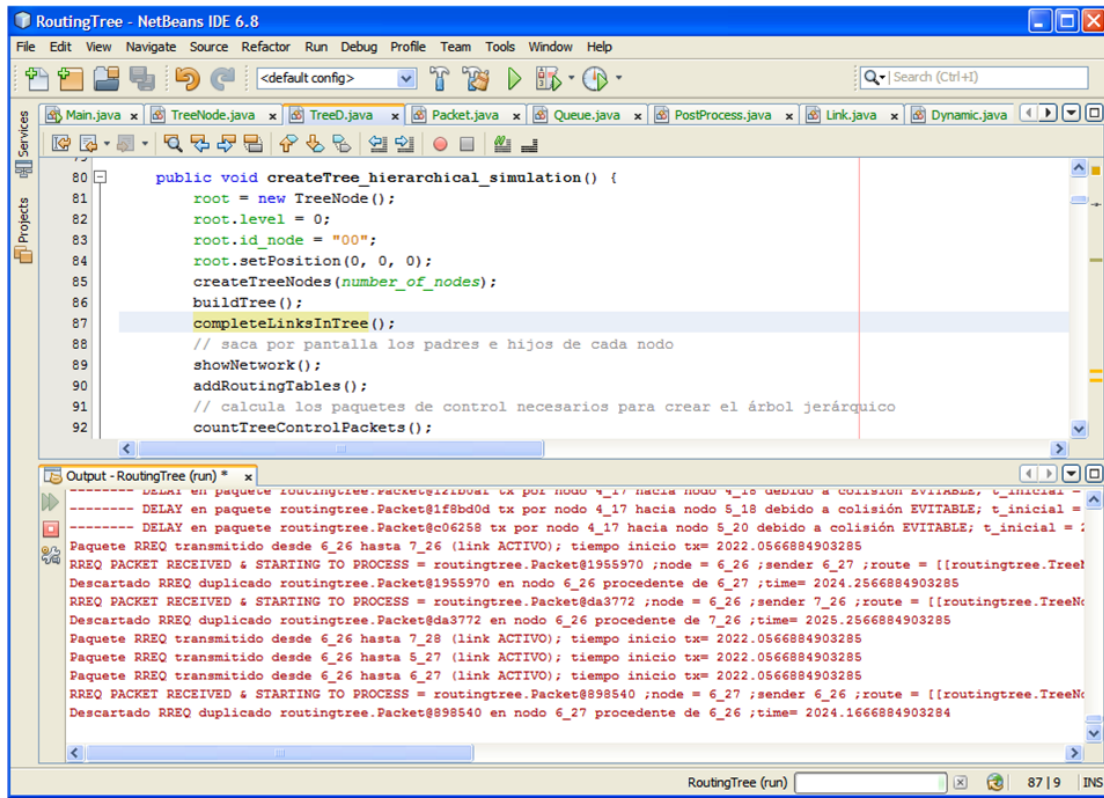


Figura 4.1: Ejemplo de simulación con WiNS utilizando la herramienta Netbeans sobre máquina virtual Java: la ventana superior contiene el código de cada una de las clases Java programadas, mientras que la ventana inferior muestra los mensajes de salida con las acciones que se realizan mientras se ejecuta una simulación.

planificados en un instante determinado y los posteriores, solucionando de esta forma el problema de la simulación de eventos que ocurren en paralelo a través de un tratamiento asíncrono de los mismos [53].

Cada evento de la simulación lleva asociado un instante concreto en el que sucede y a cada uno de ellos se le atribuye una duración determinada, ya sea de forma determinista (valor constante) o probabilístico (mediante una variable aleatoria). Los eventos se ordenan en la cola en función de este instante de generación, de menor a mayor, y cada uno de los diferentes tipos de eventos definidos tiene asignado un gestor o *handler* que se ocupa de ejecutar una serie de acciones específicas asociadas a ese evento cuando éste es procesado por la cola. De esta forma, cada

```

9_resultados_AODV.txt - WordPad
Archivo Edición Ver Insertar Formato Ayuda

R/L: 0.7027686718431019
DELAY_OF_DELIVERY_PACKET: 41.127467670587066
average of time per hop: 8.225493534117414
R/L: 0.5847388802584684
DELAY_OF_DELIVERY_PACKET: 58.90110177180759
average of time per hop: 11.780220354361518
R/L: 0.6990551754751603
DELAY_OF_DELIVERY_PACKET: 56.434024868423876
average of time per hop: 9.40567081140398
R/L: 0.5158274451331591
DELAY_OF_DELIVERY_PACKET: 31.41432619345869
average of time per hop: 7.853581548364673
R/L: 0.851155197370571
DELAY_OF_DELIVERY_PACKET: 40.14078370366042
average of time per hop: 8.028156740732083
R/L: 0.7639320225002103
% control packets: 95.0
% delivered packets: 100.0
number of delivered packets: 60
directividades: [100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0, 100.0]
DIRECTIVIDAD MEDIA: 100.0
BUFFER: [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 3, 4, 5, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
BUFFER r: [1001.11, 1004.4601273917309, 1010.5501390719702, 1012.4808079959447, 1014.5908079959447, 1015.0177772
RTabsize: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27,
Delay in queue: [0.0, 3.56561224822849, 0.0, 0.0, 0.0, 0.0, 0.5730307086116682, 0.9869637461644061, 0.0, 0.1614894799
BUFFER: [1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 1, 1, 2, 3, 4, 1, 1, 1, 2, 3, 4, 5, 1, 1, 1, 2, 3, 4, 5, 1, 1, 1, 1, 1, 1, 1]
BUFFER r: [1001.11, 1005.5714667947308, 1010.9331578253643, 2001.11, 2006.6591856067787, 2007.8416747538104, 201
RTabsize: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
Delay in queue: [0.0, 0.0, 0.0, 0.0, 3.639393695156059, 3.4569045481243847, 0.0, 0.0, 0.0, 8.716098218354546, 9.
BUFFER: [1, 1, 1, 1, 1, 1, 1, 2, 3, 4, 5, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 3, 4, 1, 1, 1, 1, 2, 3, 4, 5, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
BUFFER r: [1001.11, 1004.4601273917309, 1007.44, 1010.9331578253643, 1020.7625815205325, 2001.11, 2007.439999999
RTabsize: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Delay in queue: [0.0, 1.2013394029999063, 0.0, 0.0, 0.0, 0.0, 11.927478655544292, 12.525803901733525, 12.4885699
BUFFER: [1, 1, 1, 2, 3, 4, 5, 6, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 3, 4, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
RUFFFR r: [1001.11, 1005.5714667947308, 1006.931525584233, 1007.44, 1009.0415255842331, 1015.8275684575802, 1018

```

(a)

```

9_resultados_AODV.txt - WordPad
Archivo Edición Ver Insertar Formato Ayuda

node in path: 1_4
node in path: 2_12
node in path: 3_15
node in path: 4_15
node in path: 4_13
delivered packet: routingtree.Packet@139f97b ;origin= 00 ;dest= 4_14
node in path: 00
node in path: 1_3
node in path: 2_10
node in path: 3_16
node in path: 4_13
node in path: 4_14
delivered packet: routingtree.Packet@1073d48 ;origin= 00 ;dest= 4_15
node in path: 00
node in path: 1_3
node in path: 2_11
node in path: 2_12
node in path: 3_18
node in path: 4_16
node in path: 4_15
delivered packet: routingtree.Packet@17e222c ;origin= 00 ;dest= 4_16
node in path: 00
node in path: 1_4
node in path: 2_12
node in path: 3_15
node in path: 4_16
delivered packet: routingtree.Packet@17600c2 ;origin= 00 ;dest= 4_17
node in path: 00
node in path: 1_3
node in path: 2_10
node in path: 3_10
node in path: 4_11
node in path: 4_17
number of hops: 202
% CONTROL: 4291

```

(b)

Figura 4.2: Fragmentos de una traza de resultados obtenidos en una simulación del protocolo AODV.

evento de la cola de eventos deberá esperar a ser el primero para poder ser procesado, momento en el cual la cola iniciará el *handler* correspondiente a través de una llamada a su método. Como resultado de la ejecución de este proceso, puede ocurrir que no se genere ningún otro evento (por ejemplo, cuando el paquete de datos ha llegado a su destino y ha sido procesado), o que sí se genere un nuevo evento, en cuyo caso será añadido a la cola de eventos en el orden cronológico que le corresponda (por ejemplo, cuando un nodo recibe un paquete y debe reenviarlo a otro nodo). Una vez el evento ha sido procesado por la cola de eventos será eliminado de la misma y el reloj del sistema avanzará hasta el instante determinado por el siguiente evento, es decir, el más inminente.

La situación anterior se ilustra en la Figura 4.3, donde la cola de eventos o *eventqueue* gestiona un paquete transmitido desde el objeto *txnode* hacia el objeto *rxnode*. En el modelo propuesto, el gestor de paquetes de la cola se inicia mediante una llamada al método *processPacket()* del objeto *eventqueue*, es decir, mediante la ejecución de *eventqueue.processPacket()*. Este gestor procesa el primer evento de la misma, que se corresponde con el inicio de transmisión del paquete P_1 y, como resultado, el simulador llama al método *transmitPacket()* definido en la clase **TreeNode** para comenzar dicha transmisión. En este procedimiento el gestor deberá hacer uso de los módulos existentes en el simulador necesarios para poder acceder al canal y efectuar dicha transmisión, ejecutando las acciones oportunas en función de las características del mismo, resultando la actualización de algunas variables del sistema tales como retardo de propagación o pérdida de potencia en la transmisión. A continuación, se efectúa una llamada al método *receivePacket()* de la clase **TreeNode**, el cual se hará cargo de la recepción del paquete tras el mencionado retardo, evaluando si esta recepción es exitosa o no en función de las variables calculadas y otras predefinidas tales como ocupación máxima del buffer de entrada del nodo. Una vez recibido el paquete, el siguiente paso es su procesado, con lo que se ejecutará el método *processPacket()* de la clase **TreeNode** (que no

es el mismo que el de la clase *eventqueue*). Si como resultado de este procesado se generase un nuevo evento que simule la retransmisión del paquete a otro nodo, se efectuará en ese momento una llamada al método *movePacket()* de la cola de eventos que se encargará de insertar el nuevo evento de transmisión en dicha cola en el orden que le corresponda en función de su instante temporal ($t + t_1 + t_2 + t_3$). Esta situación se repite hasta que no queden eventos por procesar en la cola del sistema.

Además de los eventos descritos, en el simulador se definen otro tipo de entidades destinados a modificar las condiciones de la simulación, tales como fallos en nodos, cambios en el estado de los enlaces o en las condiciones de propagación y movilidad de nodos. Estos eventos se denominan *paquetes dinámicos* y se crean y gestionan a través de la clase **Dynamic**. Para diferenciarlos de los paquetes de datos y de control, estos paquetes serán de tipo *dynamic* y tendrán este valor en el campo correspondiente. La cola de eventos gestionará todos los eventos de este tipo en el instante que les corresponda mediante una llamada a su *handler* correspondiente, definido como un método perteneciente a la clase **Dynamic**.

4.3.2. Estructura de clases

El diagrama de jerarquía de clases definidas en WiNS y las relaciones entre ellas se muestra en la Figura 4.4. A continuación se hará una breve descripción de las clases Java de las que se compone el simulador WiNS.

4.3.2.1. Main.java

La clase **Main**, como su propio nombre indica, es la clase principal del simulador a partir de la cual se construyen todos los objetos que participarán en el proceso de simulación. Sus funciones principales son: i) crear los ficheros de

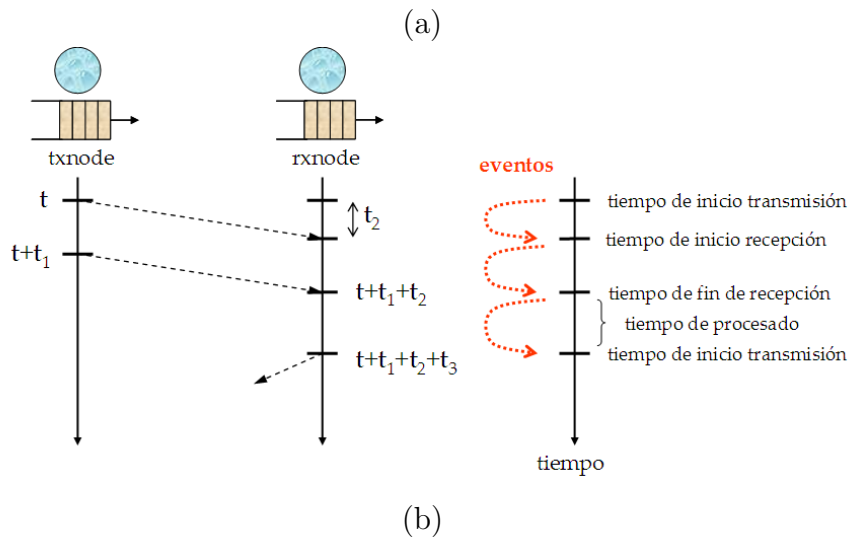
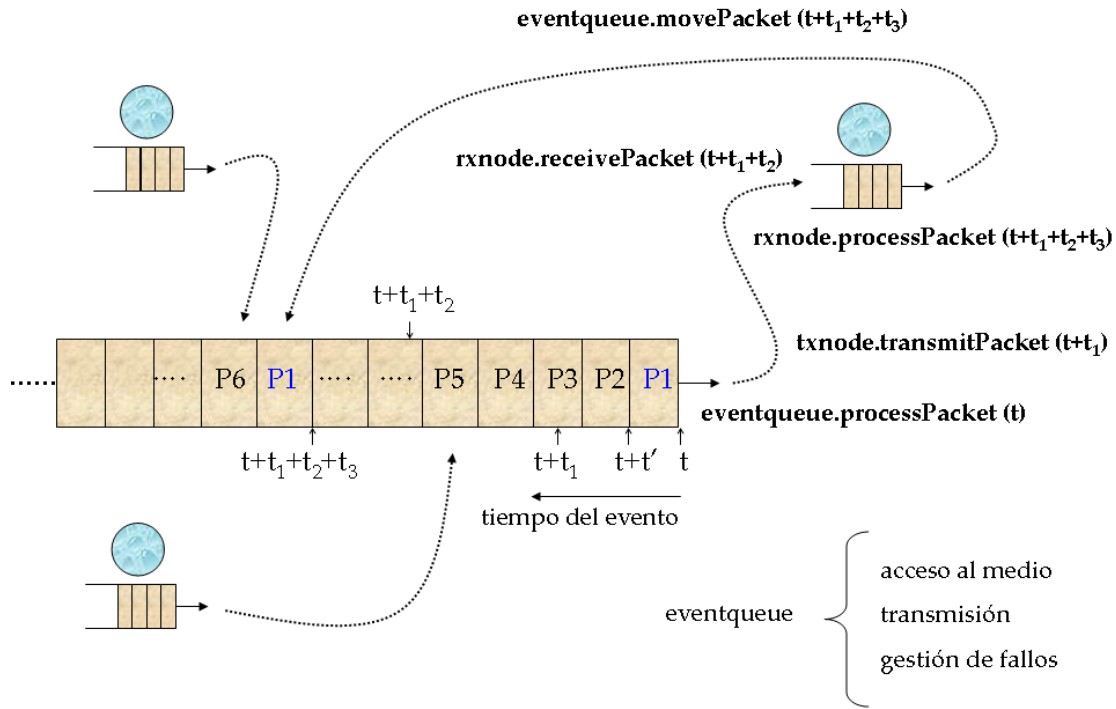


Figura 4.3: Proceso de simulación de la transmisión y recepción de un paquete entre dos nodos: a) cronología de acciones y procesos ejecutados en la cola de eventos del sistema asociados a la retransmisión del paquete P_1 ; b) diagrama temporal de eventos del sistema: inicio de transmisión, inicio de recepción, fin de transmisión, fin de recepción, inicio de procesado, fin de procesado e inicio de retransmisión.

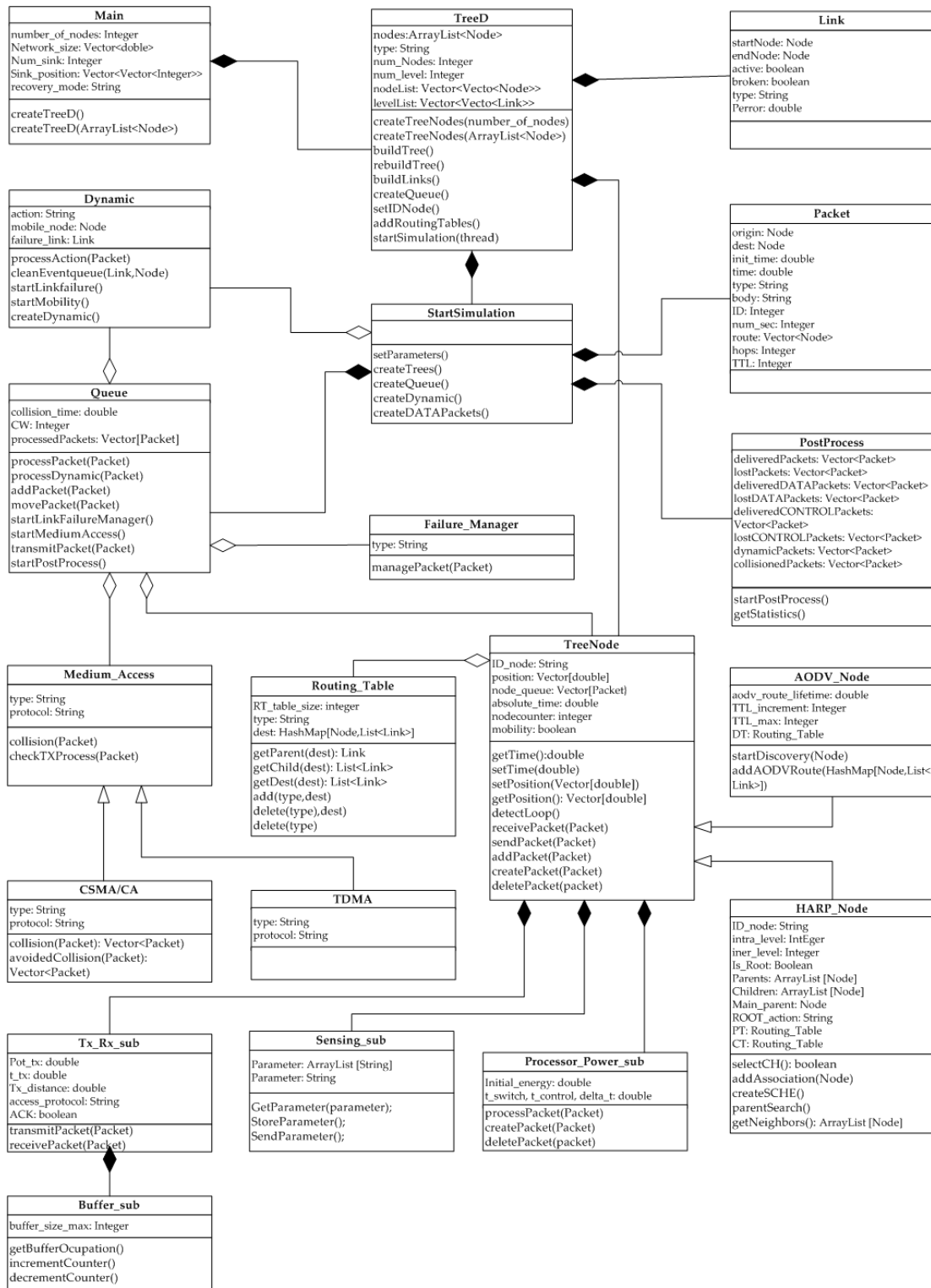


Figura 4.4: Diagrama de clases de WiNS en el que se muestran las relaciones entre las mismas y los métodos y parámetros principales de cada una de ellas.

traza para que el simulador pueda escribir en ellos todos los datos y resultados obtenidos de la simulación; ii) crear completamente la red de nodos y todos los objetos asociados a la simulación a través de la construcción de un objeto de la clase **TreeD** (que se ocupará de estas acciones); iii) iniciar la simulación a través de la creación de una tarea (*thread* en Java) de la clase **StartSimulation**. En caso de que se estén comparando mediante simulación dos o más protocolos de encaminamiento, la clase **Main** procederá a crear tantas estructuras idénticas de simulación como comparaciones se realicen, iniciando cada una de ellas una tarea diferente. Esta posibilidad de crear diferentes tareas para ejecutar varias simulaciones simultáneamente es una de las ventajas de utilizar un entorno de desarrollo basado en Java.

4.3.2.2. TreeD.java

Esta clase Java tiene por objeto la creación de la estructura completa de red: nodos, enlaces inalámbricos y cola del sistema. En HARP, además de componer la red, construye el árbol jerárquico al inicio de la simulación, asignando diferentes niveles a los nodos en función de su posición y distancia al nodo sumidero, y actualizando las tablas de encaminamiento de cada uno de ellos con la información relativa a padres e hijos dentro de la jerarquía del árbol.

Para la creación de los nodos, el simulador procede a la lectura de un fichero de entrada en formato texto (*.txt*) con las coordenadas (x, y, z) de cada uno de los mismos. Una vez creados los nodos, se inicia el proceso de generación de enlaces inalámbricos entre los mismos, de tal forma que un nodo pueda comunicarse con cualquier otro nodo de la red a través de uno de estos enlaces. Por último, el sistema les asigna un identificador (ID) para distinguirlos del resto de nodos de la red. Todos los nodos y enlaces creados son objetos en Java y se almacenan en

vectores ordenados por niveles para permitir un acceso fácil a los mismos a lo largo de la simulación.

4.3.2.3. StartSimulation.java

Definida como “*public class StartSimulation extends Thread ...*”, esta clase contiene las variables y métodos necesarios para comenzar la simulación una vez se ha creado toda la estructura de objetos necesaria. Su primera acción será la construcción de los paquetes de datos como objetos de la clase **Packet**. Estos paquetes simulan la lectura por parte de un nodo sensor de un parámetro cualquiera del entorno, así como cualquier otro evento que pueda ser captado por un dispositivo sensor y que necesite ser transmitido a través de la red. A su vez, pueden ser utilizados en aplicaciones de diseminación de datos desde el nodo sumidero hacia los nodos sensores de la red y también para el intercambio de información entre dos nodos cualquiera de la misma. La creación de paquetes es configurable por el usuario, pudiendo ser de tasa constante (por ejemplo, en simulaciones para monitorización de campos de cultivo con WSN) o siguiendo algún tipo de distribución estadística (exponencial, uniforme, etc.).

El procedimiento general consistirá en crear al principio de la simulación todos los paquetes de datos que deberán ser encaminados a través de la red. Sin embargo, también es posible la creación al inicio únicamente de algunos de ellos, mientras que el resto se generan de forma dinámica conforme avance la simulación. Cada uno de estos paquetes contendrá campos (atributos del objeto) en los que se indicarán sus características, tales como tipo, origen, destino, tiempo inicial, TTL o identificador.

De la misma forma que los procesos de creación de paquetes de datos, esta clase está destinada a la creación de eventos de la clase **Dynamic** que cambian las condiciones iniciales de los enlaces e introducen movilidad en los nodos. Se han desarrollado diversos algoritmos con objeto de modificar estas condiciones en dife-

rentes circunstancias. Algunos de ellos establecen todos los cambios y los patrones de movilidad de cada nodo al principio de la simulación. Sin embargo, debido a la complejidad existente en modelar cada uno de estos procesos al inicio de la misma, se contempla como mejor opción establecer unos patrones de actividad para cada uno de los objetos de la red (nodos y enlaces) y, mediante la programación de un algoritmo específico, ir creando este tipo de eventos a lo largo de la simulación en función de los eventos procesados por la cola, es decir, que un *handler* específico de la clase **Dynamic** corra este algoritmo de cálculo de nuevos eventos futuros para cada uno de los objetos de red. Esta es la opción escogida para las simulaciones finales incluidas en esta Tesis Doctoral.

Una vez creados los eventos (paquetes de datos o paquetes *dynamic*), la tarea de inicialización introduce en la cola del sistema cada uno de ellos ordenadamente según su tiempo de generación. Por último, para empezar a procesar los eventos de la cola se efectúa una llamada al método *processPacket()* definido en la clase **Queue**, el cual extraerá el primer evento de la misma y le asignará un *handler* en función de su tipo. Esta tarea no se detendrá hasta que la cola de eventos no quede totalmente vacía, en cuyo caso se inicia el postprocesado de la información relativa a la simulación con objeto extraer los resultados finales.

4.3.2.4. Packet.java

Durante el desarrollo de la simulación se crearán instancias de esta clase para modelar tanto los paquetes de datos como los de control transmitidos y recibidos entre los diferentes nodos de la red. Cada objeto de esta clase contiene atributos en los nodos que los generan, procesan o reenvían introducen información relativa al encaminamiento del mismo, como son nodos origen y destino, su tipo, nodos intermedios por los que pasa o información acerca del camino seguido, número de saltos hacia destino, tiempo inicial y tiempo actual, identificadores o números de

secuencia, vector con nodos hijo, *flags* de transmisión, TTL, cuerpo del paquete, etc. Los métodos definidos para esta clase tienen únicamente como objetivo permitir la lectura y escritura de los campos anteriores por parte de las instancias de los nodos de la red.

4.3.2.5. **Dynamic.java**

Como se ha introducido anteriormente, esta clase gestiona los eventos creados a partir de *paquetes dinámicos* en la red cuya misión es la de modificar, por una parte, las características y condiciones de propagación en los enlaces (cambios en el entorno, obstáculos, condiciones ambientales, etc.) y, por otra parte, establecer los patrones de fallo, movimiento y nuevos despliegues de los nodos en la red.

Estos eventos contienen atributos relativos al tipo de acción requerida y a sus características, así como un atributo adicional que indica el instante en el que deben procesarse por parte de la cola de eventos del sistema.

La clase **Dynamic** contiene los métodos necesarios para gestionar cada uno de los distintos tipos de paquetes dinámicos que se pueden generar, es decir, cada vez que la cola de eventos del sistema procese uno de estos paquetes efectuará una llamada al *handler* definido e implementado dentro de esta clase Java a través del comando *processAction()* con el objeto paquete como parámetro. Este método efectuará a su vez otra llamada al método correspondiente, el cual gestionará el *paquete dinámico* específicamente de una u otra forma según sea su naturaleza.

A su vez, como consecuencia del método escogido en WiNS para la creación de eventos de cambio de condiciones de simulación, la clase **Dynamic** posee métodos específicos con diferentes algoritmos que generan el siguiente evento a uno dado y lo introduce en la cola de eventos para su futuro procesado. De esta forma, se permite que la simulación vaya generando este tipo de eventos de forma secuen-

cial y adaptativa en función del instante de tiempo y del algoritmo seleccionado, evitando la creación de todos ellos al principio de la simulación.

Por último, es necesario también incluir mecanismos que bajo determinadas circunstancias puedan acceder a la cola de eventos y eliminar algunos de ellos. Estas circunstancias se producen por ejemplo cuando un nodo agota totalmente su batería (lo que se conoce comúnmente como muerte del nodo) o cuando un nodo se mueve de una posición a otra y los enlaces inalámbricos con los nodos vecinos se rompen. En consecuencia, **Dynamic** incluye métodos específicos para buscar en la cola de eventos y detectar eventos futuros asociados a nodos muertos o enlaces rotos debidos a movimiento de nodos para proceder a su eliminación de la misma.

4.3.2.6. **TreeNode.java**

La misión de esta clase es la de definir los atributos, parámetros, funciones y métodos necesarios para modelar el comportamiento de un nodo sensor inalámbrico en una WSN. Para hacer el sistema más modular y facilitar la gestión del código, se han definido algunas clases adicionales que se ocupan del modelado de algunos sistemas específicos. Estas clases son **Sensing_sub.java**, **Processor_Power_sub**, **Tx_Rx_sub.java** y **Buffer_sub.java** y se describen en un apartado posterior. A través de la definición de este modelo la clase **TreeNode** puede ocuparse específicamente de la gestión del encaminamiento y efectuar llamadas a procedimientos definidos en las clases anteriores cada vez que lo necesite.

TreeNode implementa los métodos generales de funcionamiento de un nodo sensor y crea las variables y objetos necesarios para la simulación de estos dispositivos, tales como tablas de encaminamiento o instancias para cada uno de los subsistemas asociados a las clases indicadas en el párrafo anterior (subsistema de

transmisión y recepción, subsistema de memoria, batería y procesado y subsistema de recogida de datos del entorno).

Para la definición de los métodos específicos de los diferentes protocolos de encaminamiento, incluido HARP, WiNS hace uso del concepto de herencia en Java, característica muy potente que permite definir una clase tomando como base otra clase ya existente. Al heredar de una clase base, las clases inferiores heredan tanto los atributos como los métodos definidos en aquélla, lo que permite la reutilización de código en el simulador. Según esto, se crearán tantas clases como protocolos de encaminamiento sean necesarios, heredando todas ellas los métodos y atributos definidos en **TreeNode** e implementando de forma individual sus propias funcionalidades mediante métodos específicos asociados a cada uno de los protocolos. En el diagrama de clases de la Figura 4.4 están representadas dos de estas clases, que son `AODV_node` y `HARP_node`, mostrando que dos protocolos de encaminamiento en WinS se diseñan como módulos separados dentro de la estructura del simulador.

Tal y como puede observarse en la Figura 4.3 a), cada nodo de la red lleva asociado una cola o buffer de entrada para gestionar los paquetes recibidos de otros nodos de la red. Esta cola se modela a través de un objeto perteneciente a la clase `Bufere_sub` que se describirá a continuación y que se crea cuando un objeto de la clase **TreeNode** es creado. La gestión de esta cola es crítica a la hora de permitir una perfecta integración dentro del modelo de simulación secuencial propuesto. En concreto, hay que tener en cuenta que un paquete que ha sido recibido y se encuentra en este buffer de entrada no puede ser procesado hasta que el anterior haya sido transmitido, y esto sólo ocurre cuando alcanza la primera posición de la cola de eventos y es procesado por la misma, momento en el cual se ejecutarán los procesos necesarios para simular su acceso al medio y su transmisión a través del canal inalámbrico. En consecuencia, es necesario definir un *flag* específico e iniciar una tarea que compruebe este hecho, de tal forma que cuando el paquete anterior

haya sido enviado, el *flag* correspondiente en el nodo transmisor cambie de estado y éste pueda proceder a la gestión del siguiente paquete de su buffer de entrada. Obviamente, si este proceso se demora o el nodo transmisor debe esperar un tiempo para poder transmitir, puede suceder que el número de paquetes recibidos por el nodo en ese intervalo alcance el límite máximo de tamaño de buffer definido como parámetro de simulación, momento en el cual se descartará el siguiente paquete que llegue a la cola de recepción del nodo.

4.3.2.7. **Routing_Table.java**

Esta clase permite crear un objeto que hará las funciones de tabla de encaminamiento asociado a cada instancia de **TreeNode**. Esta clase definirá la estructura general de una de estas tablas, creándose tantas como sean necesarias para el funcionamiento del protocolo. En HARP se crearán dos objetos de esta clase para cada nodo, una correspondiente a la tabla de padres (PT) y otra para la de hijos (CT). En AODV, sin embargo, sólo será necesaria una tabla para gestionar la información de todas las rutas conocidas por un determinado nodo de la red.

Routing_Table.java contiene diferentes métodos para que todos los protocolos puedan leer y escribir en las tablas de encaminamiento, ya sea para buscar una ruta hacia un nodo destino, para añadir una nueva o para borrar una o más rutas asociadas a un destino inalcanzable.

4.3.2.8. ***_sub.java**

Tal y como se ha mencionado en la descripción de la clase **TreeNode**, este conjunto de clases, formado por **Sensing_sub.java**, **Processor_Power_sub**, **Tx_Rx_sub.java** y **Buffer_sub.java**, tienen como objetivo la creación de objetos que modelen cada uno de los subsistemas asociados a un nodo sensor, por lo que

se creará una instancia de cada una de ellas en el momento que lo haga una de la clase **TreeNode**. Para gestionar estos subsistemas durante la simulación el objeto nodo creado efectuará llamadas a los métodos implementados en cada una de ellas incluyendo los parámetros definidos por dichos métodos, lo que libera a la clase **TreeNode** de implementarlos y favorece la modularidad del sistema.

4.3.2.9. Link.java

El modelado de los enlaces o canales inalámbricos de transmisión entre nodos se define a través de la implementación de esta clase Java, cuyos objetos están compuestos por los siguientes atributos:

- *startNode*: nodo origen para establecer el enlace
- *endNode*: nodo en el otro extremo del enlace
- *active*: estado del enlace
- *broken*: condición de enlace roto (movimiento de nodos, etc.)
- *type*: tipo de enlace con relación al modelo de propagación especificado: *free-space model*, *two-ray model*, *shadowing model*, etc.
- *Error*: probabilidad de error del enlace

4.3.2.10. Queue.java

La clase **Queue.java** implementa las funciones de la cola de eventos del sistema definida en el Sección 4.3.1, conteniendo los métodos necesarios para procesar y distinguir los diferentes eventos de la cola y asignarles un *handler* específico a cada uno de ellos según su tipo.

Cuando el objeto de esta clase (*eventqueue*) procesa los eventos asociados a transmisiones de paquetes deberá hacer uso de funcionalidades implementadas en la clase **Medium_Access** con objeto de modelar el acceso al medio de transmisión para poder transmitir dicho paquete. Es necesario que el acceso al medio lo gestione esta clase ya que es ella quien maneja la cola de eventos y cuando se utilizan protocolos de acceso por contienda como CSMA es frecuente que ocurran colisiones entre los paquetes transmitidos. En estos casos, esta clase será capaz de gestionar las colisiones producidas (evitables y no evitables) en la transmisión de un paquete simulando este fenómeno a través del retraso (retardo en el tiempo) dentro de la cola de eventos de los paquetes involucrados en una colisión según se especifique en el protocolo utilizado (tiempo de *backoff*, etc.).

Otra de las misiones de esta clase es la de almacenar todos los eventos una vez procesados para poder realizar el postprocesado de los mismos cuando haya finalizado la simulación y extraer datos relevantes acerca de la misma relacionados con las medidas de eficiencia de los diferentes protocolos de red simulados.

4.3.2.11. **Medium_Acess.java**

El acceso al medio de transmisión se modela con esta clase Java que simula los comportamientos de las capas física y de enlace del modelo OSI. Una vez que el nodo quiere iniciar una transmisión a través de un enlace, es necesario que el nivel de enlace de datos se haga cargo del acceso al medio de dicho paquete. WiNS implementa métodos que simulan el envío de este paquete al interfaz de red correspondiente, en el que se calcula la energía de transmisión necesaria para una correcta recepción del paquete teniendo en cuenta el modelo de propagación definido para ese canal inalámbrico. En la capa física de cada nodo existe un umbral de potencia recibida definido por la sensibilidad del receptor para una tasa de bit determinada y por la SNR recibida. Cuando se recibe un paquete, si la

potencia de su señal en recepción está por debajo de este umbral, se asume que la recepción ha sido errónea y no se genera el paquete de confirmación de recepción ACK.

WiNS implementa dos tecnologías de acceso al medio: CSMA/CA y TDMA. Cada una de ellas se define en una clase que hereda de la clase **Medium_Acess** e implementa sus propias funcionalidades.

4.3.2.12. **Failure_Manager.java**

Esta clase se ocupa de la gestión de los paquetes de transmisión que no han alcanzado con éxito el destino. En función del tipo de paquete de que se trate se realizarán unas acciones determinadas o simplemente no se hará nada. Entre las acciones a iniciar por parte de esta clase se encuentran el reintento del envío del paquete, la llamada a métodos que borren de las tablas de encaminamiento el nodo siguiente al que no se puede transmitir de las rutas hacia destino o la generación de un paquete de error, entre otras.

4.3.2.13. **PostProcess.java**

Por último, una vez ejecutada la simulación, cuando ya no quedan eventos por procesar en la cola de eventos, ésta hace una llamada al método *startProcess()* de esta clase **PostProcess** para iniciar el procesado de toda la información relacionada con la simulación y la extracción de los datos necesarios con objeto de evaluar los protocolos simulados.

Para definir qué parámetros son necesarios a la hora de evaluar el rendimiento de un protocolo de encaminamiento se puede recurrir a al estándar RFC 2501 “*Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*” [58] que define para redes ad hoc las variables o **paráme-**

tros generales que han de tenerse en cuenta a la hora de evaluar las prestaciones del protocolo propuesto:

- *End-to-End Data Throughput and Delay*: rendimiento y retardo extremo a extremo, que están relacionados con la efectividad del protocolo en cuanto a las reglas de encaminamiento.
- *Route Acquisition Time*: definido para los protocolos *on-demand*, es el tiempo que tarda el protocolo en la adquisición de una ruta cuando es demandada y está estrechamente relacionado con el retardo extremo a extremo.
- *Percentage Out-of-Order Delivery*: es una medida de la conectividad total que proporciona el protocolo de encaminamiento.
- *Efficiency*: si la *efectividad* es la medida externa del rendimiento del protocolo, la *eficiencia* es la medida interna de esta efectividad. Para medir esta *eficiencia* es útil recurrir a otros parámetros más concretos, como son:
 - *Número medio de paquetes (o bits) de datos transmitidos / número medio de paquetes (o bits) de datos entregados con éxito al destino.*
 - *Número medio de saltos por paquete entregado con éxito.*
 - *Número medio de paquetes (o bits) de control transmitidos / número medio de paquetes (o bits) de datos entregados con éxito al destino:* es una medida de la sobrecarga de encaminamiento (*routing overhead*) de la red debida a la necesidad de utilizar paquetes de control para encontrar nuevas rutas hacia el destino.
 - *Número medio de paquetes de control y de datos transmitidos / número medio de paquetes de datos entregados con éxito al destino.*

En concreto, este simulador ofrece información completa relativa, entre otros, a los siguientes parámetros:

- Número de paquetes de datos transmitidos en la simulación (total y por nodo)
- Número de paquetes de control y tipo transmitidos (total y por nodo)
- Número y porcentaje de paquetes de datos que alcanzan el destino y los que se pierden o descartan
- Características de los paquetes enviados: número de saltos, ruta seguida, etc.
- Retardo extremo a extremo de los paquetes de datos entregados con éxito
- Camino recorrido y número de saltos de los paquetes de datos entregados
- Estado de cada enlace inalámbrico creado en la simulación (evolución temporal)
- Tamaño del buffer de recepción para cada nodo (evolución temporal)
- Tiempo de espera en el buffer de cada paquete recibido
- Tamaño de las tablas de rutas de cada nodo (evolución temporal)
- Movilidad de los nodos

y, específicamente para WSN:

- Energía total consumida por cada nodo y por la red
- Energía consumida en cada ronda por un nodo y por la red
- Número de CH seleccionados en cada ronda
- Número de nodos asociados a cada CH cada ronda
- Tasa de paquetes o de bits transmitidos en cada ronda
- Número de nodos muertos cada ronda

Para finalizar esta sección, en la Figura 4.5 se muestra el diagrama de diseño de la simulación del protocolo HARP en WiNS en el cual se incluyen todas las llamadas entre las instancias de las clases definidas en el simulador desde que se inicia la simulación hasta que se procesan los datos y se obtienen los resultados.

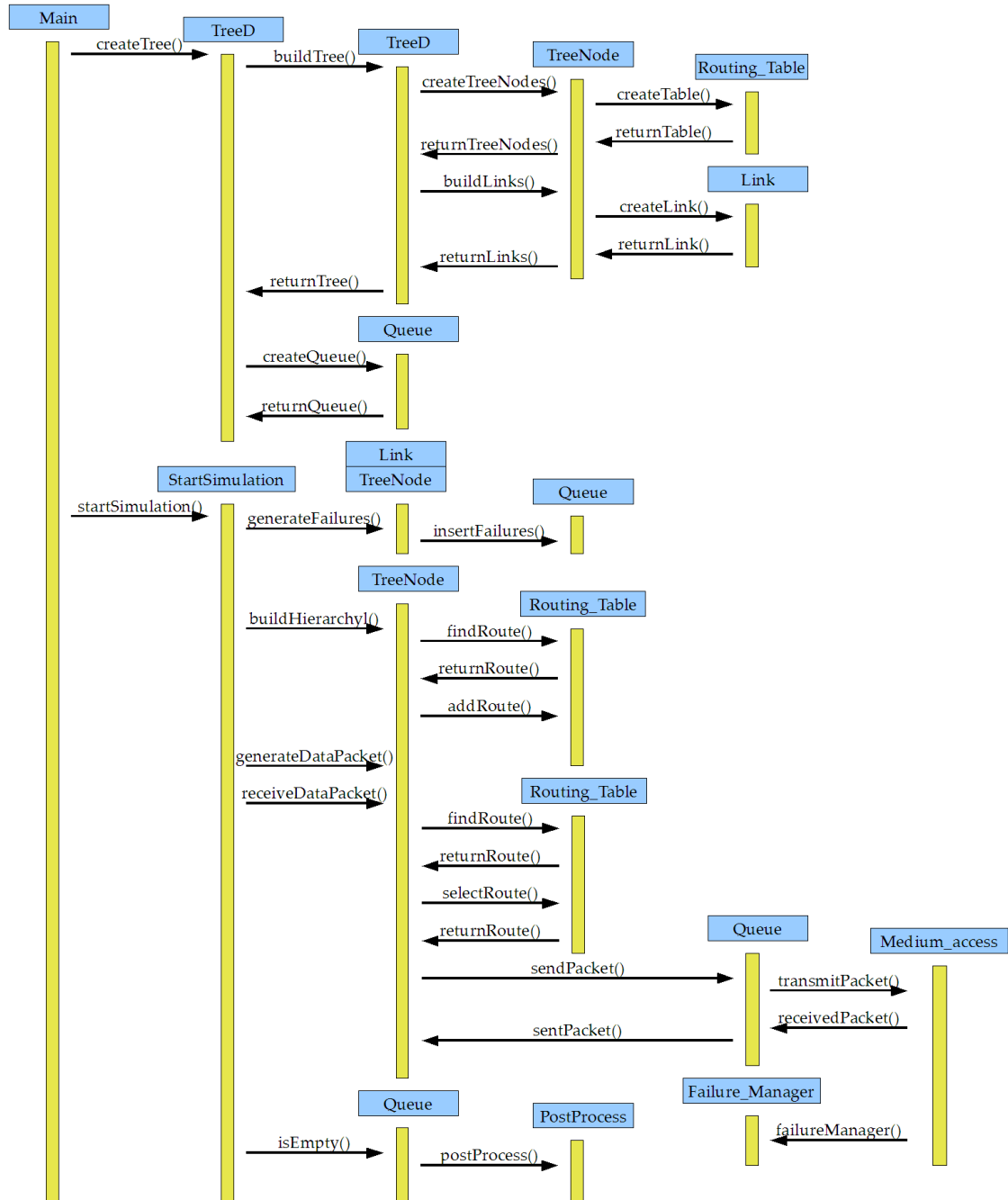


Figura 4.5: Diagrama de diseño de WiNS para la construcción del árbol jerárquico y simulación del protocolo HARP.

4.4. Descripción de los experimentos

El diseño de experimentos juega un papel fundamental a la hora de conseguir los resultados deseados para la caracterización de las propiedades del protocolo de encaminamiento propuesto en esta Tesis. Para lograr que los resultados obtenidos gocen de una representatividad adecuada, es necesaria la aplicación de una metodología, sobre todo en lo concerniente a la selección de parámetros [135]. Existe un área específica de la Estadística, el Diseño de Experimentos, que se centra en el desarrollo de estas metodologías [140], de tal forma que los resultados que se obtengan cumplan unas premisas básicas:

- Ser representativos del objeto a estudio.
- Ser presentados con base en métricas generalizables y objetivas.
- Ser fruto de la influencia de una única variable o de un conjunto separable de las mismas.

En este apartado se detalla la metodología seguida para la realización de las simulaciones incluidas en toda la presente Tesis, así como para presentar las variables particulares las mismas.

4.4.1. Características generales

En el diseño de los experimentos realizados en esta Tesis se definen las variables que deben observarse y el listado de parámetros obtenidos en cada una de las simulaciones que se llevan a cabo en WiNS. A partir de las mismas, se describen en detalle las simulaciones realizadas junto con los datos de entrada elegidos, lo que facilita la reproducción de los mismos y la posibilidad de comparación objetiva de los resultados obtenidos.

Toda simulación necesita unos parámetros o datos de entrada al sistema para poder ofrecer resultados, los cuales definirán un escenario particular de simulación. En las simulaciones de esta Tesis existe un conjunto de diferentes tipos de datos de entrada que son representativos de una red inalámbrica y que, en consecuencia, es necesario definir previamente a la ejecución de cada simulación. Estos **parámetros principales de entrada al sistema** son los siguientes:

- Coordenadas (x,y,z) de todos los nodos de la red que se pretende simular, las cuales definen el área y la topología de la misma (triangular, rectangular, aleatoria, etc.).
- Configuración de los nodos de la red: parámetros asociados a cada nodo, como son el tipo (sensor o sumidero), su energía inicial, los protocolos utilizados y sus parámetros.
- Patrón de tráfico que se va a cursar en la red, especificando para cada paquete de datos generado su origen, destino y tiempo de generación, entre otros parámetros.
- Patrón de fallos en la red o de movimiento de los nodos, caracterizados por un probabilidad de error de enlace.

Los resultados obtenidos en cada simulación son almacenados en un fichero de traza en formato texto que proporciona información completa acerca de los parámetros descritos en la Sección 4.3.2.13.

4.4.2. Características específicas

Cada simulación incluida en la presente Tesis será definida según una estructura de información común que engloba las siguientes categorías:

- **Objetivos.** Se detallarán los objetivos perseguidos en cada experimento, como elemento de partida para su configuración particular.
- **Variabes analizadas.** Los objetivos descritos anteriormente serán estudiados con base en una serie de variables representativas de las prestaciones a medir como el tiempo de vida de las rutas o la energía consumida.
- **Parámetros.** Conjunto de valores de configuración de las características específicas del experimento. Estos parámetros incluyen datos básicos sobre la red como topología o número de nodos, sobre los nodos como situación geográfica o energía inicial, y sobre las propias simulaciones como el número de repeticiones ejecutadas.

Estas categorías serán especificadas de forma particular en la descripción de cada experimento realizado a lo largo de esta Tesis aportando claridad y permitiendo la reproducción de los resultados obtenidos.

Capítulo 5

Análisis teórico de prestaciones

El diseño de un protocolo de encaminamiento eficiente es un desafío en redes ad hoc inalámbricas, caracterizadas por la movilidad de los nodos, la existencia de comunicaciones multi-salto y la ausencia de elementos específicos en la infraestructura de red encargados de dicha tarea [98], [146].

Los protocolos de encaminamiento juegan un papel muy importante en las comunicaciones de la red y su rendimiento puede medirse a través de parámetros tales como eficiencia y retardo extremo a extremo descritos en la Sección 4.3.2.13. Los protocolos reactivos (*on demand*) son un subconjunto dentro del encaminamiento en MANET, de amplio uso debido a la baja sobrecarga de paquetes que introducen en la red, ya que una ruta es utilizada únicamente cuando es necesario transmitir a través de ella. HARP pertenece a este tipo de protocolos. El carácter reactivo de HARP se extiende al proceso de descubrimiento de rutas, que se desencadena cuando se detectan fallos en las existentes o no se conoce ninguna válida hacia el destino. En general, estos procesos de descubrimiento de una nueva ruta utilizan la difusión de paquetes de control o la inundación de la red o *flooding* para proporcionar una nueva ruta hacia el nodo destino, lo que consume una parte sustancial del ancho de banda disponible en la red y aumenta el consumo energético.

co de los nodos y la sobrecarga de paquetes en la misma. Diferentes técnicas de difusión, como la denominada *probabilistic broadcast* en la que el reenvío de un paquete está condicionado por una cierta probabilidad, han sido desarrolladas para reducir el impacto de esta inundación de la red [20], [131], [33], [141], [75]), aunque cualquiera de estas técnicas hace uso de paquetes de control para establecer nuevos caminos hacia el destino. En consecuencia, para disminuir la sobrecarga de la red es necesario limitar el uso que se hace de estos procesos de descubrimiento utilizados en la fase de mantenimiento de rutas del protocolo y que inundan la red con mensajes de control. El uso frecuente de estos mecanismos de inundación no es una buena solución para el mantenimiento dinámico de las rutas en el sistema [44].

Con el objetivo de limitar el número de nuevos procesos de descubrimiento de rutas y hacer más eficiente el mantenimiento de las mismas en la red, los protocolos multi-ruta ofrecen mejores prestaciones que los que únicamente almacenan una ruta hacia el destino (*single-path*), tal y como se ha indicado en la Sección 3.1. Esta mejora del rendimiento se debe principalmente al aumento que estos protocolos proporcionan en cuanto al tiempo máximo en el que alguna de las rutas hacia el destino es válida, denominado *tiempo de vida* de las rutas, haciendo disminuir el número de nuevos procesos de descubrimiento y, en consecuencia, la sobrecarga de la red.

Para evaluar las prestaciones de HARP se tomará como referencia el tiempo de vida de las rutas y se analizará estadísticamente este parámetro para compararlo con el proporcionado por otras estructuras de rutas como *single-path*. Este estudio, que se incluye en la Sección 5.1, está enfocado en dos direcciones: por un lado, proporcionar una medida objetiva del tiempo de vida de las rutas dentro de la estructura multi-camino implementada por HARP y, por otro, analizar la influencia que tiene el proceso de recuperación local (RL) de rutas del protocolo propuesto en el aumento del tiempo de vida de las mismas. Para finalizar este capítulo, en la Sección 5.2 se incluyen diferentes experimentos utilizando el simulador WiNS

propuesto, los cuales, por un lado, validan los resultados teóricos obtenidos y el funcionamiento del propio simulador y, por el otro, muestran las prestaciones de HARP respecto de otras filosofías basadas en *single-path* y multi-camino y su mejora mediante la integración de mecanismos de RL.

5.1. Marco teórico

5.1.1. Introducción

Para analizar estadísticamente el tiempo de vida de las rutas entre dos nodos es necesario establecer alguna clasificación acerca de la estructura de rutas proporcionada por el protocolo de encaminamiento. Para ello se asumirá como definición de ruta disjunta la especificada en la Sección 3.1 para las rutas *node-disjoint*, es decir, que dos rutas se considerarán disjuntas si no tienen nodos ni enlaces en común. Por tanto, se puede establecer la siguiente **clasificación en cuanto a la estructura de las rutas** entre dos nodos proporcionadas por un protocolo:

- *Single-Path Routing* (o *Unipath Routing*): el nodo origen almacena una única ruta hacia el destino, generalmente la más corta o la que menos saltos requiere.
- *Multi-Path Routing*: para cada destino pueden conocerse y almacenarse varias rutas alternativas. Dentro de esta opción, existen dos alternativas:
 - *Multi-Path Routing con rutas no disjuntas*: posee una ruta principal y otras rutas que no son disjuntas con respecto a la principal, es decir, el nodo origen puede conocer varios caminos alternativos a la ruta principal pero todos ellos contienen al menos un nodo en común con ella.

- *Multi-Path Routing con rutas disjuntas*: todas las rutas almacenadas para un destino particular son disjuntas entre sí. En consecuencia, existen diferentes caminos independientes hacia un destino.
- *Multi-Path Routing con rutas disjuntas y no disjuntas*: además de rutas disjuntas entre sí, pueden existir otras rutas alternativas que compartan algún nodo o enlace en común con otra. Esta configuración más general es la que aporta un mayor grado de redundancia en la red y es utilizada por HARP a la hora de proponer la estructura de caminos.

La razón de establecer esta clasificación se basa en la idea de que si un nodo conoce diferentes rutas hacia un destino de la red, será más probable que alguna de ellas siga estando activa en el momento de iniciar la transmisión que en el caso de que solamente se conozca un único camino hacia el nodo destino. Es más, al disponer de varias rutas alternativas puede reducirse la sobrecarga en algunos nodos o enlaces evitando que se formen cuellos de botella y posibilitando el balanceo de las transmisiones de paquetes entre todas las rutas disponibles.

En las siguientes secciones se describe el marco teórico desarrollado en esta Tesis para analizar las diferencias entre los distintos tipos de protocolos de encaminamiento según la clasificación anterior. No se incluye en este estudio el caso de *Multi-Path Routing con rutas no disjuntas* por ser un caso intermedio entre los dos primeros y no aportar datos relevantes en cuanto al análisis de las prestaciones del protocolo propuesto.

Tal y como se ha explicado al inicio de este capítulo, uno de los principales parámetros utilizados por la comunidad científica para analizar el rendimiento de un protocolo de encaminamiento es la sobrecarga introducida en la red, la cual está relacionada con el número medio de paquetes de control transmitidos con respecto al número medio de paquetes de datos entregados con éxito al destino. Sabiendo que dos de las fuentes principales de generación de paquetes de control

son los procesos de descubrimiento y mantenimiento de rutas, puede inferirse que cuanto menor sea el número de estos procesos, menor será también el número de paquetes de control que se propagan por la red y, en consecuencia, menor será la sobrecarga en la misma. Dicho esto, si se determina de forma estadística la frecuencia relativa del *descubrimiento de rutas* para cada uno de los tipos de protocolos descritos en el apartado anterior estudiando el tiempo de vida de las rutas en cada uno de ellos, será posible comparar las respectivas sobrecargas producidas y, por tanto, extraer conclusiones teóricas relativas a la eficiencia de HARP comparado con otras propuestas de encaminamiento.

En la sección siguiente se desarrolla este estudio teórico para diferentes configuraciones de red, comenzando por la que propone una única ruta hacia el destino y finalizando con el análisis de la configuración creada por HARP para el encaminamiento en la red. Dicha formulación parte de una estructura de red conocida en cuyas expresiones se suponen conocidos a priori el número y la configuración de rutas disjuntas de la red. Más tarde se propone un algoritmo para cálculo de la probabilidad de encontrar un número de rutas alternativas disjuntas desde cada nodo de la ruta principal hasta el nodo destino, proporcionando una formulación general para la expresión de la *cdf* del tiempo de vida de las rutas entre dos nodos de la red.

5.1.2. Estudio del tiempo de vida de las rutas

5.1.2.1. Encaminamiento *single-path*

Dados una fuente, S, y un destino, D, un paquete que se transmite de S a D mediante un protocolo de encaminamiento que siga una filosofía de este tipo recorrerá el camino mostrado en la Figura 5.1.

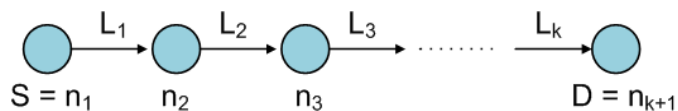


Figura 5.1: Estructura de nodos y enlaces de un protocolo con Single-Path Routing

Este camino entre dos puntos, denotado como ruta P , estará formado tanto por los nodos intermedios como por los enlaces inalámbricos que unen unos con otros. En consecuencia, una ruta P entre un origen y un destino se dice que se rompe, o que no está operativa cuando se produzcan un fallo en uno o más nodos o roturas de uno o más enlaces inalámbricos.

Por lo general, un nodo puede fallar a lo largo de una ruta debido a tres causas principales:

- *Agotamiento de la batería.*
- *Mal funcionamiento del nodo.*
- *Apagado temporal* (generalmente con el objetivo de ahorrar energía).

A su vez, se dice que un enlace inalámbrico se rompe cuando se produce alguna de las siguientes circunstancias:

- *Falla al menos uno de los nodos que lo forman.*
- *Se producen cambios en el entorno de los nodos que lo forman:* aparición de obstáculos, cambios importantes en las condiciones de transmisión, etc.
- *Movimiento de alguno de los nodos a otra posición alejada en la red.*

En adelante, a la hora de tener en cuenta los fallos que se producen en la red, se considerarán únicamente los de enlaces ya que, funcionalmente hablando, el fallo de un nodo es equivalente al fallo de todos los enlaces inalámbricos que lo

conectan con otros nodos. De esta forma es posible tratar todos los fallos de la red como fallos de enlaces y se simplifica el modelado del comportamiento de la red inalámbrica.

De acuerdo con la Figura 5.1, una ruta P que se inicie en el nodo S y termine en D estará compuesta por una secuencia de k enlaces inalámbricos y por $k - 1$ nodos intermedios. Se denota por L_i el enlace i -ésimo de la ruta P y se representa **el tiempo de vida de un enlace inalámbrico entre dos nodos** mediante una variable aleatoria. En consecuencia, el tiempo de vida del enlace L_i vendrá representado por la variable aleatoria X_{L_i} .

Si se asume que todos los enlaces inalámbricos tienen estadísticamente las mismas propiedades y que son independientes entre sí¹, es decir, que lo que ocurre en uno de ellos no afecta al comportamiento del resto, puede asumirse que X_{L_i} con $i = 1, 2, \dots, k$ son variables aleatorias independientes e idénticamente distribuidas (iid), cada una de ellas con media [98]:

$$E[X_{L_i}] = l \quad (5.1)$$

Partiendo de que una ruta completa se rompe cuando lo hace alguno de los enlaces inalámbricos que la forman², el tiempo de vida de la ruta P de la Figura 5.1 es una variable aleatoria que se denota como X_p y que puede expresarse de la siguiente forma:

$$X_p = \min(X_{L_1}, X_{L_2}, \dots, X_{L_k}) \quad (5.2)$$

¹Aunque esta asunción no sea del todo cierta debido a fenómenos de correlación espacial, se ha demostrado en [98] que los resultados obtenidos con estas premisas no difieren en gran medida con los resultados exactos obtenidos mediante experimentos o simulaciones

²Ya se ha indicado anteriormente la relación entre el fallo de un nodo y la rotura de los enlaces inalámbricos que lo unen a otros nodos

En [98], [32], [81] y [113] se pone de manifiesto que la distribución X_p resultante es una variable aleatoria exponencial con media:

$$E[X_P] = \frac{l}{k} \quad (5.3)$$

Es decir, para una ruta P la función densidad de probabilidad (*pdf, probability density function*) puede expresarse de la siguiente forma:

$$f_{X_P}(t) = \begin{cases} \lambda e^{-\lambda t} & , \text{ si } t \geq 0 \\ 0 & , \text{ si } t < 0 \end{cases} \quad (5.4)$$

Integrando la función anterior es posible calcular la función de distribución acumulativa (*cdf, cumulative distribution function*) de una variable aleatoria exponencial:

$$F_{X_P}(t) = P(X_P \leq t) = \int_{-\infty}^t f_{X_P}(x) dx = \begin{cases} 1 - e^{-\lambda t} & , \text{ si } t \geq 0 \\ 0 & , \text{ si } t < 0 \end{cases} \quad (5.5)$$

donde el valor medio y el valor del parámetro λ se expresan en la ecuación siguiente:

$$E[X_P] = \frac{l}{k} = \frac{1}{\lambda} \quad (5.6)$$

En la Figura 5.2 a) se muestra el resultado de una simulación en Matlab que incluye la *pdf* del tiempo de vida de una ruta cuando varía el número de enlaces que la forman. Puede observarse que a mayor número de enlaces, mayor es la probabilidad de que la ruta se rompa (conclusión similar a la obtenida teóricamente y mediante simulaciones en otros trabajos de referencia como [132]) y, en consecuencia, mayor es también la probabilidad de que el tiempo de vida de la ruta sea más pequeño. El tiempo medio de duración de las rutas para cada valor de k

se incluye en la Figura 5.3 a), donde queda evidenciada la disminución de dicho tiempo de vida con el aumento del número de saltos.

5.1.2.2. Encaminamiento multi-camino: N rutas disjuntas en paralelo

Dados una fuente S y un destino D , se considera en este apartado el caso en el que existan N rutas disjuntas P_1, P_2, \dots, P_N que parten de S hasta alcanzar el nodo D , tal y como se muestra en la Figura 5.4.

En el desarrollo de esta sección se asume que las variables aleatorias que modelan los tiempos de vida de las diferentes rutas son estadísticamente independientes entre sí. Al final de la misma se incluye una explicación sin tener en cuenta esta premisa asumiendo que dichas variables aleatorias no son independientes.

A cada una de las rutas P_i disjuntas se le asocia un tiempo de vida X_{P_i} que será función del tiempo de vida de los distintos enlaces inalámbricos que la forman, tal y como se ha indicado en la sección anterior. Si se tiene en cuenta que todos los enlaces son iid y que las distintas rutas disjuntas no contendrán ningún enlace en común, es consecuente asumir que el tiempo de vida de cada ruta individual es independiente del tiempo de vida del resto.

Se define el *tiempo de vida del sistema completo* y se denota por T como el intervalo de tiempo en el que existe al menos una ruta válida desde la fuente S al destino D . Para calcularlo es necesario obtener la duración de la última ruta válida o, lo que es lo mismo, el tiempo transcurrido hasta que ninguna de las rutas P_1, P_2, \dots, P_N sea útil, es decir, todas ellas son fallidas. En consecuencia, se puede expresar T como el máximo de los tiempos de vida de cada una de las rutas alternativas entre los dos nodos, es decir:

$$T = \max(X_{P_1}, X_{P_2}, \dots, X_{P_N}) \quad (5.7)$$

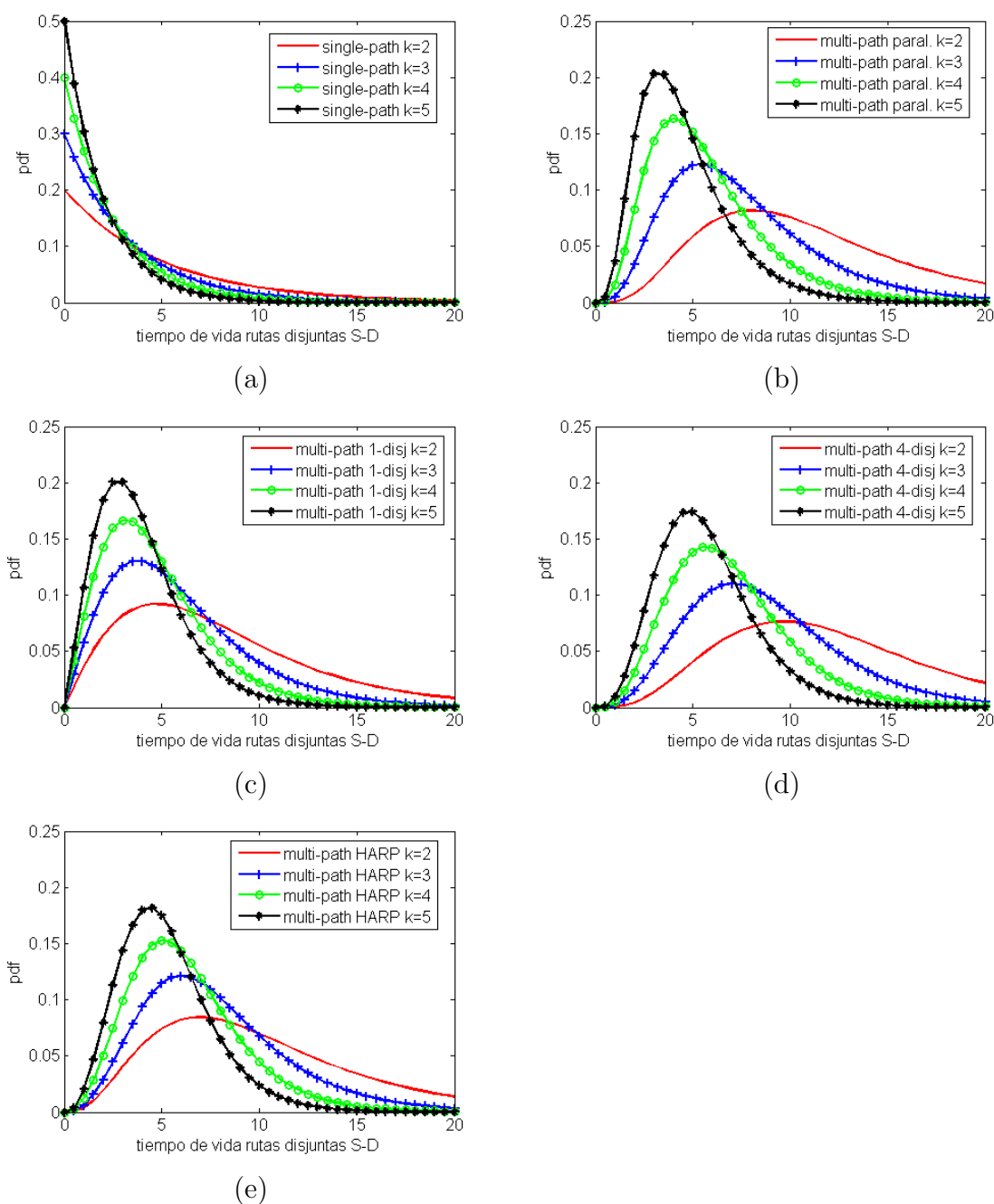
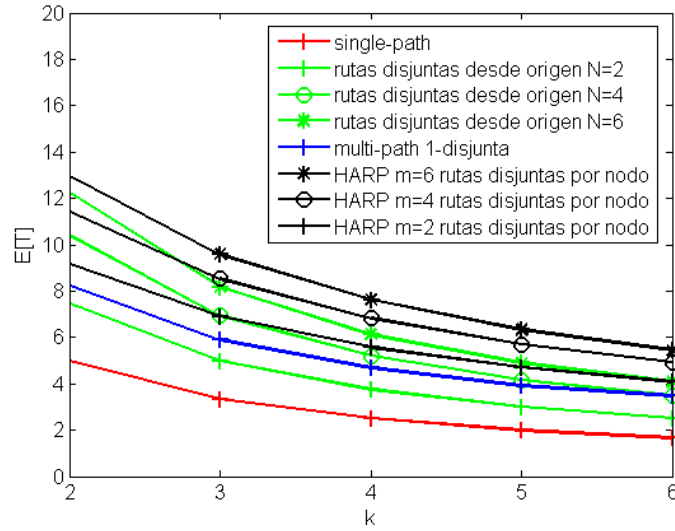
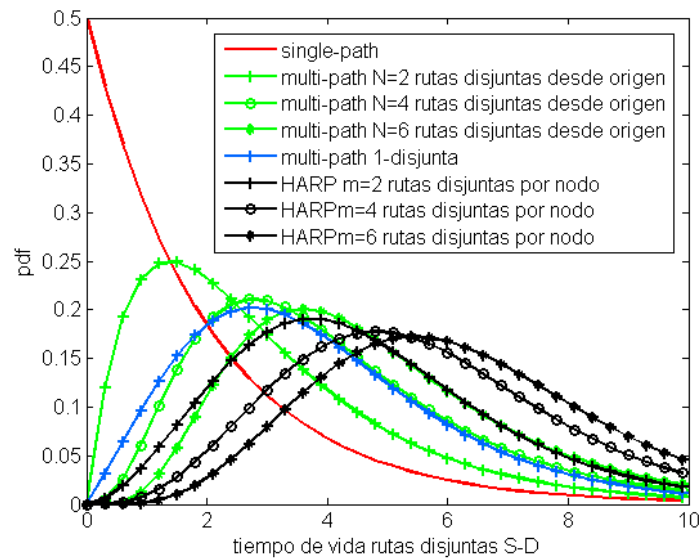


Figura 5.2: Pdf del tiempo de vida de las rutas correspondiente a las diferentes configuraciones de red en función del número de saltos hasta el destino (k) para $l = 10$ s: a) Single-path; b) Multi-camino con $N = 4$ rutas disjuntas desde el origen; c) Multi-camino con 1 ruta disjunta desde cada nodo de la ruta principal; d) Multi-camino con $m = 4$ rutas disjuntas por nodo de la ruta principal; e) HARP con $m = 4$ rutas disjuntas por nodo de la ruta principal excepto el último.



(a)



(b)

Figura 5.3: Comparación entre las diferentes configuraciones de rutas analizadas: a) tiempo medio de vida de las rutas S-D para las distintas configuraciones estudiadas en función de la longitud de la ruta principal (k) para $l = 10$ s; b) Pdf de cada configuración para $k = 5$ saltos.

En la Figura 5.5 se muestra un ejemplo para el caso $N = 4$ rutas alternativas disjuntas. En esta situación, existirá un camino de comunicación entre S y D y el tiempo de vida T del sistema vendrá determinado por el intervalo temporal

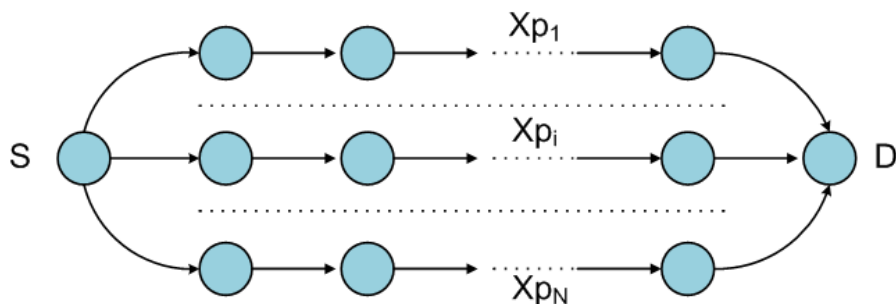


Figura 5.4: Encaminamiento multi-camino con N rutas alternativas disjuntas.

correspondiente a la ruta más longeva, que en este ejemplo es la ruta 3. A partir de este instante, el origen ya no conoce ninguna ruta válida hacia el destino y debe iniciar un nuevo proceso de descubrimiento de ruta es caso de pretender utilizarla.

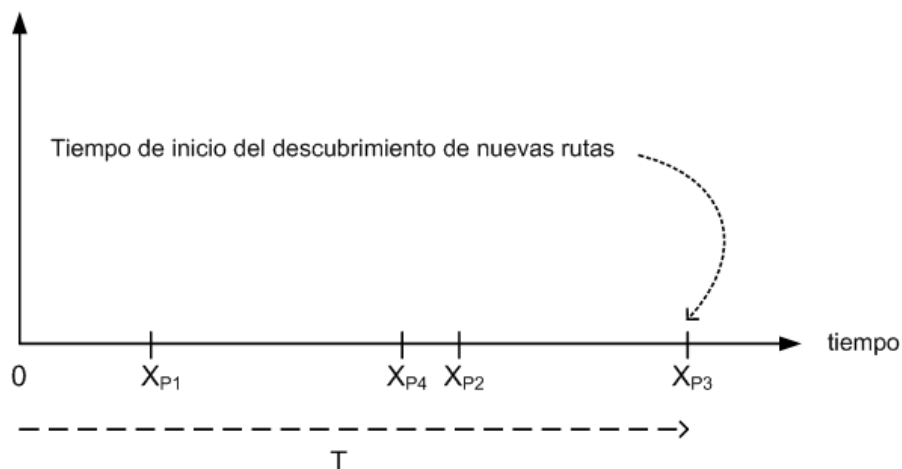


Figura 5.5: Ejemplo de tiempo de vida de un sistema con 4 rutas disjuntas en función de las variables aleatorias que modelan el tiempo de vida de cada una de ellas.

Por tanto, se puede afirmar que T representa el tiempo transcurrido entre sucesivos descubrimientos de ruta entre S y D, siempre que se asuma que el retardo extremo a extremo de un paquete en la red es muy pequeño comparado con el tiempo entre cambios en las rutas³. Partiendo de la Ecuación (5.7) y teniendo

³Tal y como se explica en [98], esta premisa es razonable y permite ignorar el tiempo de descubrimiento de nuevas rutas o de propagación de paquetes de error en la red con respecto al tiempo de duración de las rutas

en cuenta que las variables aleatorias X_{P_i} son independientes entre sí, se puede calcular la *cdf* de T de la siguiente forma:

$$\begin{aligned}
 F_T(t) &= P(T \leq t) = P[\max(X_{P_1}, X_{P_2}, \dots, X_{P_N}) \leq t] = \\
 &= P[(X_{P_1} \leq t) \cap (X_{P_2} \leq t) \cap \dots \cap (X_{P_N} \leq t)] = \\
 &= \prod_{i=1}^N F_{X_{P_i}}(t)
 \end{aligned} \tag{5.8}$$

siendo $F_{X_{P_i}}(t) = 1 - e^{-\lambda_i t}$, con $t \geq 0$.

Derivando la Ecuación (5.8) con respecto a la variable t se obtiene la *pdf* del tiempo de vida del sistema que, según la Ecuación (5.5), puede expresarse de la siguiente forma:

$$\begin{aligned}
 f_T(t) &= \lambda_1 e^{-\lambda_1 t} \cdot (1 - e^{-\lambda_2 t}) \cdot (1 - e^{-\lambda_3 t}) \cdot \dots \cdot (1 - e^{-\lambda_N t}) + \\
 &+ \lambda_2 e^{-\lambda_2 t} \cdot (1 - e^{-\lambda_1 t}) \cdot (1 - e^{-\lambda_3 t}) \cdot \dots \cdot (1 - e^{-\lambda_N t}) + \dots \\
 &+ \lambda_N e^{-\lambda_N t} \cdot (1 - e^{-\lambda_1 t}) \cdot (1 - e^{-\lambda_2 t}) \cdot \dots \cdot (1 - e^{-\lambda_{N-1} t})
 \end{aligned} \tag{5.9}$$

Por último, agrupando los términos anteriores se llega a la siguiente ecuación:

$$f_T(t) = \sum_{i=1}^N \lambda_i e^{-\lambda_i t} \cdot \prod_{\substack{j=1 \\ j \neq i}}^N (1 - \lambda_j e^{-\lambda_j t}) \tag{5.10}$$

con $\lambda_i = k_i/l$, denotando el inverso del tiempo de vida medio de la ruta i -ésima.

Nótese que no existe ninguna restricción en cuanto al número de enlaces de las rutas P_i , es decir, que pueden contener un número distinto al del resto. En el caso particular de que todas ellas contengan el mismo número de enlaces, el parámetro λ_i con $i = 1, 2, \dots, N$ será igual para todas las rutas. Así, denotando λ como el tiempo de vida medio de una ruta, la expresión de la *pdf* del tiempo de vida del

sistema resultaría de la siguiente forma:

$$f_T(t) = N \cdot \lambda e^{-\lambda t} \cdot (1 - \lambda e^{-\lambda t})^{N-1} \quad (5.11)$$

La Figura 5.2 b) muestra cómo varía la distribución de probabilidad de esta configuración para $N = 4$ rutas con enlaces iid y con el mismo número de saltos (mismo λ) cuando varía el número de saltos desde el nodo origen S hasta el nodo destino D. Comparándola con el caso *single-path*, en esta configuración existe mayor probabilidad de que el tiempo de vida de las rutas sea mayor para cualquier longitud de las mismas. Esta afirmación es coherente con el resultado de las simulaciones teóricas de la Figura 5.3 a), en donde se observa que esta configuración ofrece mejores rendimientos que la *single-path*, que puede considerarse un caso particular de esta configuración con $N = 1$.

Para finalizar esta sección se menciona el caso de que se asuma correlación entre las variables aleatorias que modelan los tiempos de vida de las diferentes rutas. Este caso, en el que las Ecuaciones (5.8) y (5.10) dejan de tener validez, ha sido estudiado con anterioridad [19]. El método desarrollado [112] consiste en la búsqueda de un límite superior para el valor medio del tiempo de vida del sistema $E[T]$ partiendo de la desigualdad de Jensen. Como resultado de aplicar la no independencia de dichas variables aleatorias se llega a la siguiente expresión, que sustituye a la Ecuación (5.11), para calcular el límite superior del tiempo medio de vida del sistema asumiendo que existen N rutas alternativas con variables aleatorias iid con distribución exponencial de media $1/\lambda$:

$$E[T] \leq \frac{1}{\lambda} (\ln N + 1) \quad (5.12)$$

Este límite superior incluido en la Ecuación 5.12 es también válido cuando los tiempos de vida de las rutas individuales de un sistema multi-camino son variables aleatorias independientes.

5.1.2.3. Encaminamiento multi-camino: K rutas alternativas disjuntas

Como en el caso anterior, para una fuente, S , y un destino, D , determinados, se supone la existencia de una ruta principal formada por los enlaces $\langle L_1 \rightarrow L_2 \rightarrow \dots \rightarrow L_k \rangle$. Se estudiará en esta sección el caso de que cada nodo de la ruta principal n_i (incluida la fuente S) conozca una ruta alternativa P_i disjunta tanto de la ruta principal como de las demás rutas alternativas, tal y como se muestra en la Figura 5.6. Por este motivo, esta configuración se denomina en esta Tesis *multi-camino 1-disjunta*. Puede observarse fácilmente que el número total de rutas disjuntas de la ruta principal es k .

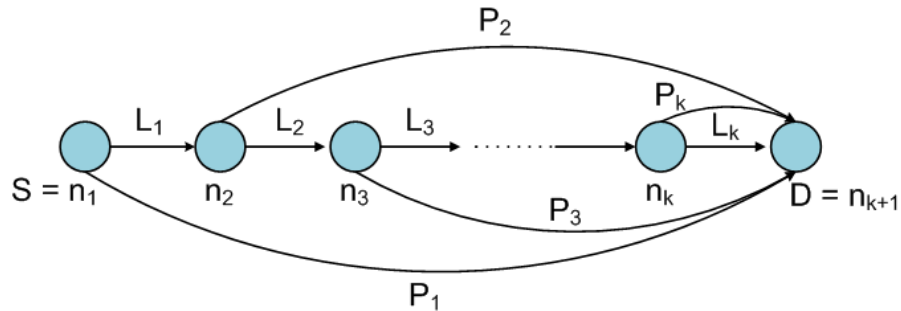


Figura 5.6: Encaminamiento multi-camino con k rutas alternativas disjuntas, una desde cada nodo de la ruta principal.

Para analizar las prestaciones de este y otros sistemas pueden aplicarse diversos métodos de evaluación de la fiabilidad de sistemas (denominada *network reliability*) aplicando teoría de grafos, como son el método de inclusión-exclusión, los productos disjuntos, la factorización o reducciones serie-paralelo, entre otros. Estos métodos se describen con detalle en [37], [122] y [109]. Otro de estos métodos consiste calcular el conjunto mínimo de *cutsets* que hacen que la red quede

desconectada desde S hasta D si alguno de ellos se produce. Un (S, T) -*cutset* o (S, T) -*cut* es un conjunto de enlaces inalámbricos cuya rotura simultánea acarrea la desconexión del grafo desde S hasta D, con la pérdida de las rutas válidas entre los dos nodos.

Aplicando el método de *cutsets* mínimos a la Figura 5.6 es posible calcular una expresión para la *pdf* y la *cdf* de esta configuración de red. Para ello se define el siguiente evento:

$$V = \{\text{no es posible encontrar una ruta válida desde S hasta D}\}$$

El tiempo de vida del sistema S-D completo se relaciona con el evento V de la siguiente forma:

$$T = \{\text{tiempo en el que existe al menos una ruta válida desde S hasta D}\} = \{\text{tiempo transcurrido hasta que se cumple el evento V}\}$$

El evento V se cumple cuando se produce al menos uno de los *cutsets* CS del conjunto de *cutsets* mínimo denotado por MCT . Por tanto:

$$MCT = \{CS_1, CS_2, \dots, CS_s\}$$

$$V = \bigcup_{i=1}^s CS_i \quad | \quad CS_i \in MCT$$

En [98], para calcular el conjunto de *cutsets* mínimo se recorren uno a uno todos los enlaces inalámbricos de la ruta principal comenzando por L_1 e identificando para cada uno de ellos los enlaces necesarios para producir un *cutset* cuando alguno de los enlaces principales se rompe. Por tanto, denotando como \overline{L}_i el evento ocurrido cuando el enlace L_i se rompe, y como \overline{P}_j el evento ocurrido cuando la ruta alternativa P_j deja de estar activa, se puede llegar a la siguiente expresión:

$$V = (\overline{L}_1 \cap \overline{P}_1) \cup (\overline{L}_2 \cap (\overline{L}_1 \cup \overline{P}_2) \cap \overline{P}_1) \cup (\overline{L}_3 \cap (\overline{L}_1 \cup (\overline{L}_2 \cup \overline{P}_3) \cap \overline{P}_2) \cap \overline{P}_1) \cup \dots \quad (5.13)$$

En la estructura de red mostrada en la Figura 5.6 se puede observar que ninguna de las rutas alternativas tiene enlaces en común entre sí ni con la ruta principal, por lo que en estas condiciones puede asumirse que existe independencia en el comportamiento de todos ellos, resultando la Ecuación (5.13) de la siguiente forma:

$$V = (\overline{L_1 P_1}) + (\overline{L_2(L_1 + P_2)P_1}) + (\overline{L_3(L_1 + (L_2 + P_3)P_2)P_1}) + \dots \quad (5.14)$$

La Ecuación 5.14 anterior puede utilizarse para calcular el tiempo de vida de las rutas desde S hasta D, que es equivalente al transcurrido hasta el siguiente descubrimiento de ruta o, lo que es lo mismo, el tiempo que transcurre hasta que el evento V es cierto. Por ejemplo, el segundo término representa la siguiente secuencia de eventos:

- L_2 se rompe en la ruta principal, obligando a S a utilizar la ruta alternativa $L_1 - P_2$.
- La ruta alternativa se rompe si lo hacen L_1 o P_2 , obligando a S a utilizar la ruta alternativa P_1 .
- La ruta falla cuando P_1 se rompe, obligando a S a iniciar un nuevo proceso de descubrimiento de rutas.

Simplificando y agrupando términos se llega a una expresión bastante compacta y fácil de generalizar, como es:

$$V = \overline{L_1 P_1} + \overline{L_2 P_2 P_1} + \overline{L_3 P_3 P_2 P_1} + \dots \quad (5.15)$$

En consecuencia, el tiempo de vida del sistema S-D será el tiempo en el que se produzca el primero de los *cutsets* que componen V , es decir:

$$T = \min[\max(X_{L_1}, X_{P_1}), \max(X_{L_2}, X_{P_2}, X_{P_1}), \dots, \dots, \max(X_{L_k}, X_{P_k}, X_{P_{k-1}}, \dots, X_{P_1})] \quad (5.16)$$

Denotando la variable aleatoria $\max(X_{L_i}, X_{P_i}, X_{P_{i-1}}, \dots, X_{P_1})$ por Z_i la ecuación anterior quedaría:

$$T = \min [Z_1, Z_2, \dots, Z_k] \quad (5.17)$$

De la misma forma que en la Ecuación (5.8), para N rutas disjuntas independientes, asumiendo que X_{L_i}, X_{P_i} son estadísticamente independientes entre sí, se puede llegar a una expresión para la *cdf* de este sistema de la siguiente forma:

$$\begin{aligned} F_{Z_i}(t) &= P(Z_i \leq t) = P(\max(X_{L_i}, X_{P_i}, X_{P_{i-1}}, \dots, X_{P_1}) \leq t) = \\ &= P[(X_{L_i} \leq t) \cap (X_{P_i} \leq t) \cap (X_{P_{i-1}} \leq t) \cap \dots \cap (X_{P_1} \leq t)] = \\ &= P(X_{L_i} \leq t) \cdot P(X_{P_i} \leq t) \cdot P(X_{P_{i-1}} \leq t) \cdots P(X_{P_1} \leq t) = \\ &= F_{L_i}(t) \cdot \prod_{j=1}^i F_{X_{P_j}}(t) \end{aligned} \quad (5.18)$$

donde $f_{Z_i}(t) = \frac{dF_{Z_i}(t)}{dt}$.

Si se asume que las variables aleatorias $Z_i(t)$ con $i = 1, 2, \dots, k$ son estadísticamente independientes (aunque esto no es estrictamente cierto, los resultados obtenidos no difieren en gran medida del caso exacto), se puede calcular la *pdf* del tiempo de vida del sistema partiendo de la siguiente ecuación:

$$F_T(t) = \lim_{dt \rightarrow 0} \frac{P(t \leq T \leq t + dt)}{dt} \quad (5.19)$$

El término de probabilidad de la ecuación anterior se calcula de la siguiente forma:

$$\begin{aligned}
P(t \leq T \leq t + dt) &= P[t \leq \min(Z_1, Z_2, \dots, Z_k) \leq t + dt] = \\
&= \sum_{i=1}^k P[t \leq Z_i \leq t + dt \mid Z_i = \min(Z_1, Z_2, \dots, Z_k)] \cdot P[Z_i = \min(Z_1, Z_2, \dots, Z_k)] = \\
&= \sum_{i=1}^k P[t \leq Z_i \leq t + dt] \cdot P[Z_i = \min(Z_1, Z_2, \dots, Z_k)]
\end{aligned} \tag{5.20}$$

Dividiendo ambos lados de la igualdad por dt y tomando el límite cuando dt tiende a cero, se llega a la expresión deseada de la *pdf* del tiempo de vida del sistema S-D:

$$\begin{aligned}
f_T(t) &= \sum_{i=1}^k f_{Z_i}(t) \cdot \prod_{\substack{j=1 \\ j \neq i}}^k P[Z_j > Z_i] = \sum_{i=1}^k f_{Z_i}(t) \cdot \prod_{\substack{j=1 \\ j \neq i}}^k P[Z_j > t] = \\
&= \sum_{i=1}^k f_{Z_i}(t) \cdot \prod_{\substack{j=1 \\ j \neq i}}^k (1 - F_{Z_j}(t))
\end{aligned} \tag{5.21}$$

Finalmente, para conocer cómo varía la *pdf* indicada en la Ecuación (5.21) en función del número de saltos hasta el destino, basta con fijarse en la Figura 5.2 c), en la cual puede comprobarse que para cualquier valor de k existe mayor probabilidad de que el tiempo de vida de las rutas sea mayor que en el caso *single-path*. En cuanto a la comparación con el caso anterior de N rutas disjuntas desde el origen, la eficacia de esta configuración dependerá del valor de N , tal y como se muestra en la Figura 5.3 a). En ella se puede observar que el tiempo de vida medio de las rutas para la configuración 1-disjunta es mayor que el proporcionado por las rutas paralelas con $N = 2$, ya que proporciona un mayor número de rutas alternativas, siendo menor para el caso de que N aumente.

Para completar esta comparación se muestra la Figura 5.7, donde se ha establecido el mismo número total de rutas alternativas hacia el destino en cada una de las dos configuraciones con objeto de estudiar cuál de ellas es más eficiente. Para ello, dado que la configuración multi-camino 1-disjunta analizada en este apartado presenta tantas rutas disjuntas como saltos hasta el destino (k) además de una ruta principal, se ha simulado la configuración anterior con $k + 1$ rutas disjuntas paralelas que parten desde S hasta D. El resultado de dichas simulaciones muestra que para cada valor de k la configuración de rutas disjuntas desde el origen consigue proporcionar una mayor probabilidad de alargar el tiempo de vida de las rutas con respecto a la estructura multi-camino 1-disjunta al encontrarse la *pdf* desplazada más a la derecha, por lo que se puede concluir que en general será más beneficioso disponer de rutas disjuntas lo más cerca posible del nodo origen que tenerlas repartidas entre varios nodos intermedios.

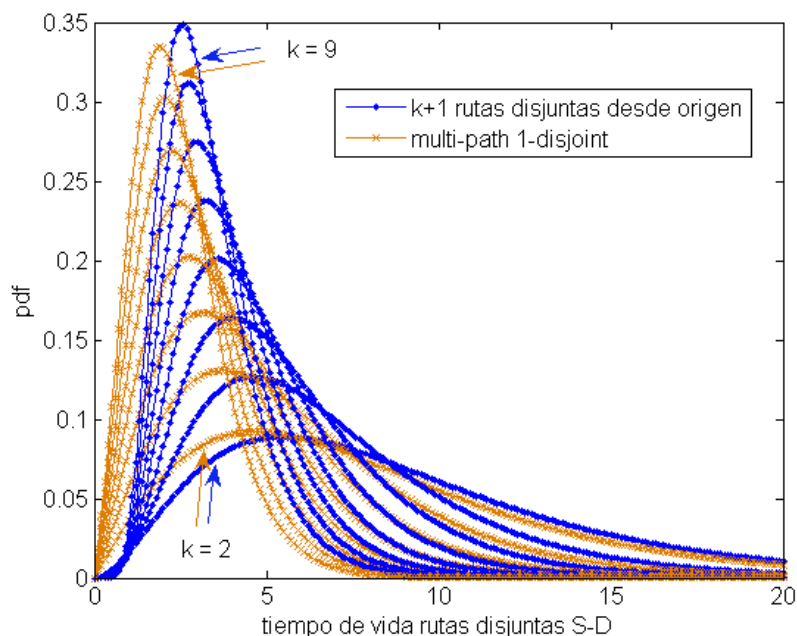


Figura 5.7: Comparación entre encaminamiento multi-camino con rutas paralelas disjuntas y con una ruta disjunta desde cada nodo de la ruta principal para $l = 10$ s.

5.1.2.4. Encaminamiento multi-camino: $m_1 + m_2 + \dots + m_k$ rutas alternativas disjuntas

Se considera ahora el caso en el que para cada nodo de la ruta principal n_i con $i = 1, \dots, k$ existen m_i rutas alternativas $\{P_{i,1}, \dots, P_{i,m_i}\}$ hacia el nodo destino disjuntas entre sí, disjuntas con la ruta principal formada por la secuencia de enlaces $\langle L_1 \rightarrow L_2 \rightarrow \dots \rightarrow L_k \rangle$ y disjuntas también con todas las rutas alternativas del resto de nodos de la ruta principal. Esta configuración es una combinación de las estudiadas en las Secciones 5.1.2.2 y 5.1.2.3 anteriores, en la que las restricciones anteriores se pueden expresar de la siguiente forma:

$$\forall n_i \in \langle L_1 \rightarrow L_2 \rightarrow \dots \rightarrow L_k \rangle, \exists \{P_{i,1}, \dots, P_{i,m_i}\} \quad (5.22)$$

tal que:

1. $P_{i,m_j} \cap L_h = \phi ; j = \{1, \dots, i\}; h = \{1, \dots, k\}$
2. $P_{i,m_j} \cap P_{i,m_n} = \phi ; j \neq n$
3. $P_{i,m_j} \cap P_{r,m_n} = \phi ; i \neq r ; j = \{1, \dots, i\}; n = \{1, \dots, r\}$

Este esquema se corresponde con la red mostrada en la Figura 5.8.

Análogamente al análisis teórico realizado para la red de la sección anterior, es posible aplicar los mismos pasos y consiguientemente llegar a una expresión del tiempo de vida del sistema S-D para esta configuración, que tendrá la forma indicada a continuación:

$$T = \min[\max(X_{L_1}, X_{P_1}^*), \max(X_{L_2}, X_{P_2}^*, X_{P_1}^*), \dots, \max(X_{L_k}, X_{P_k}^*, X_{P_{k-1}}^*, \dots, X_{P_1}^*)] \quad (5.23)$$

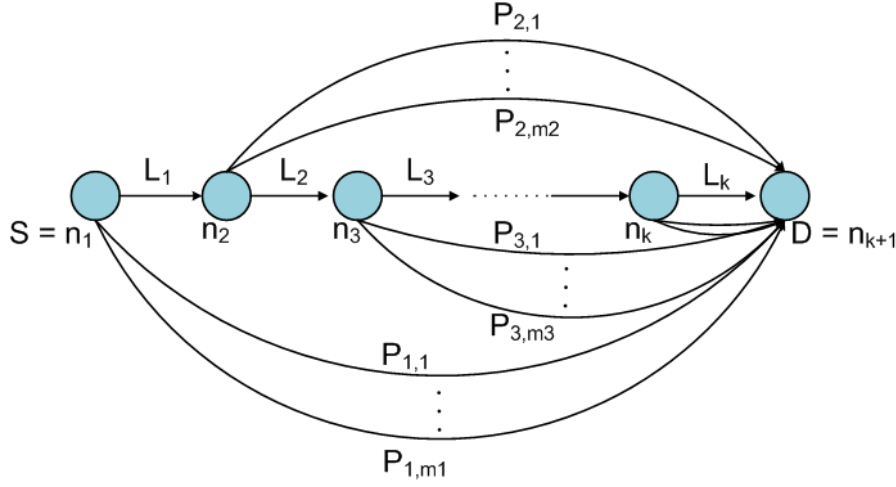


Figura 5.8: Encaminamiento multi-camino con $m_1 + m_2 + \dots + m_k$ rutas alternativas disjuntas

siendo la variable aleatoria $X_{P_i}^*$ el tiempo de vida resultante de las m_i rutas que parten del nodo n_i , denotadas como $P_{i,j}$, con $j = \{1, \dots, m_i\}$ y que se calcula como:

$$X_{P_i}^* = \max(X_{P_{i,1}}, \dots, X_{P_{i,m_i}}) \quad (5.24)$$

Por tanto, análogamente al caso de N rutas paralelas partiendo desde el origen, es posible calcular la *cdf* de la variable aleatoria resultante de considerar m_i variables aleatorias en paralelo como:

$$\begin{aligned} F_{X_{P_i}^*}(t) &= P(X_{P_i}^* \leq t) = P[\max(X_{P_{i,1}}, \dots, X_{P_{i,m_i}}) \leq t] = \\ &= P[(X_{P_{i,1}} \leq t) \cap \dots \cap (X_{P_{i,m_i}} \leq t)] = \\ &= P(X_{P_{i,1}} \leq t) \dots P(X_{P_{i,m_i}} \leq t) = \\ &= \prod_{r=1}^{m_i} F_{X_{P_{i,r}}}(t) \end{aligned} \quad (5.25)$$

Partiendo de la Ecuación (5.23), tomando $Z_i = \max(X_{L_i}, X_{P_i}^*, X_{P_{i-1}}^*, \dots, X_{P_1}^*)$ y siguiendo los mismos pasos desarrollados en la Ecuación (5.18), se llega a una

expresión para la *cdf* de Z_i como la siguiente:

$$\begin{aligned}
F_{Z_i}(t) &= P(Z_i \leq t) = P(\max(X_{L_i}, X_{P_i}^*, X_{P_{i-1}}^*, \dots, X_{P_1}^*) \leq t) = \\
&= P\left[(X_{L_i} \leq t) \cap (X_{P_i}^* \leq t) \cap (X_{P_{i-1}}^* \leq t) \cap \dots \cap (X_{P_1}^* \leq t)\right] = \\
&= P(X_{L_i} \leq t) \cdot P(X_{P_i}^* \leq t) \cdot P(X_{P_{i-1}}^* \leq t) \cdot \dots \cdot P(X_{P_1}^* \leq t) = \quad (5.26) \\
&= F_{L_i}(t) \cdot \prod_{j=1}^i F_{X_{P_j}^*}(t) = F_{L_i}(t) \cdot \prod_{j=1}^i \prod_{r=1}^{m_i} F_{X_{P_j,r}}(t)
\end{aligned}$$

Finalmente, para calcular la *pdf* del sistema completo sigue siendo válida la Ecuación (5.21), es decir:

$$f_T(t) = \sum_{i=1}^k f_{Z_i}(t) \cdot \prod_{\substack{j=1 \\ j \neq i}}^k (1 - F_{Z_j}(t)) \quad (5.27)$$

en donde $F_{Z_j}(t)$ y $f_{Z_i}(t)$ vienen dadas por la Ecuación (5.26) y por su derivada con respecto a t , respectivamente.

En la Figura 5.2 d) se presenta la *pdf* de esta configuración de rutas disjuntas. Puede observarse que, con respecto a las tres configuraciones analizadas anteriormente, esta configuración es más robusta frente a fallos aleatorios en los enlaces inalámbricos entre los nodos intermedios desde el origen S hasta el destino D, al proveer a dichos nodos de más rutas alternativas que el resto de configuraciones, lo que hace que la probabilidad de que aumente el tiempo de vida de las rutas es mayor que en las anteriores.

5.1.2.5. HARP: encaminamiento multi-camino con $m_1 + m_2 + \dots + m_{k-1}$ rutas alternativas disjuntas

En HARP, para cada uno de los saltos desde S hasta D, cada nodo intermedio buscará en su tabla de rutas una que sea válida hacia el destino y encaminará el

paquete al siguiente nodo indicado en la ruta escogida. En esta situación, para cada nodo intermedio podrá existir un número determinado de rutas disjuntas entre sí, de tal forma que en caso de fallo de una de ellas, dicho nodo estará en condiciones de encaminar el paquete por cualquiera de las otras rutas, en cuyo caso la probabilidad de fallo puede considerarse independiente del fallo producido en la anterior. Además de rutas disjuntas, cada nodo intermedio podrá usar enlaces que también pueden ser utilizados por otros nodos, proporcionando al sistema rutas no disjuntas entre S y D.

Esta situación es equivalente al caso en que algunas de las P_{i,m_i} rutas alternativas del nodo i no cumplan al menos una de las tres condiciones indicadas en la Ecuación (5.22), es decir, pueden existir rutas alternativas que tienen enlaces inalámbricos en común con otras rutas alternativas, incluida la principal. Esta configuración de rutas alternativas disjuntas y no disjuntas que caracteriza a HARP posee una mayor tolerancia a fallos de rutas que una configuración similar en la que únicamente los nodos almacenan las rutas disjuntas hacia el destino, descartando las que no lo son, tal y como se demuestra mediante simulaciones en [108].

Las variables aleatorias $X_{P_{i,m_i}}$ en esta configuración de rutas no son estadísticamente independientes entre sí, debido a que pueden contener nodos y enlaces inalámbricos en común unas con otras. En esta situación existen dos formas de modelar teóricamente las prestaciones del protocolo en cuanto a la fiabilidad del mismo, que son las siguientes:

1. Encontrar una formulación exacta para la *cdf* o la *pdf* en esta configuración utilizando cualquier método como los de inclusión-exclusión, descomposición en factores, etc. En este caso, en la Ecuación (5.18) no se puede considerar válido que la probabilidad de una intersección de sucesos sea el producto de las probabilidades de cada uno de ellos, debido a que estos sucesos pueden no ser independientes entre sí.

2. Establecer un límite inferior para las prestaciones del protocolo, de tal forma que el tiempo de vida real de las rutas entre el nodo origen y el destino proporcionado por esta configuración sea mayor o igual que el establecido por dicho límite inferior.

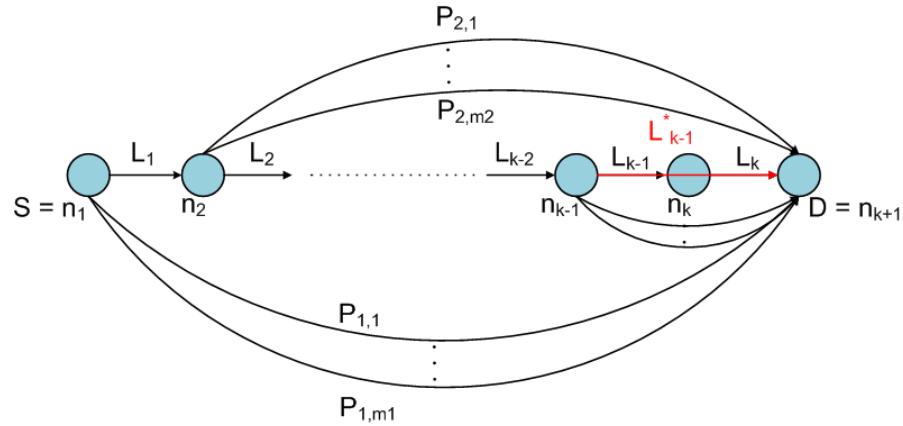


Figura 5.9: Configuración de rutas disjuntas para el encaminamiento multi-camino en HARP.

El método seleccionado en este análisis será el mencionado en segundo lugar, con lo que al eliminar las rutas redundantes esta configuración se reduce a la estudiada en el caso anterior con m_1, m_2, \dots, m_k rutas disjuntas desde cada nodo de la ruta principal. En consecuencia el límite inferior del tiempo de vida de las rutas en HARP puede ser calculado a priori a partir de los resultados obtenidos en las Ecuaciones (5.23) a (5.27). Esto es cierto salvo por una restricción que puede observarse en la Figura 5.9: por la propia definición del protocolo, a la hora de establecer asociaciones entre niveles consecutivos, los nodos del nivel inmediatamente anterior al nodo destino únicamente pueden almacenar una ruta hacia el mismo, ya que en su estructura de tablas solamente puede existir una entrada para cada nodo hijo directo. En consecuencia, la formulación del caso anterior aplicada

a HARP quedaría de la siguiente forma:

$$\begin{aligned}
T &= \min [Z_1, Z_2, \dots, Z_{k-1}] = \\
&= \min [\max(X_{L_1}, X_{P_1}^*), \max(X_{L_2}, X_{P_2}^*, X_{P_1}^*), \dots \\
&\quad \dots, \max(X_{L_{k-1}}, X_{P_{k-1}}^*, X_{P_{k-2}}^*, \dots, X_{P_1}^*)]
\end{aligned} \tag{5.28}$$

siendo $X_{P_i}^* = \max(X_{P_{i,1}}, \dots, X_{P_{i,m_i}})$, y el término $X_{L_{k-1}}^*$ una variable aleatoria que modela el tiempo de vida conjunto de los enlaces L_{k-1} y L_k y que, según se justifica en la Sección 5.1.2.1, tiene una distribución exponencial con media $l/2$. Por tanto, se puede calcular la *cdf* del conjunto de estos dos enlaces consecutivos asumiendo que son estadísticamente independientes entre sí, denotada por $F_{L_{k-1}}^*$, de la siguiente forma:

$$\begin{aligned}
F_{L_{k-1}}^* &= P(X_{L_{k-1}}^* \leq t) = P[\min(X_{L_{k-1}}, X_{L_k}) \leq t] = \\
&= 1 - P[\min(X_{L_{k-1}}, X_{L_k}) > t] = \\
&= 1 - P(X_{L_{k-1}} > t) \cdot P(X_{L_k} > t) = \\
&= 1 - (1 - F_{L_{k-1}}) \cdot (1 - F_{L_k})
\end{aligned} \tag{5.29}$$

siendo $F_{L_{k-1}}$ y F_{L_k} las *cdf* de las variables aleatorias que modelan el tiempo de vida de los enlaces L_{k-1} y L_k , respectivamente.

En la Figura 5.10 se muestra un ejemplo de Z_i para el caso de $k = 4$ saltos entre el origen y el destino. En este caso, los valores de Z_1 , Z_2 y Z_3 serían:

$$\begin{aligned}
Z_1 &= \max [X_{L_1}, X_{P_1}^*] \\
Z_2 &= \max [X_{L_2}, X_{P_1}^*, X_{P_2}^*] \\
Z_3 &= \max [X_{L_3}, X_{P_1}^*, X_{P_2}^*, X_{P_3}^*]
\end{aligned} \tag{5.30}$$

En consecuencia, la Ecuación (5.21) sigue siendo válida para calcular la *pdf* del tiempo de vida del sistema completo S-D, aunque teniendo en cuenta la restricción

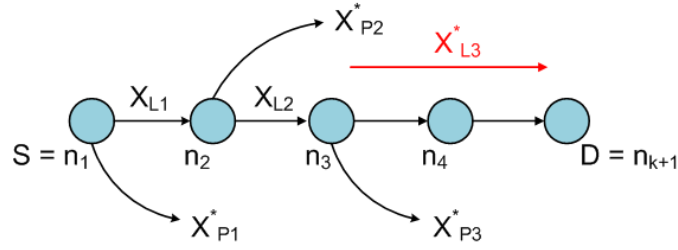


Figura 5.10: Configuración de rutas disjuntas para HARP con $k = 4$ saltos entre el origen y el destino.

impuesta en el último enlace, es decir:

$$f_T(t) = \sum_{i=1}^{k-1} f_{Z_i}(t) \cdot \prod_{\substack{j=1 \\ j \neq i}}^{k-1} (1 - F_{Z_j}(t)) \quad (5.31)$$

siendo:

$$F_{Z_i}(t) = \begin{cases} F_{L_i}(t) \cdot \prod_{j=1}^i F_{X_{P_j}^*}(t) & , \text{ si } 1 \leq i \leq k-2 \\ F_{L_i}^*(t) \cdot \prod_{j=1}^i F_{X_{P_j}^*}(t) & , \text{ si } i = k-1 \end{cases}$$

$$F_{X_{P_j}^*}(t) = \prod_{r=1}^{m_i} F_{X_{P_j,r}}(t) \quad (5.32)$$

$$F_{L_i}^*(t) = 1 - (1 - F_{L_{k-1}}) \cdot (1 - F_{L_k})$$

Por último, en el caso particular de que los enlaces inalámbricos sean estadísticamente iguales, con su *cdf* denotada por $F_L(t)$, se puede reescribir la *cdf* de las

funciones F_{Z_i} de la siguiente forma:

$$F_{Z_i}(t) = \begin{cases} F_L(t) \cdot \prod_{j=1}^i F_{X_{P_j}^*}(t) & , \text{ si } 1 \leq i \leq k-2 \\ [1 - (1 - F_L(t))^2] \cdot \prod_{j=1}^i F_{X_{P_j}^*}(t) & , \text{ si } i = k-1 \end{cases} \quad (5.33)$$

$$F_{X_{P_j}^*}(t) = \prod_{r=1}^{m_i} F_{X_{P_j,r}}(t)$$

En las Figuras 5.2 e) y 5.3 a) se puede observar por un lado la variación de la *pdf* de esta configuración de rutas disjuntas que establece un límite inferior de las prestaciones de HARP y, por el otro, el tiempo medio de vida de las rutas comparado con el resto de configuraciones. Si se compara con las cuatro configuraciones analizadas anteriormente, puede observarse que a igual número de rutas disjuntas totales, la configuración que HARP propone es sensiblemente más robusta que el resto, excepto para el caso d) en el que el último nodo n_k también conoce otras rutas alternativas disjuntas hacia el destino. Por otra parte, cuando varía el número de rutas disjuntas, la estructura que establece un límite inferior para HARP para $m = 2$ rutas en la Figura 5.3 a) tiene un rendimiento similar al de cuatro rutas paralelas que parten del origen, llegando a superarla conforme aumenta el número de saltos. Lo mismo ocurre para HARP con $m = 4$ y $N = 6$ rutas paralelas desde el origen, con lo que puede inferirse que, a mayor número de saltos no sólo basta con conocer rutas disjuntas desde el origen sino que se hace necesario establecer otras rutas que partan de los nodos intermedios y que sean disjuntas con el resto. En conclusión, se puede decir que, en general, el rendimiento de HARP es mejor cuanto mayor sea el número de saltos de la ruta comparada con otras aproximaciones.

En la Figura 5.3 b) se muestra una comparación directa para $k = 5$ saltos entre S y D, en función del número de rutas disjuntas en cada configuración. Los

resultados relativos al tiempo medio de vida de las rutas contenidos en la gráfica b) son consecuentes con las posiciones relativas de las *pdf* de cada una de las configuraciones analizadas.

Para complementar el análisis de las prestaciones teóricas de HARP se incluyen algunos gráficos que tratan de aclarar el comportamiento de HARP en función del número de rutas disjuntas encontradas por nodo de la ruta principal y en función del número de saltos hasta el destino. En la Figura 5.11 se observa cómo varía la *pdf* de HARP para distintos valores del número de rutas disjuntas que parten desde cada nodo, suponiendo que todos ellos excepto el último tienen el mismo número. Puede comprobarse que a mayor número de rutas y menor número de saltos aumenta la probabilidad de que el tiempo de vida de las rutas sea mayor.

Por otra parte, en esta Tesis se define la *sensibilidad del tiempo de vida total de las rutas entre S y D con respecto al número de rutas disjuntas que parten desde un nodo del nivel i-ésimo*, denotado por m_i , de la siguiente forma:

$$S_{m_i} = \frac{dF_t(t)}{dm_i} \quad (5.34)$$

En relación con este parámetro, la Figura 5.12 muestra la sensibilidad de la *pdf* con respecto a la variación del número de rutas disjuntas en cada uno de los niveles intermedios. A la vista de la misma se puede concluir que las prestaciones del protocolo son más sensibles a la variación de rutas disjuntas en los niveles superiores de la ruta S-D, es decir, los más cercanos al origen. Las variaciones en el número de rutas disjuntas encontradas en niveles próximos al destino tienen una influencia mucho menor en el tiempo de vida total. Este hecho ha de tenerse en cuenta a la hora del despliegue de los nodos y de la selección de su ubicación definitiva en la red.

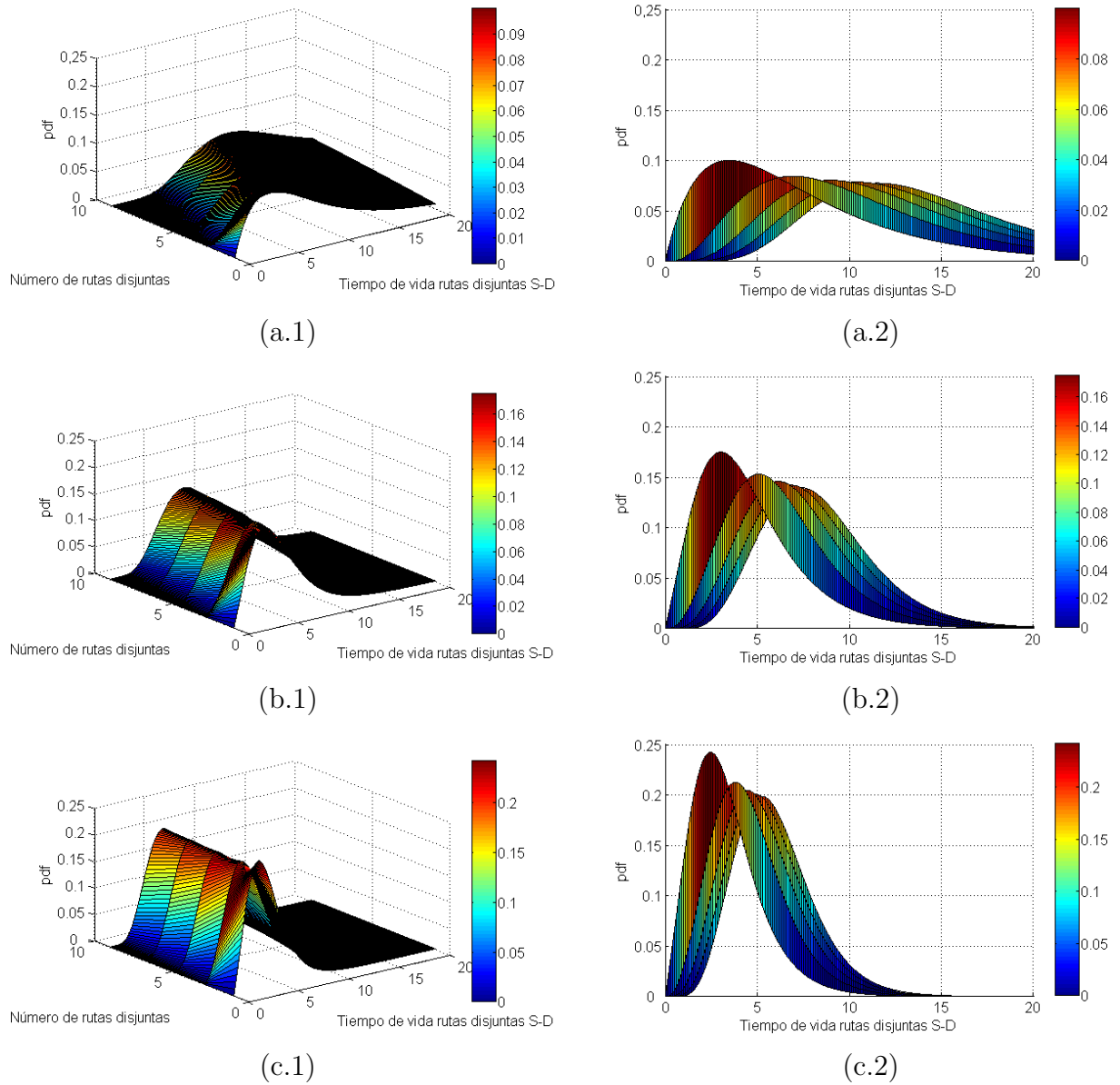


Figura 5.11: Pdf de HARP para $l = 10$ s y m variando entre 1 y 9 rutas disjuntas por nivel: a) $k = 2$ saltos entre S y D ; b) $k = 4$ saltos entre S y D ; c) $k = 6$ saltos entre S y D .

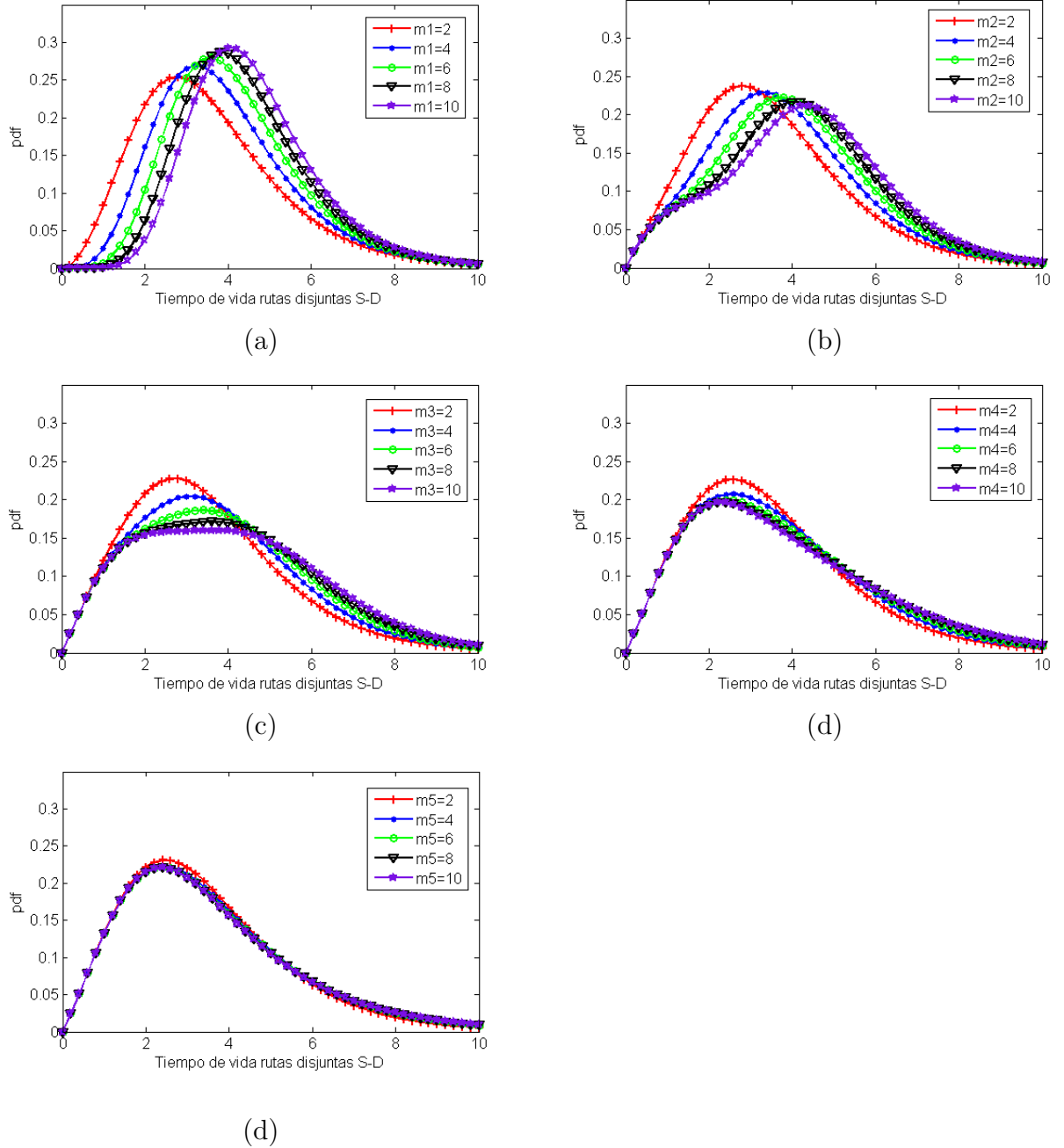


Figura 5.12: Pdf de HARP para $l = 10s$ y $k = 6$ saltos: a) m_1 varía y $m_2 = m_3 = m_4 = m_5 = 1, m_6 = 0$; b) m_2 varía y $m_1 = m_3 = m_4 = m_5 = 1, m_6 = 0$; c) m_3 varía y $m_1 = m_2 = m_4 = m_5 = 1, m_6 = 0$; d) m_4 varía y $m_1 = m_2 = m_3 = m_5 = 1, m_6 = 0$; e) m_5 varía y $m_1 = m_2 = m_3 = m_4 = 1, m_6 = 0$.

Para finalizar esta sección, en las fórmulas proporcionadas en este apartado para el cálculo de $f_T(t)$ es necesario conocer previamente los valores del número de rutas disjuntas desde cada nodo de la ruta principal m_i , con $i = 1, 2, \dots, k - 1$. En la sección siguiente se propone un procedimiento para calcular la probabilidad de encontrar este número de rutas alternativas disjuntas desde cada nodo de la ruta principal hasta el nodo destino y poder de esta forma calcular la expresión de la *pdf* del sistema para cualquier estructura de red.

5.1.3. Algoritmo de cálculo de probabilidad de rutas disjuntas

Para desarrollar esta sección se comenzará por estudiar la probabilidad de existencia de rutas disjuntas que parten desde el nodo origen S hasta el destino D. A continuación se extenderán los resultados obtenidos en este primer análisis calculando una expresión para la probabilidad de que existan rutas disjuntas repartidas entre todos los nodos de la ruta principal, para finalizar este estudio con la estructura de HARP en la que puedan existir $m_1 + m_2 + \dots + m_{k-1}$ rutas alternativas disjuntas desde los correspondientes nodos de la ruta principal hacia el nodo destino.

5.1.3.1. Probabilidad de encontrar h rutas disjuntas desde S hasta D

El cálculo de la probabilidad de encontrar un número determinado de rutas disjuntas que partan desde el nodo origen S hasta el nodo destino D requiere que se tenga en cuenta la división que el protocolo establece de los nodos de la red en diferentes niveles funcionales, los cuales imponen la estructura del árbol jerárquico explicada en esta Tesis. En la Figura 5.13 se muestra un ejemplo de esta estructura por niveles con la existencia de 2 rutas disjuntas entre S y D.

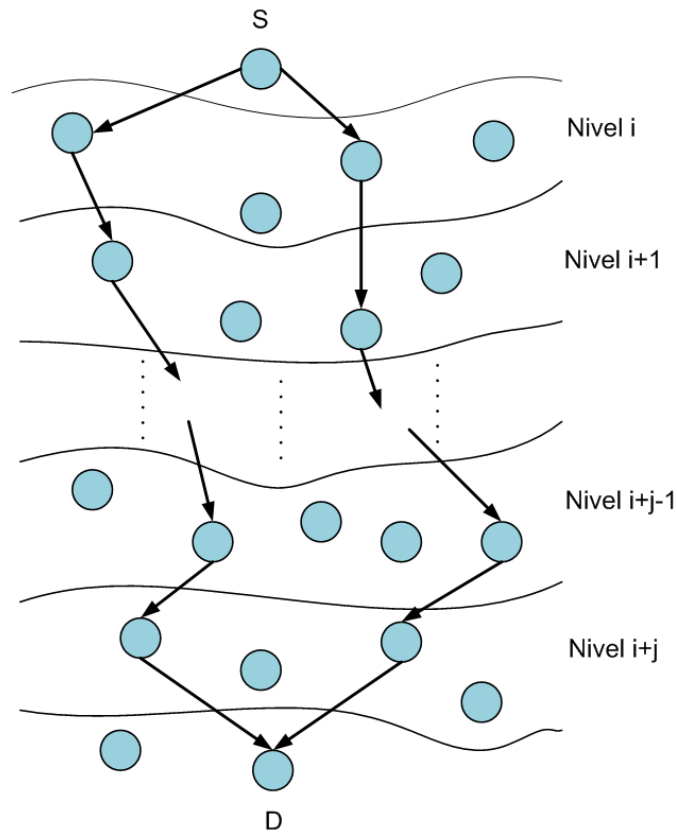


Figura 5.13: Ejemplo de la estructura por niveles de HARP con dos rutas disjuntas entre S y D .

La probabilidad de encontrar h rutas disjuntas entre S y D puede calcularse de forma probabilística a partir de la distribución de los nodos en la red y de la estructura jerárquica de nodos en niveles creada en la primera fase de puesta en marcha de la red. Para ello hay que tener en cuenta el número de combinaciones posibles de rutas disjuntas que pueden formarse entre un origen y un destino determinados en función del número de nodos que conocen el destino en cada uno de los niveles intermedios, y dividirlo por el número total de rutas (disjuntas y no disjuntas) que pueden formarse entre esos dos nodos S y D . Sin embargo, con la probabilidad anterior no es suficiente, según se indica en [18] y [17]. Para completar este cálculo es necesario además tener en cuenta la probabilidad de que pueda establecerse una conexión entre dos nodos consecutivos de esa ruta, ya que

en función de la posición de cada uno en la red y de su potencia de transmisión podrán comunicarse o no a través de un enlace inalámbrico entre ambos.

En consecuencia, si los niveles intermedios entre S y D se denotan por i hasta $i + j$ (es decir, existen $j + 1$ niveles intermedios entre S y D), siendo n_i cualquier nodo perteneciente al nivel i que conoce al menos una ruta hacia el destino, tal y como se muestra en la Figura 5.13, se tiene:

$$\begin{aligned}
 P(\exists h \text{ combinaciones de rutas disjuntas } S \rightarrow D) &= \\
 &= \frac{\prod_{l=i}^{i+j} C_{n_l,1} \cdot \prod_{l=i}^{i+j} C_{n_{l-1},1} \cdot \dots \cdot \prod_{l=i}^{i+j} C_{n_{l-(h-1)},1}}{\prod_{l=1}^{i+j} C_{n_l,1} \cdot \prod_{l=1}^{i+j} C_{n_l,1} \cdot \dots \cdot \prod_{l=1}^{i+j} C_{n_l,1}} = \\
 &= \frac{\prod_{m=0}^{h-1} \prod_{l=i}^{i+j} C_{n_{l-m},1}}{\left(\prod_{l=1}^{i+j} C_{n_l,1}\right)^h} = \prod_{m=0}^{h-1} \prod_{l=i}^{i+j} \frac{C_{n_{l-m},1}}{C_{n_l,1}}
 \end{aligned} \tag{5.35}$$

que indica la probabilidad de que un nodo pueda establecer una conexión con algún otro del nivel siguiente que no haya sido utilizado anteriormente para formar otras rutas, siendo:

$$C_{m,n} = \binom{m}{n} = \frac{m!}{n! \cdot (m-n)!} \tag{5.36}$$

Utilizando la igualdad de la Ecuación (5.36), se puede reescribir la Ecuación (5.35) de la siguiente forma:

$$\begin{aligned}
 P(\exists h \text{ combinaciones de rutas disjuntas } S \rightarrow D) &= \\
 &= \prod_{m=0}^{h-1} \prod_{l=i}^{i+j} \frac{\binom{n_l-m}{1}}{\binom{n_l}{1}} = \prod_{m=0}^{h-1} \prod_{l=i}^{i+j} \frac{n_l - m}{n_l}
 \end{aligned} \tag{5.37}$$

La demostración de esta formulación puede consultarse en [18].

Teniendo en cuenta la distribución física de los nodos en la red, se denota por $\varphi_{x,y}$ a la probabilidad de que exista conexión entre los nodos x e y , es decir, que exista un enlace inalámbrico entre ambos. Se puede calcular la probabilidad de

que exista una ruta desde S hasta D, que se denominará Ω , como la probabilidad de que exista un enlace entre S y un nodo del nivel i , un enlace entre los $j + 1$ nodos intermedios y un enlace entre el nodo del nivel $i + j$ y el nodo destino D. En consecuencia:

$$\Omega = \varphi_{S,n_i} \cdot \varphi_{n_i,n_{i+1}} \cdot \varphi_{n_{i+1},n_{i+2}} \cdots \varphi_{n_{i+j-1},n_{i+j}} \cdot \varphi_{n_{i+j},D} \quad (5.38)$$

Es decir, que una ruta existirá si y sólo si existen todos los enlaces inalámbricos que la forman. Si cualquiera de estas conexiones no existiera, la ruta tampoco existiría. En el caso particular de que la probabilidad de conexión sea la misma para todos los niveles involucrados, se tiene que $\varphi_{x,y} = \varphi, \forall x, y$. Por tanto, tal y como se asume en [19]:

$$\Omega = \varphi^{j+2} \quad (5.39)$$

Teniendo en cuenta la distribución de los nodos en la red, para cada una de las rutas disjuntas, es necesario multiplicar el resultado de la Ecuación (5.37) por la probabilidad de que exista conexión entre todos los nodos consecutivos de la ruta dadas las posiciones de cada nodo, es decir, que la ruta sea capaz de conectar la fuente S con el destino D. Por tanto, teniendo en cuenta dicho término que viene dado por la igualdad de la Ecuación (5.38), se puede escribir:

$$\begin{aligned} P(\exists h \text{ rutas disjuntas } S \rightarrow D) = \\ P(\exists h \text{ combinaciones de rutas disjuntas } S \rightarrow D) \cap P(h \text{ rutas conecten } S \rightarrow D) \end{aligned} \quad (5.40)$$

Lo que conduce a la siguiente expresión:

$$P(\exists h \text{ rutas disjuntas } S \rightarrow D) = \prod_{m=0}^{h-1} \prod_{l=i}^{i+j} \frac{(n_l - m) \cdot \Omega}{n_l} \quad (5.41)$$

En el caso más simple, se podrá establecer una conexión entre dos nodos x e y siempre que la distancia que los separa $d(x, y)$ sea menor que la distancia máxima alcanzada en el rango de transmisión, R , del nodo emisor, es decir, utilizando un modelo radio en el que la potencia recibida depende únicamente de la distancia al nodo transmisor, se tiene:

$$\varphi_{x,y} = \begin{cases} 1 & , \text{ si } d(x, y) \leq R \\ 0 & , \text{ si } d(x, y) > R \end{cases} \quad (5.42)$$

En consecuencia, para este modelo se tiene que Ω puede tomar únicamente los valores 0 y 1, por lo que su contribución en la Ecuación (5.41) es 0 ó 1 según sea o no posible establecer una ruta disjunta nueva entre los nodos S y D.

Sin embargo, el modelo anterior de pérdidas puede resultar bastante impreciso en la realidad ya que los niveles de potencia recibidos en un entorno del emisor pueden experimentar variaciones importantes a pesar de que no existan diferencias significativas en cuanto a la distancia al origen. Existen por tanto modelos radio más realistas como que tienen en cuenta las variaciones en la potencia de la señal radio causadas por obstáculos o irregularidades de algún tipo alrededor de las antenas transmisora o receptora y que son, en general, más complejos de analizar. Un ejemplo de este tipo de modelos de propagación radio es el llamado *log-normal shadowing* [67], en el que la probabilidad de conexión entre dos nodos de la red no depende exclusivamente de la distancia al nodo transmisor, sino que tiene en cuenta otras circunstancias como las indicadas anteriormente. En este modelo la probabilidad de conexión viene determinada por el parámetro ξ , que tiene en cuenta los efectos que el entorno produce en una transmisión determinada y que se indica en la Figura 5.14 extraída del mencionado trabajo. El caso $\xi = 0$ se corresponde con el modelo simple de pérdidas basado en la distancia al nodo transmisor.

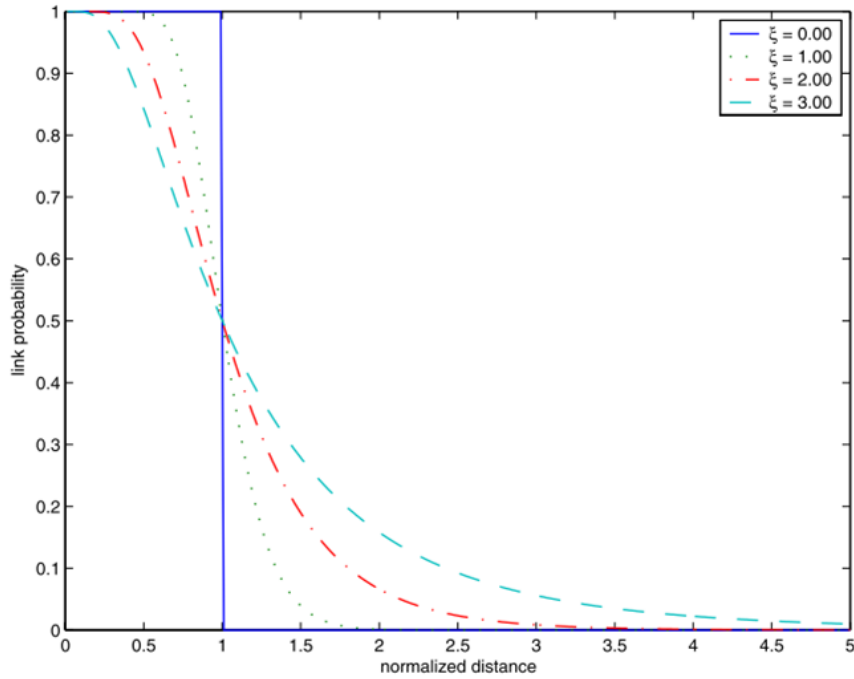
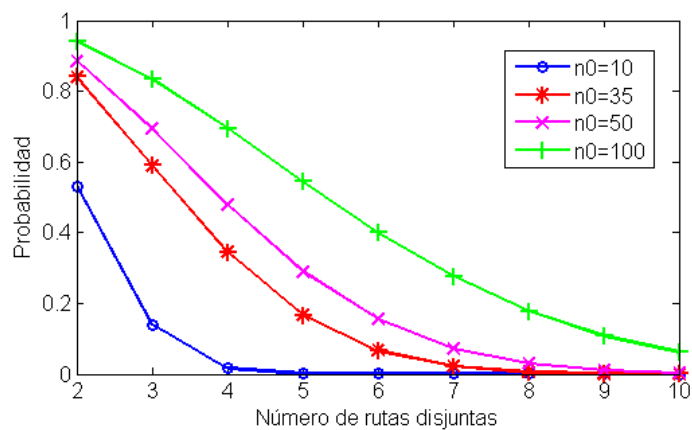
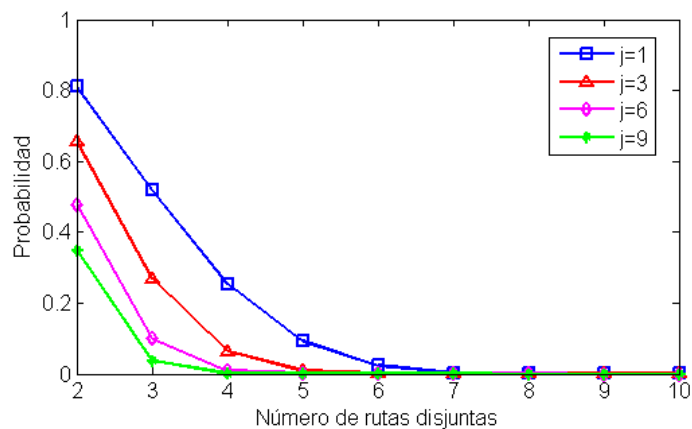


Figura 5.14: Probabilidad de conexión entre dos nodos según el modelo log-normal shadowing incluido en [67] para distintos valores de ξ .

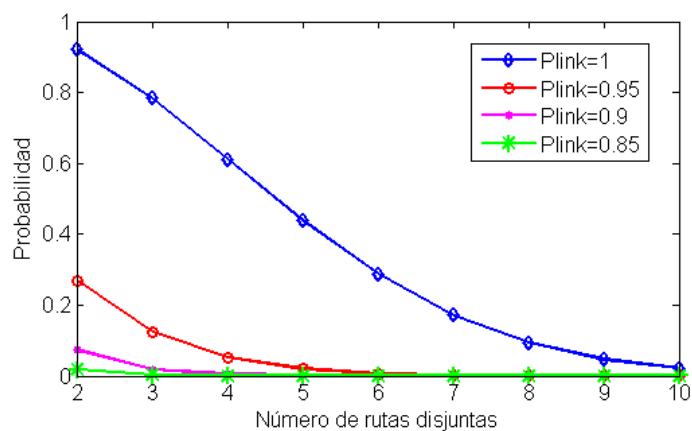
Para finalizar esta sección, en la Figura 5.15 se indica, a partir de la Ecuación (5.41), cómo varía la probabilidad de encontrar rutas disjuntas que partan de S en función del número de nodos por nivel que conocen al menos una ruta hacia el destino (n_0), del número de niveles intermedios (j) y de la probabilidad de conexión de nodos de niveles consecutivos (P_{link}). Puede observarse que mayor será la probabilidad de encontrar rutas disjuntas cuanto mayor sea el número de nodos por nivel, menor el número de saltos entre origen y destino y mayor sea la probabilidad de que existan los enlaces entre los nodos intermedios.



(a)



(b)



(c)

Figura 5.15: Probabilidade de encontrar rotas disjuntas que partem de S hasta D : a) para $j = 5$ niveles intermedios y $\Omega = 1$; b) para $n_0 = 10$ nodos por nivel y $\Omega = 1$; c) para $j = 3$ niveles intermedios y $n = 50$ nodos por nivel.

5.1.3.2. Probabilidad de encontrar m_i rutas disjuntas en el camino desde S hasta D

En esta sección se generalizará la Ecuación (5.41) para poder aplicarla a las rutas disjuntas que parten desde cualquier nodo situado en los niveles intermedios entre S y D.

Denotando como n_q a un nodo del nivel q -ésimo ($i - 1 \leq q \leq i + j - 1$) que conoce al menos una ruta hacia el destino, se tiene que:

$$P(\exists m_q \text{ rutas disjuntas } n_q \rightarrow D) = \prod_{m=0}^{m_q-1} \prod_{l=q+1}^{i+j} \frac{(n_l - m) \cdot \Omega_q}{n_l} \quad (5.43)$$

siendo:

$$\Omega_q = \varphi_{n_q, n_{q+1}} \cdot \varphi_{n_{q+1}, n_{q+2}} \cdots \varphi_{n_{i+j-1}, n_{i+j}} \cdot \varphi_{n_{i+j}, D} \quad (5.44)$$

A continuación se muestra el desarrollo anterior para cada uno de los niveles intermedios entre S y D con la notación empleada en la Figura 5.8:

$$\begin{aligned} i &= 2 \\ i + j &= k \end{aligned} \quad (5.45)$$

En este caso, siendo ($1 \leq q \leq k - 1$) se tiene que:

$$P(\exists m_q \text{ rutas disjuntas } n_q \rightarrow D) = \prod_{m=0}^{m_q-1} \prod_{l=q+1}^k \frac{(n_l - m) \cdot \Omega_q}{n_l} \quad (5.46)$$

siendo:

$$\Omega_q = \varphi_{n_q, n_{q+1}} \cdot \varphi_{n_{q+1}, n_{q+2}} \cdots \varphi_{n_{k-1}, n_k} \cdot \varphi_{n_k, D} \quad (5.47)$$

Particularizando para cada uno de los niveles intermedios y asumiendo que la probabilidad de que dos nodos de niveles consecutivos estén conectados es la

Nodo	Nº saltos hasta D	Probabilidad
S=n ₁	k	$\prod_{m=0}^{m_1-1} \prod_{l=2}^k \frac{(n_l-m) \cdot \varphi^k}{n_l}$
n ₂	k-1	$\prod_{m=0}^{m_2-1} \prod_{l=3}^k \frac{(n_l-m) \cdot \varphi^{k-1}}{n_l}$
...
n _{k-2}	3	$\prod_{m=0}^{m_{(k-2)}-1} \prod_{l=k-1}^k \frac{(n_l-m) \cdot \varphi^{k-(k-3)}}{n_l}$
n _{k-1}	2	$\prod_{m=0}^{m_{(k-1)}-1} \prod_{l=k}^k \frac{(n_l-m) \cdot \varphi^{k-(k-2)}}{n_l}$
n _k	1	0

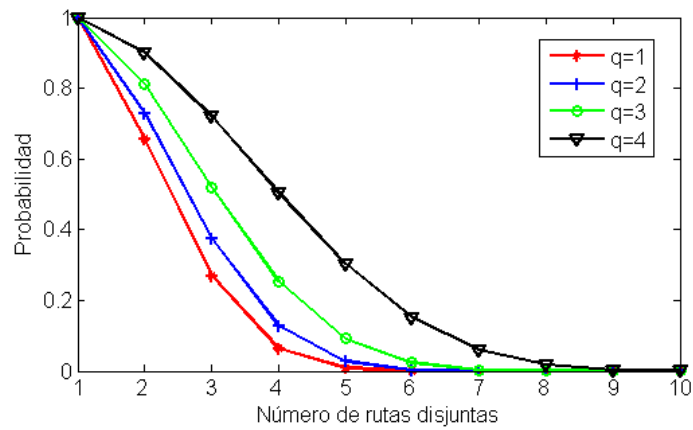
Tabla 5.1: Probabilidad de encontrar rutas alternativas disjuntas desde cada nodo de la ruta principal hasta el nodo destino.

misma para todos ellos, la Ecuación (5.46) puede particularizarse para cada uno de los nodos n_i , con $i = 1, 2, \dots, k$ situados en los niveles desde el origen hasta el destino, tal y como se muestra en la Tabla 5.1.

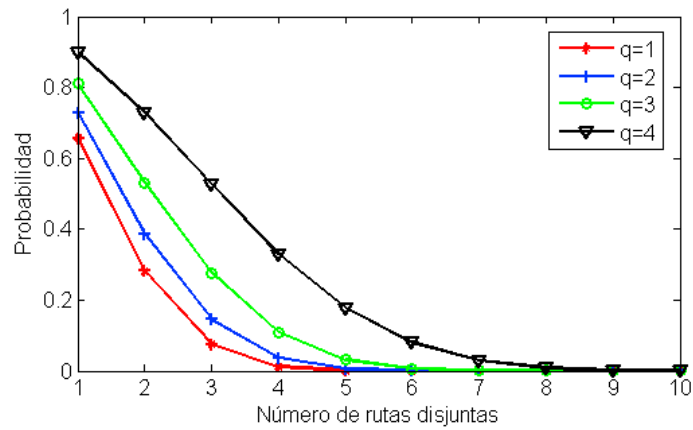
En la Figura 5.16 se analiza la Ecuación (5.46) a través de las curvas que indican cómo varía la probabilidad de encontrar rutas disjuntas a partir de un nodo según sea el nivel q en el que se encuentre, para un total de $k = 5$ saltos entre origen y destino, $n = 10$ nodos por nivel y para diferentes valores de Ω . El caso $q = 1$ corresponde al origen S. Se puede observar con cierta claridad que la probabilidad de encontrar rutas disjuntas es mayor cuanto menor es la diferencia de niveles entre el nodo intermedio y el destino, es decir, que la probabilidad de encontrar rutas disjuntas disminuye a medida que el nodo intermedio esté más alejado del nodo destino.

5.1.3.3. Cálculo de $m_1 + m_2 + \dots + m_{k-1}$ rutas alternativas disjuntas

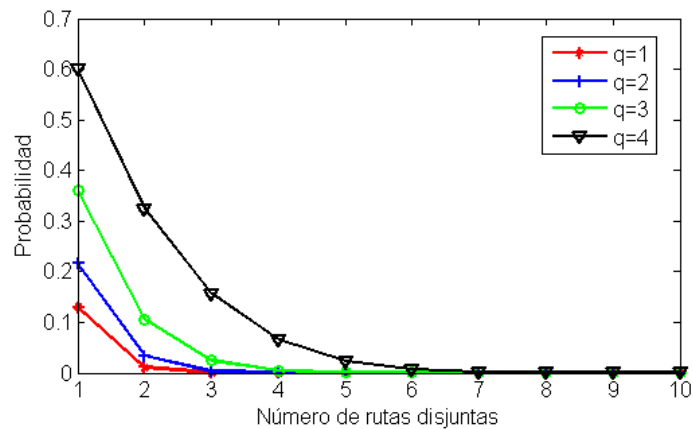
En la sección anterior se ha proporcionado una formulación para calcular la probabilidad de encontrar un número determinado de rutas disjuntas entre sí desde cada uno de los nodos de la ruta principal hacia un destino concreto. Sin embargo, no se consideraba ninguna restricción acerca de la relación entre las rutas que parten desde un nodo con las que parten desde el resto. En esta sección se incluye



(a)



(b)



(c)

Figura 5.16: Probabilidad de encontrar rutas disjuntas en función del nivel del nodo para $k = 5$ saltos S-D y $n = 10$ nodos por nivel: a) para $\Omega = 1$; b) $\Omega = 0.9$; c) $\Omega = 0.6$.

por tanto la condición de que las rutas que parten de un nodo sean disjuntas no sólo entre sí, sino también con todas las demás rutas alternativas que partan de cualquiera de los nodos de la ruta principal, es decir, que cumplan las condiciones en la Ecuación (5.22).

En estas condiciones, el desarrollo teórico anterior debe tener en cuenta para el cálculo de las rutas que partan desde un determinado nivel, los nodos que ya se estén utilizando en rutas ya formadas en niveles anteriores. En concreto, esos nodos no podrán utilizarse para formar las nuevas rutas. De esta forma, reescribiendo la Ecuación (5.46) se tiene para $i \leq q \leq k - 1$:

$$P_{m_q} = P(\exists m_q \text{ rutas disjuntas } n_q \rightarrow D) = \prod_{m=0}^{m_q-1} \prod_{l=q+1}^k \frac{(n_l - m - M) \cdot \Omega_q}{n_l} \quad (5.48)$$

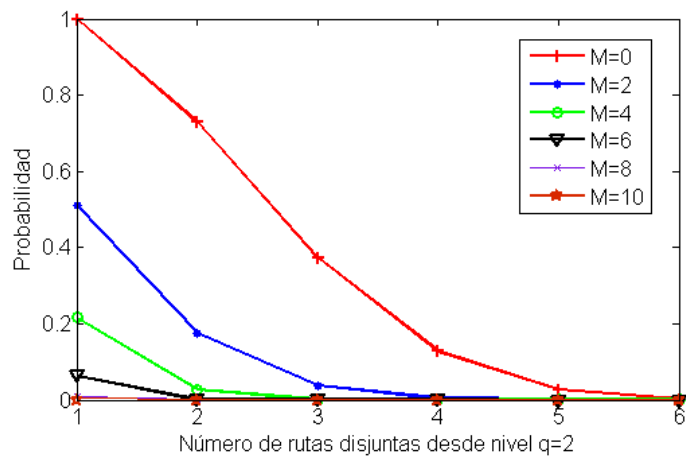
siendo:

$$\begin{aligned} \Omega_q &= \varphi_{n_q, n_{q+1}} \cdot \varphi_{n_{q+1}, n_{q+2}} \cdots \varphi_{n_{k-1}, n_k} \cdot \varphi_{n_k, D} \\ M &= \sum_{r=1}^{q-1} m_r \end{aligned} \quad (5.49)$$

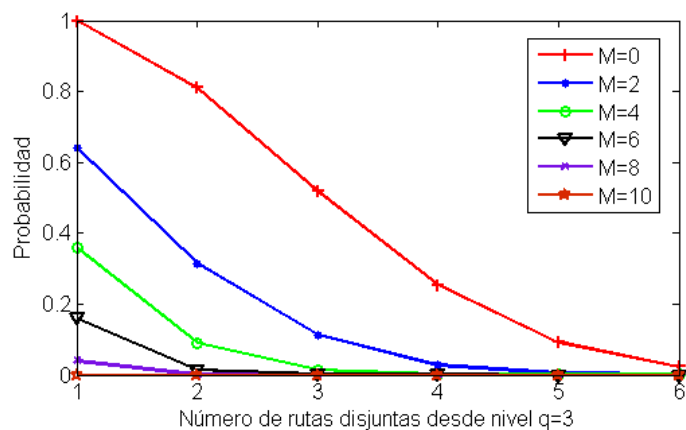
donde el parámetro m_r indica el número de rutas disjuntas hacia el destino que parten desde un nodo del nivel r . M indica, para cada nivel, los nodos que ya están siendo utilizados para formar rutas disjuntas en niveles anteriores, por lo que se calcula sumando el número de rutas disjuntas formadas en todos los niveles anteriores desde el origen hasta el nivel q . En consecuencia, para conocer el número de rutas disjuntas que pueden formarse desde un nodo del nivel q es necesario calcular previamente el número de rutas disjuntas que se pueden formar desde los nodos de niveles anteriores al q , partiendo desde S que, según la Figura 5.8, está en el nivel 1.

Es coherente pensar que a medida que se van encontrando rutas disjuntas en los niveles superiores disminuye la probabilidad de que puedan formarse más rutas

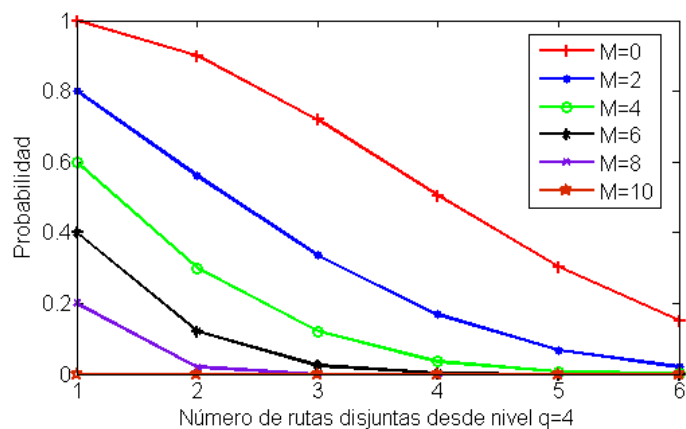
en los niveles más cercanos al destino. En la Figura 5.17 se muestran simulaciones para 3 niveles intermedios en los que se hace evidente que a medida que M aumenta y se encuentran más rutas disjuntas en nodos superiores, la probabilidad de formar nuevas rutas disjuntas en cada nivel disminuye sensiblemente.



(a)



(b)



(c)

Figura 5.17: Probabilidad de encontrar rutas disjuntas en función las rutas disjuntas de niveles superiores (M) para $k = 5$ saltos $S-D$ y $n = 10$ nodos por nivel: a) primer nivel intermedio (el más cercano a S); b) segundo nivel intermedio; c) tercer nivel intermedio.

5.1.4. Cálculo de la *pdf* del tiempo de vida del sistema

Para calcular estadísticamente el tiempo de vida de todas las rutas entre S y D hay que partir de los resultados obtenidos en la Sección 5.1.2.4, en donde se consideraban que todas las rutas alternativas son disjuntas entre sí. Las Ecuaciones (5.23) y (5.24) siguen siendo válidas en este caso, ya que todas las rutas alternativas que parten desde un nodo siguen siendo disjuntas entre sí y con el resto de rutas alternativas. A su vez, la Ecuación (5.25) que modela el tiempo de vida de las rutas alternativas que parten desde el nodo i -ésimo tiene en cuenta un número determinado de rutas disjuntas hacia el destino, por lo que puede reescribirse introduciendo ese número de rutas disjuntas como una nueva variable aleatoria, la cual ha sido caracterizada en la sección anterior a través de la Ecuación (5.46). Para ello, a continuación es posible aplicar el *teorema de la probabilidad total* para calcular la *cdf* del tiempo de vida del conjunto de rutas que parten desde un nodo cualquiera de la ruta principal.

Sea M_i la variable aleatoria discreta que modela el número de rutas disjuntas que es posible encontrar entre un nodo del nivel i -ésimo y el destino D. En consecuencia, los sucesos $\{M_i = 0\}, \{M_i = 1\}, \{M_i = 2\}, \dots, \{M_i = m_i\}$ constituyen una partición del espacio muestral formado por todas las posibles rutas disjuntas existentes entre un nodo del nivel i -ésimo y el destino D.

Se puede calcular la *cdf* del tiempo de vida de las rutas que parten desde un nodo cualquiera de la ruta principal a partir de la siguiente expresión:

$$\begin{aligned}
F_{X_{P_i}^*}(t) &= P(X_{P_i}^* \leq t) = \sum_{m_i=1}^{z_i} P[(X_{P_i} \leq t) \cap (M_i = m_i)] = \\
&= \sum_{m_i=1}^{z_{m_i}} P[(X_{P_i} \leq t) \mid (M_i = m_i)] \cdot P[M_i = m_i] = \\
&= P[(X_{P_i} \leq t) \mid (M_i = 0)] \cdot P[M_i = 0] + \\
&+ P[(X_{P_i} \leq t) \mid (M_i = 1)] \cdot P[M_i = 1] + \dots \\
&\dots + P[(X_{P_i} \leq t) \mid (M_i = z_{m_i})] \cdot P[M_i = z_{m_i}]
\end{aligned} \tag{5.50}$$

siendo z_{m_i} el número máximo de rutas disjuntas que pueden existir entre un nodo del nivel i -ésimo y el destino D, y que está limitado por el menor de dos números: el número de enlaces del nodo del nivel i -ésimo con nodos del siguiente nivel y el número de enlaces entre los nodos del nivel inmediatamente anterior al destino y el propio destino, es decir:

$$\begin{aligned}
z_{m_i} &= \min(n_{i+1}^*, n_k^*) \\
n_{i+1}^* &= n_{i+1} \cdot \varphi_{n_i, n_{i+1}} \\
n_k^* &= n_k \cdot \varphi_{n_k, D}
\end{aligned} \tag{5.51}$$

Los términos de probabilidad condicionada de la Ecuación (5.50) pueden calcularse a partir del modelo de N rutas disjuntas paralelas independientes caracterizado por la Ecuación (5.8), es decir:

$$\begin{aligned}
P[(X_{P_i} \leq t) \mid (M_i = m_i)] &= P[\max(X_{P_{i,1}}, X_{P_{i,2}}, \dots, X_{P_{i,m_i}}) \leq t] = \\
&= P[(X_{P_{i,1}} \leq t) \cap (X_{P_{i,2}} \leq t) \cap \dots \cap (X_{P_{i,m_i}} \leq t)] = \prod_{z=1}^{m_i} F_{X_{P_{i,z}}}(t)
\end{aligned} \tag{5.52}$$

siendo $F_{X_{P_{i,z}}}(t) = 1 - e^{-\lambda_i t}$, $t \geq 0$ según (5.5).

Por último, sustituyendo el término anterior en la Ecuación (5.50), se obtiene la fórmula general de la *cdf* del tiempo de vida de las rutas que parten desde cualquier nodo de la ruta principal:

$$\begin{aligned}
F_{X_{P_i}^*}(t) &= P(X_{P_i}^* \leq t) = \sum_{m_i=1}^{z_{m_i}} P[(X_{P_i}^* \leq t) \cap (M_i = m_i)] = \\
&= \sum_{m_i=1}^{z_{m_i}} P[(X_{P_i}^* \leq t) \mid (M_i = m_i)] \cdot P[M_i = m_i] = \\
&= \sum_{m_i=1}^{z_{m_i}} \left(\prod_{z=1}^{m_i} F_{X_{P_i,z}}(t) \cdot P[M_i = m_i] \right)
\end{aligned} \tag{5.53}$$

Para calcular $P[M_i = m_i]$ hay que aplicar el Algoritmo 1 desarrollado en esta Tesis descrito a continuación, el cual, a su vez, efectúa llamadas a los Algoritmos 2, 3 y 4.

Un ejemplo de aplicación de los citados algoritmos se encuentra en el Apéndice A.

Partiendo de la Ecuación (5.32) y considerando que en este caso las variables aleatorias involucradas $X_{P_i}^*$ sí son independientes entre sí ya que no contienen nodos ni enlaces comunes, se puede sustituir el término que expresa el tiempo de vida de todas las rutas disjuntas que parten desde un nodo por la Ecuación (5.53). Finalmente, se aplica la expresión general siguiente:

$$\begin{aligned}
F_{Z_i}(t) &= P(Z_i \leq t) = P(\max(X_{L_i}, X_{P_i}^*, X_{P_{i-1}}^*, \dots, X_{P_1}^*) \leq t) = \\
&= F_{L_i}(t) \cdot \prod_{j=1}^i F_{X_{P_j}^*}(t)
\end{aligned} \tag{5.54}$$

Algoritmo 1 Cálculo de $P[M_i = m_i]$

```

if  $i = 1$  {nodo S} then
  if  $m_1 = 0$  then
     $P[M_1 = 0] = 0$  {siempre existe al menos una ruta desde S hasta D}
  else
    for  $m_1 = 1$  to  $z_{m_1}$  do
       $P[M_1 = m_1] = T_{11}(m_1) \cdot T_{12}(m_1)$ 
    end for
  end if
else
  for  $i = 2$  to  $k - 1$  {para el resto de niveles} do
    for  $m_i = 0$  to  $z_{m_i}$  do
       $P[M_i = m_i] = \sum_{M=1}^{z_m} P[(M_i = m_i) \cap (M_{T_i} = M)] =$ 
       $= \sum_{M=1}^{z_m} P[(M_i = m_i) \mid (M_{T_i} = M)] \cdot P[M_{T_i} = M]$ 
       $z_m = \max(z_{m_1}, \dots, z_{m_{i-1}})$ 
       $P[M_i = m_i \mid (M_{T_i} = M)] = T_{i1}(m_i, M) \cdot T_{i2}(m_i, M)$ 
    end for
  end for
end if

```

Algoritmo 2 Cálculo de $P[M_{T_i} = M]$

```

if  $i = 1$  then
   $P[M_{T_1} = 0] = 1$ 
   $P[M_{T_1} \neq 0] = 0$  {en el primer nivel no existen rutas disjuntas previas}
else
  for  $i = 2$  to  $k - 1$  do
     $P[M_{T_i} = M] = \sum_{j=0}^M P[m_i = j] \cdot P[M_{T_{i-1}} = M]$ 
  end for
end if

```

Algoritmo 3 Cálculo de T_{11} y T_{12}

$$T_{11}(m_1) = \prod_{m=0}^{m_1-1} \left(\prod_{l=2}^k \frac{(n_l - m) \cdot \Omega_1}{n_l} \right)$$

$T_{12}(m_1)$:

Número de elementos: $C_{R1} = C_{R(m_1, z_{m_1} - m_1)}$

C_R : combinaciones de elementos con repetición, $C_{R(m,n)} = \frac{(m+n-1)!}{n! \cdot (m-1)!}$

Elementos = $\{1, 2, \dots, m_1\}$

Se forma la matriz U: por filas las combinaciones posibles

if $C_{R1} = 0$ {si $m_1 = z_{m_1}$ } **then**

$$T_{12}(m_1) = 1$$

else

$$T_{12}(m_1) = \sum_{h=1}^{C_{R1}} \prod_{j=1}^{z_{m_1} - m_1} \left(1 - \prod_{l=2}^k \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right)$$

end if

Algoritmo 4 Cálculo de T_{i1} y T_{i2}

$T_{i1}(m_i, M)$

if $m_i = 0$ **then**

$$T_{i1}(0, M) = 1$$

else if $m_i = z_{m_i}$ **then**

$$T_{i1}(z_{m_i}, M) = 0$$

else

$$T_{i1}(m_i, M) = \prod_{m=M}^{m_i+M-1} \left(\prod_{l=i+1}^k \frac{(n_l - m) \cdot \Omega_i}{n_l} \right)$$

end if

$T_{i2}(m_i, M)$

Número de elementos: $C_{Ri} = C_{R(m_i+1, z_{m_i} - (m_i+M))}$

Elementos = $\{M, M+1, \dots, M+m_i\}$

Se forma la matriz U: por filas las combinaciones posibles

if $C_{Ri} = 0$ {si $m_i + M = z_{m_i}$ } **then**

$$T_{i2}(z_{m_i}, M) = 1$$

else if $m_i + M > z_{m_i}$ **then**

$$T_{i2}(z_{m_i}, M) = 0$$

else

$$T_{i2}(m_i, M) = \sum_{h=1}^{C_{Ri}} \prod_{j=1}^{z_{m_i} - (m_i+M)} \left(1 - \prod_{l=i+1}^k \frac{(n_l - u_{hj}) \cdot \Omega_i}{n_l} \right)$$

end if

Para calcular la *pdf* del tiempo de vida del sistema completo se aplica el resultado ya conocido de la Ecuación (5.31):

$$f_T(t) = \sum_{i=1}^k f_{Z_i}(t) \cdot \prod_{\substack{j=1 \\ j \neq i}}^k (1 - F_{Z_j}(t)) \quad (5.55)$$

donde, para el caso de que los enlaces sean estadísticamente iguales, se tiene:

$$F_{L_i}(t) = \begin{cases} F_L(t) & , \text{ si } 1 \leq i \leq k - 2 \\ 1 - (1 - F_L(t))^2 & , \text{ si } i = k - 1 \end{cases} \quad (5.56)$$

5.1.5. Análisis del proceso de Recuperación Local

Cuando un paquete se está transmitiendo a través de la red y encuentra un enlace inalámbrico por el cual no puede propagarse (bien por las razones indicadas en la Sección 5.1.2.1, bien porque hay congestión en esa zona de la red) se considera que se ha producido un error en la transmisión del mismo. Este error tiene como consecuencia la invalidez de la ruta escogida hacia el destino, lo que hace necesario encontrar una nueva ruta para poder alcanzar el nodo al cual va dirigido.

En general, los protocolos de encaminamiento reactivos tienen dos formas principales de enfrentarse a estas situaciones de error en la transmisión del paquete o en la ruptura de algún enlace inalámbrico en la red:

- *Recuperación extremo a extremo (end-to-end recovery)*: es la alternativa utilizada por algunos protocolos de encaminamiento en la cual, cuando un nodo detecta un error la transmisión de un paquete, informa al nodo origen a través de un mensaje de error, quien iniciará la búsqueda de una ruta alternativa en caso de no conocer ninguna otra ruta válida.

- *Recuperación local (RL) (local recovery)*: cuando alguno de los nodos intermedios de una ruta detecta la ruptura de algún enlace inalámbrico procede a su reparación mediante la búsqueda de rutas alternativas a partir de él, sin que el nodo origen intervenga necesariamente en dicho proceso.

Supóngase la situación mostrada en la Figura 5.18 a) en la cual un paquete se origina en el nodo A con destino el nodo D. Siguiendo el algoritmo de operación de HARP, el nodo A buscará primero en su tabla de hijos y después en la de padres al nodo destino y, al no encontrarlo, enviará el paquete a alguno de sus nodos padre situados en un nivel superior en la jerarquía de la red (que, en este caso, se reducen al nodo B). Por tanto, suponiendo que el enlace inalámbrico entre A y B está activo, el nodo B recibirá el paquete transmitido por A y repetirá el mismo proceso, procediendo entonces a la retransmisión del paquete hacia el nodo D. Si en ese momento el enlace entre B y D estuviese roto y B no dispusiese de ninguna ruta alternativa hacia el nodo D, el paquete nunca alcanzaría su destino, a menos que se iniciara un nuevo procedimiento de búsqueda de rutas hacia el nodo D o se ejecutara algún mecanismo de reparación de la ruta dañada, como es el caso de HARP con el proceso de *recuperación local* de rutas descrito en el Capítulo 3.

Este proceso de recuperación finalizará según esta figura con la asociación del nodo B con el nodo C como nuevo padre, por lo que la conexión del nodo B con los niveles superiores de la red volvería a estar activa, pasando a establecerse una estructura de niveles como la que aparece en la figura 5.18 b).

En esta situación el nodo B ya está en condiciones de reenviar el paquete recibido al nodo C, y éste finalmente lo transmitirá al nodo D, con lo que la transmisión entre el origen A y el destino D ha sido completada con éxito. Nótese que en esta situación el nodo D no actúa en ningún caso e incluso no es consciente de la recuperación que está llevando a cabo⁴, únicamente recibirá los paquetes

⁴Solamente sería consciente del error en caso de que pretendiese enviar un paquete hacia el nodo B mientras se está ejecutando el proceso de recuperación local

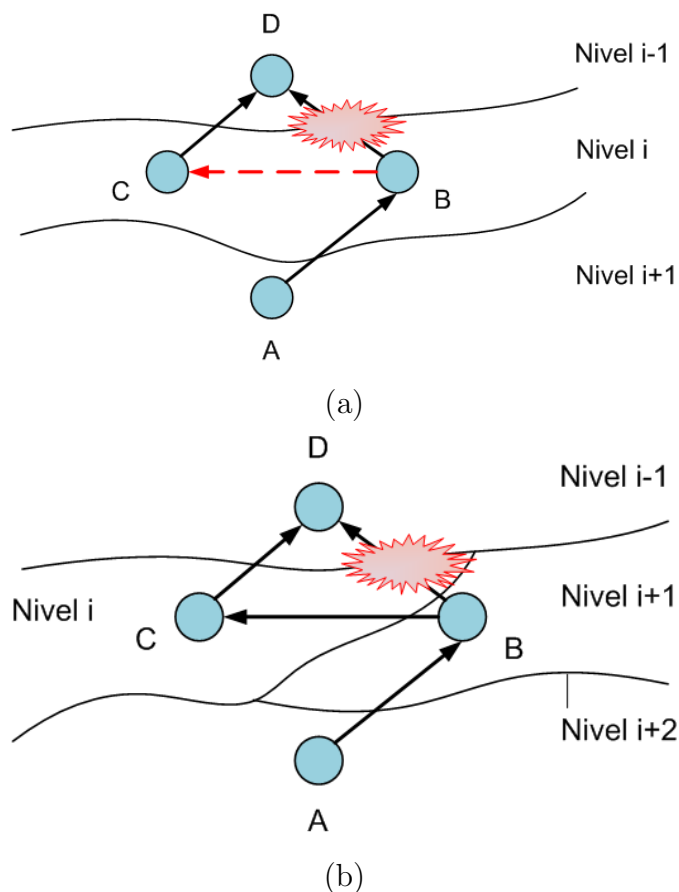


Figura 5.18: Proceso de RL de un enlace inalámbrico roto en HARP: a) situación inicial; b) configuración final de niveles con la nueva ruta creada.

de control necesarios para adaptar sus tablas a la nueva situación, sin perder conocimiento de la red inferior. Con este proceso de reconfiguración dinámica se consigue aumentar el tiempo de vida de las rutas entre A y D con respecto al de la ruta original ABD a través de la creación de una nueva ruta alternativa ABCD.

Para que la ejecución del proceso de RL implique un aumento real del tiempo de vida de las rutas entre A y D, éste debe completarse antes de que ambos nodos sean conscientes del error producido en la ruta. Para ello deben cumplirse las siguientes dos condiciones:

1. El fallo en el enlace B-D se debe producir antes de que B inicie una transmisión hacia D.

2. El instante de recuperación del enlace (en el que se ha creado y está operativa la ruta alternativa B-C-D) debe ser anterior al instante en el que D pretenda enviar un nuevo paquete hacia el nodo B.

A continuación se presenta un análisis teórico del impacto que tiene la RL de errores en el aumento de tiempo de vida de una ruta y las condiciones que han de cumplirse para que ello ocurra, lo que implica la existencia de un valor de probabilidad de RL denotada por P_{rec} en función de los instantes de transmisión de diferentes nodos a través de los enlaces que forman la ruta.

5.1.5.1. Caso 1: cálculo de P_{rec} con un solo flujo de paquetes hacia el destino y un único nodo intermedio

En primer lugar se analizará la situación operativa de una ruta entre dos nodos D y A en donde se tienen en cuenta, por un lado, los instantes de tiempo en los cuales el nodo D pretende iniciar una transmisión hacia el nodo B, por otro lado, el momento en el que el enlace que une los nodos B y D falla y, por último, el instante temporal en el que B pretende transmitir un paquete hacia el nodo D. Tomando como origen de tiempo (instante 0) la transmisión con origen en D previa al fallo del enlace y considerando los instantes anteriormente descritos, se produciría una situación descrita por el diagrama de tiempo que se indica en la Figura 5.19:

Siendo:

M : instante de tiempo en el que se produce el error en el enlace B-D (con respecto a la última transmisión desde D hacia B).

T_M : duración del error en el enlace B-D.

T_B : instante de tiempo en el que B inicia la transmisión hacia D a través del enlace B-D.

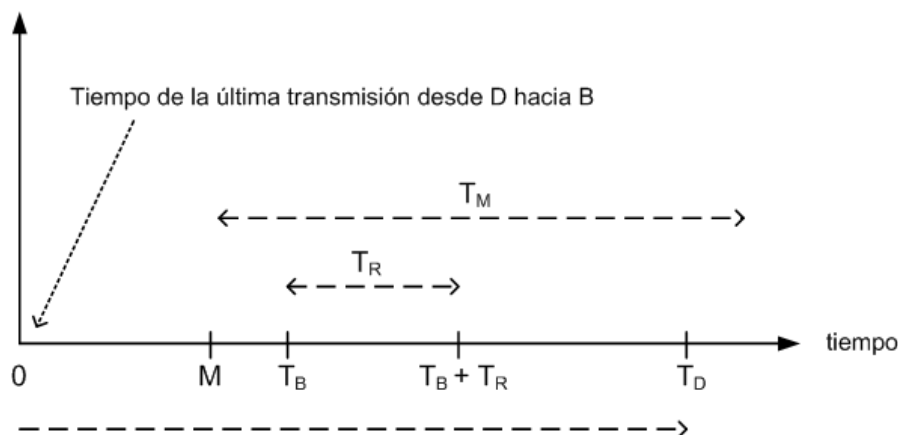


Figura 5.19: Esquema de tiempos en un proceso de RL.

T_R : duración de la recuperación local del enlace (tiempo transcurrido hasta que se forma la ruta alternativa B-C-D).

T_D : instante de tiempo en el que D inicia la transmisión de un nuevo paquete con destino B con respecto a la última (tiempo entre transmisiones).

En este esquema se considera que el tiempo de duración del fallo del enlace inalámbrico T_M es suficientemente grande comparado con el tiempo T_D entre sucesivas transmisiones desde D hacia B. Si no fuese así y el enlace se recuperara antes de que D iniciase una nueva transmisión hacia B, ésta podría llevarse a cabo utilizando dicho enlace y la ruta directa original sería por tanto reutilizable (en este caso D no sería consciente de que se ha producido un error en el enlace con B).

Para el cálculo analítico de las prestaciones del protocolo, se considerarán M , T_B y T_D como variables aleatorias que modelan cada uno de los conceptos anteriormente descritos.

Por otra parte, el tiempo de duración de la RL, T_R , es una variable que depende de los niveles jerárquicos de los nodos involucrados, de la configuración de la red (número de nodos vecinos), del rango o potencia de transmisión de dichos nodos

y de los tiempos de propagación y procesado de los paquetes. Este valor T_R se calcula para cada caso particular en función de los parámetros indicados y puede considerarse constante para un nodo determinado en una situación determinada. En el caso en que no pueda encontrarse una ruta alternativa, el valor de T_R sería infinito y, en consecuencia, no existiría recuperación local del error.

De esta forma, considerando el esquema de la Figura 5.19, se puede expresar la probabilidad de RL ante el fallo del enlace B-D, P_{rec} , como:

$$P_{rec} = P \{(M \leq T_B) \cap (T_B + T_R \leq T_D)\} \quad (5.57)$$

Teniendo en cuenta que no existe a priori relación alguna entre los instantes en los que se producen los fallos en los enlaces y los instantes en los que cada nodo de la red genera un paquete e inicia una transmisión, se puede asumir que las variables aleatorias M , T_B y T_D son independientes entre sí, por lo que, aplicando el *teorema de la probabilidad total*, se puede expresar la Ecuación (5.57) anterior como:

$$\begin{aligned} P_{rec} &= P(T_B + T_R \leq T_D \mid M \leq T_B) \cdot P(M \leq T_B) = \\ &= P(M \leq T_B) \cdot P(T_B + T_R \leq T_D) \end{aligned} \quad (5.58)$$

Para el cálculo de cada uno de los dos términos de la Ecuación (5.58) debe resolverse una desigualdad entre dos variables aleatorias, resultando las siguientes expresiones:

$$\begin{aligned} P(M \leq T_B) &= \int_0^{\infty} P(T_B = t) \cdot P(M \leq t) dt = \\ &= \int_0^{\infty} P(M \leq T_B \mid T_B = t) \cdot f_{T_B}(t) dt = \\ &= \int_0^{\infty} F_M(t) \cdot f_{T_B}(t) dt \end{aligned} \quad (5.59)$$

para el primer término; desarrollando la segunda desigualdad se tendría el siguiente resultado:

$$\begin{aligned}
 P(T_B + T_R \leq T_D) &= \int_0^{\infty} P(T_D = t) \cdot P(T_B + T_R \leq t) dt = \\
 &= \int_0^{\infty} P(T_B + T_R \leq T_D | T_D = t) \cdot f_{T_D}(t) dt = \\
 &= \int_0^{\infty} P(T_B + T_R \leq t) \cdot f_{T_D}(t) dt = \tag{5.60} \\
 &= \int_0^{\infty} P(T_B \leq t - T_R) \cdot f_{T_D}(t) dt = \\
 &= \int_0^{\infty} F_{T_B}(t - T_R) \cdot f_{T_D}(t) dt
 \end{aligned}$$

Al igual que en otros trabajos de investigación como [31], [87] y [85], se puede particularizar la formulación anterior para el caso de que los procesos tanto de generación de paquetes como de inicio de fallos en enlaces sean en ambos casos procesos de Poisson. En esta situación, dado que la distribución de probabilidad de los tiempos entre ocurrencias de sucesos de un proceso de Poisson es una función exponencial [25], las variables aleatorias que modelan el tiempo hasta el siguiente evento (transmisión o fallo) M , T_B y T_D seguirán una distribución de este tipo, es decir, que las expresiones de las *pdf* asociadas a dichos eventos serán las siguientes:

$$\begin{aligned}
 f_M(t) &= \begin{cases} \alpha e^{-\alpha t} & , \text{ si } t \geq 0 \\ 0 & , \text{ si } t < 0 \end{cases} \\
 f_{T_B}(t) &= \begin{cases} \lambda_B e^{-\lambda_B t} & , \text{ si } t \geq 0 \\ 0 & , \text{ si } t < 0 \end{cases} \\
 f_{T_D}(t) &= \begin{cases} \lambda_D e^{-\lambda_D t} & , \text{ si } t \geq 0 \\ 0 & , \text{ si } t < 0 \end{cases}
 \end{aligned} \tag{5.61}$$

donde $1/\alpha$ representa el valor medio del tiempo entre fallos consecutivos del enlace inalámbrico; por su parte, $1/\lambda_B$ y $1/\lambda_D$ se identifican con el valor medio del tiempo entre transmisiones a través del enlace D-B de los nodos B y D, respectivamente.

Sustituyendo los términos correspondientes a las variables aleatorias exponenciales en la Ecuación (5.59) y desarrollando, se obtiene la siguiente expresión:

$$P(M \leq T_B) = \int_0^{\infty} (\lambda_B e^{-\lambda_B t}) \cdot (1 - e^{-\alpha t}) dt = \frac{\alpha}{\alpha + \lambda_B} \quad (5.62)$$

Con respecto al término correspondiente a la Ecuación 5.60 el resultado obtenido es:

$$\begin{aligned} P(T_B + T_R \leq T_D) &= \int_0^{\infty} (\lambda_D e^{-\lambda_D t}) \cdot (1 - e^{-\lambda_B \cdot (t - T_R)}) dt = \\ &= \int_0^{\infty} \lambda_D e^{-\lambda_D t} dt - e^{\lambda_B \cdot T_R} \int_0^{\infty} (\lambda_D e^{-(\lambda_D + \lambda_B)t}) dt = \\ &= 1 - \frac{\lambda_D}{\lambda_D + \lambda_B} \cdot e^{\lambda_B \cdot T_R} \end{aligned} \quad (5.63)$$

En consecuencia, según la Ecuación (5.57) se puede expresar la probabilidad de RL ante el fallo del enlace B-D como:

$$\begin{aligned} P_{rec} &= P\{(M \leq T_B) \cap (T_B + T_R \leq T_D)\} = \\ &= \left(\frac{\alpha}{\alpha + \lambda_B}\right) \cdot \left(1 - \frac{\lambda_D}{\lambda_D + \lambda_B} \cdot e^{\lambda_B \cdot T_R}\right) \end{aligned} \quad (5.64)$$

En el caso ideal $T_R = 0$, la probabilidad de recuperación dependería únicamente de los valores medios de las tasas de fallos y de transmisión a través del enlace fallido, resultando la siguiente expresión:

$$P_{rec} \approx \left(\frac{\alpha}{\alpha + \lambda_B}\right) \cdot \left(\frac{\lambda_B}{\lambda_B + \lambda_D}\right) \quad (5.65)$$

5.1.5.2. Caso 2: cálculo de P_{rec} con múltiples flujos de paquetes hacia el destino y un único nodo intermedio

En el apartado anterior se ha definido la variable aleatoria T_B para modelar el instante de tiempo en el que el nodo B inicia el reenvío a través del enlace roto del paquete de datos enviado por A. Sin embargo, en el caso más general, en una red inalámbrica real, el nodo B recibirá, además de tráfico de paquetes procedente del nodo A, otro flujo de paquetes procedentes de cualquier otro nodo vecino (nodo H) con destino el nodo D y tendrá, a su vez, la capacidad de generar sus propios paquetes con este mismo destino, por lo que se transmitirán otros dos flujos adicionales a través de la ruta A-D. Esta situación aparece reflejada en la Figura 5.20.

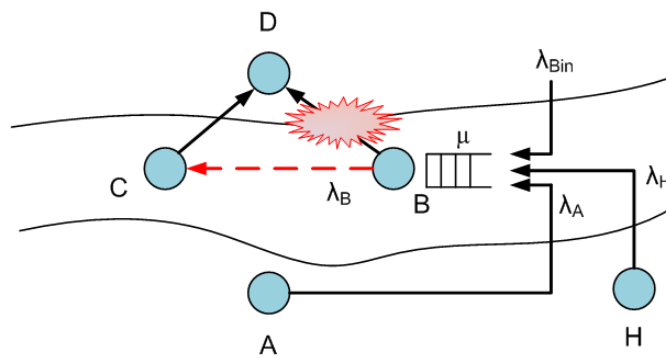


Figura 5.20: Nodo B con diversos flujos de entrada de datos con destino el nodo D.

En adelante se asumirá que la tasa de generación de paquetes por parte de los diferentes nodos es independiente entre sí y, en consecuencia, cada uno de estos tráficos es independiente del resto. Dado que la suma de procesos de Poisson independientes es otro proceso de Poisson cuya tasa de llegada es la suma de las tasas de llegadas de ambos procesos y que la descomposición de un proceso de Poisson da lugar a otros procesos de Poisson cuya suma de caudales es igual al caudal original [25], es posible modelar en estas condiciones la llegada de paquetes al nodo B como una cola $M/M/1$ con las siguientes características [78]:

- La tasa de entrada (recepción) de paquetes al nodo B, λ_{in} , es la suma de tres tasas de entrada: la procedente del nodo A (λ_A), de otros nodos vecinos (λ_H) y la generada por él mismo ($\lambda_{B_{in}}$)⁵, es decir:

$$\lambda_{in} = \lambda_A + \lambda_H + \lambda_{B_{in}} \quad (5.66)$$

- La variable aleatoria $T_{B_{IN}}$ que modela el instante de tiempo en el que el nodo B recibe un paquete en su cola de entrada (tiempo entre llegadas) tiene una distribución exponencial.
- El tiempo medio de espera de un paquete en el nodo B (desde que se recibe en la cola hasta que se transmite, es decir, el tiempo de espera en cola sumado al tiempo de procesado) es inversamente proporcional a la diferencia entre μ y λ_{in} , siendo $1/\mu$ el tiempo medio de procesado de un paquete por parte del nodo B, equivalente al tiempo medio de servicio.

En consecuencia, se puede expresar la variable aleatoria T_B a partir de $T_{B_{in}}$, de la siguiente forma:

$$\begin{aligned} T_{B_{in}} &= T_{AB} + T_{HB} + T_{B_{in}} \\ T_B &= T_{B_{in}} + T \end{aligned} \quad (5.67)$$

donde:

T_{AB} : instante de tiempo en el que B recibe un paquete procedente del nodo A (con respecto a la última transmisión realizada desde A hasta B).

T_{HB} : instante de tiempo en el que B recibe un paquete procedente de otro nodo de la red que no se encuentra en la ruta A-D.

⁵Asumiendo $M/M/1$ en el que no se pierden ni se descartan paquetes en la cola por tener capacidad infinita

$T_{B_{in}}$: instante de tiempo en el que B genera un paquete propio y lo introduce en su cola con destino un nodo de jerarquía superior de la ruta A-D.

T : tiempo de espera del paquete en el sistema desde que es recibido hasta que el nodo B inicia su retransmisión, equivalente al tiempo de espera en cola sumado al tiempo de procesado.

Para aplicaciones de WSN puede asumirse que los tiempos de transmisión y propagación en los enlaces (desde que se inicia la transmisión hasta que el destino recibe el paquete), así como el tiempo de recuperación del enlace y el tiempo T de espera en cola se pueden despreciar con respecto al tiempo entre recepción de paquetes $1/\lambda_{in}$ ⁶, es decir:

- $\mu \gg \lambda_{in}$.
- Capacidad de la cola es infinita (modelo $M/M/1$).
- tiempo de transmisión + tiempo de propagación $\ll 1/\lambda_{in}$.
- tiempo de RL del enlace $\ll 1/\lambda_{in}$.

Con estas suposiciones se puede considerar que T_B es también un proceso de Poisson cuya tasa de paquetes transmitidos a través del enlace B-D es la suma de las tasas de entrada λ_{in} , es decir, que también sigue una distribución exponencial.

Un ejemplo de esta situación se muestra en el diagrama de tiempos de la Figura 5.21, en la cual el nodo B realiza 4 transmisiones con destino el nodo D en el intervalo indicado. Las dos primeras se producen antes de que el enlace B-D se rompa, con lo que el paquete alcanzará su destino con éxito. La tercera transmisión se produce después del que el enlace falle, por lo que será la que origine el inicio del proceso de recuperación local en este caso. Por último, la cuarta transmisión

⁶En este segundo caso, $T = 1/(\mu - \lambda_{in})$ es muy pequeño comparado con $T_{B_{in}}$

se produce una vez que se ha encontrado una ruta alternativa entre B y D, por lo que el paquete alcanzará el destino a través de la ruta alternativa creada.

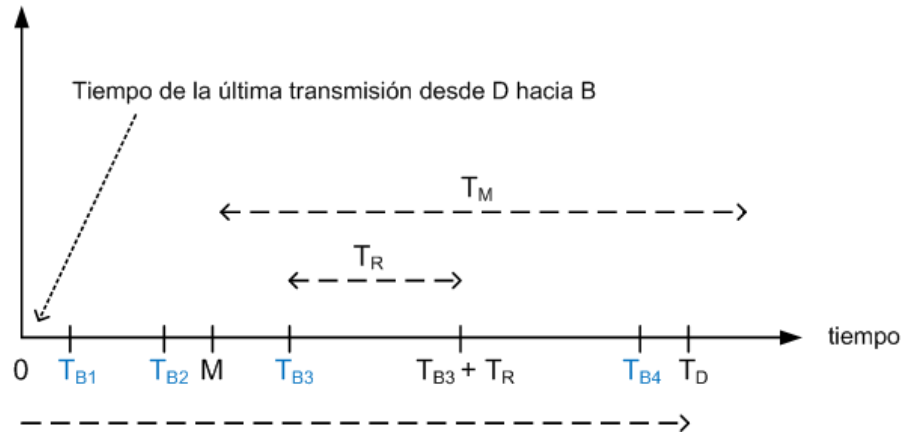


Figura 5.21: Esquema de tiempos en un proceso de RL: ejemplo con 4 transmisiones a través del enlace roto por parte del nodo que inicia el proceso.

En este caso no puede aplicarse la Ecuación (5.57) ya que las dos primeras transmisiones del nodo B a través del enlace se producen antes de que dicho enlace se rompa, por lo que no originan ningún proceso de RL. Si en este caso se aplicara la condición $M \leq T_B$ se estaría asumiendo que la primera transmisión de B denotada por T_{B1} debe producirse después de que se produzca el fallo, lo que no es cierto. Por lo tanto, para desencadenar el proceso de RL del enlace la primera condición es que una de las transmisiones del nodo B con destino el nodo D se inicie después del fallo del enlace, es decir:

$$\begin{aligned} P(M \leq T_B) &= 1 - P(M > T_B) = \\ &= 1 - \{P(M > T_{B1}) \cap \dots \cap P(M > T_{B4})\} \end{aligned} \quad (5.68)$$

La segunda condición es que esta transmisión desde B hacia D se produzca antes de que D inicie el envío de un paquete hacia B:

$$P(T_B + T_R \leq T_D) = P(T_B^* + T_R \leq T_D) \quad (5.69)$$

siendo:

$$T_B^* = \min(T_{Bi} \mid T_{Bi} > M) = T_{B3} \quad (5.70)$$

Para calcular la probabilidad de recuperación del enlace, P_{rec} , asumiendo que la distribución de tiempos entre envíos de paquetes desde el nodo B hacia el nodo D es exponencial y que el tiempo entre fallos de un enlace es también un proceso de Poisson según la Ecuación (5.61), se puede hacer uso de la *propiedad de ausencia de memoria o propiedad de Markov* de este tipo de distribución y puede establecerse que la distribución del tiempo hasta el próximo envío es independiente de cuándo se produjo el último envío y también del instante en el que se produjo el fallo del enlace, es decir:

$$P(T_B \leq M + T_D \mid T_B > M) = P(T_B \leq T_D) \quad (5.71)$$

En conclusión, la probabilidad de que B realice una transmisión entre el momento del fallo del enlace (M) y del envío de D (T_D), desencadenante del proceso de recuperación local, es igual a la probabilidad de que la variable aleatoria T_B sea menor o igual que T_D . Aplicando los conceptos incluidos en las Ecuaciones (5.69) y (5.70), se puede calcular P_{rec} a partir de las Ecuaciones (5.58) y (5.63) de la siguiente forma:

$$\begin{aligned} P_{rec} &= P(T_B + T_R \leq T_D \mid M \leq T_B) \cdot P(M \leq T_B) = \\ &= P(T_B^* + T_R \leq T_D) = P(T_B + T_R \leq T_D) = \\ &= 1 - \frac{\lambda_D}{\lambda_D + \lambda_B} \cdot e^{\lambda_B \cdot T_R} \end{aligned} \quad (5.72)$$

Finalmente, suponiendo, tal y como se ha hecho anteriormente, que el tiempo de recuperación local del enlace T_R es muy pequeño comparado con el tiempo

entre dos transmisiones consecutivas por parte del nodo B, se puede escribir la Ecuación (5.72) como:

$$P_{rec} \approx P(T_B \leq T_D) = \frac{\lambda_B}{\lambda_B + \lambda_D} \quad (5.73)$$

5.1.5.3. Caso 3: cálculo de P_{rec} con múltiples flujos de paquetes hacia el destino y múltiples nodos intermedios

En la Ecuación (5.66) se ponía de manifiesto que el flujo de paquetes que llega al último nodo de la ruta D-A (el más cercano a A) se componía de la suma de varios flujos de paquetes. El cálculo de esta tasa de paquetes a la entrada se puede generalizar para cualquier nodo de la ruta principal de forma análoga a como se ha calculado para este primer nodo, resultando las siguientes expresiones a la luz de la Figura 5.22:

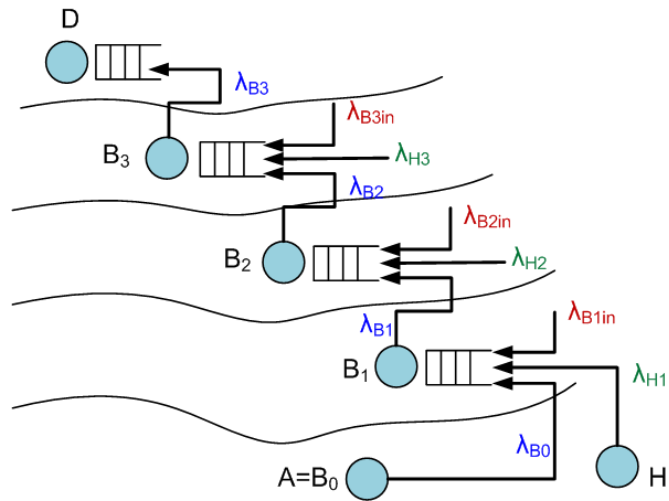


Figura 5.22: Esquema general de una ruta en la que existen múltiples flujos de paquetes hacia cada nodo de la ruta principal.

Para el último nodo de la ruta D-A (B_1):

$$\lambda_{B_1} \approx \lambda_{B_1in} = \lambda_{B_0} + \lambda_{H_1} + \lambda_{B_1in} \quad (5.74)$$

Para el penúltimo nodo de la ruta D-A (B_2):

$$\lambda_{B_2} \approx \lambda_{B_2in} = \lambda_{B_1} + \lambda_{H_2} + \lambda_{B_2in} = (\lambda_{B_0} + \lambda_{H_1} + \lambda_{B_1in}) + \lambda_{H_2} + \lambda_{B_2in} \quad (5.75)$$

Para el nodo j -ésimo de la ruta D-A (B_j):

$$\lambda_{B_j} \approx \lambda_{B_jin} = \lambda_{B_0} + \sum_{k=1}^j \lambda_{H_k} + \sum_{k=1}^j \lambda_{B_kin} \quad (5.76)$$

En consecuencia, teniendo en cuenta la *propiedad de superposición de los procesos de Poisson* puede afirmarse que la tasa de paquetes que llegan a un nodo con destino el nodo destino D es la suma de las contribuciones de las tasas que llegan a todos los nodos de la ruta principal que están en niveles jerárquicos inferiores. Introduciendo la probabilidad de recuperación de los enlaces inferiores de la ruta, la expresión general para la tasa media de paquetes que genera el nodo del nivel j de la ruta sería la siguiente:

$$\lambda_j^* = \lambda_j + \sum_{i=1}^{j-1} \lambda_i \cdot \prod_{z=i}^{j-1} P_{recz} \quad (5.77)$$

siendo $P_{recj} \approx \lambda_j / (\lambda_j + \lambda_D)$. Para $j = 1$ (primer salto) se tiene $\lambda_1^* = \lambda_1$.

Considerando este resultado, se puede concluir que la probabilidad de recuperación de un enlace, que depende de la tasa de paquetes que se envíen a través de él tal y como puede deducirse de la Ecuación (5.73) de la sección anterior, será mayor cuanto menor sea el nivel del nodo de la ruta principal involucrado, es decir, cuanto más cerca esté del nodo D:

$$\lambda_{B_1} \leq \lambda_{B_2} \leq \lambda_{B_3} \leq \dots \implies P_{rec}(\overline{B_0B_1}) \leq P_{rec}(\overline{B_1B_2}) \leq P_{rec}(\overline{B_2B_3}) \leq \dots \quad (5.78)$$

5.1.5.4. Cálculo final del tiempo de vida de una ruta con recuperación local de errores

Tal y como se ha explicado anteriormente, el proceso de RL lleva implícito la búsqueda de una ruta alternativa al enlace roto, lo que conlleva que el tiempo de vida de una ruta entre los nodos D y A aumente, estando determinada su nueva duración por el instante en que se produzca el siguiente fallo en cualquiera de los enlaces de la ruta siempre que éste no pueda ser reparado.

El siguiente y último paso, una vez conocida la probabilidad de recuperación local del enlace, P_{rec} , será por tanto calcular en términos probabilísticos cuánto aumenta el tiempo de vida de las rutas entre D y A con respecto al tiempo original en estas circunstancias. Para analizar este caso, se tomará como referencia la configuración de nodos y niveles indicada en la Figura 5.23, en la que en un momento determinado se produce un fallo en uno de los enlaces de la ruta entre D y A.

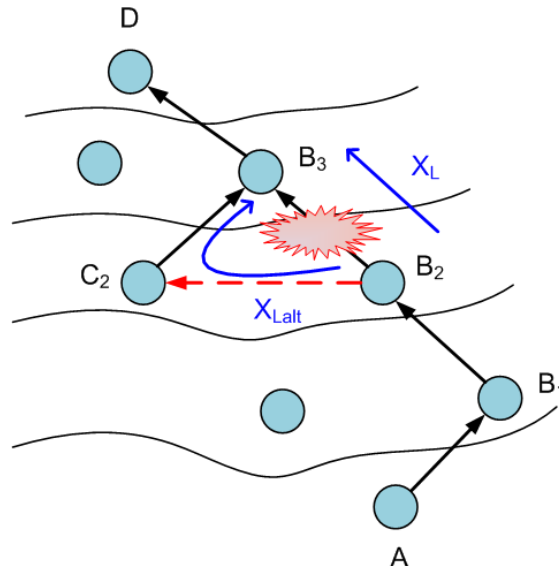


Figura 5.23: Ruta con un fallo en uno de sus enlaces.

Cuando el enlace entre B_2 y B_3 de la Figura 5.23 se rompe, asumiendo la existencia del nodo C_2 , se creará la ruta alternativa $B_2-C_2-B_3$ cuyo $E[X_{L_{alt}}] = E[X_L]/2$ [98], [113]. En este caso la *cdf* del tiempo de vida de las rutas entre B_2 y B_3 vendrá dada por la expresión:

$$F_{X_L}^*(t) = P[\max(X_L, X_{L_{alt}}) \leq t] = F_{X_L}(t) \cdot F_{X_{L_{alt}}}(t) \quad (5.79)$$

Generalizando para una ruta con k enlaces y suponiendo que cada uno de ellos tiene un nodo alternativo C_i con $i = 1, \dots, k$ para completar el proceso de recuperación, puede calcularse su $P_{rec,i}$, resultando la Ecuación (5.79) de la siguiente forma:

$$F_{X_{L_i}}^* = F_{X_L} \cdot (1 - P_{rec,i}) + F_{X_L} \cdot F_{X_{L_{alt}}} \cdot P_{rec,i} \quad (5.80)$$

Finalmente, la *cdf* del tiempo de vida de las rutas entre A y D se calcula aplicando esta última ecuación a la siguiente expresión:

$$\begin{aligned} F_{X_p}(t) &= P(X_p \leq t) = P[\min(X_{L_1}^*, \dots, X_{L_k}^*) \leq t] = \\ &= 1 - [(1 - F_{X_{L_1}}^*(t)) \cdot \dots \cdot (1 - F_{X_{L_k}}^*(t))] = \\ &= 1 - \prod_{z=1}^j (1 - F_{X_{L_i}}^*(t)) \end{aligned} \quad (5.81)$$

Para comparar cuánto aumenta el tiempo de vida de una ruta cuando existe la recuperación local de rutas propuesta en esta Tesis se han realizado las simulaciones teóricas de la Figura (5.24). En ellas se analiza cómo afecta el proceso de RL al tiempo de vida de las rutas con respecto al caso de que no exista tal mecanismo. Considerando flujos *upstream* en los cuales el nodo D no envía paquetes, se tiene que $P_{rec} = 1$. En estas gráficas se puede apreciar que la *pdf* en caso de RL para

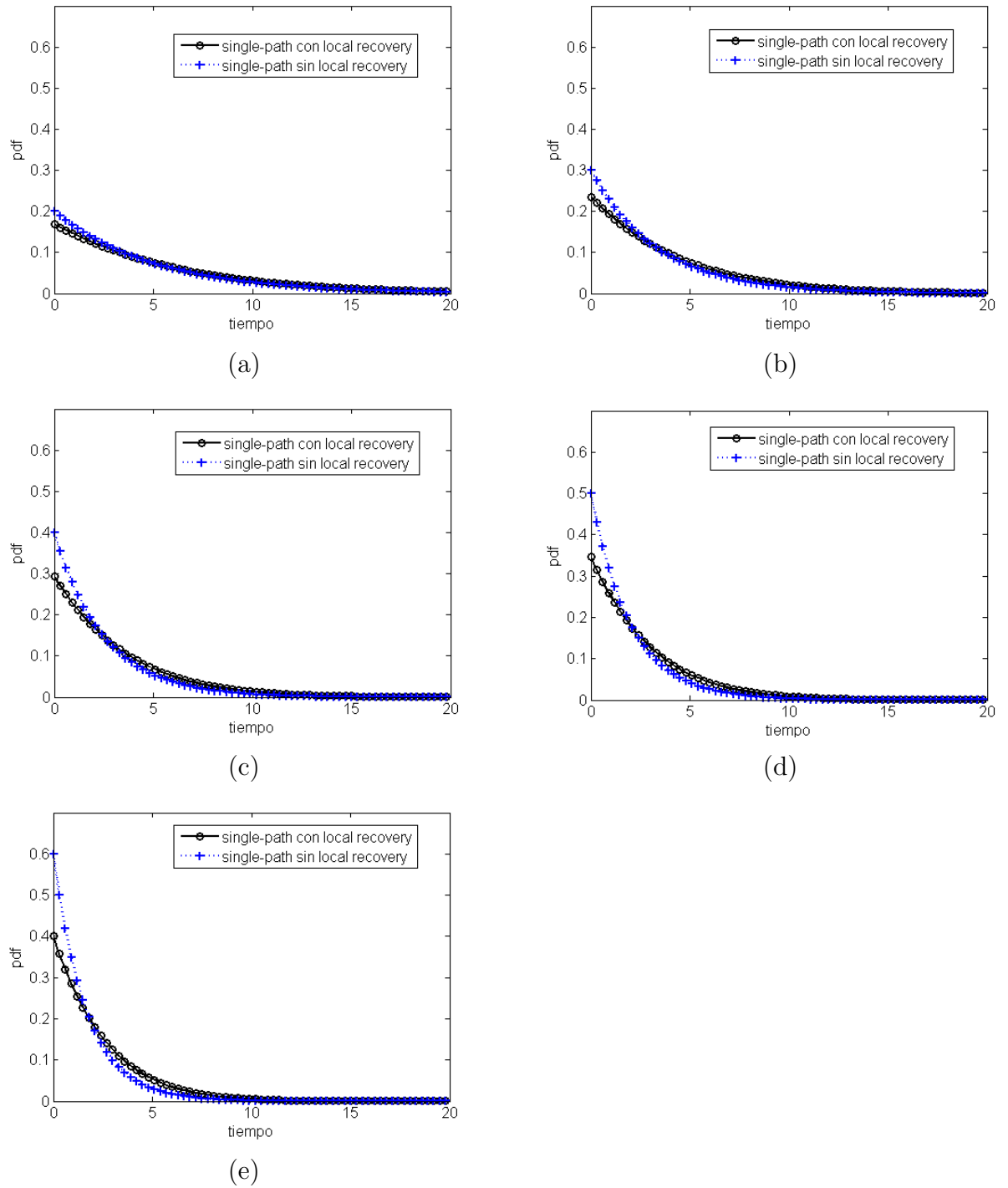


Figura 5.24: Comparación de las pdf de una configuración single-path con y sin utilizar el mecanismo de recuperación local de errores para $l = 10$ s. a) $k = 2$ saltos; b) $k = 3$ saltos; c) $k = 4$ saltos; d) $k = 5$ saltos; e) $k = 6$ saltos.

cualquier longitud de la ruta está desplazada hacia valores de tiempo crecientes, lo que se traduce en la existencia de una mayor probabilidad de que la duración de las rutas sea mayor con respecto al caso de no existencia de RL.

En cuanto a la duración media de las rutas para este valor de P_{rec} , se han realizado las simulaciones teóricas de la Figura 5.25, en la cual se puede comprobar que el tiempo de vida medio de una ruta aumenta con los procesos de recuperación local para cualquier valor de longitud de ruta.

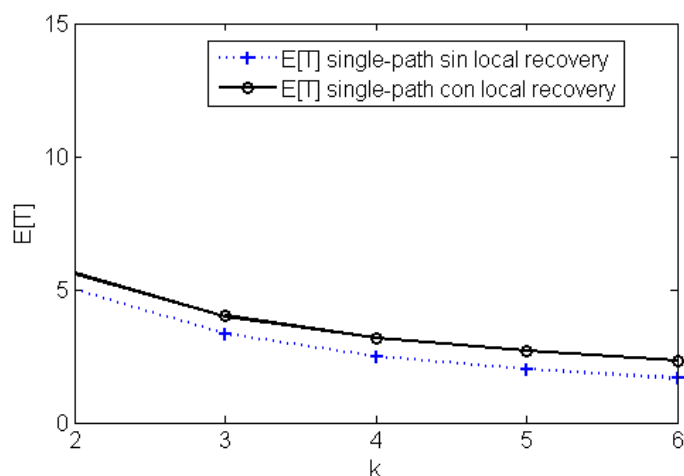


Figura 5.25: Tiempo medio de vida de una ruta en función de su número de enlaces k : comparación con y sin RL en enlaces para $l = 10$ s.

De forma análoga al caso de una sola ruta entre A y D, se han analizado los resultados teóricos de este apartado aplicados al protocolo HARP para demostrar que el tiempo de vida total de las rutas aumenta al utilizar el mecanismo de recuperación propuesto, mejorando en consecuencia las prestaciones del protocolo. La Figura (5.26) muestra que, al igual que en *single-path*, la *pdf* del tiempo de vida de las rutas en HARP está desplazada hacia la derecha con respecto a la *pdf* original sin RL, para las tres las configuraciones de rutas estudiadas.

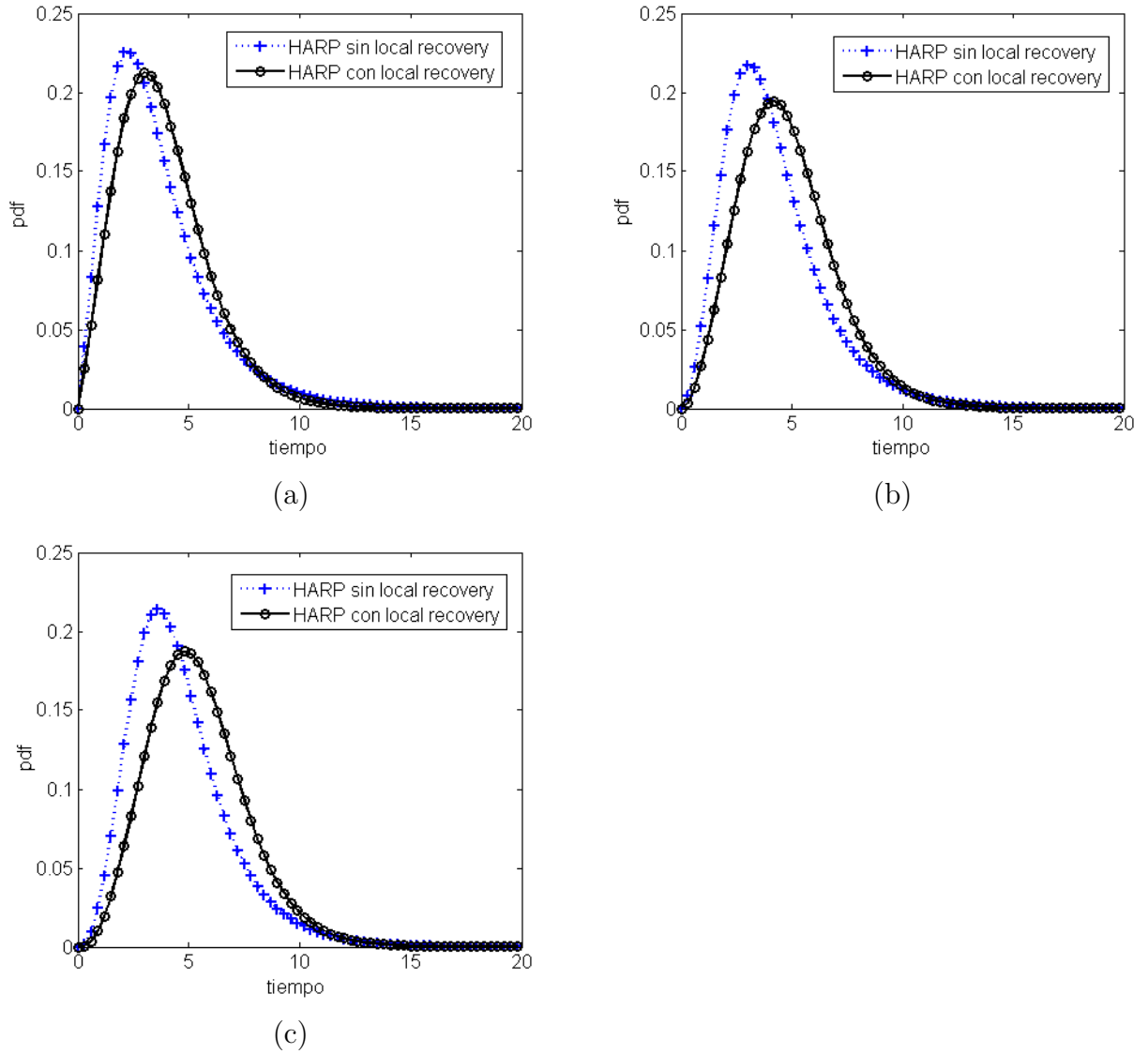


Figura 5.26: Comparación de las funciones densidad de probabilidad de HARP con y sin recuperación local para $l=10s.$ y $k=6$ saltos a) $m1=m2=m3=m4=m5=1$; b) $m1=m2=m3=m4=m5=2$; c) $m1=m2=m3=m4=m5=3$.

Para finalizar este apartado se muestra en la figura 5.27 la variación del tiempo medio de vida de las rutas en función del número de saltos de la ruta principal para *single-path* y HARP. En ella puede observarse la diferencia del tiempo medio de vida de rutas en dos aplicaciones con distinta P_{rec} y con respecto al caso de que no exista recuperación local: 1) aplicaciones *data gathering* con tráfico exclusivamente *upstream* ($\lambda_{Root} = 0, P_{rec} = 1$); 2) *data gathering* y *data dissemination* con tráfico

tanto *upstream* como *downstream*, con $\lambda_j = 10$, $j = 1, \dots, k$, $\lambda_{Root} = \lambda_j/2$ ($P_{rec} < 1$).

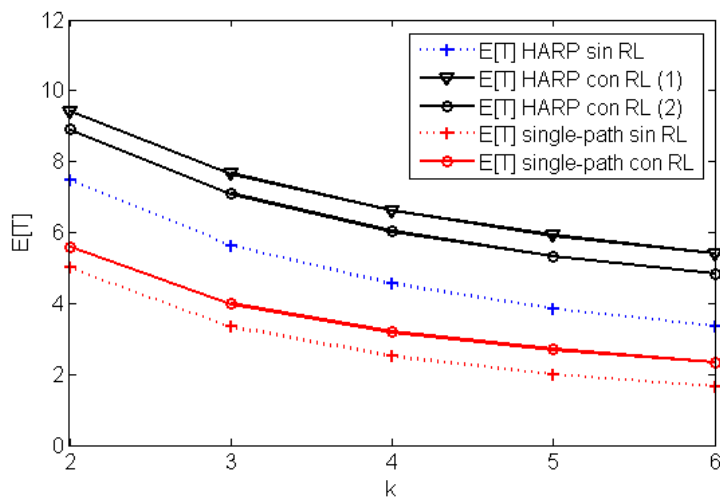


Figura 5.27: Comparativa del aumento del tiempo medio de vida de una ruta en función de la aplicación: (1) tráfico upstream con $P_{rec} = 1$; (2) tráfico upstream y downstream con $\lambda_j = 10$, $j = 1, \dots, k$, $\lambda_{Root} = \lambda_j/2$ ($P_{rec} < 1$).

5.2. Simulaciones

En este último apartado se presentan los resultados de las simulaciones realizadas en el simulador WiNS con tres objetivos fundamentales: (i) validar el modelo teórico desarrollado basado en el cálculo del tiempo de vida de las rutas; (ii) realizar una comparativa preliminar de las prestaciones de HARP respecto de filosofías *single-path* y multi-camino; y (iii) analizar las prestaciones del mecanismo de RL.

La descripción de los diferentes experimentos sigue la estructura de información detallada en la Sección 4.4.

5.2.1. Prestaciones de HARP

5.2.1.1. Descripción de experimentos

Objetivos

1. Validación de los resultados teóricos obtenidos para la operación de los protocolos de encaminamiento.
2. Comparación de prestaciones de HARP frente a filosofías *single-path* y multicamino con rutas disjuntas desde el origen.

VARIABLES ANALIZADAS

La variable analizada será el valor medio del tiempo de vida de las rutas, $E[t]$. Este tiempo de vida se representará para rutas de distinta longitud, medida en número de saltos, k , parámetro que variará entre 2 y 6. El tiempo de vida se limitará introduciendo fallos en los enlaces de la red. Se asume que los fallos en cada enlace de una misma ruta son independientes entre sí y que todos ellos siguen una distribución de Poisson, con *pdf* exponencial de media $l = 10$ s. Para efectuar el cálculo de este parámetro mediante simulación se han seguido los siguientes pasos:

1. Establecimiento de tiempos nulos en la red: los tiempos de procesado, transmisión, propagación, etc. se ponen con valor igual a cero.
2. Se establece un patrón independiente de fallos en los enlaces siguiendo una distribución de Poisson: con ello se define un tiempo de fallo para cada uno de los enlaces de la red.
3. Se generan paquetes con tasa constante (CBR, *Constant Bit Rate*) desde el origen hacia el destino, cuyo valor marcará el error máximo cometido

en la medición del tiempo de vida, ya que se tomará como tiempo de vida de las rutas en cada simulación la del último paquete capaz de alcanzar el destino (el tiempo de vida real estará limitado inferiormente por este tiempo y tendrá como límite superior el tiempo de generación correspondiente al siguiente paquete creado, el cual ya no alcanza el destino).

4. Por último, se repite el experimento un número determinado de veces y se extrae la media estadística de todos ellos.

Parámetros

Se realizaron dos tipos de simulaciones: una primera con una estructura sencilla de 13 nodos en la que se analizó el protocolo *single-path*; y una segunda con 25 nodos en las que se estudiaron los protocolos multi-camino con rutas desde el origen y HARP. El protocolo multi-camino se diseña con $N = 3$ rutas disjuntas desde el origen hasta el destino, siguiendo el esquema mostrado en la Figura 5.4. Por su parte, HARP se configura con $m = 1$ ruta disjunta desde cada nodo de la ruta principal según la disposición que puede observarse en la Figura 5.9. Se realizan un total de 1000 simulaciones en WiNS de cada uno de los protocolos analizados mediante la generación de paquetes con tasa constante de 100 paquetes por segundo, lo que establece un error de medida del tiempo de vida de 0,01 segundos. Las Tablas 5.2 y 5.3 muestran los parámetros seleccionados.

Parámetro	Valor
Número de nodos sensores	13
Distribución de los sensores en la red	predefinida
Protocolo 1	<i>single-path</i>

Tabla 5.2: Parámetros de simulación de las prestaciones en *single-path*.

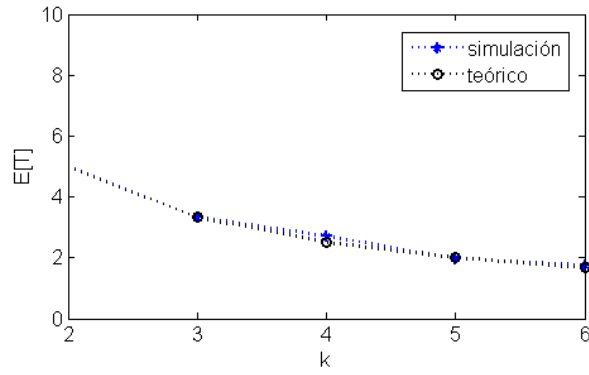
Parámetro	Valor
Número de nodos sensores	25
Distribución de los sensores en la red	predefinida
Protocolo 1	Multi-camino ($N = 3$)
Protocolo 2	HARP ($m = 1$)

Tabla 5.3: Parámetros de simulación de las prestaciones en HARP y multi-camino con rutas disjuntas desde el origen.

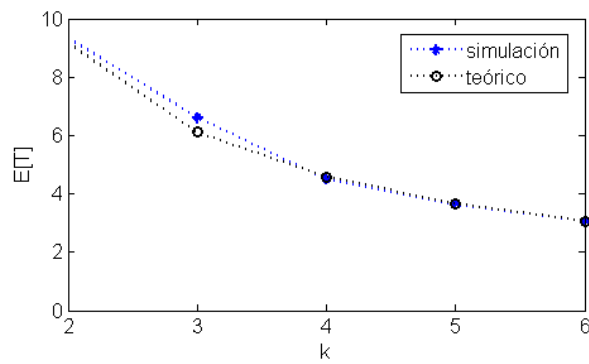
5.2.1.2. Resultados

La Figura 5.28 muestra los tiempos medios de vida de las rutas obtenidos mediante simulación junto con los obtenidos teóricamente utilizando la herramienta Matlab para cada una de las configuraciones indicadas anteriormente. A la vista de los mismos, se puede concluir que:

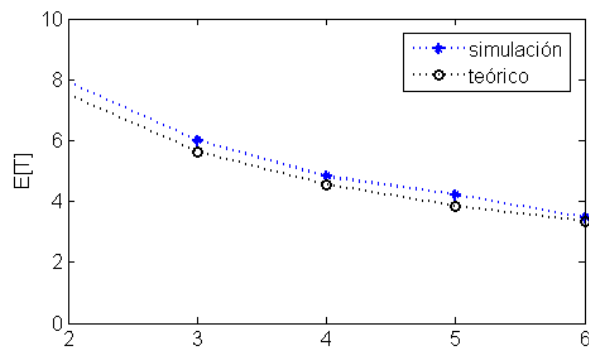
- Las simulaciones con WiNS representan fielmente los resultados teóricos obtenidos. Con respecto a HARP, el resultado teórico representado es el correspondiente a la Ecuación (5.21) que, como ya se indicó, es una aproximación y además tiene en cuenta únicamente rutas disjuntas, con lo esta gráfica teórica establece un límite inferior en las prestaciones de HARP. Esta es la explicación a la ligera diferencia de valores entre la curva teórica y la simulada en WiNS.
- El protocolo *single-path* ofrece unas prestaciones muy bajas en cuanto al tiempo de vida de sus rutas. Por su parte, HARP mejora el tiempo de vida de las rutas ofrecido por multi-camino a partir de rutas con más de 4 saltos.



(a)



(b)



(c)

Figura 5.28: Valor medio del tiempo de vida de las rutas teórico y simulado para $l = 10$ s.
 a) Single-path; b) Multi-camino ($N = 3$); c) HARP ($m = 1$).

5.2.2. Prestaciones de Recuperación Local

Este experimento pretende analizar las prestaciones del mecanismo de Recuperación Local en cuanto a su capacidad para extender el tiempo de vida de las rutas.

5.2.2.1. Descripción de experimentos

Objetivos

1. Validación de los resultados teóricos de las prestaciones del mecanismo de RL.
2. Análisis de la mejora de prestaciones por la inclusión del algoritmo de RL.

VARIABLES ANALIZADAS

La variable analizada será de nuevo el valor medio del tiempo de vida de las rutas, $E[t]$, representado para distintos valores de k variando entre 2 y 6. El tiempo de vida se limitará introduciendo fallos en los enlaces de la red y el proceso de cálculo será el mismo que el indicado en el caso anterior, excepto por una única variación: en este caso será el nodo con mayor nivel el encargado de generar los paquetes, de tal forma que se cree un flujo *upstream* necesario para que se pueda iniciar el proceso de RL cuando falla la ruta.

Parámetros

El mecanismo de RL definido en esta Tesis se aplica tanto al protocolo *single-path* como a HARP para estudiar la posible mejora de las prestaciones en ambos casos. La Tabla 5.5 muestra los parámetros utilizados en las simulaciones.

Parámetro	Valor
Número de nodos sensores	13
Distribución de los sensores en la red	predefinida
Protocolo	<i>Single-path</i> + RL

Tabla 5.4: Parámetros de simulación de las prestaciones de RL en *single-path*.

Parámetro	Valor
Número de nodos sensores	25
Distribución de los sensores en la red	predefinida
Protocolo	HARP + RL

Tabla 5.5: Parámetros de simulación de las prestaciones de RL en HARP.

5.2.2.2. Resultados

Los resultados obtenidos mediante simulación se comparan asimismo con los teóricos calculados previamente para reforzar la validación de éstos. En el caso del mecanismo de RL integrado en HARP, esta comparación se realiza frente a los resultados teóricos generados: (i) de forma *aproximada*, aplicando el cálculo de la Ecuación (5.21), la cual constituye un límite inferior de las prestaciones del protocolo); y (ii) de forma *exacta*, calculando la *cdf* de la configuración de red mediante la técnica de factorización utilizada para el cálculo de la fiabilidad de una red [37], [122], [109].

Los resultados de la incorporación del mecanismo de RL en *single-path* y HARP pueden observarse en las Figuras 5.29 y 5.30 respectivamente.

Los resultados obtenidos indican que:

- Las simulaciones con WiNS representan fielmente los resultados teóricos obtenidos en todos los casos considerados.
- La aproximación de utilizar la Ecuación (5.21) es bastante buena si se compara con el caso exacto, tanto con RL como sin RL.

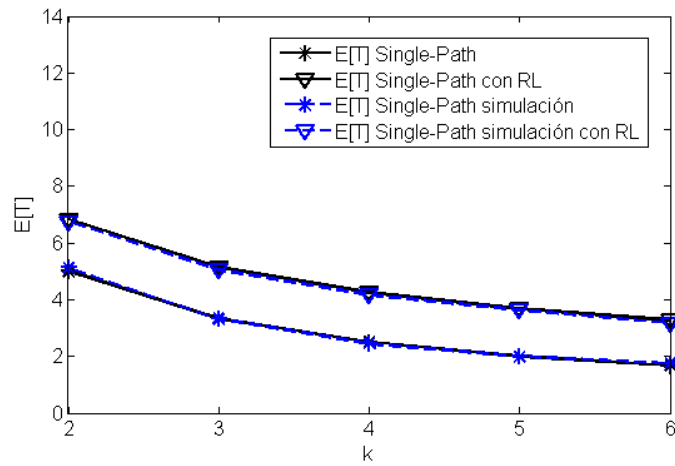


Figura 5.29: Tiempos de vida de rutas en protocolo single-path con y sin mecanismo de RL (teórico y simulado).

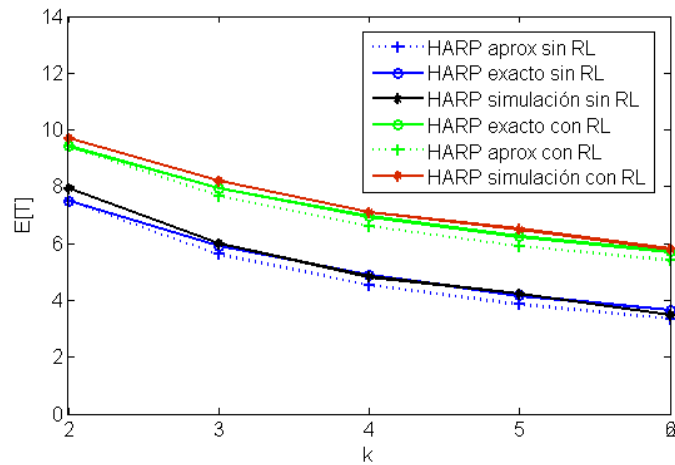


Figura 5.30: Tiempos de vida de rutas en HARP con y sin mecanismo de RL (teóricos y simulado).

- El mecanismo de RL mejora significativamente el tiempo de vida de las rutas, independientemente de su longitud, en los dos protocolos analizados (valores cercanos a un 25 % de mejora).

Capítulo 6

HARP en WSN

6.1. Introducción

En este capítulo se describe la propuesta de arquitectura del protocolo HARP para WSN y en cuyo diseño se han seguido los criterios y requerimientos indicados en la Sección 2.2 de esta Tesis. En concreto, algunas de las **características de HARP que son deseables en un protocolo de encaminamiento en WSN** son las siguientes:

- Arquitectura dinámica basada en *clusters* con rotación periódica de roles en la red con objeto de repartir el gasto energético entre todos los nodos y extender el tiempo de vida útil de la misma.
- Creación de estructuras multi-salto que optimizan el consumo energético en las transmisiones.
- Planificación efectiva de las comunicaciones que permite la disminución de las colisiones y las transiciones de los nodos a estados de bajo consumo de energía (modo *sleep*).

- Estructura robusta de tablas de encaminamiento que permite la comunicación entre cualquier par de nodos de la red y cualquier tipo de tráfico cursado.
- Tolerancia a fallos en la red y soporte para rutas alternativas y balanceo de carga.
- Posibilidad de utilizar fusión o agregación de la información monitorizada en las transmisiones hacia el nodo sumidero con el objetivo de minimizar aquéllas que sean innecesarias.
- Gestión eficiente de movilidad de nodos.

6.2. Arquitectura del protocolo

Los avances recientes en tecnología para sensores inalámbricos han hecho posible la disponibilidad de estos dispositivos a un coste relativamente pequeño y con capacidad de muy bajo consumo energético. Como se ha mencionado en los capítulos iniciales de esta Tesis, las WSN pueden contener cientos o incluso miles de estos nodos sensores que pueden estar distribuidos sobre grandes superficies, lo que requiere que los nodos sean capaces de comunicarse unos con otros para transmitir los datos recogidos del entorno en ausencia de infraestructura de red. Al estar limitada esta capacidad energética de los dispositivos sensores, se hace necesario el diseño de mecanismos eficientes de comunicación y encaminamiento de la información con objeto de minimizar el consumo energético de la red y maximizar por tanto el tiempo de vida de los elementos del sistema [64].

Debido a la circunstancia de que las WSN pueden extenderse a grandes superficies para recoger y medir diversos parámetros del entorno, existirá a priori una gran cantidad de datos que deben ser enviados al usuario o a la aplicación final

para su correspondiente procesado. En consecuencia, la utilización de técnicas automatizadas para combinar, correlar y agregar esos datos en unidades mínimas de información se hace indispensable para incrementar la eficiencia energética en la red y la efectividad de las comunicaciones en la misma [61], [77]. Debido a que los datos leídos por un nodo pueden estar altamente correlados con los recogidos por sus nodos vecinos, la agregación o fusión de datos, que depende de la aplicación específica para la que ha sido diseñada la WSN, es capaz de combinar diferentes técnicas que consiguen evitar la sobrecarga de la red con información redundante o medidas poco significativas.

En esta sección se incluye una descripción del modelo de red utilizado para calcular el gasto energético de una comunicación entre dos nodos de la red y que servirá como punto de partida para la definición y evaluación de los mecanismos de construcción de árboles jerárquicos en HARP. A continuación se describirán los distintos tipos de nodos que se pueden encontrar en una WSN. La estructura de mensajes completa que se utiliza en HARP ha sido ya incluida en la Sección 3.2, por lo que se remitirá a la misma a la hora de describir los diferentes procesos que tienen lugar en la configuración y operación del protocolo propuesto. Para finalizar, se identificarán los diferentes roles y estructuras que están presentes en HARP y que constituyen la arquitectura general del protocolo.

6.2.1. Modelo energético

Para llevar a cabo la comunicación con otros nodos, los sensores utilizan paquetes de datos y de control de una longitud en bytes determinada, lo que lleva asociado un gasto energético tanto en la transmisión como en la recepción de los mismos. Como se ha mencionado en los párrafos de introducción a este capítulo, la fusión o agregación de la información recibida de otros nodos en un solo paquete de datos también debe considerarse dentro del cálculo del balance energético

de las comunicaciones en la red. Aunque el resto de actividades del nodo sensor, como el procesado de un paquete, la actualización de tablas de encaminamiento, la monitorización del entorno, permanecer en modo de escucha del canal o pasar a estado *sleep*, también llevan asociado un gasto energético, las contribuciones de estos tres términos (transmisión, recepción y agregación) al gasto total del nodo son muy elevadas [91], [105].

En general, el circuito de transmisión/recepción de un dispositivo sensor consta de los siguientes tres estados:

- *Modo sleep*: el dispositivo apaga su radio y permanece en modo de bajo consumo.
- *Modo idle*: se enciende la etapa radio y permanece a la espera para transmitir o recibir (escucha).
- *Modo transmisión/recepción*: el dispositivo transmite o recibe una secuencia de bits a través de su interfaz radio.

Para poder calcular la energía gastada en la comunicación de un paquete de información entre dos nodos se adoptará en esta Tesis el modelo simplificado de disipación de energía propuesto en [64] y cuya utilización está generalizada en la literatura científica relativa al encaminamiento en WSN ([35], [129], [128], [84], [149], [143], [36], [70], [86], [130], [40], [24], [119], [127], [142], [72], [151]). Este modelo tiene en cuenta las características del canal inalámbrico a la hora de calcular el balance energético de un enlace. A la vista de la Figura 6.1, el gasto energético del circuito radio de transmisión asociado a la transferencia de un paquete de k bits entre dos nodos separados una distancia d se denota por E_{TX} y tiene la forma siguiente:

$$E_{TX}(k, d) = k \cdot E_{elect} + k \cdot \epsilon \cdot d^\beta \quad (6.1)$$

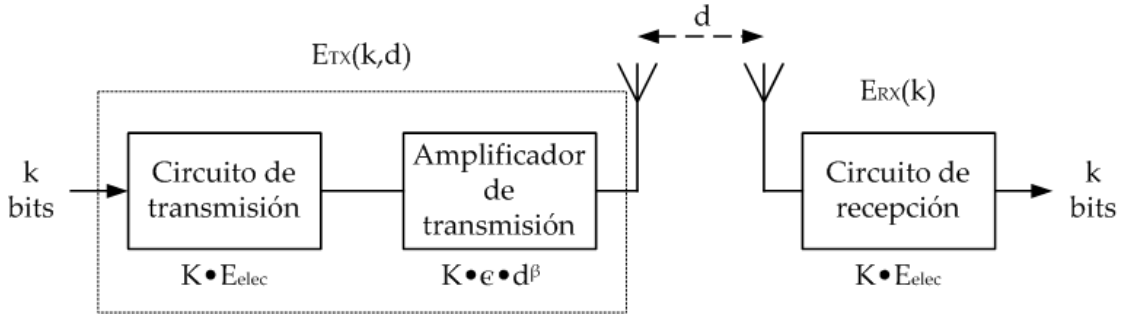


Figura 6.1: Modelo utilizado para el consumo energético en la transmisión de un paquete de k bits entre dos nodos separados una distancia d .

siendo E_{elect} la energía necesaria para enviar un bit del paquete desde el circuito transmisor al amplificador de transmisión, ϵ el valor de la constante de amplificación que consigue un valor requerido de E_b/N_0 y de SNR en recepción, d la distancia de transmisión expresada en metros y β el exponente de propagación. Este exponente es función de las condiciones del canal inalámbrico y depende de la distancia de transmisión.

El modelo simplificado utilizado asume dos tipos de efectos de canal: para distancias r cercanas al nodo transmisor y menores que un umbral predeterminado el modelo utiliza la atenuación en espacio libre o modelo *Friss-Space*, que asume una dependencia de las pérdidas como $1/r^2$. Sin embargo, para distancias grandes al nodo transmisor se tienen en cuenta los efectos producidos por la propagación multi-camino y se utiliza una aproximación del modelo de 2 rayos o *Two-ray Ground Reflection Model*, con una dependencia $1/r^4$, asumiendo reflexión especular en un plano horizontal. En consecuencia, la Ecuación (6.1) puede reescribirse en función de estos dos modelos de canal de la forma:

$$E_{TX}(k, d) = \begin{cases} k \cdot E_{elect} + k \cdot \epsilon_{fs} \cdot d^2, & d < d_0 \\ k \cdot E_{elect} + k \cdot \epsilon_{mp} \cdot d^4, & d \geq d_0 \end{cases} \quad (6.2)$$

donde ϵ_{fs} y ϵ_{mp} son la particularización de ϵ para el caso de espacio libre (*fs*, *free space*) y multi-camino (*mp*, *multi-path*), siendo $d_0 = \sqrt{\epsilon_{fs}/\epsilon_{mp}}$. El primer término de cada igualdad representa la energía consumida en el circuito de transmisión y el segundo indica la energía consumida en el proceso de amplificación y transmisión de la señal.

Por su parte, la energía consumida por el circuito receptor en la recepción de un paquete de k bits, denotada como E_{RX} , puede escribirse como:

$$E_{RX}(k) = k \cdot E_{elect} \quad (6.3)$$

Finalmente, los nodos pueden agregar o fusionar los datos recibidos de un nodo vecino con el suyo propio (*data fusion*), lo que llevará asociado un consumo energético adicional. Este consumo puede ser apreciable si los datos recogidos por los nodos tienen una alta correlación y se hace necesario comprimir la información recibida en un solo paquete de datos. En consecuencia, siendo E_{fuse} la energía necesaria para agregar un bit de datos del paquete recibido, se puede escribir la energía consumida en la agregación de un paquete de k bits de forma inmediata, mediante la expresión:

$$E_{AG}(k) = k \cdot E_{fuse} \quad (6.4)$$

6.2.2. Tipos de nodos

HARP construye su estructura jerárquica en función de dos tipos de nodos dentro de la red de sensores: nodo sensor y nodo sumidero.

6.2.2.1. Nodo sensor

Tal y como se ha indicado anteriormente en esta Tesis, un nodo sensor es un dispositivo con capacidades limitadas en cuanto a la duración de la batería, la capacidad de procesamiento de información, la memoria y los recursos de computación. La Figura 6.2 muestra **los subsistemas que componen un nodo sensor** y que se describen a continuación:

- *Subsistema de procesamiento*: con el microcontrolador como elemento principal, este subsistema se ocupa de la ejecución de las líneas de código, programas o aplicaciones, de la gestión de recursos (memoria, baterías, etc.) y de la interacción con los distintos periféricos y con la placa donde se encuentran los dispositivos sensores.
- *Subsistema de sensores*: incorpora los distintos sensores de medida: luz, humedad, temperatura, aceleración, sonido, etc.
- *Subsistema de memoria*: se compone de una memoria principal en la cual se almacenan los datos necesarios para la ejecución de los distintos algoritmos, y una memoria secundaria para el almacenamiento de los datos físicos recogidos del entorno en las operaciones de monitorización. Esta memoria secundaria puede también almacenar temporalmente las imágenes o distintas versiones de programas o firmware que se distribuyen a través de la red.
- *Baterías*: sistema de alimentación autónomo y limitado que regula la tensión que proporciona a los demás subsistemas.
- *Transceptor RF*: es el encargado de la transmisión y recepción a través del medio inalámbrico de los paquetes de comunicaciones entre los distintos nodos de la red. Incluye todos los aspectos necesarios para el envío de la información: modulación/demodulación, codificación, etc.

Los nodos sensores poseen capacidades autónomas tanto para reencaminar un paquete como para fusionar su información con los paquetes propios, comprimiéndola en uno solo.

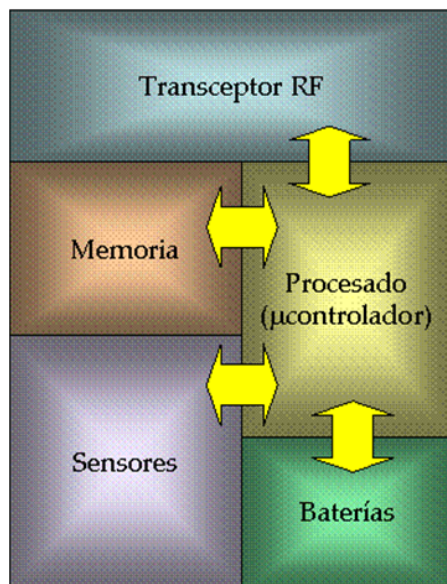


Figura 6.2: Subsistemas que componen un nodo sensor, con el microprocesador como unidad principal.

6.2.2.2. Nodo sumidero

El nodo sumidero, también denominado fuente o *sink*, es la frontera entre la red de nodos sensores y el mundo exterior. Su misión es la de hacer de pasarela entre los dos sistemas, recogiendo, agregando, procesando y transmitiendo la información procedente de la red hacia la aplicación de destino, así como tomando las acciones necesarias para la reconfiguración, recuperación o envío de información a lo largo de la red de sensores.

El nodo sumidero consta de uno o varios microcontroladores o procesadores, una o varias memorias, un sistema de alimentación generalmente a través de red eléctrica o paneles solares que doten al equipo de capacidad energética continua

y sin restricciones de capacidad, y un sistema de comunicaciones. Este último sistema consta de transceptor radio para la comunicación con los nodos de la red y opcionalmente con los sistemas externos cuando esta comunicación no sea a través de un medio cableado, o bien una tarjeta de red para el caso de que sí lo sea. Este sistema de comunicaciones debe proporcionar al nodo sumidero la capacidad de comunicarse con cualquier nodo de la red, así como poder alcanzar grandes distancias de comunicación si el sistema gestor está alejado del mismo. Las tecnologías que pueden adaptarse para este tipo de nodo varían desde Ethernet para LAN hasta WiFi, WiMAX o UMTS para redes de larga distancia.

HARP está diseñado para su utilización en WSN con uno o varios nodos sumidero.

6.2.3. Arquitectura propuesta

La estructura propuesta por HARP para WSN se basa en la organización de la red en *clusters* y árboles jerárquicos. El objetivo de esta estructura es permitir mecanismos de encaminamiento energéticamente eficientes que se adapten a las circunstancias cambiantes de la red y permitan su escalabilidad sin incurrir en una degradación significativa de las prestaciones del mismo.

La estructura de *clusters* creada debe ser dinámica con objeto de dividir el gasto energético entre todos los nodos de la red, como consecuencia del mayor gasto energético realizado por los CH para establecer, controlar y recoger información, tal y como se describirá en los apartados siguientes.

HARP propone **tres tipos de roles en función de su operación** en el funcionamiento del protocolo de la red:

- *Nodo sumidero o sink.*

- *Nodo CH* o *Cluster-Head*: es el nodo encargado de la creación del *cluster* y de organizar a los nodos normales más cercanos para la transmisión eficiente de los datos recogidos dentro del mismo. Para aplicaciones de diseminación de datos o consultas, es el encargado de encaminar los paquetes a los nodos que se encuentran adscritos a su *cluster*.
- *Nodo nonCH* o *nodo normal*: es el nodo que no actúa como CH y cuya misión es la de recoger los datos del entorno e interactuar con sus nodos vecinos en función de la organización dentro del *cluster*, bien para enviar los paquetes de datos con destino al CH o bien para encaminar los paquetes hacia otro nodo vecino.

El protocolo propone la creación de **tres tipos diferentes de estructuras jerárquicas**:

- *Árbol inter-sink*: cuando la WSN se compone de varios nodos sumidero debido tanto a su extensión como a razones meramente operativas como puede ser la redundancia en la red, puede construirse una estructura jerárquica en la que uno de los nodos sumidero es seleccionado en función de un criterio específico y actúa como nodo principal, situándose en la cabecera del árbol como nodo raíz. El resto de nodos se organiza por niveles jerárquicos en función de la distancia al nodo raíz y de las condiciones inalámbricas de cada enlace con sus nodos vecinos.
- *Árbol inter-cluster*: estructura jerárquica que tiene como nodo raíz del árbol a un nodo sumidero y como nodos del árbol a los CH que asumen las funciones de liderazgo dentro de cada *cluster*.
- *Árbol intra-cluster*: árbol creado dentro de cada *cluster* en el cual el CH ocupa la posición de nodo raíz y los nodos normales del *cluster* ocupan el resto de posiciones dentro de la estructura jerárquica del mismo.

En la Figura 6.3 se muestra un ejemplo de la arquitectura propuesta en HARP, con las tres estructuras jerárquicas definidas anteriormente.

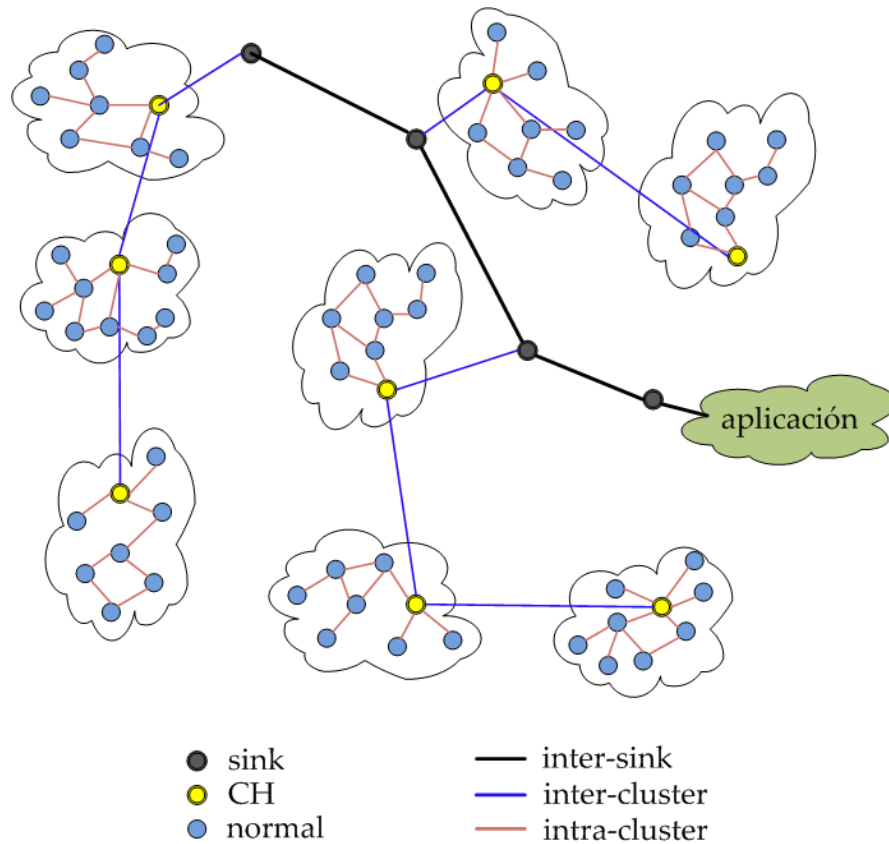


Figura 6.3: Arquitectura HARP: ejemplo de construcción de los tres tipos de árboles jerárquicos

Si bien el árbol *inter-sink* es parte de la estructura de la red de sensores, los nodos que lo forman no tienen las restricciones impuestas por este tipo de dispositivos y entre ellos puede crearse una red ad hoc que encamine los paquetes en función de cada aplicación (centralizada, redundante, etc.). Por este motivo, esta Tesis doctoral se centrará en los otros dos tipos de estructuras jerárquicas, si bien lo indicado para ambas puede aplicarse también al árbol creado entre los nodos sumidero.

6.3. Fases de operación del protocolo

HARP es un protocolo de encaminamiento adaptativo y dinámico que tiene como objetivo la creación de estructuras jerárquicas que minimicen el consumo energético de los nodos de la red. En HARP los nodos se organizan en *clusters* en torno a un nodo que actúa como nodo líder o CH. Como se ha indicado en el apartado anterior, el CH tiene la responsabilidad de organizar todas las comunicaciones del *cluster*, recolectando y agregando en su caso la información recogida por todos los nodos asociados al mismo y enviándola a otros nodos CH vecinos para que llegue al nodo sumidero. A su vez, el CH recibirá otros paquetes de datos procedentes de CH vecinos y será el encargado de gestionar las entregas de paquetes a los nodos normales dentro de su *cluster* en el caso de aplicaciones de diseminación de datos o consultas a los nodos sensores.

En el proceso de creación y organización de los *clusters*, que se describirá con detalle a lo largo de este apartado, es evidente que el CH gastará más energía que cualquier otro nodo, ya sea por el mayor número de paquetes de control que intercambia con los nodos, como por el gasto adicional que supone la agregación de los datos recibidos por todos los nodos del cluster y por las comunicaciones a través del árbol inter-*cluster* creado. En consecuencia, para que estas estructuras jerárquicas sean eficientes y el consumo energético se distribuya en mayor o menor medida entre todos los nodos de la red, es necesario que la funcionalidad de CH vaya rotando entre diferentes nodos atendiendo a criterios específicos. En la revisión del estado del arte incluida en el Capítulo 2 se hacía referencia a multitud de trabajos de investigación que proponen métodos específicos para calcular esta distribución de CH en función de parámetros tales como energía disponible, distancia al nodo sumidero, número deseado de CH en la red, etc. En la Sección 6.6 de este capítulo se propondrá en s-HARP un método específico para la selección de CH.

Según este criterio, es necesario definir un intervalo temporal en el cual, tras la selección inicial del conjunto de nodos que actuarán como CH y la creación de las estructuras jerárquicas definidas por HARP, la transmisión de información entre los nodos de la red sea estable, sin que haya modificaciones en el rol asumido por los nodos ni en las estructuras jerárquicas a menos que sean las causadas por fallos en la red. Una vez transcurrido ese intervalo, se procederá a la renovación de los nodos con rol de CH y al inicio de un nuevo intervalo estable de transmisión. A dicho intervalo se le denomina *ronda de operación* o *round*.

Cada ronda de operación se divide a su vez en varias etapas, las cuales se muestran en la Figura 6.4. Se ha querido respetar la nomenclatura en idioma inglés por ser ampliamente utilizada en la bibliografía relacionada con este tema. En los apartados siguientes se describen detalladamente cada una de ellas.

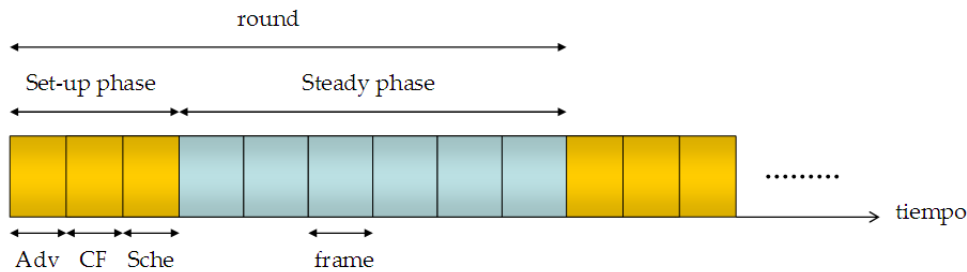


Figura 6.4: Etapas que constituyen una ronda de operación o round en HARP.

En el caso de que existan diferentes nodos sumidero, la estructura jerárquica creada entre ellos no está sujeta a restricciones energéticas por lo que no es necesaria su modificación al final de cada ronda.

6.3.1. Fase de inicialización o *Set-up phase*

Al principio de cada ronda se inicia la formación de los *clusters* y la construcción de los árboles jerárquicos dentro de cada uno de ellos (*intra-cluster*), así como el correspondiente a la estructura *inter-cluster* entre los CH seleccionados. Esta fase

se divide a su vez en tres fases diferentes: fase de elección y notificación de CH, fase de formación de los *clusters* y fase de planificación.

6.3.1.1. Fase de elección y notificación de rol o *Advertisement phase* (Adv)

Al inicio de cada ronda los nodos que se encontraban en modo *sleep* o *idle* despiertan de ese estado y se lleva a cabo el proceso de selección de los nodos que actuarán con rol de CH a lo largo de toda ella. Existen **dos procedimientos principales para la elección de CH**:

1. *Selección mediante umbral*: los nodos calculan mediante una formulación específica un valor umbral o *threshold* comprendido entre 0 y 1 y que se utilizará como probabilidad de elegirse a sí mismo como CH durante esta ronda. Si este valor es elevado, existirá alta probabilidad de ser elegido, mientras que si el umbral calculado está muy cerca del cero el nodo tendrá pocas posibilidades de ser CH. Los parámetros utilizados para el cálculo del umbral pueden ser tan variados como la distancia al sumidero, la energía restante, un valor de probabilidad predefinido, el número de ronda actual, un valor aleatorio, etc. Este procedimiento elige los CH sin tener en cuenta a los nodos de alrededor. Un ejemplo de protocolo que utiliza este tipo de método de elección es LEACH ([64]), en el que este valor umbral se calcula en función de una probabilidad preestablecida y del número de rondas desde la última en la que el nodo actuó como CH.
2. *Selección mediante comparación de pesos*: cada nodo difunde un valor calculado previamente llamado *peso* y escucha el correspondiente a sus nodos vecinos. La elección o no como CH en cada nodo se realiza comparando su peso con el de los demás nodos.

Además de estos dos métodos, puede aplicarse un procedimiento híbrido entre ambas propuestas. En este caso, el nodo difunde a los nodos de su alrededor algún parámetro propio como la energía restante que almacena y recibe el mismo parámetro de los demás. El cálculo del umbral tendrá ahora además este dato como parámetro, con lo que permitirá al nodo tomar su decisión de ser o no CH teniendo además en cuenta la situación particular y las características de los nodos de su alrededor. Esta metodología híbrida es la que utilizará s-HARP para la elección de los CH cada ronda tal y como se describirá en la Sección 6.6.

HARP define estructuras jerárquicas y de organización de *clusters* que son independientes del método escogido para seleccionar los CH. En consecuencia, puede utilizarse cualquiera de ellos durante esta fase de notificación. Algunos de estos métodos que optimizan el utilizado por LEACH son [107], [138], [50], [127], [147], [148], [102], [74], [47] y [41].

Para finalizar esta fase, los nodos que han sido seleccionados para ser CH durante esta ronda difundirán un mensaje CHADV de notificación que en HARP deberá alcanzar como mínimo al nodo sumidero, aunque este alcance es configurable para cada aplicación. En este mensaje de notificación cada CH incluirá su dirección o ID en la red, su potencia de transmisión y un código CDMA elegido de entre una lista para su utilización en las comunicaciones intra-*cluster* (excepto el seleccionado para el árbol inter-*cluster*) con objeto de evitar interferencias con otros *clusters* cercanos. Como alternativa al código CDMA se pueden utilizar canales de radiofrecuencia diferentes. Para el envío de este paquete de notificación los nodos competirán por el canal utilizando el protocolo de capa MAC CSMA/CA, mediante el cual escuchan el canal antes de transmitir y pueden evitar de esta forma colisiones con otras transmisiones simultáneas.

Por último, el nodo sumidero puede cambiar al principio de esta fase las condiciones de selección de CH y el código CDMA para las comunicaciones entre los

nodos CH a través de la difusión en la red de un paquete CHCON con los nuevos valores a utilizar.

6.3.1.2. Fase de formación de *cluster* o *Cluster formation phase* (CF)

Todos los nodos de la red permanecerán a la escucha de los mensajes de notificación CHADV, incluidos los CH. Estos mensajes tienen dos utilidades: por una parte crear la estructura inter-*cluster* entre nodos CH y, por otra, la formación interna de los *clusters* mediante la asociación de los nodos normales con uno de los CH elegidos. A continuación se describe cada uno de estos dos procesos.

Formación del árbol jerárquico intra-*cluster*

Tras permanecer a la escucha y recibir los mensajes de notificación difundidos por los CH, los nodos que no han sido seleccionados pasarán a asumir el rol de nodos normales o *nonCH* y elegirán a uno de los CH para unirse a su *cluster*. El criterio escogido para esta elección es el de mínimo consumo energético, aunque por las condiciones del canal inalámbrico esta condición no es siempre equivalente a la de mínima distancia. Es decir, el *nonCH* escogerá a un CH comparando las pérdidas de potencia en cada uno de los paquetes CHADV recibidos, que puede calcularse como el valor de la potencia de transmisión (incluido en el propio paquete CHADV) y el valor de la potencia de recepción a través de la RSSI de la señal recibida. Hay que hacer constar que en realidad esta elección no implica necesariamente el mínimo gasto energético para el *nonCH* debido a la asimetría que presentan los enlaces inalámbricos y que hace que las condiciones de propagación y las pérdidas no sean las mismas en los dos sentidos. Para que esta elección fuese real sería necesario que el *nonCH* enviase un paquete de control a cada uno de los CH (o, al menos, a los candidatos a tener las menores pérdidas en el enlace ascendente) y éstos le respondiesen con otro que incluyera el valor de la potencia recibida. De esta forma, el *nonCH* podría estimar con qué nodo CH la asociación

conlleva un menor gasto energético por su parte. Sin embargo, en general esta aproximación de enlace simétrico en esta situación está bastante extendida y con ella es posible ahorrar un número elevado de mensajes en cada fase de inicialización de la red.

Una vez escogido el CH, los *nonCH* accederán al canal mediante el protocolo MAC CSMA/CA y le enviarán un paquete JOINRQ de petición de unión al *cluster* utilizando el código CDMA indicado por el CH en su paquete de notificación. Al incluirse en este paquete la potencia de recepción del paquete CHADV, el CH será capaz de adaptar la potencia de transmisión en función de las pérdidas calculadas en cada uno de los enlaces descendentes con los nodos asociados.

Transcurrido un cierto tiempo de espera y tras la recepción de todos los mensajes JOINRQ, el CH responderá con un único mensaje ASSOCH de confirmación, el cual incluirá un listado de los ID de cada *nonCH* que se dispone a asociar ordenado de menor a mayor según la pérdida de potencia del enlace con cada uno de ellos. Esta lista es la denominada CST (*Child Strength List*) y se complementa con otro listado indicando en el mismo orden el valor de la potencia de la señal recibida de cada uno de ellos, con objeto de que los *nonCH* puedan calcular las pérdidas del enlace en sentido *upstream* y ajustar de esta forma su potencia de transmisión.

El CH tiene la capacidad de limitar el número de *nonCH* que asocia a su *cluster*, con lo que si uno de estos nodos no recibe confirmación por parte de un CH escogerá al siguiente en la lista de candidatos en función de la potencia y la calidad de la señal recibida.

En esta fase los *nonCH* se comunican con el CH pero además deben estar despiertos para escuchar el resto de mensajes de petición de asociación que envían otros *nonCH* al mismo CH. De esta forma cada nodo podrá calcular el valor de RSSI de los paquetes JOINRQ recibidos procedentes de otros *nonCH* y, en conse-

cuencia, cada nodo del *cluster* puede estimar la calidad del enlace con cada uno de ellos y las pérdidas que introduce. Este dato, junto con la lista CST contenida en el paquete ASSOCH, permite decidir a cada nodo del *cluster* si existe algún *nonCH* a través del cual la transmisión de un paquete con destino el CH puede realizarse de forma energéticamente más eficiente que la transmisión directa al CH. Estos *nonCH*, si existe más de uno, serán candidatos a convertirse en nodos padre. En esta comunicación a través de nodos intermedios los *nonCH* deben tener en cuenta el nivel de agregación de datos requerido para las transmisiones de datos a través del árbol intra-*cluster*. En general, para aplicaciones de monitorización del entorno los datos recogidos por los *nonCH* tendrán una gran correlación espacial, por lo que como norma general y a menos que la aplicación imponga otros requisitos, el *nonCH* intermedio agregará el paquete recibido junto con el suyo propio en uno de longitud fija y procederá a su envío al siguiente nivel en el árbol jerárquico.

Una vez seleccionados uno o varios nodos *nonCH* como candidatos a padre, el nodo original enviará por orden (primero al que mejor comportamiento haya detectado en el enlace) un mensaje JOINPAR para asociarse como hijo y recibirá en su caso una confirmación mediante un mensaje JOINREP. La recepción de este paquete enviado por cada candidato a padre, en el cual se incluye la potencia de recepción del paquete JOINPAR, permitirá al *nonCH* calcular el balance energético de todos sus enlaces *upstream*. En función de estos cálculos, el *nonCH* enviará un nuevo paquete JOINPAR con el flag *main_parent* activado al nodo padre que conlleve menores pérdidas para confirmarlo como nodo padre principal.

El proceso de decisión por parte de un nodo candidato a padre en el que considera si asocia o no al nodo como hijo se realiza en función de la energía remanente del mismo, ya que se establece un umbral mínimo que si no es alcanzado no se procederá con la asociación, evitando que el gasto energético extra de la agregación disminuya su ya mermada capacidad energética. En caso de que la

decisión sea positiva, tras este proceso ambos nodos actualizarán sus tabla de rutas tal y como se ha descrito en la Sección 3.3.

El *nonCH* encaminará preferentemente todas las transmisiones a través del padre principal, ya que este enlace permite el mayor ahorro energético de entre todos. Como cada nodo adaptará su potencia de transmisión a las condiciones del canal establecido, este proceso permitirá reducir la energía necesaria para la transmisión de los datos recogidos a lo largo de la red.

En los mensajes JOINRQ y JOINPAR está desactivado por defecto el flag *mandatory*. Sin embargo, en caso de que un *nonCH* sea rechazado en todas sus peticiones de asociación, tanto con los CH como con los *nonCH* padres, puede en último caso activar este flag para forzar a uno de ellos a completar la asociación, siempre que sea como última opción para no quedarse aislado sin nodo padre a quien transmitir sus datos. Por defecto, en HARP se configura de tal forma que la obligatoriedad de aceptar a un *nonCH* se imponga al primer CH seleccionado por él.

Por último, una vez se haya completado el proceso de creación del árbol *intra-cluster*, los nodos hoja del mismo (los que no tienen asociado ningún hijo) generarán un mensaje CHROUTE y lo enviarán a sus nodos padre, el cual se propagará hasta llegar al CH, actualizando las tablas CT de los nodos intermedios con la información correspondiente a sus ramas hijas, tal y como se ha descrito en la Sección 3.3.

El diagrama de flujo que detalla la asociación de un nodo normal a un *cluster* y la creación del árbol *intra-cluster* es el indicado en la Figura 6.5. En este diagrama existe una peculiaridad que lo diferencia del procedimiento general explicado en este apartado y que favorece su utilización en aplicaciones de captura de datos: a la hora de seleccionar un nodo padre dentro de la estructura *intra-cluster*, los *nonCH* únicamente enviarán el paquete JOINPAR y se asociarán con un nodo padre y

no con todos los candidatos. Esta alternativa, que en principio parece que puede limitar la estructura multi-ruta del protocolo, es más eficiente energéticamente ya que ahorra paquetes de control y, en caso de fallo de este enlace, utiliza el *slot de recuperación* descrito en la Sección 6.4 para asociarse con el siguiente en la lista, lo que en definitiva no implica pérdida de capacidades y sí un ahorro energético. Como se ha mencionado al inicio de esta Tesis, las aplicaciones son las que marcan en gran medida el funcionamiento de los protocolos en WSN, de ahí la importancia de que HARP sea fácilmente adaptable a los requerimientos impuestos por las mismas.

Formación del árbol jerárquico inter-*cluster*

La creación del árbol inter-*cluster* puede realizarse en paralelo a la creación de los diferentes árboles intra-*cluster* con objeto de reducir la latencia total del proceso. El procedimiento de creación se describirá a continuación y es similar al indicado en el apartado anterior para la estructura entre nodos normales y CH.

Una vez que el nodo sumidero ha recibido en la fase de notificación todos los mensajes CHADV de los CH, generará un paquete de confirmación ASSOCH en el que se incluye la lista CST con un listado de los ID de cada CH ordenado por la potencia de la señal recibida de cada uno de ellos. Para ello el nodo sumidero realiza un cálculo de las pérdidas del enlace a través de la diferencia entre la potencia de transmisión de los mensajes CHADV y la potencia con la que se recibe cada uno de ellos. En caso de que los enlaces inalámbricos se consideren estadísticamente iguales, esta ordenación generalmente coincidirá con la distancia a la que se encuentra cada nodo CH del nodo sumidero.

Los nodos CH, que han escuchado el resto de mensajes CHADV y calculado el RSSI de cada uno de ellos, reciben entonces el mensaje ASSOCH y pueden calcular las pérdidas del enlace directo con el nodo sumidero. Es importante reseñar que,

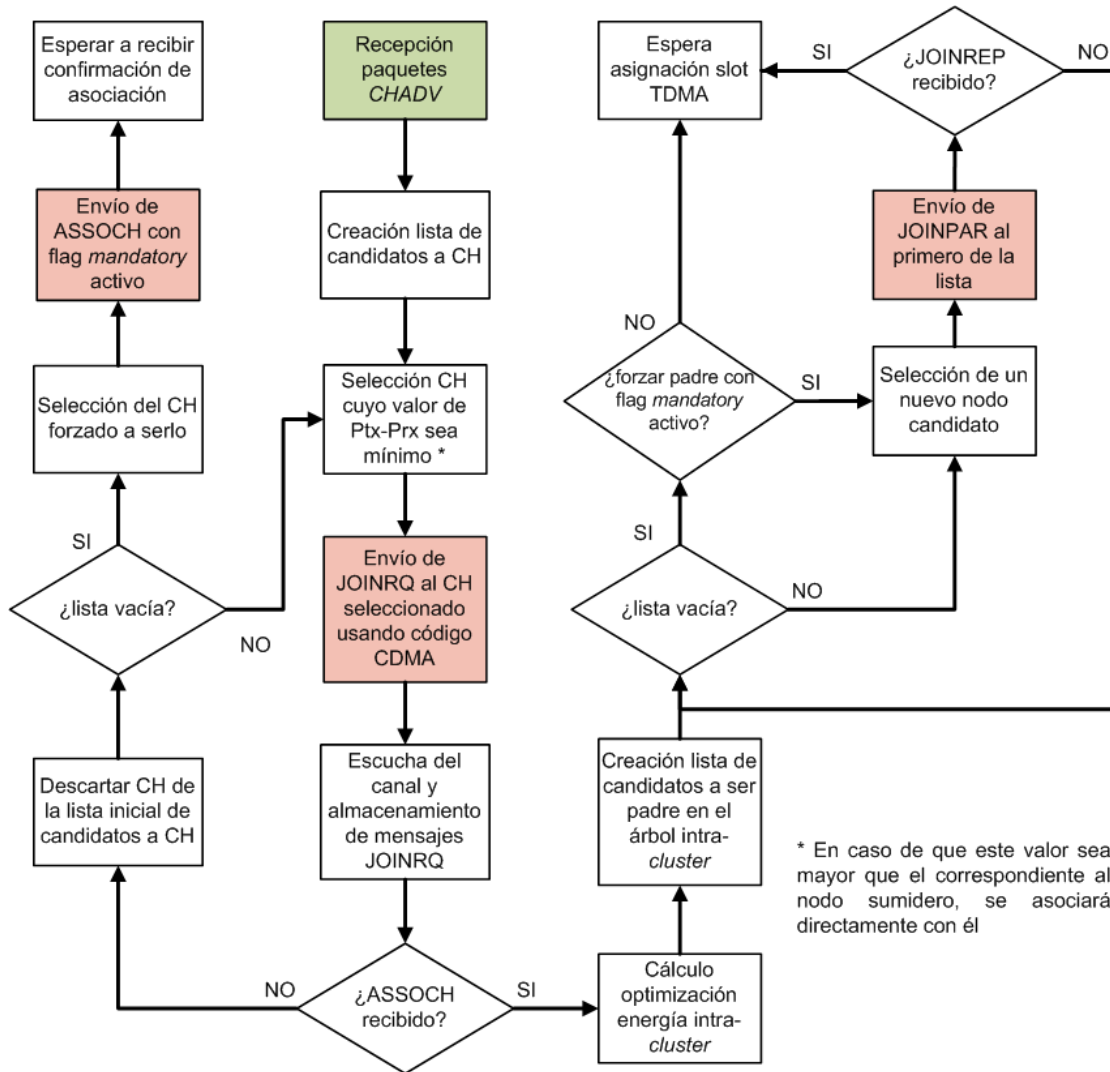


Figura 6.5: Diagrama de flujo del proceso de creación del árbol intra-cluster para aplicaciones data gathering a partir de la recepción de los mensajes CHADV de notificación de los CH.

al igual que en el caso intra-cluster, es necesario calcular el balance energético en ambos sentidos del enlace.

De esta forma, los nodos CH serán capaces de decidir si existen otros nodos CH a través de los cuales la comunicación multi-salto con el nodo sumidero puede realizarse de forma más eficiente energéticamente que la comunicación directa. En esta transmisión a través de nodos intermedios hay que tener también en cuenta el

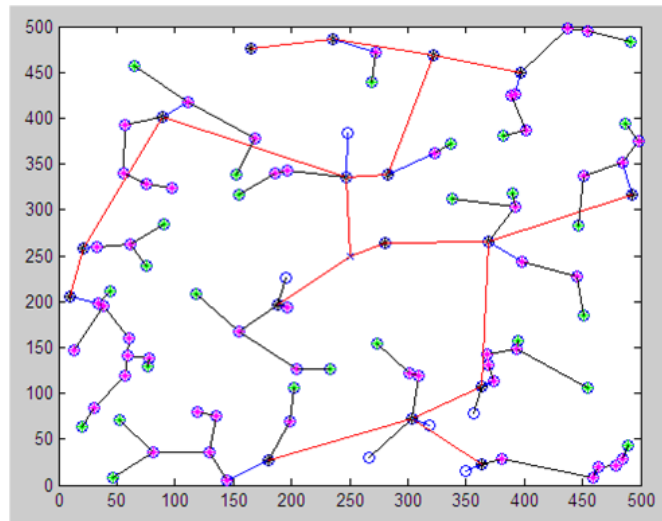
nivel de agregación requerido en el árbol *inter-cluster*, que generalmente será diferente al impuesto para las estructuras *intra-cluster*. Por su parte, el diagrama de flujo de la formación de este árbol es similar al caso de *intra-cluster* sustituyendo el CH por el nodo sumidero y los *nonCH* por los CH.

Por último, en caso de que existan nodos normales asociados directamente al nodo sumidero, bien porque no han podido asociarse con ningún CH, o bien porque energéticamente es más eficiente de esta forma, se procederá a la creación entre todos ellos de un nuevo árbol jerárquico con objeto de optimizar la energía consumida en el proceso de transmisión. El nodo raíz de este nuevo árbol será el sumidero, quien integrará la planificación de esta estructura junto con la del árbol *inter-cluster*. Esta planificación es sencilla, ya que los nodos normales pueden transmitir sus datos en los mismos *slots* que se utilizan en los *clusters* para las comunicaciones *intra-cluster*, mientras que permanecerán en estado *sleep* durante las transmisiones *inter-cluster*, evitando de esta forma colisiones e interferencias con otros *clusters* y con las transmisiones entre CH.

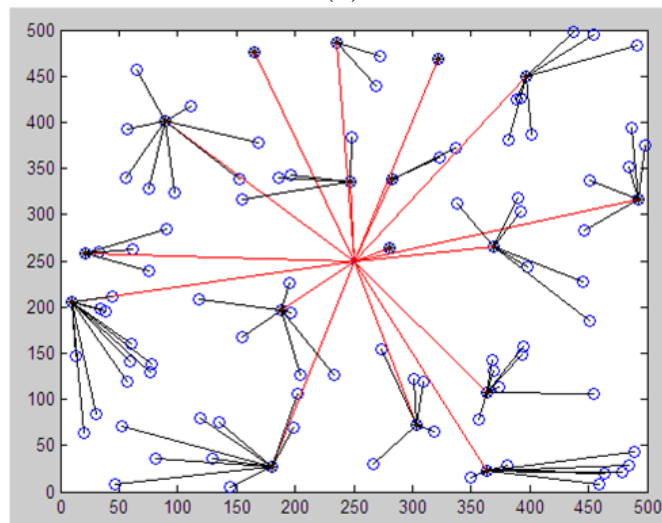
La Figura 6.6 muestra un ejemplo de las estructuras *inter-cluster* e *intra-cluster* creadas en HARP comparadas con la arquitectura de LEACH en una red con 100 nodos sensores, considerando el caso de una única asociación con un nodo padre. HARP crea estructuras con enlaces de menor longitud y menor consumo energético.

6.3.1.3. Fase de planificación o *Scheduling phase* (Sche)

Una vez creadas las estructuras jerárquicas *inter* e *intra-cluster*, el siguiente proceso es el de planificar las transmisiones de paquetes de datos en ellas con objeto de maximizar la eficiencia energética de la red. Para ello, comenzando por el interior de los *clusters*, el nodo CH iniciará la asignación de un *slot* o intervalo de transmisión a cada uno de los nodos normales asociados, conformando de esta



(a)



(b)

Figura 6.6: Ejemplo de topología creada para la transmisión durante una ronda en una red de 500x500m con 100 nodos y el nodo sumidero en el centro: a) estructuras intra-cluster (líneas negras y azules) e inter-cluster (líneas rojas) en HARP; b) protocolo LEACH, con la creación de enlaces entre el CH y los nonCH (líneas negras) y enlaces entre los CH y el nodo sumidero (líneas rojas).

forma una trama TDMA para que cada uno haga uso del *slot* asignado para la transmisión de sus datos.

La asignación de *slots* puede hacerse des dos formas distintas: centralizada o distribuida.

Asignación centralizada

El CH ha recibido todos los paquetes CHROUTE procedentes de los nodos del árbol, por lo que conoce perfectamente la estructura del mismo. En consecuencia, puede proceder a la creación de la trama TDMA en donde, comenzando por los nodos hoja, va asignando los *slots* de transmisión consecutivamente teniendo en cuenta que, para cualquier nodo del árbol, los *slots* de los nodos hijos directos deberán preceder al suyo propio con objeto de garantizar la recepción y agregación de los datos de los nodos hijo junto con los suyos propios antes de proceder a su envío. Dentro de este proceso, se deben dejar ventanas temporales de seguridad para compensar posibles derivas dentro de cada transmisión.

La asignación de *slots* dentro de la trama conlleva el envío por parte del CH de un paquete SCHE asignando consecutivamente los mismos en el orden propuesto a cada uno de los nodos del *cluster*. Cada nodo del árbol deberá guardar información no sólo de su *slot*, sino también de los *slots* de los nodos hijo directos para poder recibir el paquete de datos de cada uno de ellos. En estos *slots* de transmisión y recepción asignados cada nodo deberá estar despierto, pasando a estado *sleep* o de bajo consumo en los restantes en los que no tenga que realizar ninguna actividad.

Asignación distribuida o basada en *token*

Opcionalmente HARP dispone de un método distribuido para realizar la planificación de la trama TDMA en el árbol jerárquico. La razón de definir este método es que existen determinadas aplicaciones en las que puede no ser necesaria la utilización de todas las capacidades de encaminamiento que HARP pone a disposición de la red. En este tipo de aplicaciones, tales como la monitorización del entorno o la detección de eventos como pueden ser terremotos, erupciones volcánicas o aumentos de temperatura, no es necesario que el CH conozca la estructura de rutas hacia todos los nodos del *cluster*. En este caso, esta alternativa permitiría suprimir el envío de los paquetes CHROUTE de información hacia el propio CH y evitaría

de esta forma el sobre coste energético que conlleva el método anteriormente descrito, en el que es necesario que el CH conozca la estructura del árbol para poder realizar la planificación y después difundirla. Obviamente también puede utilizarse junto con los paquetes CHROUTE, cuando se quiera disponer de un proceso distribuido que permita al nodo CH realizar simultáneamente otras tareas como las asociadas a la planificación de la trama TDMA del árbol inter-*cluster*.

El método propuesto a continuación es un método de asignación que puede denominarse *basado en token* o *paso de testigo*. En él, el CH cede la gestión en la asignación de *slots* a los nodos del árbol a través de un *testigo* o *token* que va propagándose a lo largo de la estructura jerárquica ya creada entre los nodos.

En primer lugar, el CH crea un paquete SCHE específico que actuará como *token* virtual y en el que define el número y la duración de tantos *slots* consecutivos como el número de nodos *nonCH* asociados al *cluster* (esta información sí la posee, ya que ha procedido anteriormente a confirmar esta asociación con cada uno de ellos en la fase de formación del *cluster*). Una vez conformado este paquete, procede a su envío con destino el primer nodo de la lista CST que, generalmente, será el más cercano.

El nodo que recibe el *token* actuará siguiendo el proceso descrito en el diagrama de flujo de la Figura 6.7. Básicamente este método consiste en el reenvío del *token* entre los nodos del árbol y en la auto-asignación consecutiva del primer *slot* libre por parte de los nodos hijo de cada nodo en la rama hasta llegar de nuevo al CH. En primer lugar, el nodo que recibe el *token* comprobará si tiene nodos hijos de los que él es el padre principal (asociación con menor gasto energético); en caso afirmativo seleccionará uno de ellos y le enviará el *token*; en caso negativo, tomará posesión del primer *slot* libre incluido en el *token*, guardará esa información en el propio *token* para que luego el CH pueda rescatarla y reenviará dicho paquete a su nodo padre principal. El nodo padre que lo reciba, lo encaminará a través de otro nodo

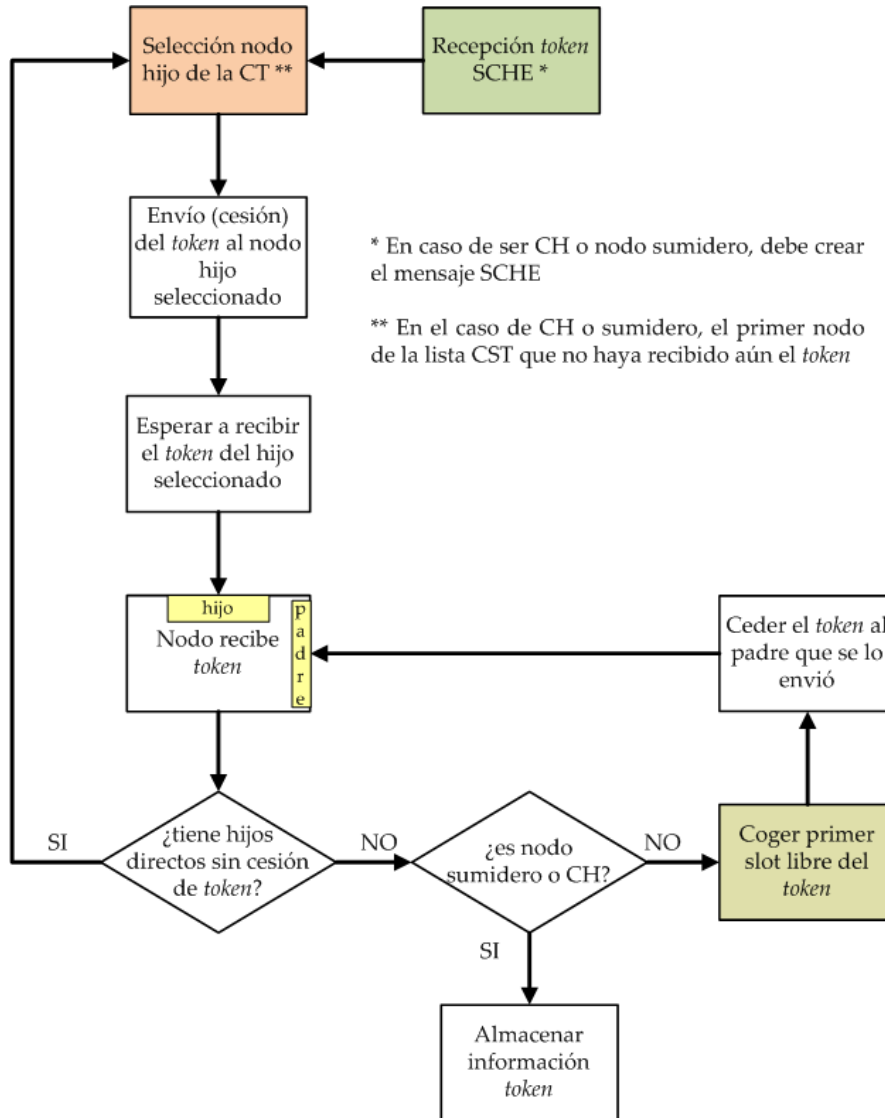


Figura 6.7: Diagrama de flujo del método de asignación basada en token para un árbol intra-cluster.

hijo, y así consecutivamente hasta que todos los nodos de sus ramas a través de sus hijos tengan un *slot* asignado. En ese momento, este nodo se asignará el siguiente *slot* libre, que siempre será posterior al de todos sus hijos, y reenviará el *token* hacia el padre del que lo recibió. Cuando el CH recibe el *token* comprueba si queda alguna rama por asignar *slot*. En caso positivo vuelve a ceder el *token* hasta que todos los nodos del árbol tengan su *slot* asignado.

Una vez planificadas las tramas TDMA en cada uno de los *clusters*, se procederá a la asignación de *slots* en la estructura inter-*cluster* entre los CH y el nodo sumidero, si bien esta planificación puede realizarse también en paralelo a la correspondiente al interior de cada *cluster*. El nodo sumidero deberá tener en cuenta los criterios de agregación de datos en la estructura inter-*cluster* a la hora de planificar los *slots* de transmisión. De esta forma, en caso de que los CH no agreguen los datos recibidos de su CH hijo sino que los reenvíe (coeficiente de agregación igual a cero), la duración de su *slot* de transmisión deberá ser mayor que el de su hijo para poder hacer frente a esta situación. Esta característica puede observarse, junto con el esquema de planificación descrito en este apartado, en el esquema general de una ronda de operación incluido en la Figura 6.8.

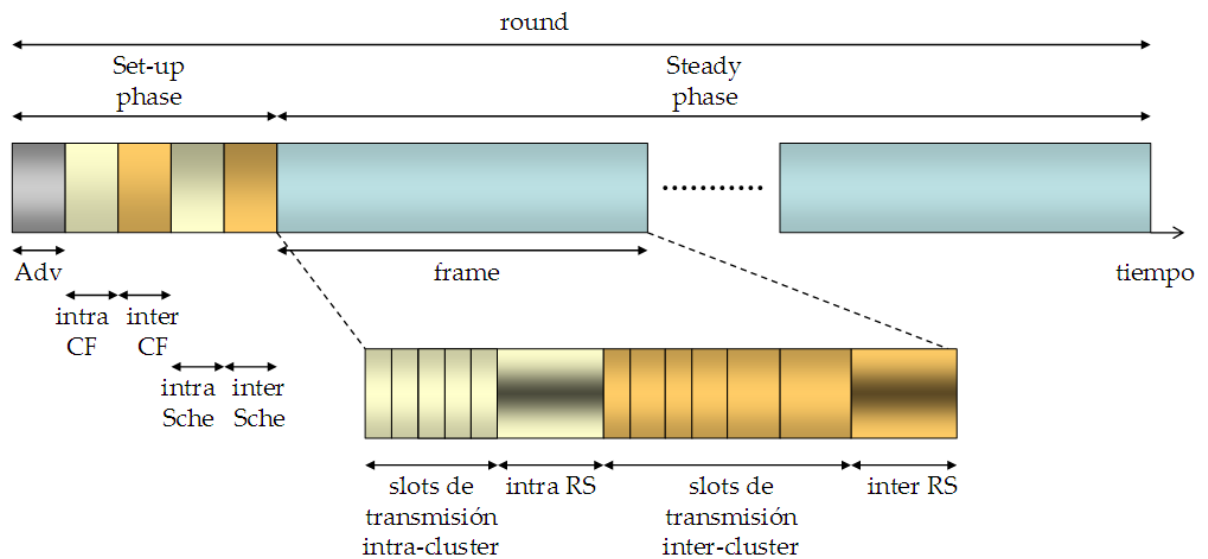


Figura 6.8: Esquema general de una ronda de operación o round en HARP

6.3.2. Fase de transmisión de datos o *Steady phase*

Una vez que se ha establecida la estructura completa de la red y se han planificado todas las tramas TDMA inter e intra-*cluster*, se inicia una fase estable en la

cual se aprovechan las estructuras de red creadas durante un intervalo de tiempo predeterminado por la aplicación.

Esta fase de transmisión se divide en franjas temporales o *frames*, cada una de las cuales está formada por los *slots* asignados y en donde se producen las transmisiones tanto desde los nodos del *cluster* hacia el CH, como entre los CH entre sí y el nodo sumidero.

En este proceso cíclico los nodos sensores deben recoger o medir la información de su entorno y enviarla a su nodo padre en el *slot* asignado tal y como se ha descrito en los apartados anteriores. El nodo padre que lo recibe iniciará el proceso de agregación o fusión de los datos de todos los nodos hijo con el suyo propio, y reencaminará esta información a través de un nuevo paquete de datos hacia su nodo padre en el árbol.

Cuando el CH recibe todos los paquetes procedentes de cada una de las ramas del árbol, repite el proceso de agregación y reenvío de esa información hacia su nodo padre dentro de la estructura inter-*cluster*. Este proceso se repite, teniendo en cuenta las condiciones y el grado de agregación de la información obtenida en cada una de las estructuras, hasta que los paquetes de datos llegan al nodo sumidero.

Una optimización del procedimiento anterior puede tenerse en cuenta en aplicaciones en las que no se requiera una monitorización continua del entorno, sino que sólo se transmitirá información cuando el nodo sensor haya detectado algún evento o se haya cumplido determinada condición. Un ejemplo de este tipo de aplicaciones son las de detección y seguimiento de intrusos, detección de catástrofes naturales tales como terremotos o erupciones volcánicas, monitorización de un campo de cultivo con objeto de detectar los puntos en los que se supera un umbral de humedad y, en ese caso, activación de los mecanismos apropiados para el cierre de válvulas de riego, etc. En estos casos, existirá un número determinado de nodos

que no tendrán datos que transmitir cuando llegue su *slot* correspondiente, con lo que el nodo padre que escucha no recibirá paquete alguno. En esta situación, el padre simplemente ignorará esta ausencia de información y continuará con su proceso normal de agregación del resto de paquetes recibidos y el reenvío de la misma. Esta situación puede dar lugar al uso ineficiente del ancho de banda asignado a la red de sensores debido a la utilización de TDMA para la transmisión de información en la red.

En consecuencia, HARP puede ser fácilmente construido sobre otras alternativas en el uso del ancho de banda como es el acceso aleatorio al canal, simplemente eliminando la fase de planificación descrita anteriormente. En concreto, en esta estructura de trama definida, pueden sustituirse los *slots* TDMA por un único *slot* temporal en el que todos los nodos utilicen técnicas de acceso aleatorio como CSMA/CA para transmitir a través del canal cuando lo necesite. Esto puede conllevar problemas a la hora de despertar al nodo padre cuando se vaya a transmitir un paquete de datos, por lo que deben utilizarse técnicas adicionales para lograr este objetivo. Una de estas posibilidades es la utilización por parte del nodo emisor, previamente al envío de su paquete de datos, de una señal específica que fuerce al nodo padre a despertar de su estado de bajo consumo y pueda recibir así el paquete que se va a enviar. Este tipo de señal descrita en [59] y utilizada en otros protocolos de encaminamiento en WSN como [42] se denomina *radio_triggered_wake_up* y es capaz de excitar el circuito de recepción de un nodo que se encuentre en estado *sleep* o de bajo consumo haciendo que se despierte en ese momento.

Otra técnica alternativa a la planificación con TDMA es la de TDMA-*polling* o sondeo, en la cual no se definen *slots* de transmisión sino que en cada *frame* cada nodo padre, comenzando por el CH en la estructura *intra-cluster*, sondea a sus hijos preguntándoles si tienen datos que transmitir. Éstos, a su vez, sondearán a sus hijos y así consecutivamente hasta alcanzar los nodos hoja del árbol. Cuando un nodo tiene datos preparados para transmitir, se lo comunica a su padre y los

transmite, propagándose este proceso hasta alcanzar de nuevo el CH. El mismo proceso se repite en la estructura inter-*cluster* para la transmisión de datos hacia el nodo sumidero. Un análisis y una descripción más detallada de esta técnica puede consultarse en [144].

Al proceso completo de formación y planificación de la red y a la transmisión de información durante la etapa estable se le ha denominado ronda o *round*. Tras alcanzar el número de *frames* preestablecido, la estructura de red se vuelve obsoleta si se tienen en cuenta consideraciones energéticas, por lo que se inicia una nueva ronda de transmisión comenzando por la elección y notificación de nuevos nodos con el rol de CH dentro de la fase de inicialización de la red. Cuando acaba la última *frame* los CH avisan de este hecho a sus nodos asociados y por tanto éstos no pasarán a estado de bajo consumo ya que se iniciará de nuevo el proceso de creación de los *clusters*. El tiempo de duración de las rondas dependerá de la aplicación específica para la que se utilice la WSN. Para cada una de ellas, en función de las condiciones de la red (número de nodos sensores, extensión, etc.) deberá tenerse en cuenta el sobrecoste energético que supone la formación de las estructuras jerárquicas en la fase de inicialización.

Sin embargo, con lo indicado hasta ahora en este capítulo, el esquema de fases contenidas en una ronda no estaría completo aún ya que es necesario incorporar a la estructura de tramas y *frames* algún mecanismo que permita la reestructuración de la red ante fallos de transmisión que se puedan producir en el transcurso de una ronda y que puedan afectar al rendimiento del sistema completo, así como gestionar la movilidad de los nodos y el despliegue de nuevos dispositivos en la red. Según esto, HARP incluye un *slot* adicional al final de la trama temporal de transmisiones en el árbol jerárquico denominado *slot de recuperación* o *recovery slot* (RS) y que se describirá con detalle en la sección siguiente. En la Figura 6.8 puede observarse el diagrama completo con todas las fases incluidas en una ronda de operación completa y los *slots* definidos en cada *frame*.

6.4. Mecanismos de recuperación ante fallos y gestión de movilidad

Tal y como se ha introducido en el apartado anterior, las condiciones cambiantes en la red así como en el medio inalámbrico, pueden producir fallos de comunicación entre los nodos. Estos fallos de red pueden ser causados por diferentes motivos, algunos relacionados con las condiciones dinámicas de los canales inalámbricos establecidos y otras asociadas a fallos en los nodos de la red. La detección de estos fallos en enlaces se producirá cuando el mensaje de confirmación de recepción ACK no llegue al nodo destino correctamente. Así, un protocolo de encaminamiento en WSN debe poder responder a situaciones que varíen las condiciones de transmisión de la red tales como obstáculos, desvanecimientos en el canal, agotamiento de la batería o mal funcionamiento de los nodos, cambio de posición de los mismos, etc. Más aun, las aplicaciones existentes exigen que los procesos de recuperación sean ejecutados de manera eficiente y con un mínimo retardo para poder mantener los requisitos impuestos por las mismas.

A pesar de que el coste de los dispositivos sensores debe ser bajo, según lo anterior se hace necesaria la definición de ciertos mecanismos de recuperación ante fallos que, aunque introducen mayor complejidad en el funcionamiento del protocolo, permiten dotar a los nodos de procedimientos eficientes de recuperación ante los fallos producidos en la red y les permiten además adaptarse a las condiciones cambiantes de la misma, incrementando a su vez la robustez del protocolo sin incidir de manera apreciable en el consumo energético total de los dispositivos.

HARP proporciona dos mecanismos de recuperación local ante fallos de red: el *slot de recuperación* y el *nodo sustituto*. Si bien el concepto de nodo sustituto no es nuevo, su utilización en HARP debe ser definida de forma distinta a como

se realiza en otros protocolos de encaminamiento como [42]. A continuación se describe detalladamente cada uno de estos mecanismos.

6.4.1. *Slot* de recuperación

El *slot* de recuperación (RS) es un intervalo temporal que se utiliza para informar a otros nodos tanto de los cambios detectados en la estructura de red o de *cluster* como de los fallos en transmisiones producidas entre nodos. A la vista de la Figura 6.8 de la sección anterior, existen dos tipos de RS: RS intra-*cluster* y RS inter-*cluster*, según la estructura jerárquica en la que se utilice.

En las transmisiones intra-*cluster*, al final de los *slots* de transmisión de los *nonCH*, se inicia un nuevo *slot* en donde los nodos que no hayan podido transmitir en su *slot* correspondiente acceden mediante CSMA para informar del error producido. Para ello, cuando detectan la transmisión fallida, almacenan el paquete de datos en un *buffer* de memoria, borran de su PT la entrada correspondiente al nodo padre inalcanzable, seleccionan de esta tabla de rutas el siguiente de la lista como candidato a padre principal y pasan a estado de bajo consumo hasta que comience el RS. El instante de comienzo de este *slot* ha sido indicado por el CH anteriormente, tal y como se explicará más adelante. En la Figura 6.9 se muestra el diagrama de flujo del procedimiento de recuperación propuesto por HARP ante la detección de un fallo en la transmisión de un paquete de datos.

Con el inicio del RS, el CH difunde un paquete CLADV con el *flag* RS_start activado para indicar que es el principio del *slot* y con información relativa al instante de finalización del RS de la *frame* actual, entre otros campos. El nodo que no haya podido transmitir sus datos accede entonces al canal mediante CSMA/CA para evitar colisiones y transmite un paquete JOINPAR de petición de asociación con destino al nodo candidato a padre principal seleccionado con anterioridad. Si el candidato le contesta con un JOINREP, la nueva asociación se habrá establecido;

6.4 Mecanismos de recuperación ante fallos y gestión de movilidad 221

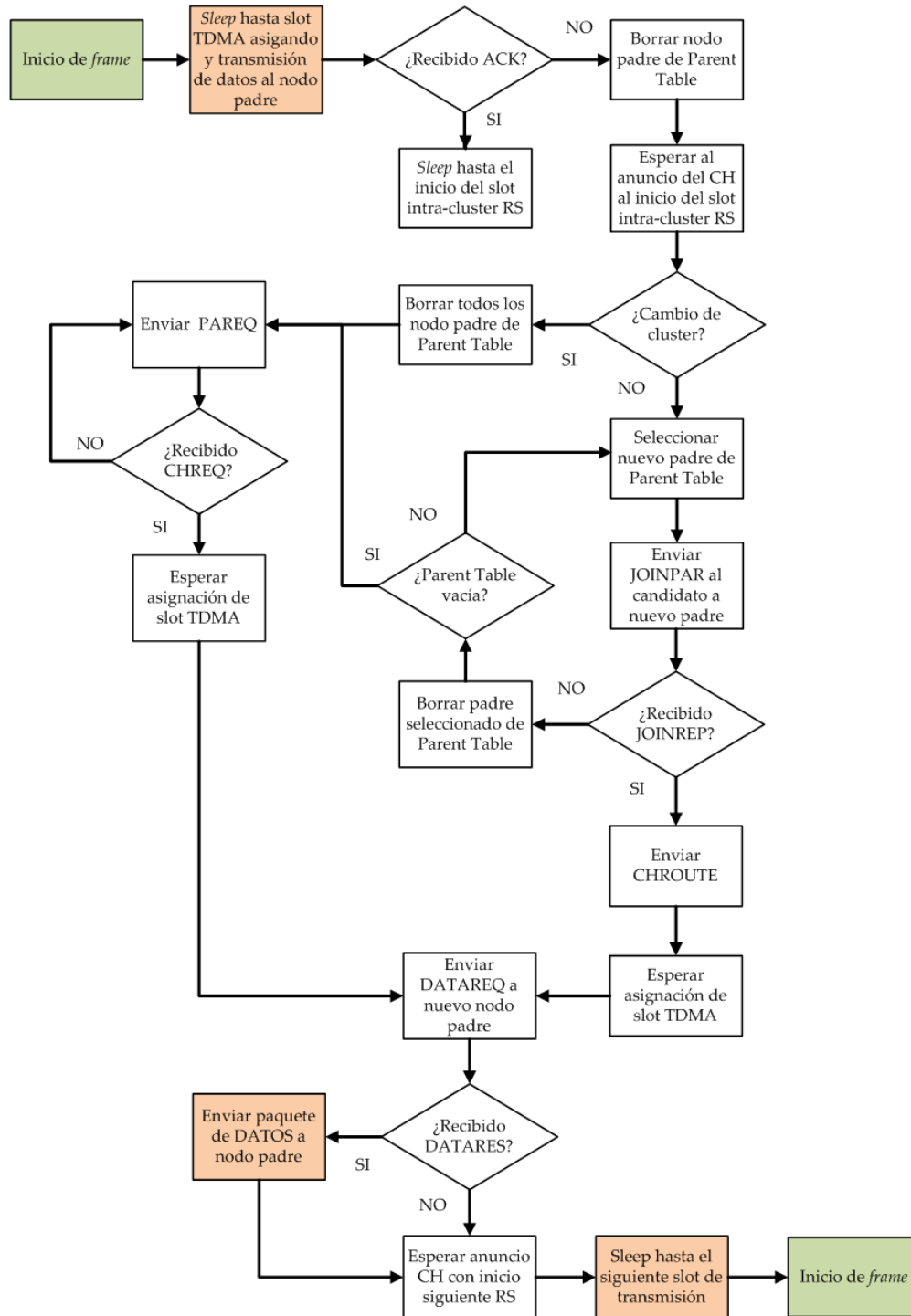


Figura 6.9: Diagrama de flujo del procedimiento de recuperación ante un fallo de transmisión utilizando el RS.

de lo contrario, el nodo original deberá seleccionar el siguiente nodo de la PT y repetir el mismo proceso de envío. El último de esta lista será el CH. En caso de que ninguno le conteste, el nodo asumirá el rol de *nodo huérfano* e iniciará el procedimiento de unión al árbol conocido como *joining mechanism*.

El proceso *joining mechanism* adaptado a WSN consiste en la difusión por parte del nodo huérfano de un paquete PAREQ de petición de asociación con un nuevo padre en la red utilizando el código CDMA del *cluster*. Los nodos que escuchen este mensaje responderán obligatoriamente enviando un paquete CHREQ. El nodo huérfano almacenará en su PT todos los nodos que le han enviado el CHREQ como candidatos a padres, pero solamente uno de ellos (el que conlleve mayor ahorro energético en las transmisiones) será seleccionado como principal. A continuación, el nodo que acaba de reinsertarse a la estructura del árbol enviará a su nuevo padre un paquete DATAREQ solicitando autorización para enviarle los datos monitorizados del entorno y que no pudieron ser transmitidos anteriormente a su antiguo nodo padre. Si recibe un DATARES, procederá a su envío; de lo contrario, si el nodo padre decide que el tiempo que queda del *slot* RS no es suficiente para permitir esta transmisión, el nodo original descartará los datos recogidos y esperará a que se le asigne *slot* para la siguiente *frame*. Si por algún motivo el nodo ha quedado aislado y no recibe contestación de ningún miembro de su antiguo *cluster*, deberá permanecer a la escucha para recibir los paquetes CLADV procedentes de los CH al inicio del siguiente RS. En ese momento difundirá un paquete PAREQ de petición de asociación al *cluster*. En caso de que siga sin poder recibir ninguno de estos paquetes por razones de distancia, incrementará proporcionalmente la potencia de emisión del PAREQ hasta que reciba contestación por parte de alguno de los nodos de la red.

Este mecanismo también es de aplicación cuando se gestione la movilidad de los nodos en la red. En esta situación, cuando un nodo se mueve sin que se haya completado una ronda de operación y detecta que en el momento de iniciar una

6.4 Mecanismos de recuperación ante fallos y gestión de movilidad 223

transmisión en su *slot* asignado los enlaces establecidos con los nodos padre o hijo ya no pueden utilizarse, debe permanecer a la escucha y esperar a recibir algún paquete CLADV al inicio de los RS procedente de los CH para decidir posteriormente a cuál de ellos se asocia.

En caso de que se produzca la asociación con un nuevo padre en el árbol (ya sea un nodo normal o el CH), el nodo origen enviará un paquete CHROUTE con información relativa a sus hijos y a todos los hijos que se encuentran por debajo en alguna de las ramas, con objeto de que los nodos superiores actualicen sus tablas. Además, este paquete llevará el *flag delete_old_route* activado y el campo ID del antiguo nodo padre relleno para que el resto de nodos eliminen esa entrada (nodo hijo alcanzable a través de nodo padre antiguo) de sus tablas de rutas, si es que la tienen almacenada.

Esta información contenida en el paquete CHROUTE es importante a la hora de disponer de las capacidades completas que HARP proporciona para el encañamiento en WSN. En algunas aplicaciones, además de flujo de datos entre los nodos del árbol hacia su nodo raíz (*upstream*), existirá la necesidad del envío de flujo de paquetes entre los nodos raíz y el resto de nodos del árbol, es decir, flujo *downstream*. En este caso es necesario que un nodo conozca al menos una ruta hacia todos los nodos que se encuentren por debajo de él para poder reenviarle el paquete recibido, con lo que no puede obviarse el envío de este tipo de paquetes a pesar de que conlleve un mayor consumo energético.

Al final del proceso el paquete CHROUTE llegará al CH, quien deberá actualizar también su CT. Una vez hecho esto, sólo queda pendiente comprobar si la planificación del *cluster* ha de ser alterada para tener en cuenta la reestructuración producida en el árbol jerárquico. En caso de utilizarse una planificación centralizada, el CH determinará en función de la nueva estructura si sigue siendo válida la trama TDMA anterior y, en caso de no ser así, procederá al cambio de asig-

nación en los *slots* necesarios para adaptar la trama a la nueva estructura. Para ello hará uso del mensaje SCHE con destino únicamente a los nodos involucrados en dichos cambios. En caso de que se esté utilizando una planificación basada en *token*, será el nuevo nodo padre quien primero decidirá si se necesitan cambios en los *slots* que le preceden para incluir al nuevo nodo hijo. En caso de no poder realizarlo, involucrará en esta decisión a su nodo padre, y así consecutivamente hasta que la estructura de *slots* pueda ser reconfigurada incluyendo los nuevos cambios. Un ejemplo de este tipo de reconfiguraciones se muestra en la Figura 6.10.

Finalmente, cuando el RS está próximo a su finalización, el CH tiene reservado un *slot* adicional con el que informa al resto de nodos del *cluster* a través de un paquete CLADV del tiempo de inicio del siguiente RS. Esta acción permite adaptar la duración de este intervalo al número de nodos asociados al *cluster* tras la finalización de la presente *frame*.

La utilización de este RS para reconfiguraciones inter-*cluster* es idéntica a la descrita para la trama intra-*cluster*, sustituyendo los roles de CH y nonCH por los de sumidero y CH, respectivamente. Este *slot* es un elemento original que HARP incorpora si se compara con otros protocolos similares de encaminamiento para WSN.

6.4.2. Nodo sustituto

El RS descrito en la sección anterior puede utilizarse para minimizar los efectos que tiene en la red el fallo de los enlaces con uno de los nodos del árbol jerárquico, ya sea por agotamiento de su batería, por su mal funcionamiento o debido al movimiento. Sin embargo, el RS tal y como se ha descrito no puede hacer frente a fallos en los CH de la estructura intra-*cluster* ni a fallos del sumidero en la estructura inter-*cluster*. En estos casos, se utiliza un nodo sustituto que toma el rol del nodo desaparecido cuando éste deja de estar activo.

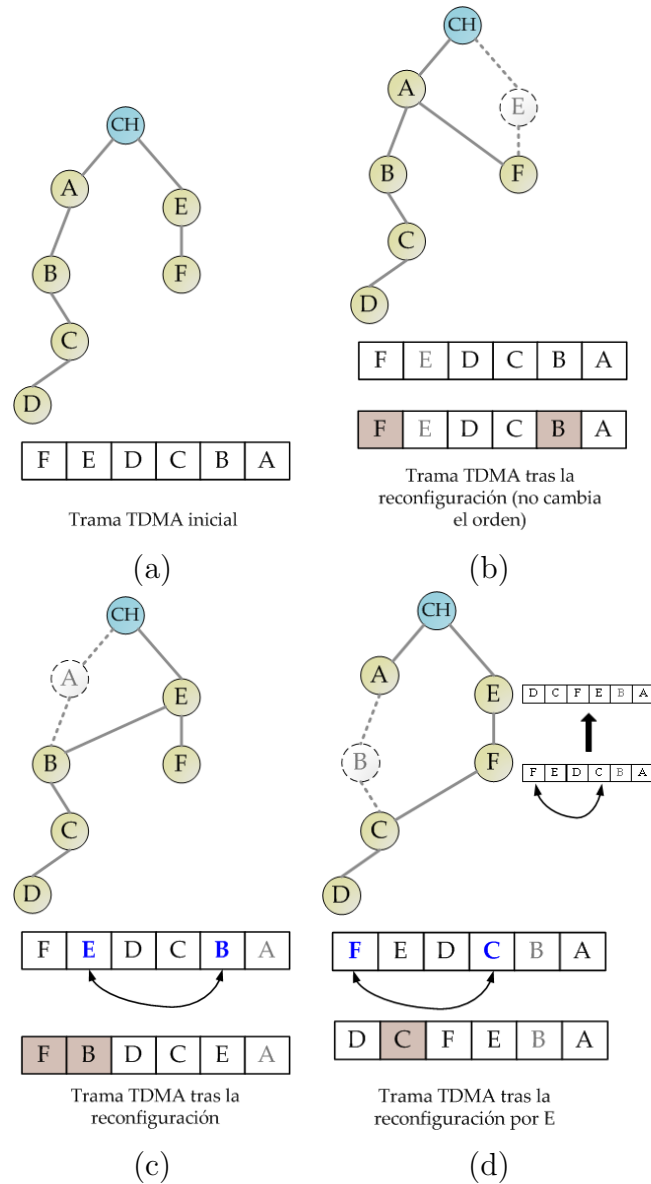


Figura 6.10: Ejemplo de reconfiguración de la trama TDMA con planificación basada en token: a) estructura y trama formadas inicialmente; b) reconfiguración de la estructura cuando falla el nodo E, en la cual no se modifica la estructura de la trama TDMA (los slots sombreados corresponden a aquéllos en los que A debe escuchar las transmisiones de B y F); c) reconfiguración de la trama en caso de que falle el nodo A, donde E debe cambiar el orden de su slot con B y debe escuchar las transmisiones de los slots sombreados (B y F); d) reconfiguración compleja donde F no es capaz de reorganizar correctamente la trama sin que afecte a nodos superiores, por lo que cede esta responsabilidad a E; una vez reconfigurada, el nodo F debe escuchar la transmisión de C (sombreada).

El *mecanismo de nodo sustituto* (SN, *substitute node*) proporcionado por HARP reduce el impacto de la desaparición o el mal funcionamiento del nodo raíz del árbol (CH en el *intra-cluster* y nodo sumidero en el *inter-cluster*). Este proceso se basa en la selección al principio de cada ronda por parte del nodo raíz de un nodo que tomará su rol en caso de que sea necesario. Para ello, el CH (en la estructura *intra-cluster*, aunque la descripción es la misma para la estructura *inter-cluster* con respecto al nodo sumidero) seleccionará en la fase de inicialización uno de los *nonCH* asociados al *cluster* y notificará esta elección en el mensaje CLADV de información enviado al final de la primera *frame* de la ronda, procediéndose al almacenamiento de su ID en la memoria del resto de los nodos del *cluster*. Si el nodo sustituto no cambia de una *frame* a otra, no será necesario incluirlo en el siguiente mensaje CLADV. Además del CLADV general, el CH enviará un mensaje SCHE con destino únicamente al nodo sustituto informándole de datos necesarios que deberá conocer en el caso de que deba asumir su rol, como son el código CDMA de las transmisiones *inter-cluster* y el ID del padre principal dentro de esta estructura.

El CH elegirá el nodo sustituto tratando de que el nodo seleccionado sea lo más similar a él en función de parámetros tales como nivel de energía (que puede conocer con la recepción de un mensaje REMEN en s-HARP, Sección 6.6) o su posición, ya que cuanto más cerca se encuentre, más probabilidades existen de que las características de propagación de los enlaces inalámbricos con el resto de nodos del *cluster* sean muy similares, lo que minimizaría el efecto del fallo del CH en las transmisiones dentro del mismo. Este nodo deberá seguir el mismo esquema de operación que el CH, es decir, deberá escuchar cuando el CH lo haga y podrá pasar a estado *sleep* cuando aquél conmute.

Cuando el nodo sustituto detecta el fallo del CH, bien porque no puede realizar una transmisión o bien porque no envía ningún mensaje CLADV en el RS, deberá tomar posesión de este rol y hacerse cargo de sus funciones. Con el envío

6.4 Mecanismos de recuperación ante fallos y gestión de movilidad 227

del mensaje CLADV notificará al resto de nodos este cambio. Además, si el fallo del CH es detectado por otros nodos de la red antes que por el nodo sustituto, éste será avisado ya que recibirá el paquete de datos que estaba destinado originalmente al CH.

HARP es consciente de que el nodo sustituto ha sido forzado a tomar el rol de CH, por lo que debe tratar de optimizar el consumo energético de este nodo. Para ello, a pesar de que el nodo asume el rol de CH en el árbol *intra-cluster*, esta obligatoriedad debe permitirle minimizar su gasto de energía dentro de la estructura *inter-cluster*. Para ello, con objeto de convertirse en un CH hoja en el árbol *inter-cluster* y ahorrarse de esta forma el gasto extra que supone para un nodo intermedio el reenvío de los datos generados por los nodos hijo, el nodo sustituto enviará los paquetes correspondientes al nodo padre como si del nodo CH fallido se tratase, pero en cambio no contestará a los de sus hijos, obligando a éstos a reconfigurar la estructura asociándose a otros padres dentro de la estructura *inter-cluster*.

En definitiva, a través de la utilización de los dos mecanismos anteriores, junto con el proceso de unión al árbol denominado *joining mechanism* y explicado en el Capítulo 3.5, HARP es capaz de adaptarse a las circunstancias específicas de la red y reconfigurar las estructuras jerárquicas existentes para hacer frente a los fallos producidos en la misma. Con estos mecanismos se consigue además incorporar a la red a los nodos que han sido desplegados en instantes posteriores a su inicio, lo que sucede en muchas aplicaciones. Con este despliegue adicional pueden conseguirse numerosos beneficios para la red como puede ser la extensión del tiempo de vida de la misma. Además de ello, el protocolo debe proporcionar un método específico para gestionar la movilidad de los nodos de la red.

Para finalizar este apartado, aunque es bastante inusual en la mayoría de las aplicaciones de WSN, puede ocurrir que existan situaciones con una alta tasa de

fallos o con gran movilidad de los nodos en la red, lo que puede dar lugar a la saturación del RS en alguno de los *clusters* formados. Ante esta situación, el RS puede no ser lo suficientemente duradero como para permitir todas las secuencias de mensajes definidos en la reconstrucción de la estructura jerárquica. En aplicaciones con tasa de recogida o monitorización de datos constante, la imposibilidad de extender la duración de las *frames* hace que deban descartarse algunos de estos procesos de reasociación que tienen lugar en el RS, con lo que deberá decidirse bien si se desea que estos datos recogidos por los sensores no lleguen al destino, o bien que los nodos que no puedan asociarse a un *cluster* por motivos de saturación puedan hacerlo a otro *cluster* próximo, aunque de esta forma no se optimice la eficiencia energética de la red. Para otras aplicaciones, HARP contempla la configuración dinámica de la duración de las *frames* y las rondas para adaptar el tiempo del RS a las condiciones cambiantes de la red.

6.5. Arquitectura de aplicación específica

Uno de los factores de diseño de HARP es la facilidad del protocolo para adaptarse a aplicaciones específicas. Tal y como se ha descrito a lo largo de este capítulo, HARP utiliza la misma estructura dinámica de *cluster* que la mayoría de los protocolos de encaminamiento definidos para WSN, lo que proporciona escalabilidad, sencillez y ahorro energético a la red. Es más, el protocolo propuesto optimiza el funcionamiento, el consumo y la reestructuración de las WSN cuando se utiliza en labores de monitorización o recolección de datos del entorno, superando en rendimiento a otros protocolos definidos con el mismo propósito.

Además de su utilidad en redes de monitorización, la estructura de tablas de encaminamiento propuesta en HARP permite su utilización en diferentes aplicaciones que requieran la comunicación entre los nodos de la red jerárquica en sentido descendente, es decir desde el nodo sumidero a los nodos hoja de cada uno de los

clusters. Estas aplicaciones se conocen comúnmente como aplicaciones de diseminación dirigida de datos (*one-to-many*), en contraposición a las de recolección, monitorización o captura de datos (también denominadas *many-to-one*) indicadas en el párrafo anterior. En ellas es requisito indispensable el encaminamiento eficiente desde el nodo sumidero o desde el CH a cualquier otro nodo de la red. Un ejemplo de este tipo de aplicaciones son las que tienen como objeto el sondeo de nodos específicos de la red para la detección de algún tipo de evento, como puede ser la detección de intrusos. Estos procesos requieren que el protocolo de encaminamiento sea capaz de encontrar de forma eficiente rutas hacia cualquier nodo de la red. A medio camino entre estos dos tipos de aplicaciones se encuentran las que son capaces de comunicar cualquier nodo de la red con cualquier otro. Son las llamadas *peer-to-peer* o *any-to-any*.

A través del uso de los *slots* asignados para la transmisión, los mensajes y la estructura de tablas descrita, HARP es capaz de establecer cualquier tipo de comunicación entre dos nodos de la red, sea en sentido *upstream* (recolección), *downstream* (diseminación) o ambas (*peer-to-peer*). Así, los nodos CH que reciben un paquete CHROUTE con información de la estructura de las rutas del *cluster*, pueden reencaminar esta información para que los nodos superiores en el árbol jerárquico y el nodo sumidero sean capaces de conocer en cada momento al menos una ruta válida hacia cualquier nodo sensor que se encuentre en cualquiera de las estructuras jerárquicas inferiores. Ésta es la esencia del encaminamiento propuesto HARP y que permite optimizar la eficiencia de la red sea cual sea el flujo de comunicaciones que se produzca en la misma.

6.6. s-HARP

A lo largo de este capítulo se ha descrito tanto la arquitectura propuesta por HARP para WSN, como las fases de operación y los mecanismos de recuperación

ante fallos y de gestión de movilidad. En la sección correspondiente a la fase de elección y notificación de rol de la Sección 6.3 se hacía referencia a que HARP, como propuesta de estructura jerárquica optimizada, podía utilizar cualquiera de los métodos propuestos en la literatura científica asociada para la elección de los CH en cada ronda de operación. La elección de CH es un proceso crítico que tiene incidencia directa en la distribución del consumo energético en la red, tal y como se demostrará en el capítulo destinado a los resultados.

Algunos de esos métodos de selección no tienen en cuenta factores decisivos para el devenir energético de la red cuando toman esta decisión. Así, por ejemplo, LEACH no considera la distribución de los nodos en la red, ni su distancia al nodo sumidero, ni la energía remanente que le queda a cada uno, resultando una distribución de CH que no equilibra el consumo energético de los nodos.

Tal y como se explica en [125], todo algoritmo de selección de CH debe ser distribuido, energéticamente eficiente, tener en cuenta la distribución de la energía de los nodos en la red y ser, a su vez, balanceado. En consecuencia, se debe buscar un método que sea capaz de elegir una lista de CH permitiendo que el consumo energético entre todos los nodos de la red sea, por un lado, el menor posible y, por otro, igualmente distribuido entre todos ellos. De esta forma se conseguirá que todos los nodos en la red vayan agotando sus baterías casi al mismo tiempo, lo que implica que el tiempo en el que la red es útil para desarrollar su labor de forma completa (es decir, que todos los nodos cumplan con su misión) aumente con relación al caso en que los nodos vayan muriendo poco a poco y exista mucha diferencia entre la muerte de los primeros y la de los últimos. En este último caso, la eficiencia y fiabilidad de la monitorización de la red se ve fuertemente limitada por el número de nodos sensores que permanezcan activos en cada ronda de operación.

En esta Tesis se propone además un método específico de selección de CH utilizando una formulación particular que tratará, por un lado, de optimizar el consumo energético total en cada *frame* y, por otro, de balancear o distribuir este consumo energético por igual entre todos los nodos de la red. Este método será un híbrido entre los dos definidos en la sección correspondiente a la fase de elección y notificación de rol de la Sección 6.3.

Para su auto-elección como CH un nodo tendrá en cuenta su energía restante en comparación con la de sus vecinos, conocida ya que en el proceso de formación de los *clusters* a través de la primera acción que realizarán los nodos consistente en difundir un mensaje REMEN con el nivel de su energía remanente. Por otro lado, la distancia a la que se encuentre el nodo con respecto al sumidero también influye en el gasto energético, según el modelo energético presentado en la Sección 6.2. Por último, con objeto de balancear aun más la carga, se incluye la formulación propuesta por LEACH que tiene en cuenta el número de rondas transcurrido desde la última en la que un nodo fue CH. Estos tres parámetros se utilizan para el cálculo del umbral de decisión a través de la ecuación:

$$T(n) = \alpha \frac{(d_{max} - d_{toBS})}{d_{max}} + \beta \frac{E_{rem}}{E_{mean}} p_{CH} + \gamma T_L(n) \quad (6.5)$$

donde d_{max} es la máxima distancia a la que puede estar un nodo del sumidero y tiene en cuenta la extensión de la red, E_{rem} es el valor de la energía restante de un nodo, E_{mean} es el valor medio calculado de la energía de los nodos vecinos de los que recibe el mensaje REMEN y p_{CH} el valor deseado de la probabilidad de ser CH en la red. Por su parte α , β y γ son los pesos que cada componente tendrá en el cálculo del umbral y que deben ser ajustados mediante simulación para cada una de las redes definidas. Finalmente, $T_L(n)$ es el umbral definido en el protocolo LEACH, que tiene en cuenta el valor p de la probabilidad de ser CH (definida en función del número deseado de CH con respecto al total) y la ronda de operación,

r , en la que se encuentre el sistema con respecto al inicio de la actividad, según la ecuación siguiente:

$$T_L(n) = \begin{cases} \frac{p}{1-p \cdot (r \cdot \text{mod} \frac{1}{p})} & , \text{ si } n \in G \\ 0 & , \text{ en cualquier otro caso} \end{cases} \quad (6.6)$$

siendo G el conjunto de nodos que no han sido CH en las últimas $1/p$ rondas de operación.

Finalmente, el candidato seleccionado para ser CH lo será finalmente sólo si su energía restante es mayor o igual que el valor del parámetro χ (ajustado mediante simulación) multiplicado por la energía media restante de los nodos de su alrededor, resultando la ecuación:

$$E_{rem} \geq \chi \cdot E_{mean} \quad (6.7)$$

6.7. Conclusiones

En este capítulo se ha definido una nueva arquitectura jerárquica para el encaminamiento de paquetes en WSN, denominada HARP. El propósito de esta propuesta es la de mejorar la eficiencia energética del encaminamiento, optimizando el gasto energético de los nodos a través de la creación de estructuras jerárquicas estables en dos niveles: una estructura entre los nodos sensores normales y el CH (árbol *intra-cluster*) y otra estructura, denominada *inter-cluster*, entre los CH y el nodo sumidero. HARP puede generalizarse para el caso de que existiesen varios nodos sumideros. Debido a la estructura de tramas TDMA definida, se evitan además colisiones en las transmisiones a través de la red, lo que optimiza aun más las prestaciones del protocolo.

Además del encaminamiento eficiente, HARP dispone de mecanismos de tolerancia ante fallos ocurridos en la red, tanto en los enlaces inalámbricos como en los propios nodos, así como procedimientos específicos para gestionar la movilidad de los nodos en la red y la adición de nuevos nodos desplegados a lo largo de la misma. Estos mecanismos son dinámicos y adaptan la estructura de red y de tramas de transmisión a los cambios producidos, todo ello sin introducir sobrecarga adicional tal y como sucedería con otros procesos de descubrimiento de nuevas rutas. Adicionalmente, al contrario que en otros protocolos para WSN denominados de *localización* o *geográficos*, en HARP no es necesario que los nodos conozcan su posición exacta en la red mediante la utilización de un dispositivo GPS o similar.

HARP se ha diseñado siguiendo el criterio de adaptabilidad a aplicaciones específicas, por lo que su particularización para cada una de ellas es sencilla de implementar, lo que abre un enorme abanico de posibilidades de utilización.

En el siguiente capítulo se describirán los experimentos realizados con HARP en WSN y se demostrará el ahorro energético y las altas prestaciones que el protocolo ofrece a la aplicación, distribuyendo entre todos los nodos el gasto de energía en cada ronda de operación y, en definitiva, extendiendo el tiempo de vida de la red de sensores.

Capítulo 7

Resultados y aplicaciones

7.1. Descripción de los experimentos

La descripción de los experimentos se llevará a cabo desarrollando la estructura principal descrita en la Sección 4.4, según la cual, para cada uno de los escenarios objeto de estudio, se describirán los objetivos propuestos, los parámetros y valores de configuración específicos y las variables de salida analizadas. En cuanto a las características generales de los experimentos llevados a cabo en este tema se exponen a continuación los datos más relevantes.

Los experimentos se llevarán a cabo a través de simulaciones realizadas en el simulador WiNS descrito en la Sección 4, algunas de las cuales han sido complementadas y corroboradas con otras nuevas ejecutadas en el entorno de simulación MATLAB. En esta herramienta se han programado las funcionalidades específicas de HARP y s-HARP y se ha incluido el código correspondiente a los protocolos LEACH [64] y SEP [123] proporcionado por los autores de este último. Por último, se ha integrado todo el código y se han lanzado simulaciones simultáneas en las que se compara, ronda por ronda, el rendimiento de los algoritmos ejecutados.

El elemento fundamental de la simulación es el nodo. Los nodos se identifican por un ID único en la red, aunque para redes extensas con un número elevado de nodos esta exclusividad puede ser inviable. En este caso, la selección de un ID por parte de cada nodo puede realizarse de forma aleatoria y con una duración temporal ligada a la duración de los *clusters*. Todos ellos disponen de una antena omni-direccional para su comunicación con otros nodos y de capacidad de agregación de datos y de encaminamiento. Esta comunicación se efectúa a través de un enlace inalámbrico bidireccional y asimétrico entre cualquier par de nodos de la red, el cual puede romperse según una probabilidad determinada. Los nodos de la red pueden ajustar su potencia de transmisión dependiendo del radio de cobertura que se desee o de la distancia a la que se encuentre el nodo receptor. En cuanto al tráfico de red se ha utilizado fuentes de flujo constante (CBR), estableciendo a priori el valor del parámetro de tasa de generación de paquetes.

Asimismo, el modelo de consumo energético y de transmisión seguido es el que se ha detallado en la Sección 6.2, en el cual se tienen en cuenta los efectos de la propagación en espacio libre para el cálculo del consumo de energía al transmitir si la distancia es menor que un umbral predefinido, mientras que si la distancia aumenta hay que tener en cuenta además los efectos producidos por la propagación multi-camino o *multipath fading*.

El principal objetivo de este estudio es la evaluación del consumo de energía del protocolo propuesto y su comparación con el de otros protocolos representativos de WSN. Además del gasto energético, se analizarán otros factores relacionados con la medida de las prestaciones del protocolo.

Para cumplir este objetivo, se ha llevado a cabo la selección de un conjunto de topologías de red, las cuales, junto con la definición de los parámetros representativos de una WSN, forman cada una de ellas un escenario de simulación. El escenario básico en WSN ha consistido en una red de 200 nodos sensores distri-

buidos aleatoriamente en un área de 500×500 metros cuadrados, situándose el nodo sumidero en el centro de la misma. Partiendo de este escenario principal, se han utilizado otros alternativos para el estudio del impacto que tienen parámetros tan fundamentales en una WSN como son el número de nodos y el área de la red de sensores en el consumo energético total del sistema. Para el primer estudio de sensibilidad, además del escenario básico, se han simulado otros dos escenarios consistentes en el despliegue aleatorio de 100 y 500 nodos distribuidos en un área de 500×500 metros cuadrados. Para el segundo caso, se han ejecutado simulaciones de 200 sensores desplegados en áreas de 200×200 y 700×700 metros cuadrados. Por último, para el estudio de aplicabilidad de HARP a otros tipos de redes se han empleado además redes triangulares de 60 y 168 nodos sensores y actuadores. En todos los escenarios se ha utilizado un solo nodo sumidero que actuará como *Root* o nodo raíz del árbol correspondiente.

7.2. Resultados

En este apartado se presentan los resultados experimentales de las simulaciones en diferentes escenarios y su comparación con otros protocolos de encaminamiento ampliamente aceptados por la comunidad científica. En primer lugar, se analizarán las prestaciones de HARP en aplicaciones típicas de utilización de las WSN como son las de monitorización, recolección o captura de datos del entorno y envío hacia el nodo sumidero para su posterior procesado. En estos casos, HARP se comparará con protocolos representativos como LEACH en redes homogéneas y SEP en redes heterogéneas. A continuación, se presentan diferentes aplicaciones de HARP que hacen uso de su capacidad para gestionar eficientemente la movilidad de los nodos y los fallos producidos en la red.

7.2.1. Prestaciones de HARP en redes homogéneas

En primer lugar se analizará el comportamiento de HARP en WSN homogéneas, con nodos sensores con las mismas propiedades y la misma energía inicial. Este comportamiento será comparado con otros protocolos basados en *clusters* por ser éstos los que mejor responden a los requerimientos en este tipo de aplicaciones y redes.

A continuación se describen los experimentos llevados a cabo, siguiendo la pauta descrita en la sección anterior.

7.2.1.1. Descripción de experimentos

Objetivos

1. Análisis de prestaciones de HARP y s-HARP frente al protocolo más representativo de WSN.

VARIABLES ANALIZADAS

Para estudiar las prestaciones de HARP y s-HARP en redes homogéneas en toda su extensión, se registrarán los valores de las siguientes variables:

- *Tiempo de vida útil de la red*: se define como el intervalo de tiempo desde el inicio de la operación de la red hasta la muerte del 75 % de los nodos de la misma, debida al agotamiento de su batería. Este porcentaje puede cambiar para diferentes aplicaciones debido a que, si bien hasta la muerte del último nodo de la red ésta puede todavía recolectar datos del entorno, a menudo la disminución del número de sensores capaces de recoger datos del entorno hace que esta información no sea realmente fiable ni describa con exactitud lo que ocurre en el entorno observado. Sin tener en cuenta otros factores

externos, se puede asumir que un nodo muere cuando su energía es igual o menor que cero después de completar una operación.

- *Región estable*: es el intervalo de tiempo que transcurre desde el inicio de la operación hasta la muerte del primer nodo de la red. Este parámetro está asociado con la estabilidad que ofrece un protocolo en cuanto a su operación en la red. Es evidente que a mayor región estable, mayor número de datos recogidos y, en consecuencia, mayor fiabilidad del algoritmo.
- *Consumo energético*: es la cantidad de energía consumida por la red en cada ronda de operación completa y el consumo agregado, es decir, la energía total agregada consumida desde el inicio de la simulación hasta una ronda determinada.
- *Rendimiento*: es la tasa total de datos enviados a través de la red en cada ronda, incluyendo los paquetes transmitidos en el árbol inter-*cluster* desde los CH hacia el nodo sumidero y también los datos intercambiados en los árboles intra-*cluster* entre los *nonCH* y los CH.
- *Número de CH por ronda*: permite evaluar la fiabilidad del protocolo y su capacidad para cubrir el área a evaluar y tomar las medidas del entorno de manera uniforme.

Parámetros

Se ha realizado un conjunto de 50 simulaciones completas de 1000 rondas de operación cada una de ellas, generando valores de las variables anteriores a partir del conjunto de parámetros incluido en la Tabla 7.1. Los resultados finales mostrados han sido calculados como el valor medio de los obtenidos en el conjunto de simulaciones ejecutadas.

Para realizar una comparación justa entre HARP y LEACH se ha simulado la arquitectura del primero con el umbral definido en LEACH para la selección de

CH, aislando de esta forma el rendimiento de la arquitectura propuesta del método de selección de CH.

Se asumirá que la información recogida en los *clusters* es redundante y está correlada espacialmente, por lo que en los árboles intra-*cluster* el CH y los *nonCH* de niveles intermedios utilizarán fusión de datos en los paquetes recibidos de los nodos asociados y siempre podrán agregar estos datos en un solo paquete. Debido a que estos datos agregados procedentes de los *clusters* son menos redundantes que los transmitidos dentro de los mismos, se asume que la agregación en el árbol inter-*cluster* no es aconsejable para mantener la fiabilidad de la información recogida, por lo que un CH reenviará a su CH padre los datos recibidos en este árbol jerárquico y no efectuará fusión ni agregación alguna de los paquetes recibidos.

Las variables α , β , γ y χ optimizan el rendimiento del protocolo propuesto y han sido ajustadas mediante simulación para este escenario en particular.

7.2.1.2. Resultados

A continuación se muestran los resultados obtenidos para cada una de las variables representativas de las prestaciones en este escenario.

Tiempo de vida útil de la red

La Figura 7.1 indica el número de nodos que no han agotado su energía y que siguen estando vivos, para cada ronda de operación. De su observación se concluye lo siguiente:

- El rendimiento de las operaciones con HARP y s-HARP supera ampliamente al correspondiente a LEACH en cuanto al tiempo de vida útil de la red, siendo cerca de 2 y 2.5 veces mayor respectivamente que el proporcionado por este protocolo representativo de las WSN.

Parámetro	Valor
Dimensiones de la red	500×500 (m ²)
Número de nodos sensores	200
Distribución de los sensores en la red	aleatoria
Protocolo 1	HARP
Protocolo 2	s-HARP
Protocolo 3	LEACH
Posición del nodo sumidero	(250, 250) m
E_0 (energía inicial)	2 J
E_{elec}	50 nJ/bit
ϵ_{fs}	10 pJ/bit·m ²
ϵ_{amp}	0.0013 pJ/bit·m ⁴
E_{elec}	50 nJ/bit
E_{fuse}	5 nJ/bit
p_{CH}	0.05
d_0	$\sqrt{\epsilon_{fs}/\epsilon_{amp}} = 87.7$ m
Tamaño del paquete de datos	525 bytes
Tamaño del paquete de control	25 bytes
Número de rondas	1000
Número de <i>frames</i> por ronda	6
α	0.3
β	0.4
γ	0.3
χ	1

Tabla 7.1: Parámetros de simulación de prestaciones de HARP en redes homogéneas.

- HARP consigue prolongar el tiempo de vida de los nodos con respecto a LEACH debido a las estructuras jerárquicas creadas, las cuales se muestran muy efectivas en el ahorro de energía. A su vez, el método de elección de CH en s-HARP aumenta la eficiencia del protocolo y le permite distribuir de forma más uniforme el consumo energético entre todos los nodos, pro-

longando el tiempo en el que muere el primero de ellos. Esta circunstancia conlleva que la muerte de la gran mayoría se produzca en un corto espacio de tiempo, maximizando la efectividad en la monitorización del entorno con respecto a LEACH, en donde existe una reducción lenta pero constante de los nodos útiles capaces de efectuar esta tarea.

Región estable

A partir de la misma Figura 7.1 se observa que:

- Con el protocolo LEACH el primer nodo muere en la ronda 21, mientras que HARP y s-HARP extienden este tiempo hasta las rondas 292 y 506 respectivamente.
- Estos datos hacen que la región estable generada por s-HARP sea cerca de 25 veces mayor que la correspondiente a LEACH.
- Por su parte, HARP presenta una región estable aproximadamente 14 veces mayor que LEACH.

Consumo energético

La Figura 7.2 muestra los resultados de las simulaciones en términos del consumo energético incurrido en cada una de las 100 primeras rondas de operación. Por su parte, la Figura 7.3 presentas los consumos agregados de la red después de 100 y 600 rondas.

Los resultados presentados generan las siguientes conclusiones:

- En cada ronda s-HARP consigue mantener un consumo energético menor y más uniforme que LEACH. Esta mejora es debida a la distribución equitativa y balanceada de la carga asignada a los nodos a través del proceso de selección

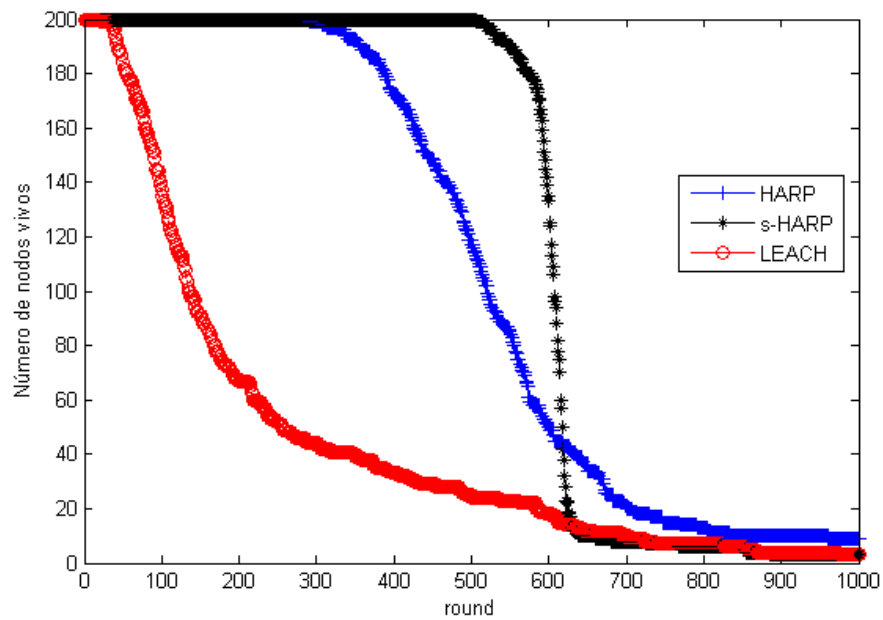


Figura 7.1: Número de nodos con energía remanente en función del número de rondas de operación.

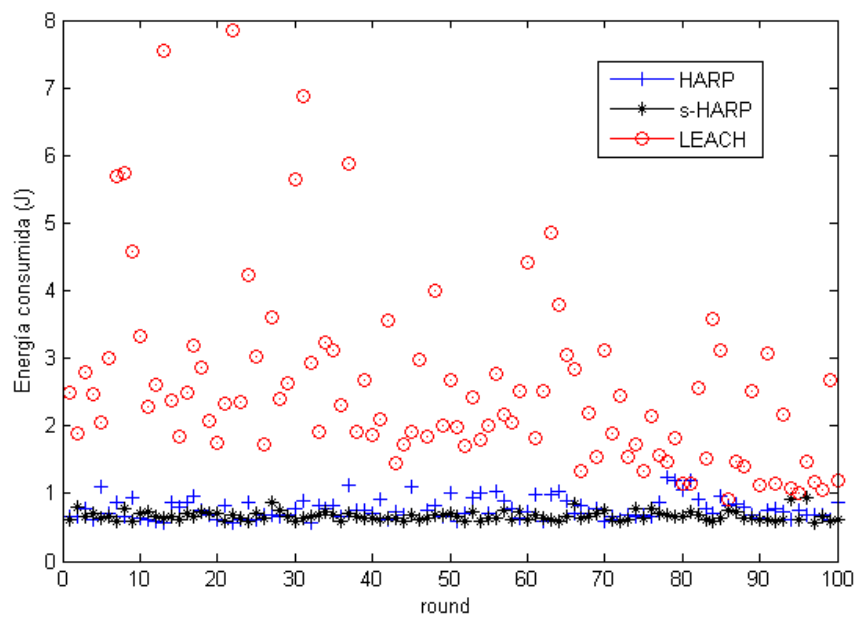
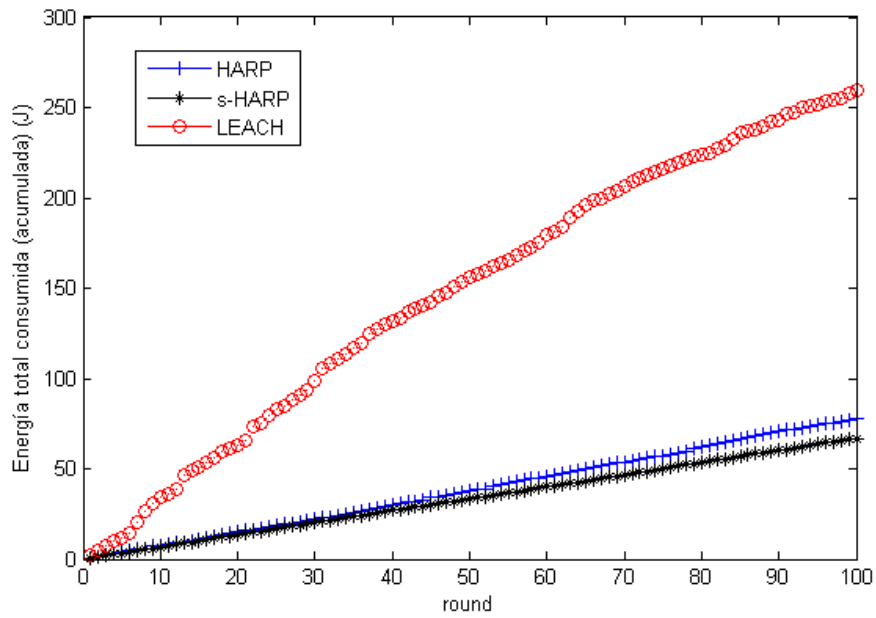
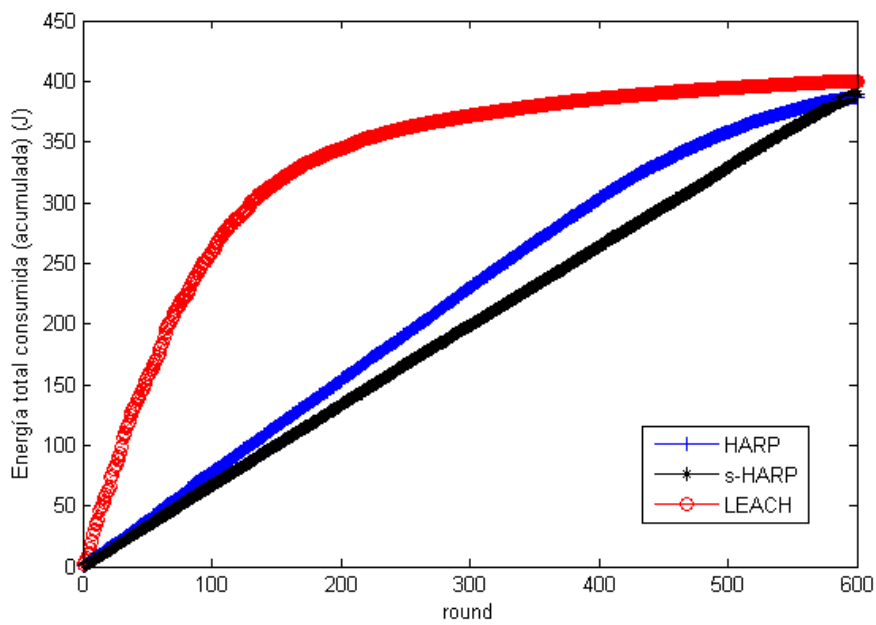


Figura 7.2: Consumo energético por ronda (primeras 100 rondas).



(a)



(b)

Figura 7.3: Consumo energético total acumulado a) después de 100 rondas; b) después de 600 rondas.

de CH de s-HARP, que considera la energía remanente del nodo. Ésta es a su vez la razón de las altas prestaciones de s-HARP relativas al tiempo de vida y la región estable mostradas anteriormente.

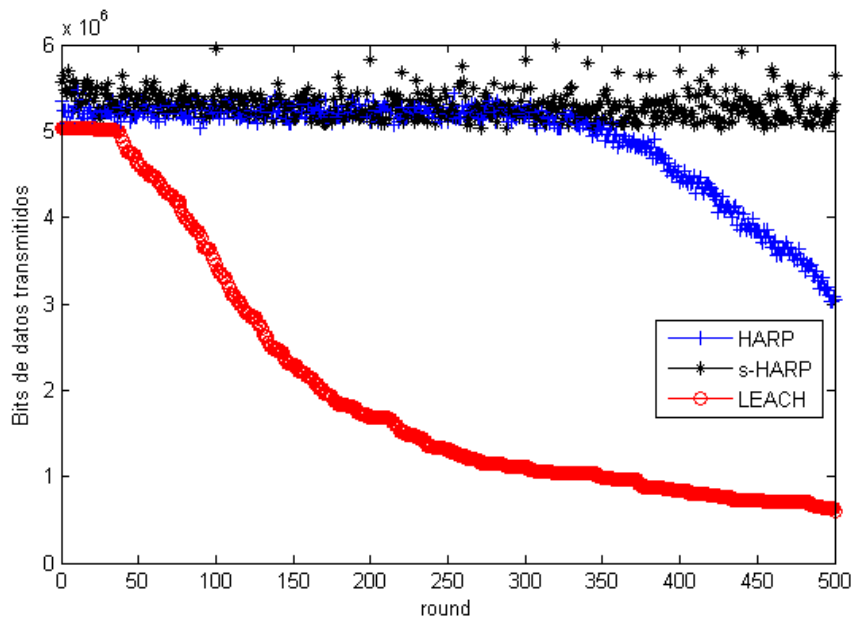
- HARP ofrece una distribución de energía estable, al minimizar los grandes consumos energéticos debidos a transmisiones de paquetes a grandes distancias a través de sus árboles jerárquicos.
- HARP y s-HARP ofrecen un consumo energético lineal en contraposición con el presentado en LEACH, según se muestra en la Figura 7.3.
- Tras 100 rondas HARP y s-HARP muestran un consumo 4 veces menor que LEACH.
- El consumo acumulado tras 600 rondas en los tres protocolos se iguala debido a que el número de nodos muertos en la misma es muy elevado y, por tanto, la optimización de la operación de HARP y s-HARP pierde su efecto.

Rendimiento

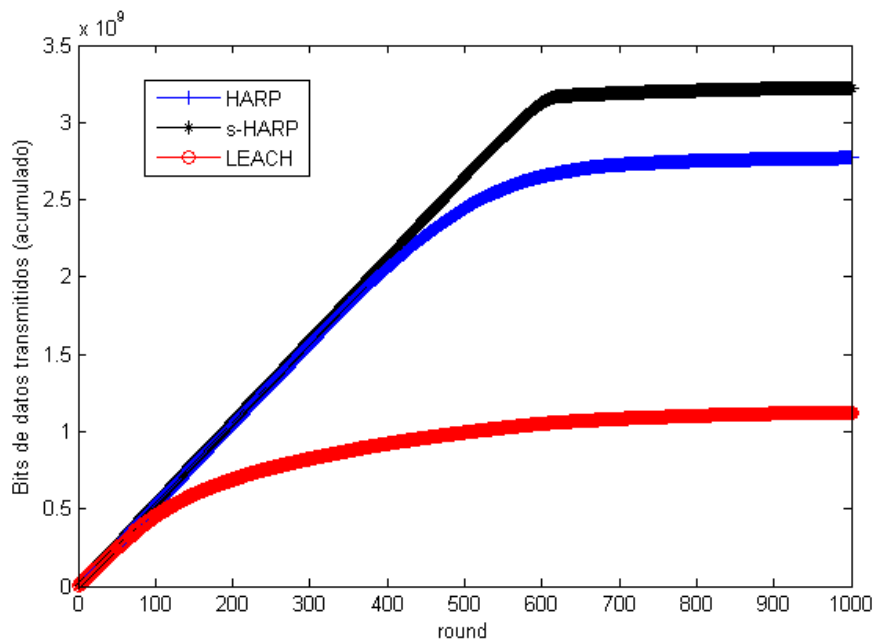
El rendimiento de los diferentes protocolos comparados se muestra en la Figura 7.4. A partir de la observación de la misma se puede concluir que:

- LEACH no es capaz de atender la demanda de datos exigida por la aplicación debido a la veloz disminución del número de nodos activos en la red, mientras que HARP y s-HARP mantienen estable durante mucho más tiempo la tasa de paquetes de datos transmitida hacia el nodo sumidero.
- El número de paquetes de datos transmitidos por HARP y s-HARP es alrededor de 2.5 y 3 veces mayor que en LEACH respectivamente.

Número de CH por ronda



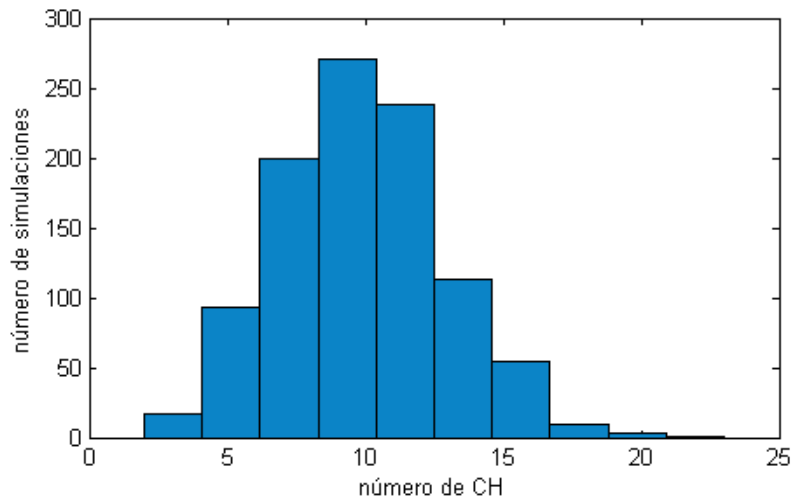
(a)



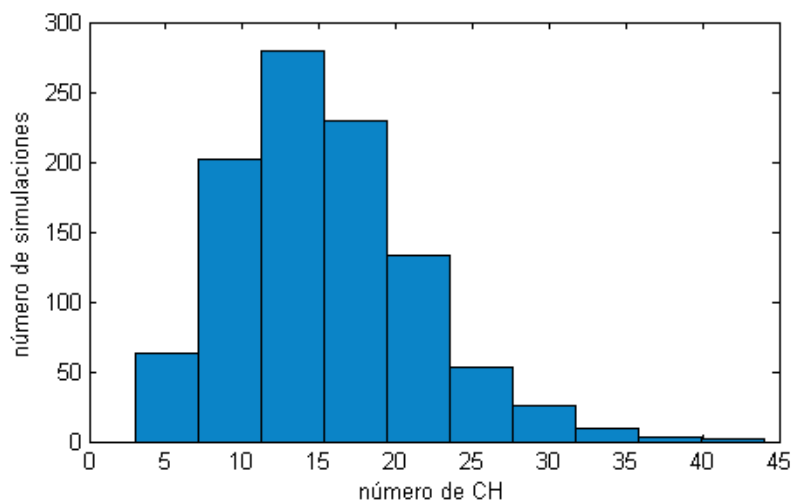
(b)

Figura 7.4: Rendimiento de LEACH, HARP y s-HARP, medido como el número total de datos recogidos y enviados por la red en cada ronda.

Por último, la Figura 7.5 muestra un histograma del número de *clusters* por ronda en LEACH y s-HARP cuando están operando en la fase estable, es decir,



(a)



(b)

Figura 7.5: Histograma del número de clusters por ronda de operación para: a) LEACH; b) s-HARP.

con todos sus nodos activos monitorizando y enviado información. La observación de esta figura muestra que:

- El número de *clusters* en LEACH tiende hacia una distribución gaussiana centrada en 10 (el valor medio calculado de las simulaciones es 9.99). Este valor es coherente con el resultado esperado ya que el valor asignado al

parámetro correspondiente a la probabilidad de ser CH es $p_{CH} = 0.05$ y el número de nodos en la red simulada es de 200, lo que implica que el número esperado de CH por ronda es de 10.

- En s-HARP se observa una distribución diferente de *clusters*, tendiendo hacia una *pdf* Rayleigh con un valor medio de CH de 15.4, un 50% superior al observado en LEACH. Este mayor número de CH favorece en gran medida las labores de monitorización de la red y aporta consistencia en la recogida de datos, ya que no hay que olvidar que el nodo sumidero en este tipo de tareas recibe los datos agregados correspondientes a cada uno de los *clusters*, lo que implica que a mayor número de *clusters* mayor será también el número de paquetes de datos recibidos y, en consecuencia, más completa será la visión que se tiene de la red. Este mayor número de CH favorece además el ahorro energético dentro de los *clusters*, en cuanto a que reduce las distancias medias de transmisión de los nodos asociados a los mismos.

Resumen

En resumen, tras la evaluación de HARP y s-HARP, se concluye que ambos protocolos ofrecen mejores prestaciones que LEACH en todas las variables representativas analizadas. HARP y s-HARP muestran un comportamiento estable y eficiente de los *clusters* y las estructuras jerárquicas formadas a lo largo de las rondas de operación de la red. Adicionalmente, son capaces de incrementar el tiempo de vida de la red y balancear la carga energética de los nodos a través de la selección eficiente de CH en cada ronda, así como de maximizar la tasa de datos enviada al nodo sumidero y distribuir de forma eficiente el número de CH a lo largo de la red.

7.2.2. Prestaciones de HARP en redes heterogéneas

Existe una multitud de aplicaciones de WSN en las que se despliega un porcentaje de nodos sensores equipados con mayor capacidad energética que el resto. Una de estas aplicaciones es la así llamada *the re-energization of sensor networks* [123]: dado que el tiempo de vida de las WSN está limitado por la energía de los nodos, puede existir la necesidad de prolongar esta vida útil a través del despliegue de nuevos nodos en la red, los cuales estarán equipados con mayor energía que los que ya estaban en la red, lo que introduce una clara heterogeneidad en términos energéticos. Como alternativa, asumiendo que el coste de un nodo sensor es mucho mayor que el de sus baterías, otro método para prolongar la vida de la red consistiría en recargar las mismas (a través, por ejemplo, de placas solares) en lugar de introducir nuevos sensores. Esta heterogeneidad energética en la red puede resultar también de la propia operación de los nodos cuando éstos consumen cantidades distintas de energía debido a las diferentes distancias de transmisión que deben salvar, a desvanecimientos a pequeña escala en algunos enlaces o a las propias características morfológicas del campo en el que están desplegados (como por ejemplo, un terreno irregular).

7.2.2.1. Descripción de experimentos

Objetivos

1. Análisis de prestaciones de HARP y s-HARP frente a protocolos representativos de WSN heterogéneas.

VARIABLES ANALIZADAS

El presente análisis de prestaciones se realiza en función de las mismas variables que la sección anterior, exceptuando el número de CH por ronda.

Parámetros

La Tabla 7.2 contiene los parámetros utilizados en las simulaciones, en las cuales se ha introducido un 10% de nodos (denominados *nodos avanzados*) con el doble de energía que el resto de los sensores de la red. El número de simulaciones es el mismo que para redes homogéneas.

En este apartado se estudia el impacto de esta heterogeneidad, analizando el comportamiento de HARP y comparándolo con el ofrecido por LEACH y SEP, protocolo específicamente diseñado para tener en cuenta esta heterogeneidad en la red en términos de energía. Al igual que en el apartado anterior, se estudian HARP y s-HARP, habiéndose ajustado las variables α , β , γ y χ mediante simulación para este último.

7.2.2.2. Resultados

A continuación se presentan los resultados de las simulaciones descritas, particularizados por cada una de las variables indicadas.

Tiempo de vida útil de la red

La Figura 7.6 muestra los nodos que no han consumido toda su energía en cada una de las rondas de operación. Las conclusiones extraídas a partir del estudio de la misma son las siguientes:

- Las redes heterogéneas prolongan el tiempo transcurrido hasta la muerte del primer nodo. El primer nodo muere en LEACH en la ronda 34, lo que supone una extensión de 13 rondas con respecto al escenario homogéneo.
- El protocolo SEP presenta la muerte de su primer nodo en la ronda 42. La mejora respecto a LEACH se debe a que SEP tiene en cuenta la diferente

Parámetro	Valor
Dimensiones de la red	500×500 (m ²)
Número de nodos sensores	200
Distribución de los sensores en la red	aleatoria
Protocolo 1	HARP
Protocolo 2	s-HARP
Protocolo 3	LEACH
Protocolo 4	SEP
Posición del nodo sumidero	(250, 250) m
E_0 (energía inicial de los nodos normales)	2 J
$E_{0_{advanced}}$ (energía inicial de los nodos avanzados)	4 J
% de nodos avanzados	10 %
E_{elec}	50 nJ/bit
ϵ_{fs}	10 pJ/bit·m ²
ϵ_{amp}	0.0013 pJ/bit·m ⁴
E_{elec}	50 nJ/bit
E_{fuse}	5 nJ/bit
$p_{CH_{LEACH}}$	0.05
p_{CH}	1
d_0	$\sqrt{\epsilon_{fs}/\epsilon_{amp}} = 87.7$ m
Tamaño del paquete de datos	525 bytes
Tamaño del paquete de control	25 bytes
Número de rondas	1000
Número de <i>frames</i> por ronda	6
α	0.2
β	0.6
γ	0.2
χ	1

Tabla 7.2: Parámetros de simulación de prestaciones de HARP en redes heterogéneas.

escala energética presente en la red, siendo capaz de realizar una selección más eficiente de los CH en cada una de las rondas de operación.

- El número de nodos vivos en SEP siempre es mayor que el de LEACH.
- El primer nodo muere en s-HARP en la ronda 565 mientras que en HARP lo hace alrededor de la ronda 300, mejorando ambos ampliamente las prestaciones de LEACH y SEP.
- HARP y s-HARP presentan similares tiempos de vida útil de la red. Esta variable es alrededor de 2.5 y 2 veces mayor que las conseguidas por LEACH y SEP respectivamente. Esta mejora es debida al mayor rendimiento de las estructuras jerárquicas formadas.

Región estable

Por su parte, a partir de la misma Figura 7.1 se observa que:

- s-HARP consigue aumentar la región estable en factores de 17 y 13 las conseguidas por LEACH y SEP respectivamente.
- HARP consigue un aumento de 8 y 6 veces la región estable alcanzada por LEACH y SEP respectivamente.

Consumo energético

La Figura 7.7 muestra los datos relativos al consumo energético de los cuatro protocolos bajo estudio en las 100 primeras rondas de operación, tanto por ronda como el total acumulado. Como conclusiones se pueden destacar:

- HARP y s-HARP consiguen una distribución mucho más uniforme del consumo de energía en la red al igual que en el caso de una WSN homogénea.
- s-HARP muestra un consumo siempre menor que los de LEACH y SEP, repartiéndolo equitativamente entre los distintos nodos debido a un proceso

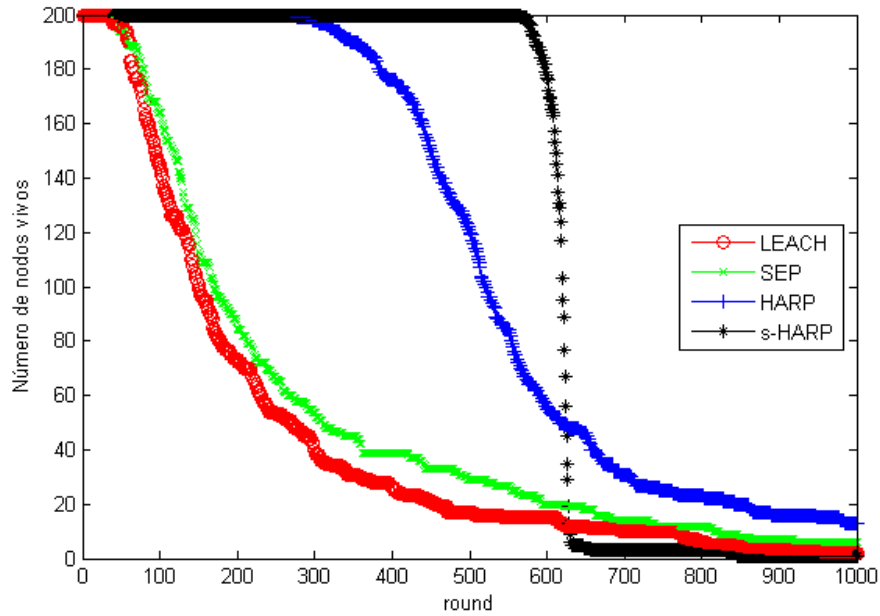


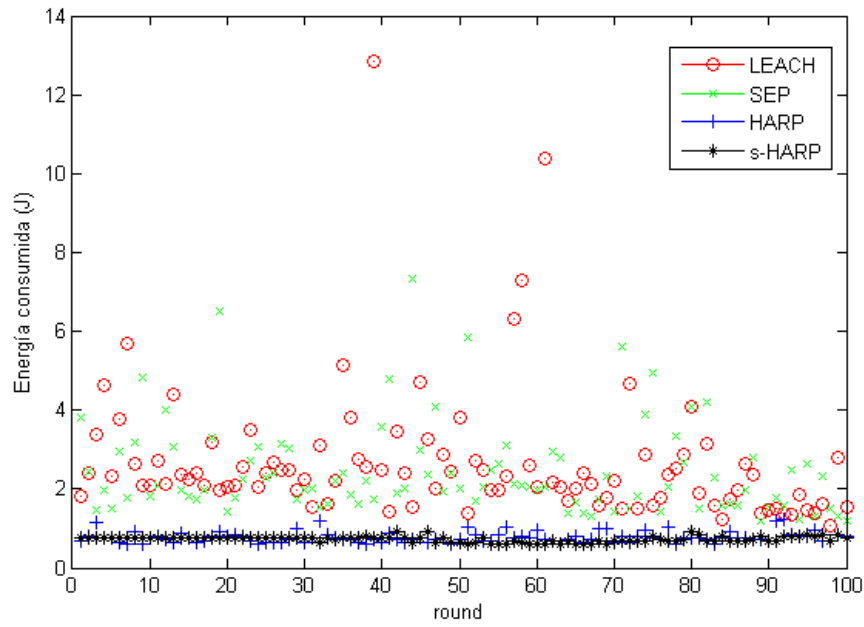
Figura 7.6: Número de nodos con energía remanente en función del número de rondas de operación.

más eficiente de selección de CH, el cual tiene en cuenta la energía de un nodo con respecto a los de su alrededor. Este aspecto adquiere mayor importancia en redes heterogéneas, en las que los protocolos deben que tratar con diferentes niveles energéticos en los nodos.

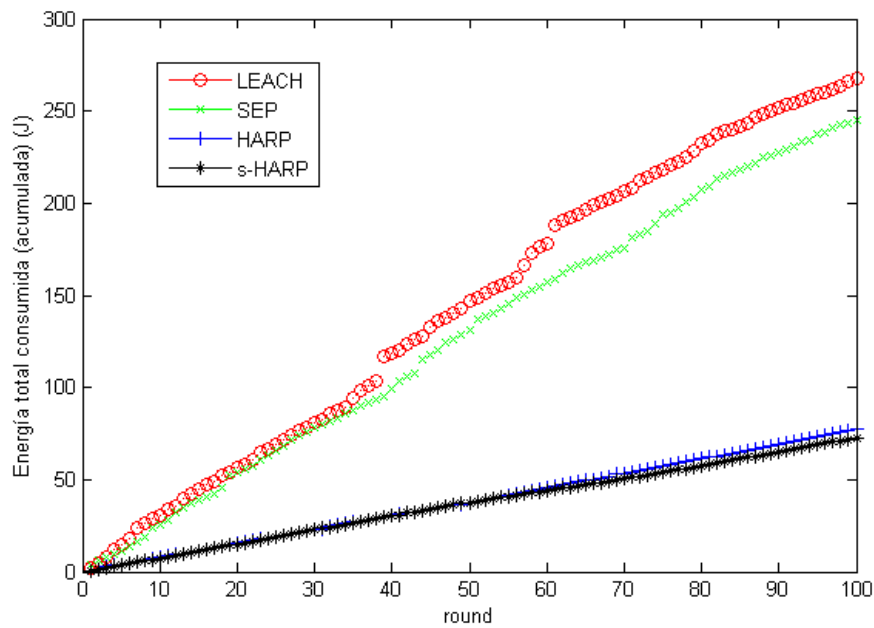
- HARP y s-HARP reducen en 3.3 y 3.5 veces los consumos energéticos totales acumulados en la red incurridos por LEACH y SEP respectivamente, según se muestra en la Figura 7.7 b).

Rendimiento

El rendimiento de los diferentes protocolos analizados se muestra en la Figura 7.8 a través del número de bits de datos transmitido en cada ronda. Su observación genera la siguiente conclusión:



(a)



(b)

Figura 7.7: Consumo energético en las primeras 100 rondas: a) por ronda; b) acumulada.

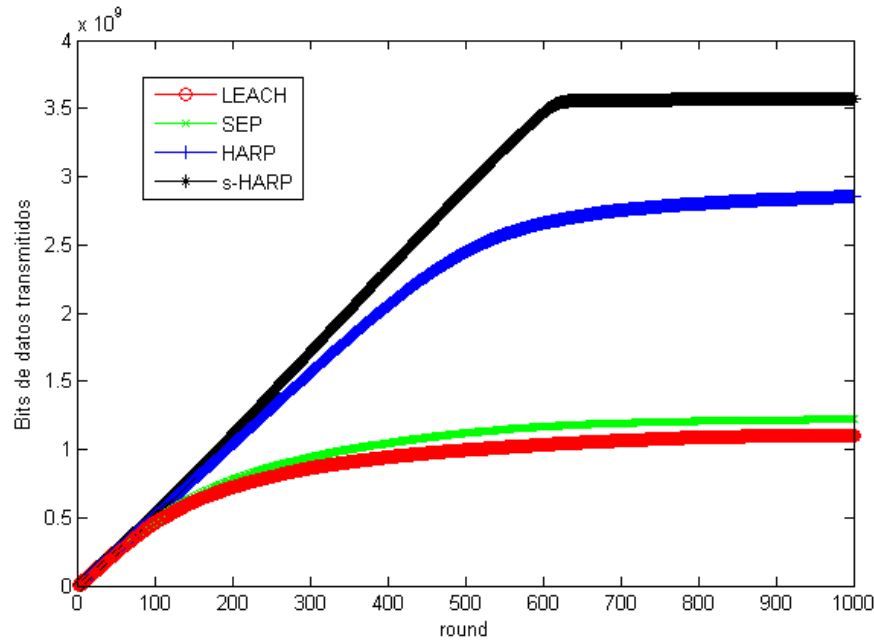


Figura 7.8: Rendimiento de LEACH, SEP, HARP y s-HARP, medido como el número total de datos recogidos y enviados por la red en cada ronda.

- s-HARP es el protocolo más efectivo entre todos los comparados debido a que consigue mantener un mayor número de nodos vivos durante un mayor espacio de tiempo con relación al resto, hasta que mueren en un número de rondas relativamente pequeño.

7.2.3. Red homogénea frente a red heterogénea

Para finalizar esta sección, la Figura 7.9 muestra una comparación del rendimiento de s-HARP y LEACH en WSN homogéneas y heterogéneas. Es interesante resaltar que el proceso de selección de CH en s-HARP consigue aprovechar mejor las ventajas que ofrece una red heterogénea con respecto a la homogénea y aumenta la región estable del protocolo en mayor medida (60 rondas) que LEACH (aumento de tan sólo 13 rondas respecto del escenario homogéneo).

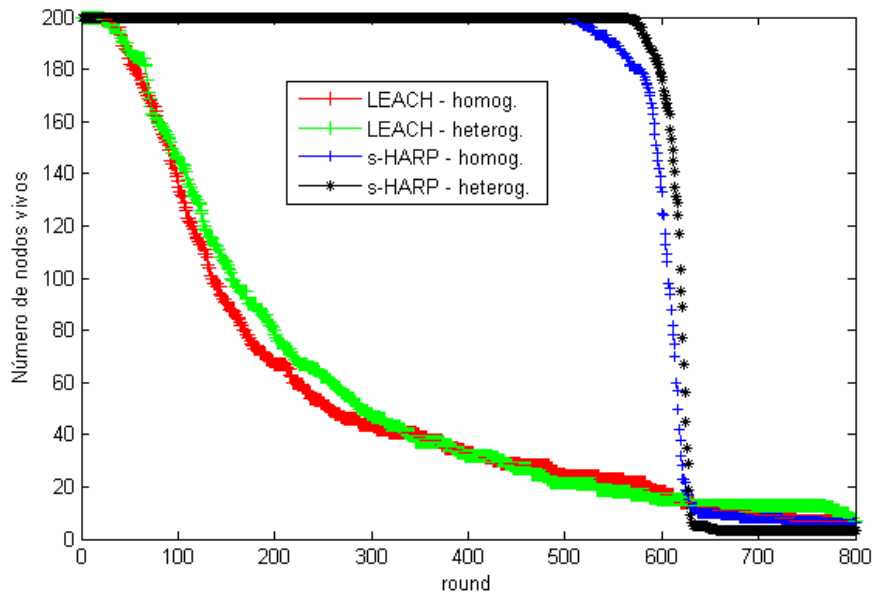


Figura 7.9: Comparación de la eficiencia energética de s-HARP y LEACH en WSN homogéneas y heterogéneas.

7.2.4. Otros escenarios de simulación

Tal y como se indicó al inicio de este capítulo, se han realizado simulaciones en otros escenarios diferentes del básico con el objetivo de estudiar el impacto que tiene en el consumo energético de la red la variación, por un lado, del número de sensores que la forman y, por el otro, del área de despliegue de la misma.

7.2.4.1. Descripción de experimentos

Objetivos

1. Estudio de sensibilidad mediante el análisis del impacto del número de nodos y del área de despliegue en el consumo energético de la red operando con HARP y LEACH.

VARIABLES ANALIZADAS

Se estudiará el consumo energético acumulado de la red de sensores transcurridas las 100 primeras rondas de operación. La variación del tamaño de la región estable en función de los parámetros descritos será otra de las variables analizadas en este apartado.

Parámetros

Los parámetros utilizados en estas simulaciones para cada uno de los escenarios y su número son los mismos que en el caso de redes homogéneas. Los distintos escenarios definidos son los siguientes: 200 sensores en 200×200 (m²), 200 sensores en 700×700 (m²), 100 sensores en 500×500 (m²) y 500 sensores en 500×500 (m²). En todos ellos se realiza un despliegue aleatorio de los nodos.

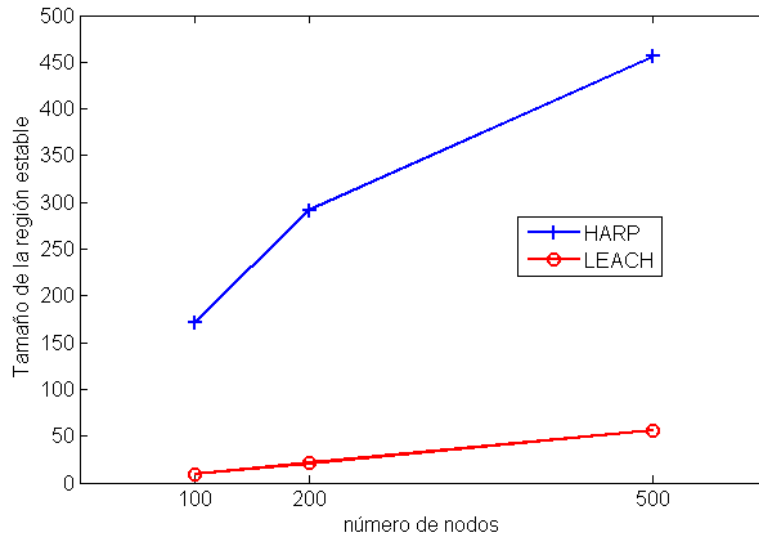
7.2.4.2. Resultados

A continuación se presentan los resultados de las simulaciones en cada uno de los escenarios descritos.

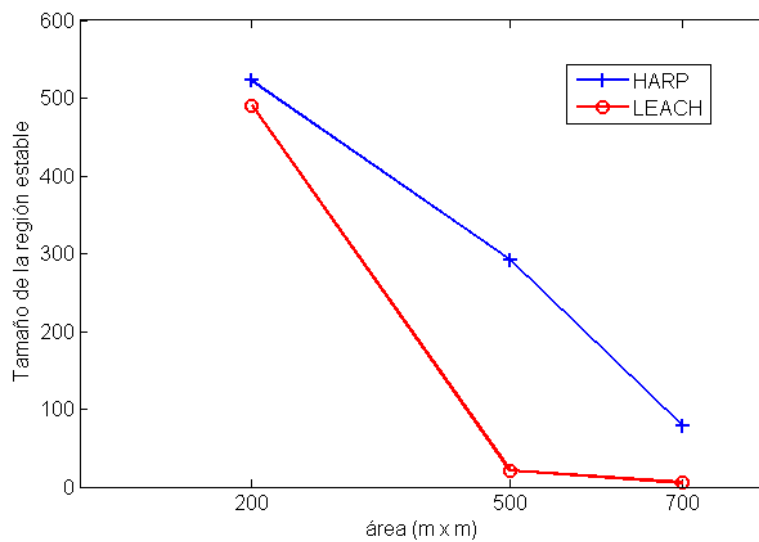
Región estable

El tamaño de la región estable se representa en las Figuras 7.10 a) y b) para distintos valores del número de nodos y del área de despliegue, respectivamente. De ellas puede deducirse lo siguiente:

- El aumento de la región estable en HARP es mucho más acentuado y abrupto conforme se aumenta el número de nodos en la red con respecto a LEACH, cuya variación es más lineal.
- Al aumentar el tamaño de la red, los nodos deben cubrir en media mayores distancias de transmisión, con lo que consumirán más energía y morirán antes. Esta muerte del primer nodo partir de 200×200 (m²) ocurre cada vez



(a)

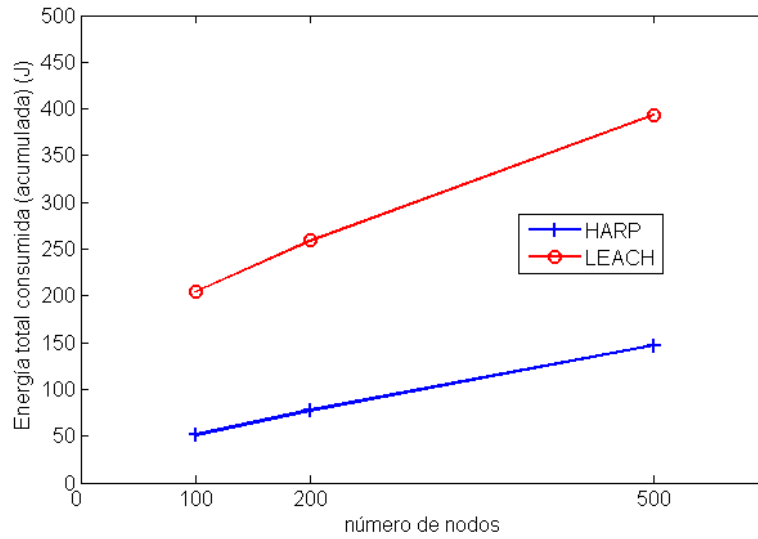


(b)

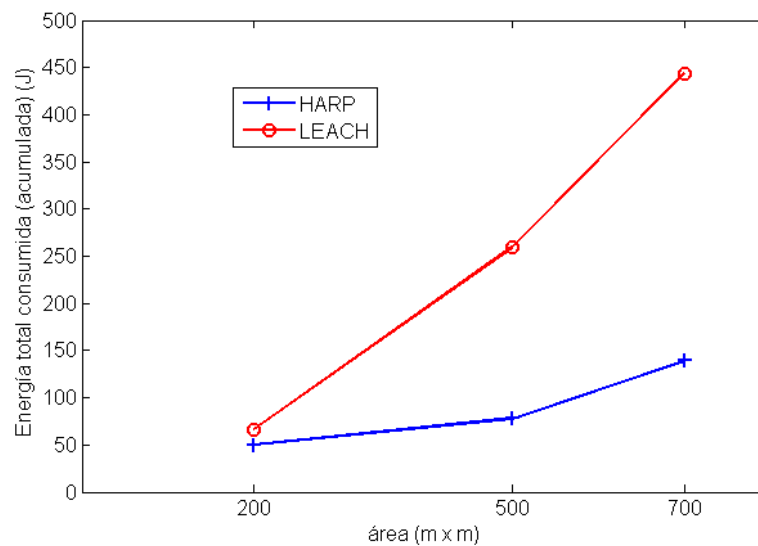
Figura 7.10: Variación del tamaño de la región estable en LEACH y HARP para distintos escenarios de simulación: a) red de 500×500 (m^2) y distinto número de sensores; b) escenarios con 200 sensores y distintas superficies de red.

mucho antes en LEACH que en HARP conforme aumenta el tamaño de la red.

Consumo energético



(a)



(b)

Figura 7.11: Variación del consumo de energía de la red en LEACH y HARP para distintos escenarios de simulación: a) red de 500×500 (m^2) y distinto número de sensores; b) escenarios con 200 sensores y distintas superficies de red.

La Figura 7.11 a) muestra los resultados obtenidos para el consumo energético acumulado en la red en las primeras 100 rondas de operación cuando se modifica el número de sensores en una red de 500×500 (m^2). Se han seleccionado 100, 200 y

500 nodos. Por su parte, en la Figura 7.11 b) se refleja el mismo parámetro cuando se fija el número de nodos de la red en 200 y se modifica el área de despliegue de la misma. A la vista de estas figuras puede concluirse lo siguiente:

- HARP consigue crear árboles jerárquicos más eficientes que ahorran energía con respecto a LEACH, por lo que a medida que hay más nodos en la red la cooperación entre ellos es mayor y en consecuencia el incremento del gasto es menor que en LEACH.
- Al aumentar el área de despliegue, LEACH es muy poco escalable debido a las distancias mayores que tienen que salvar los nodos y, en especial, los CH para alcanzar el sumidero. HARP es muy escalable debido a las estructuras jerárquicas inter e intra-*cluster* creadas, las cuales permiten cubrir áreas geográficas extensas manteniendo estable el consumo de energía.

7.3. Aplicaciones de HARP

La arquitectura del protocolo propuesta en esta Tesis Doctoral puede aplicarse en situaciones que requieran de un protocolo de encaminamiento capaz de gestionar de forma eficiente cualquier tipo de tráfico en la red y hacer frente a las circunstancias que pueden limitar la eficiencia del protocolo, tales como fallos en enlaces o movilidad de nodos. En este sentido, HARP puede ser utilizado en aplicaciones distintas de las relacionadas con WSN, para las que fue originalmente concebido, extendiendo su uso al caso genérico de las MANET.

En una red ad hoc inalámbrica HARP es capaz de crear la estructura jerárquica explicada en el Capítulo 3 y de encaminar los paquetes de forma eficiente. Ante las características propias de las MANET como movilidad de nodos y entornos cambiantes, HARP hace uso de sus mecanismos de recuperación ante fallos para hacerlas frente.

En los apartados siguientes se incluyen distintos escenarios de aplicación de HARP. Además de poder comparar su rendimiento con el de otros protocolos representativos de las redes MANET, las simulaciones realizadas permiten a su vez evaluar las prestaciones de HARP ante la presencia de fallos de la red y los efectos creados por la movilidad de los nodos. LEACH y SEP no definen ningún mecanismo para gestionar dichas situaciones.

En particular, se analizarán las prestaciones de HARP en (i) redes inalámbricas de sensores y actuadores (WSAN) basadas en ZigBee y (ii) como plataforma de encaminamiento para la gestión de comunicaciones en un sistema de transporte público entre vehículos móviles y el centro de gestión.

7.3.1. HARP en WSAN

Las WSAN tienen como finalidad la recogida de datos del entorno y la posterior ejecución de tareas en la red como resultado del análisis y procesado de dichos datos, todo ello de forma distribuida. Las WSAN suelen estar compuestas por un gran número de nodos sensores con capacidades limitadas, encargados de monitorizar la red y extraer información del entorno, enviándola posteriormente a los actuadores. Estos nodos, presentes en la red generalmente en menor cantidad que los nodos sensores, tienen capacidades aumentadas en cuanto a energía, procesado, comunicación e incluso movimiento, aunque también suelen ser limitadas. Los actuadores reciben la información procedente de los sensores sobre los eventos ocurridos y son capaces de reaccionar antes las mismas y ejecutar tareas específicas en un área determinada de la red [22]. Esta decisión acerca de la ejecución de una tarea puede tomarse de forma autónoma o coordinada con otros actuadores (arquitectura automatizada [114]), o también comunicándose con un gestor de tareas centralizado (nodo sumidero), capaz de procesar la información procedente

de uno o más actuadores y determinar cuáles de ellos deben ejecutar ciertas tareas (arquitectura semi-automatizada).

Para asegurar el éxito de este proceso de monitorización y ejecución de tareas es necesaria una comunicación efectiva y eficiente entre los nodos de la red. En concreto, se requiere que lo sean los mecanismos de coordinación entre las distintas entidades: sensor-sensor, sensor-actuador, actuador-actuador y actuador-sumidero. Debido a que estos mecanismos deben tratar a menudo con fenómenos asociados a fallos y movilidad (en particular, de los nodos actuadores), resulta interesante evaluar el comportamiento de HARP en este tipo de aplicaciones y sus prestaciones en cuanto a la gestión de tráfico tanto *upstream* como *downstream* y la recuperación ante fallos de red. En particular, este análisis se centrará en redes WSN basadas en ZigBee [153], ampliamente utilizadas para automatización de hogares [152] y gestión de energía en el hogar [62], edificios inteligentes [16], monitorización de pacientes [14] o asistencia y rehabilitación de personas [52].

Para llevar a cabo este análisis se tendrán en cuenta los mecanismos de coordinación entre actuadores y entre actuadores y el nodo sumidero, siendo HARP el encargado de proporcionar el encaminamiento en este nivel. Se asumirá que los nodos sensores se organizan en *clusters*, cada uno de ellos gestionado por un nodo actuador que toma el rol de CH dentro del mismo. Los nodos actuadores recibirán información de monitorización del entorno procedente de los nodos sensores de su *cluster* y deberán comunicarse con otros actuadores o con el nodo sumidero para evaluar la ejecución de alguna acción en su entorno. Siguiendo criterios de eficiencia energética, el protocolo propuesto será capaz de crear una estructura jerárquica según lo indicado en el Capítulo 3 entre los nodos actuadores y un nodo sumidero, el cual asumirá el papel de *Root* o nodo raíz dentro del árbol jerárquico formado, siguiendo la estructura mostrada en la Figura 6.3. De esta forma, los actuadores se organizarán por niveles y serán necesarias comunicaciones multi-salto para poder alcanzar el nodo sumidero desde cualquiera de ellos.

En el Capítulo 5 se ha modelado el tiempo de vida de un enlace inalámbrico a través de una variable aleatoria cuya distribución vendrá determinada por las condiciones del canal (obstáculos, desvanecimientos, etc.), por la probabilidad de mal funcionamiento de los nodos sensores origen y extremo y por el patrón de movilidad definido. Con relación a este último parámetro, diferentes patrones de movilidad han sido analizados en la literatura científica [32] y ha sido posible modelar la *pdf* del tiempo de vida de un enlace en función de la velocidad y del modelo de movimiento seguido por cada nodo. En consecuencia, en este escenario se puede proceder a evaluar el rendimiento de HARP asociando a cada enlace inalámbrico un valor de probabilidad de fallo en el momento en el que una transmisión es iniciada a través de dicho canal. De esta forma, los efectos de los fallos en enlaces y de la movilidad de los nodos son considerados en las simulaciones realizadas.

Un ejemplo gráfico de la estructura simulada se presenta en la Figura 7.12. Esta figura incluye la estructura jerárquica formada por los actuadores y el nodo sumidero como nodo raíz del árbol, indicando en el interior de cada uno de ellos el nivel que le corresponde dentro de la misma. Además, se muestran diferentes situaciones que alteran el funcionamiento inicial de la red, como son el establecimiento de nuevos canales inalámbricos de comunicación con objeto de recuperarse de otros enlaces rotos debido a movilidad, obstáculos o fallos de nodos, así como para asociar nuevos nodos en la red.

ZigBee y su estándar asociado 802.15.4, está diseñado para controlar los distintos dispositivos en una red de comunicaciones con baja tasa de bit y en su funcionamiento combina un protocolo jerárquico de encaminamiento denominado HRP (*Hierarchical Routing Protocol*) con el concepto de AODV (*Ad hoc On-demand Distance Vector* [103]) con objeto de proporcionar robustez de rutas y efectividad en cuanto a consumo energético y sobrecarga de la red [60]. AODV es uno de los protocolos mejor conocidos y más representativos de las redes MANET y que

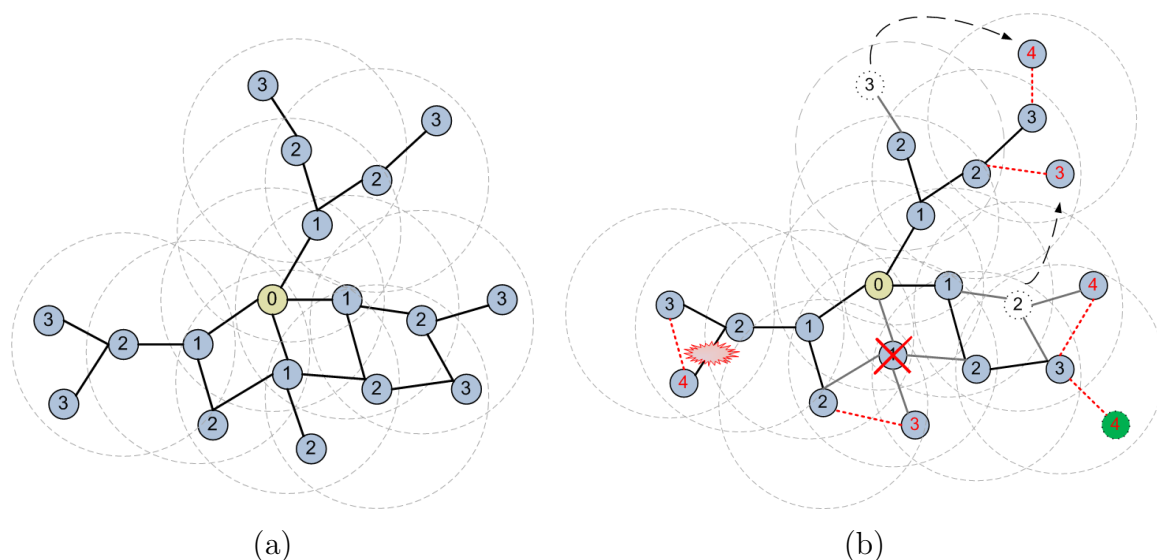


Figura 7.12: Estructura jerárquica creada por HARP en WSN: a) árbol inicial; b) estructura modificada con nuevos enlaces inalámbricos (en rojo), enlaces rotos (en gris), movilidad (nodos con fondo blanco), obstáculos (flash), fallos de nodos (aspa roja) y asociación de nuevos nodos en la red (verde).

también puede ser utilizado en numerosas aplicaciones. Éste es un protocolo de encaminamiento plano (no se crean estructuras jerárquicas en la red), *single-path* y reactivo, en el que el proceso de búsqueda de nuevas rutas se realiza mediante *broadcast* o inundación de la red y se inicia solamente cuando un nodo necesita enviar información a otro y no conoce rutas hacia el destino o éstas ya no están disponibles. Específicamente para WSN, el algoritmo de encaminamiento de ZigBee está basado en AODV para descubrir la ruta más corta hacia un destino determinado cuando de éste no se conoce ningún camino, lo que puede introducir cierta sobrecarga de paquetes de control que interfiere con el tráfico de datos de la red. ZigBee está presente en varios módulos RF de comunicaciones tales como el CC240 utilizado en los dispositivos sensores MICAz de Crossbow [4].

A continuación se describen los experimentos realizados y se presentan y analizan los resultados obtenidos, siguiendo la estructura ya utilizada en secciones anteriores.

7.3.1.1. Descripción de experimentos

Objetivos

1. Análisis de prestaciones de HARP en la coordinación entre nodos actuadores en redes WSAN basadas en ZigBee frente a AODV.

Variables analizadas

Se han evaluado las siguientes variables¹:

- *Tiempo de vida de rutas.*
- *Sobrecarga de paquetes (overhead, routing load):* medida del número de paquetes transmitidos a través de la red (paquetes de datos y paquetes de control) por paquete de datos y por nodo de la red.
- *Retardo medio extremo a extremo:* tiempo transcurrido entre la transmisión y la recepción de un paquete de datos.
- *Número medio de saltos por paquete de datos entregado.*

Parámetros

Se han realizado diferentes simulaciones para una arquitectura semi-automatizada con un nodo sumidero actuando como nodo *Root* en HARP en dos escenarios con 60 y 168 actuadores respectivamente. Estos nodos han sido situados formando una red triangular como se indica en la Figura 7.13 con una separación $a = 50$ metros, en cuyo centro se encuentra el nodo sumidero. En cada actuador se ha limitado la potencia de salida de tal forma que sólo pueda alcanzar a sus vecinos más próximos, lo que le confiere un índice de coordinación de 6, medido como el número de nodos vecinos de cada uno.

¹Dado que las WSAN pueden tratarse como un caso particular de redes heterogéneas, se obvia el estudio energético, haciendo referencia al obtenido en la Sección 7.2.

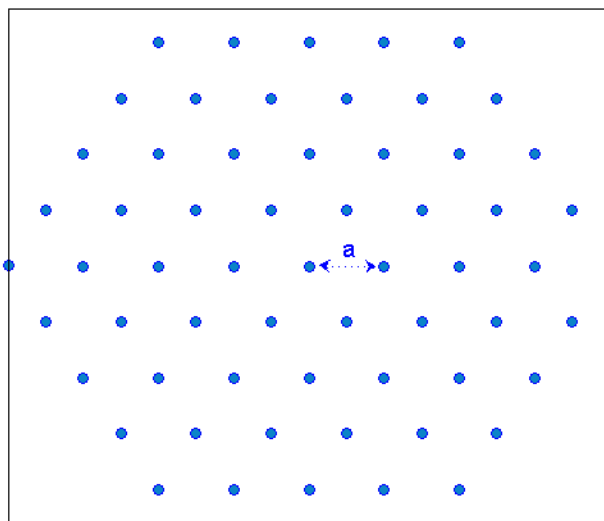


Figura 7.13: Red hexagonal triangular simulada con 60 nodos actuadores y un nodo sumidero central en una arquitectura semi-automatizada.

Para cada uno de los escenarios anteriores se han ejecutado simulaciones de los dos tipos de tráfico de datos en la red:

- *upstream* o ascendente: todos los actuadores envían la información monitorizada por los nodos sensores hacia el nodo *Root* para su procesado.
- *downstream* o descendente: se ha implementado un flujo de paquetes procedente del nodo *Root* hacia cada uno de los actuadores para simular una indicación de las acciones a ejecutar por parte de los actuadores en la WSN.

Se han realizado 100 simulaciones completas para cada uno de los escenarios (60 y 168 actuadores), para cada tipo de tráfico indicado y para cada uno de los siguientes valores de probabilidad de error de enlace: 0.0, 0.2, 0.4, 0.6, 0.8 y 1.0. Además, para la consecución de los resultados relacionados con el tiempo de vida útil de la red se han añadido simulaciones con valores adicionales de probabilidad

de error: 0.1, 0.3, 0.5, 0.7 y 0.9. La Tabla 7.3 muestra los parámetros utilizados en todas ellas.

Parámetro	Valor
Número de nodos actuadores 1	60
Número de nodos actuadores 2	168
Distribución de los actuadores en la red	triangular
Separación horizontal entre actuadores	50 m
Protocolo 1	HARP
Protocolo 2	AODV
Posición del nodo sumidero	central

Tabla 7.3: Parámetros de simulación de prestaciones de HARP en redes WSAAN.

7.3.1.2. Resultados

A continuación se muestran los resultados obtenidos para cada una de las variables indicadas anteriormente.

Tiempo de vida útil de la red

La Figura 7.14 muestra la evolución del porcentaje de paquetes recibidos con éxito en tráfico *downstream* antes de iniciar el proceso de descubrimiento de rutas. Este porcentaje de éxito tiene relación directa con el tiempo de vida de las rutas que parten del nodo sumidero hacia cualquier destino de la red en ambos protocolos. La conclusión más importante extraída de este resultado es la siguiente:

- HARP consigue mantener siempre un porcentaje de paquetes entregados mayor que AODV. Esto se debe a la estructura propuesta de las tablas de encaminamiento que proporcione rutas multi-camino entre los dos nodos, con rutas disjuntas y no disjuntas, lo que favorece la utilización de un mayor número de rutas alternativas cuando se producen fallos de transmisión en las actuales.

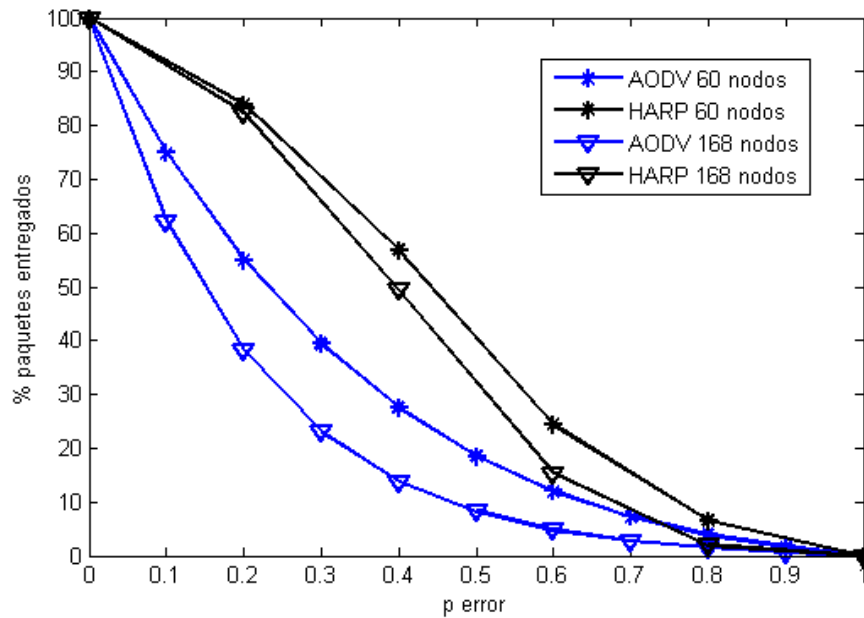
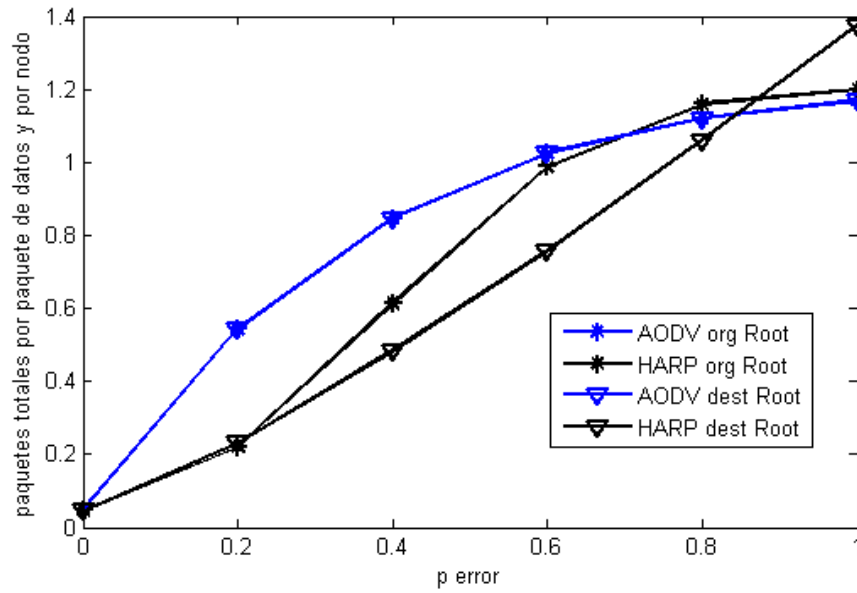


Figura 7.14: Porcentaje de paquetes de datos entregados con éxito en tráfico downstream antes del inicio de un nuevo proceso de descubrimiento de rutas.

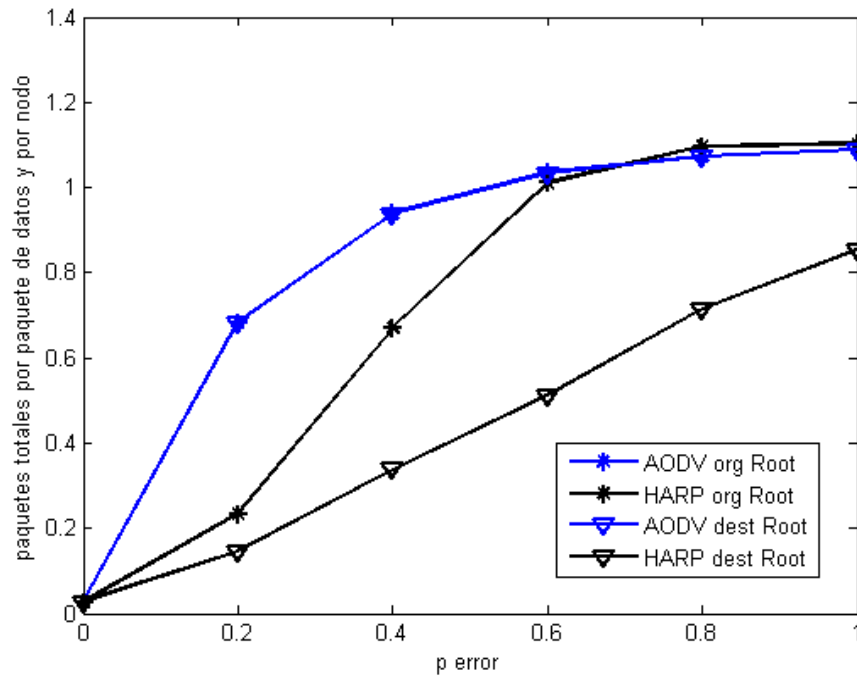
Sobrecarga de paquetes

La Figura 7.15 muestra el número total de paquetes transmitidos en la red por paquete de datos y por nodo cuando el nodo *Root* envía un paquete a cada uno de los nodos actuadores de la red (*org Root*) y cuando recibe un paquete de datos de cada uno de los actuadores (*dest Root*). Este número total de paquetes incluye tanto las transmisiones del paquete de datos a lo largo de la ruta como las transmisiones de los paquetes de control necesarios para una correcta recepción del paquete en el nodo destino (reparación de enlaces rotos, descubrimiento de nuevas rutas, etc.). A la vista de dicha figura, tanto para 60 como para 168 nodos, se puede inferir lo siguiente:

- HARP casi siempre consigue introducir menor sobrecarga de paquetes en la red que AODV, consecuencia directa de dos situaciones:



(a)



(b)

Figura 7.15: Paquetes totales transmitidos (control+datos) por paquete de datos y por nodo: a) 60 actuadores; b) 168 actuadores.

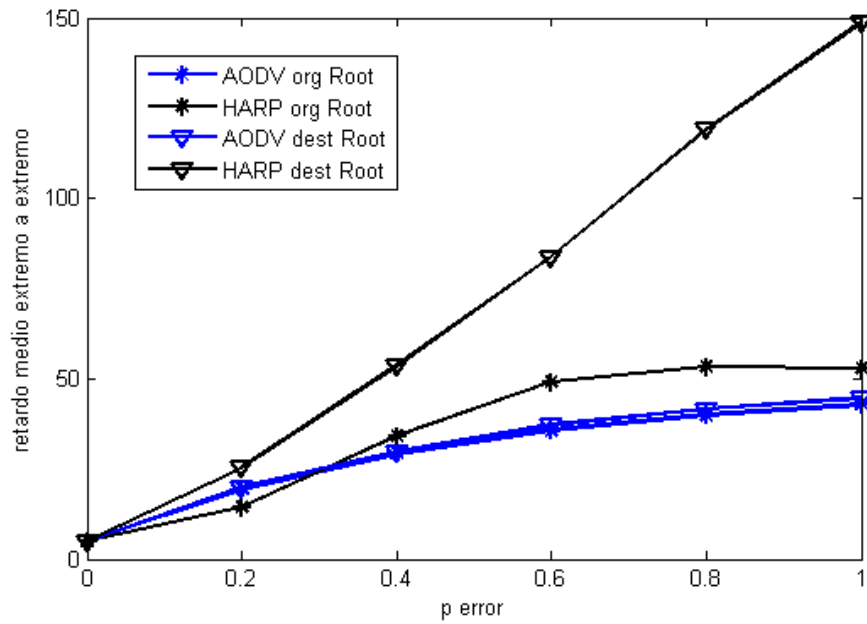
1. En *downstream* HARP consigue prolongar el tiempo de vida de las rutas y, en consecuencia, son necesarios menos procesos de descubrimiento de rutas (que en general son procesos de difusión o *bradcast* e inundan la red con paquetes de control).
 2. En *upstream* HARP hace uso del proceso de recuperación local del enlace roto, capaz de encontrar una nueva ruta hacia el nodo sumidero sin necesidad de inundar la red con mensajes de control como lo hace AODV.
- La única excepción al mejor comportamiento de HARP en cuanto a sobrecarga en la red en el caso de tráfico *upstream* se produce para una red de 60 nodos y probabilidades de error de enlace mayores de 0.85. En este caso, debido a esta alta probabilidad de fallo, HARP se ve obligado a iniciar su proceso de recuperación de rutas un número muy elevado de veces para tratar de encontrar una ruta hacia el destino, lo que hace que se incremente en gran medida el número de paquetes de control utilizados. AODV, sin embargo, ante el primer fallo en un enlace de la ruta iniciará el proceso de descubrimiento, inundando la red y encontrando, en caso de que exista, un nuevo camino alternativo. En este proceso transmitirá como máximo un número de paquetes igual al número de nodos de la red aunque, debido a la probabilidad de fallo utilizada, este número será mucho menor ya que existirán numerosos enlaces rotos en la red que impiden la retransmisión del paquete de inundación a través de la red. Para el caso de 168 nodos el proceso de recuperación de HARP encontrará una ruta hacia el destino sin tener que involucrar a muchos de los nodos de la red, con lo que la sobrecarga será menor en relación con AODV, donde el proceso de inundación ahora se produce en una red mucho más grande y con muchos más paquetes de control retransmitiéndose.

- En el caso de tráfico *downstream*, para valores de probabilidad de error grandes (mayores de 0.7), HARP introduce ligeramente mayor sobrecarga al intentar alcanzar el destino por todas las rutas alternativas de que dispone antes de proceder con el inicio del proceso de descubrimiento de rutas. AODV, sin embargo, lanza este proceso en cuanto detecta el fallo de la ruta conocida.
- Por último, resulta interesante resaltar que debido a que AODV es un protocolo plano, su método de funcionamiento no distingue los dos tipos de tráfico indicados anteriormente, con lo que en las dos gráficas de la Figura 7.15 las curvas obtenidas tanto para *upstream* como para *downstream* tienen el mismo comportamiento y, en consecuencia, están solapadas.

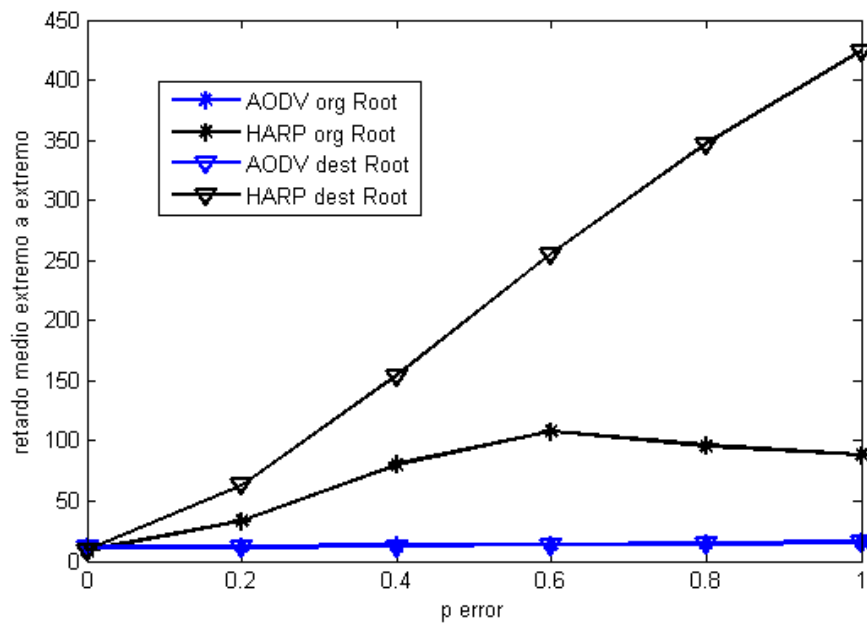
Retardo medio extremo a extremo

A la vista de la Figura 7.16 se puede concluir lo siguiente:

- En general HARP entrega los paquetes con un retardo mayor que AODV. La única excepción a este hecho es el caso de 60 nodos, tráfico *downstream* y probabilidad de error hasta 0.3 (ver Figura 7.16 a)), donde HARP encuentra menos contención en la red y utiliza las rutas alternativas para alcanzar el destino sin necesidad de iniciar el descubrimiento de otras nuevas.
- El retardo ligeramente mayor observado del tráfico *downstream* en el resto de los valores es debido a la utilización de diferentes rutas alternativas para tratar de alcanzar el destino antes de iniciar un nuevo descubrimiento, hecho que se produce en AODV rápidamente ante el primer fallo detectado en la red y que hace disminuir el retardo medio hasta alcanzar el destino. Sin embargo, para el tráfico *upstream* este retardo es considerablemente mayor por la misma razón expuesta en el análisis de la sobrecarga de la red, es decir, a mayor probabilidad de error mayor será también el número de procesos de



(a)



(b)

Figura 7.16: Retardo medio extremo a extremo (ms) de los paquetes de datos entregados: a) 60 actuadores; b) 168 actuadores.

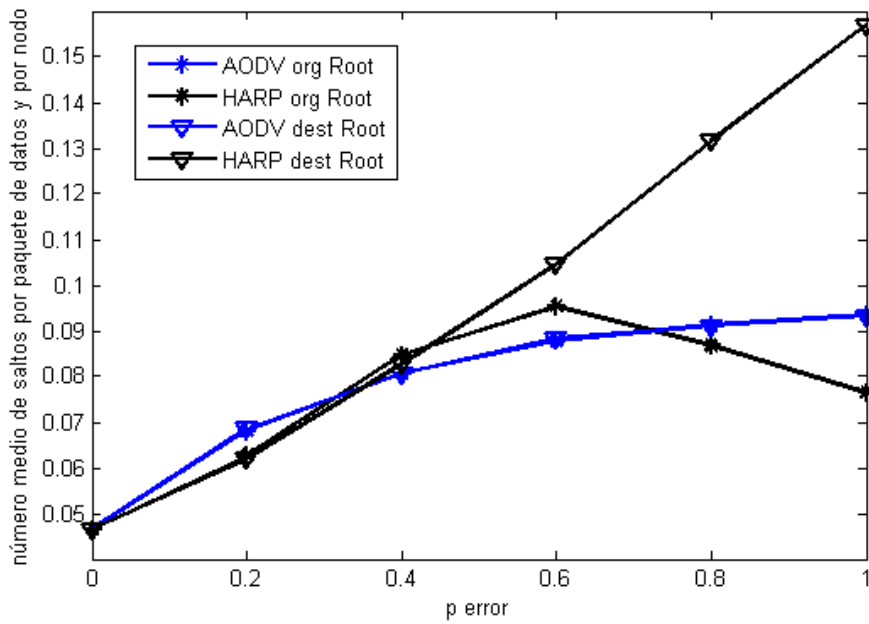
recuperación local iniciados, cada uno con su correspondientes retardo, lo que hace que la suma de estos tiempos sea mayor que la de encontrar una nueva ruta por inundación, como es el caso de AODV.

- Por su parte, este comportamiento se ve magnificado en la red de 168 nodos como puede observarse en la Figura 7.16 b).
- La evaluación conjunta de este parámetro con el asociado a la sobrecarga de la red arroja como conclusión que HARP es capaz de entregar un mayor número de paquetes sin necesidad de iniciar procesos de descubrimiento de nuevas rutas en la red, consiguiendo disminuir la sobrecarga y consecuentemente el consumo energético de los nodos, aunque para ello haya que sacrificar en parte el retardo de los paquetes en entornos muy desfavorables (con alta tasa de fallos). El hecho de que en general HARP introduzca más retardo que AODV y que otros protocolos representativos de las MANET está relacionado con el objetivo propuesto por este protocolo, en el que prima (como ya se ha justificado para WSN) el ahorro energético y la menor sobrecarga de la red al retardo en la entrega de paquetes, ya que éste no es un criterio de diseño crítico en este tipo de redes.

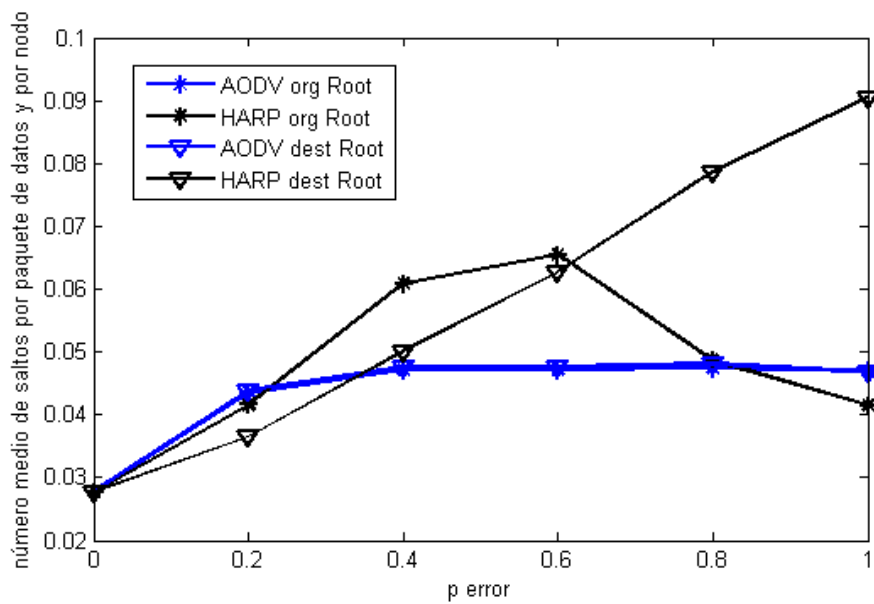
Número medio de saltos por paquete de datos entregado

Para finalizar este análisis comparativo, en la Figura 7.17 se incluyen gráficas acerca del número medio de saltos que utiliza un paquete de datos para alcanzar el destino. En las dos gráficas presentadas se observa lo siguiente:

- Para probabilidades de error de hasta aproximadamente 0.35, HARP consigue menos transmisiones que AODV, debido a las razones explicadas anteriormente.
- Cuando la probabilidad de error se incrementa, HARP intenta encontrar rutas alternativas antes de iniciar nuevas búsquedas por inundación, lo que



(a)



(b)

Figura 7.17: Número médio de saltos recorrido por los paquetes de datos entregados: a) 60 actuadores; b) 168 actuadores.

hace que los paquetes necesiten más saltos hasta encontrar una ruta que los conduzca a su destino.

- Sin embargo, para el caso de tráfico *downstream*, cuando esta probabilidad es mayor que 0.6, en ambos casos la tendencia se invierte, siendo necesarios menos saltos y llegando a superar incluso a AODV en este concepto. La razón radica en que con alta probabilidad los primeros enlaces de la ruta fallarán, con lo que HARP escogerá una nueva ruta sin necesidad de avanzar en la anterior ruta fallida, lo que le haría volver hacia atrás haciendo que se incrementara el número de saltos realizados.

7.3.2. HARP adaptado al proyecto GUIADE

A lo largo de las últimas décadas el crecimiento económico en Europa ha ido incrementando de forma continua la demanda de energía y movilidad, desembocando en una situación actual en la cual los medios de transporte consumen alrededor del 30 % de la energía total consumida en la Unión Europea. Este incremento de movilidad requiere por tanto actuaciones urgentes en términos de innovación tecnológica y políticas que conduzcan hacia un transporte más sostenible: ecológico, seguro y eficiente. El fomento de sistemas de transporte inteligentes y eficientes es una de las prioridades marcadas desde hace algunos años en la Comisión Europea [46].

En el año 2008, un consorcio formado por varias compañías privadas e instituciones y centros de investigación de reconocido prestigio puso en marcha en España el proyecto GUIADE, financiado por el Ministerio de Fomento y Fondos de Desarrollo Regional, el cual utiliza las Tecnologías de la Información y Comunicación, junto con Automatización Industrial, con el objetivo de mejorar la eficiencia energética en el transporte público de personas. En concreto, la finalidad del Proyecto GUIADE es *“la construcción de un sistema de posicionamiento y guiado automático de vehículos de transporte público, basado en una percepción multimodal del entorno que englobe tanto información obtenida desde la infraestructura como datos recogidos por los propios vehículos, con la finalidad última de optimizar su eficiencia entendida en sus diversas facetas de consumo energético, impacto medioambiental, seguridad y calidad de los servicios de transporte público”* [15].

Para conseguir este objetivo en el proyecto se utilizan nodos sensores que combinan la información de posicionamiento proporcionada por los diferentes elementos móviles del sistema con información adicional obtenida por los elementos sensores pertenecientes a la infraestructura fija de la red. Esta información es procesada por el Centro de Gestión de Tráfico (TMC, *Traffic Management Center*) y devuelta a

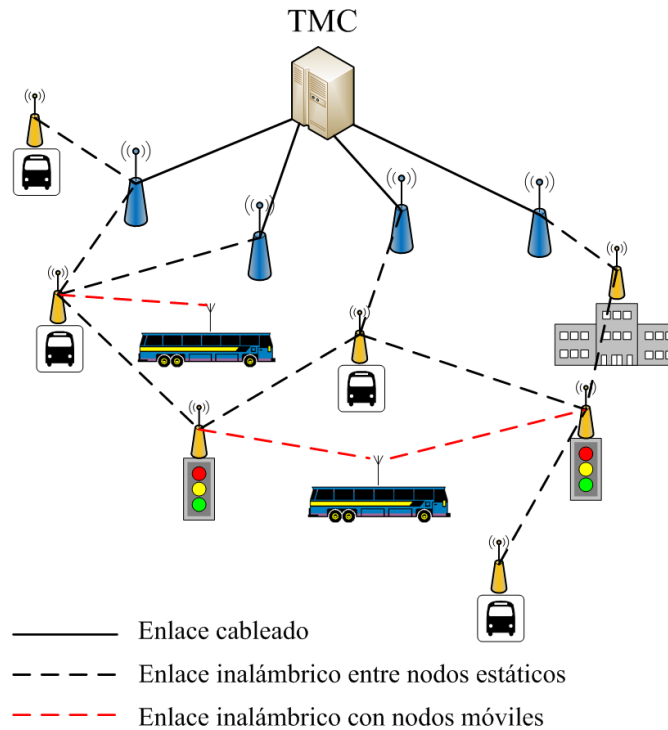


Figura 7.18: Red heterogénea para las comunicaciones entre elementos en GUIADE.

los vehículos proporcionándoles información útil acerca de las condiciones y previsiones del tráfico que se van a encontrar en su camino. El proyecto está basado en tres pilares fundamentales: por un lado, en un entorno multimodal con monitorización del tráfico y de la ruta; por otro, en un sistema inteligente de procesado de datos; y, por último, en una sofisticada plataforma de comunicaciones.

La realidad del proyecto hace que el sistema descansa sobre una red de comunicaciones heterogénea. La Figura 7.18 muestra una representación esquemática de la estructura de red diseñada para este proyecto. La red está compuesta por cuatro tipos de nodos diferentes, tal y como se describe en la Tabla 7.4, organizados según una estructura jerárquica con topología en árbol en donde el TMC ejerce de nodo raíz o *Root*. Los nodos en el nivel superior son los más cercanos al TMC y están conectados al mismo a través de enlaces fijos cableados. Estos nodos del primer nivel que, en general, no tienen limitaciones energéticas para su funcionamiento ya que

están conectados a la red eléctrica, se comunican con otros nodos situados en un nivel intermedio a través de enlaces inalámbricos. Estos dispositivos pertenecientes al nivel intermedio se sitúan en puntos tales como semáforos, paradas de autobús o cualquier otro elemento no móvil de la infraestructura. En principio, la capacidad energética de estos nodos se considera limitada, ya que no siempre será posible conectarlos a una red eléctrica externa. Por último, existen nodos sensores que se sitúan en el nivel inferior de la jerarquía de red y que van embarcados en los vehículos de transporte público y que también tienen restricciones energéticas en cuanto a la capacidad limitada de su batería.

Nivel	<i>Up-Link</i>	<i>Down-Link</i>	Tipo de nodo	Alimentación
<i>Root</i>	–	Cableado	Estático	Red eléctrica
Superior	Cableado	Inalámbrico	Estático	Red eléctrica
Intermedio	Inalámbrico	Inalámbrico	Estático	Baterías
Inferior	Inalámbrico	–	Móvil	Baterías

Tabla 7.4: Tipos de nodos y sus características.

Debido a la cantidad de datos intercambiados entre el TMC y los dispositivos situados en los diferentes elementos de la plataforma (semáforos, paradas de autobuses, etc.) y en los vehículos, el principal reto de la red de comunicaciones es la gestión eficiente del tráfico tanto en sentido infraestructura-TMC (*upstream*) como en sentido TMC-nodos sensores (*downstream*). Los requisitos que este escenario heterogéneo impone, con distintos niveles y diferentes tipos de nodo en cada uno de ellos, hacen del protocolo de encaminamiento utilizado un elemento fundamental para prolongar la vida útil de la red.

Cada nivel utiliza su propia tecnología física de comunicaciones, ya sea cableada o inalámbrica. Para esta última, puede utilizarse WiMAX y 802.11p en comunicaciones de larga distancia, aunque a priori no hay restricciones en cuanto a los protocolos de las capas inferiores de acceso al medio y transmisión. De hecho, los detalles de la tecnología específica utilizada son irrelevantes en cuanto a

la elección del protocolo de encaminamiento utilizado, siendo la probabilidad de fallo de cada uno de los enlaces la única característica que debe tenerse en cuenta a la hora de modelar el sistema completo.

7.3.2.1. Descripción de experimentos

En este escenario se ha simulado el comportamiento de HARP una vez creado el árbol jerárquico entre los diferentes nodos de la plataforma y comparado su rendimiento con AODV, protocolo tradicional de las redes MANET.

Objetivos

1. Análisis de prestaciones de HARP en una estructura de red heterogénea basada en el proyecto GUIADE.

Variables analizadas

- *Porcentaje de paquetes entregados con éxito.*
- *Paquetes totales (datos+control) por paquete de datos y nodo móvil.*
- *Retardo medio extremo a extremo: tiempo transcurrido entre la transmisión y la recepción de un paquete de datos.*

Parámetros

El simulador WiNS descrito en el Capítulo 4 ha sido utilizado para realizar diferentes simulaciones del tráfico *upstream* y *downstream* en una red compuesta por un TMC y un sistema público de transporte con 60 nodos de infraestructura y 10 nodos móviles colocados en otros tantos vehículos. La Figura 7.19 muestra el escenario utilizado con 3 de esos nodos móviles siguiendo una ruta concreta entre

la primera y la última parada de autobuses indicadas, en la cual HARP hace uso del proceso de recuperación local de rutas para adaptarse a las distintas situaciones de los vehículos móviles.

Con objeto de modelar los fallos en enlaces y la movilidad de los nodos, se han asignado diferentes valores de probabilidad de fallo a cada uno de los niveles y tipos de enlace de la red descritos anteriormente. Los enlaces cableados tendrán una probabilidad de fallo baja, de 10^{-5} , mientras que los enlaces inalámbricos entre nodos fijos han sido simulados utilizando un valor de probabilidad de fallo de 10^{-1} . Finalmente, los enlaces entre nodos móviles y la infraestructura fija son los más inestables debido a las condiciones dinámicas del canal y a la movilidad de los nodos. La probabilidad de fallo y la disponibilidad del canal inalámbrico en este nivel están relacionados con la velocidad de los nodos móviles y el tiempo transcurrido entre dos transmisiones sucesivas de paquetes a través del mismo [57], [95]. En consecuencia, para modelar distintas velocidades y tasas de transmisión se simulan los enlaces de este nivel en la ruta asumiendo una probabilidad de fallo que varía entre 0 y 1. La Tabla 7.5 muestra un resumen de las distintas probabilidades de error utilizadas en los niveles del sistema.

Enlace	Tipo de conexión	Movilidad	Perror
TMC-Nivel superior	Cableado	No	10^{-5}
Nivel superior-Nivel intermedio	Inalámbrico	No	10^{-1}
Nivel intermedio-Vehículos	Inalámbrico	Sí	(0, 1)

Tabla 7.5: Valores de probabilidad de error asociados a los diferentes tipos de enlaces de la red.

Se han realizado 100 simulaciones diferentes generando dos paquetes sucesivos para cada uno de los dos tipos de tráfico estudiados (desde el TMC a los vehículos y viceversa) y para cada valor de probabilidad de error. Esta situación proporciona información acerca de la duración de las rutas establecidas entre el TMC y el nodo móvil y de la capacidad del protocolo para hacer frente a la pérdida de un enlace de

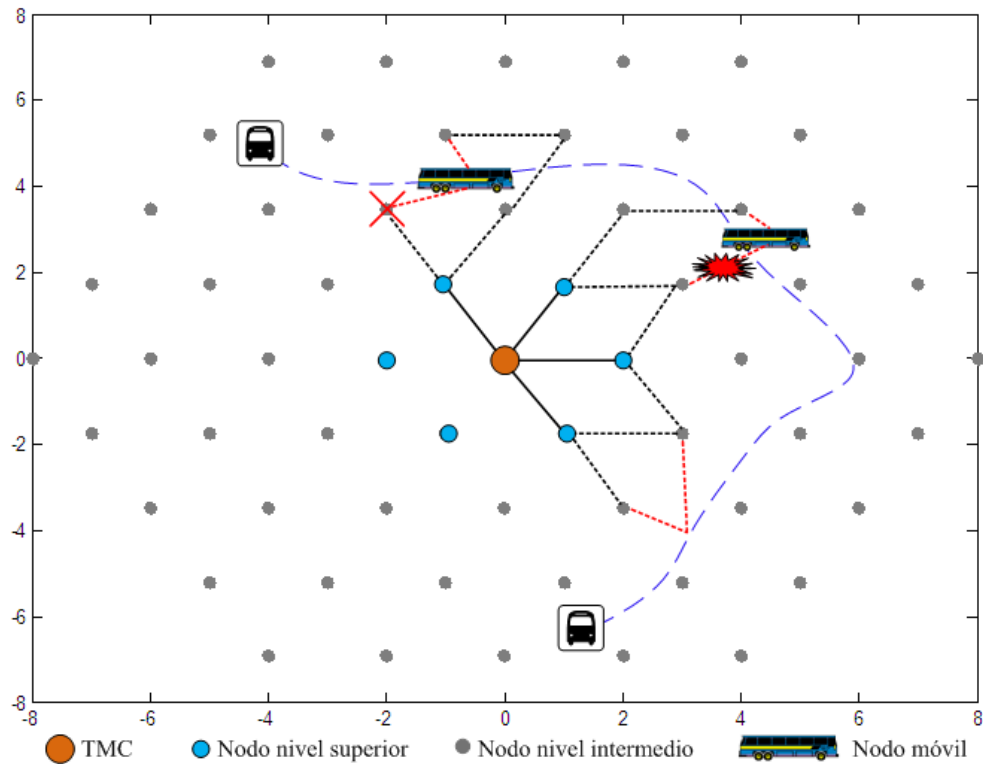


Figura 7.19: Red simulada con nodos móviles: utilización del mecanismo de recuperación local de rutas en HARP para adaptarse a la movilidad de los nodos.

la ruta entre dos transmisiones consecutivas, simulando las condiciones del canal y el cambio en la posición de los vehículos. La tasa de generación de paquetes, la velocidad de los nodos y la probabilidad de fallo en los enlaces inalámbricos con nodos móviles se ha variado para evaluar las distintas situaciones.

7.3.2.2. Resultados

Porcentaje de paquetes entregados con éxito

La Figura 7.20 muestra el porcentaje de paquetes entregados con éxito en ambos tipos de flujo de datos, a la vista de la cual puede extraerse la siguiente conclusión:

- Debido a la estructura de rutas alternativas y a los procesos de recuperación local, HARP consigue encontrar una nueva ruta sin necesidad de inundar la red y, en consecuencia, es capaz de transmitir con éxito un porcentaje mayor de paquetes tanto en *upstream* como en *downstream*, logrando aumentar el tiempo de vida útil de las rutas entre el TMC y los distintos vehículos móviles.

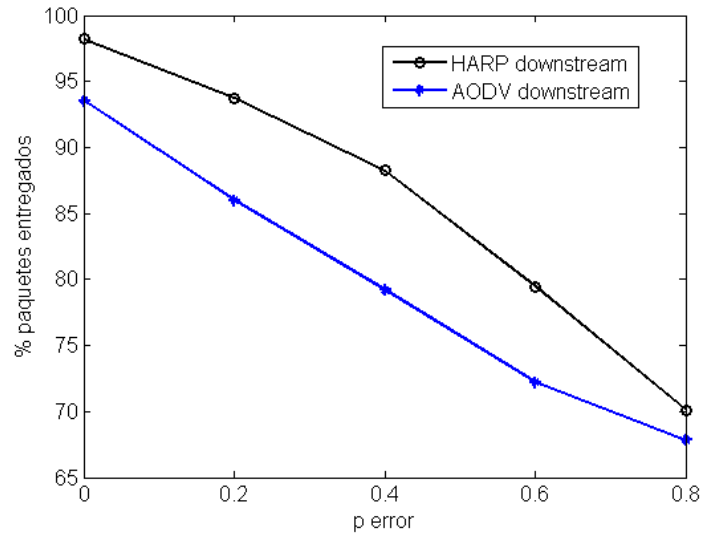
Paquetes totales (datos+control) por paquete de datos y nodo móvil y Retardo medio extremo a extremo

La sobrecarga de paquetes introducida por cada protocolo y el retardo extremo a extremo se evalúan conjuntamente debido a la interrelación existente entre estas dos variables. Las Figuras 7.21 y 7.22 muestran los resultados obtenidos para ambas, de los que se puede inferir lo siguiente:

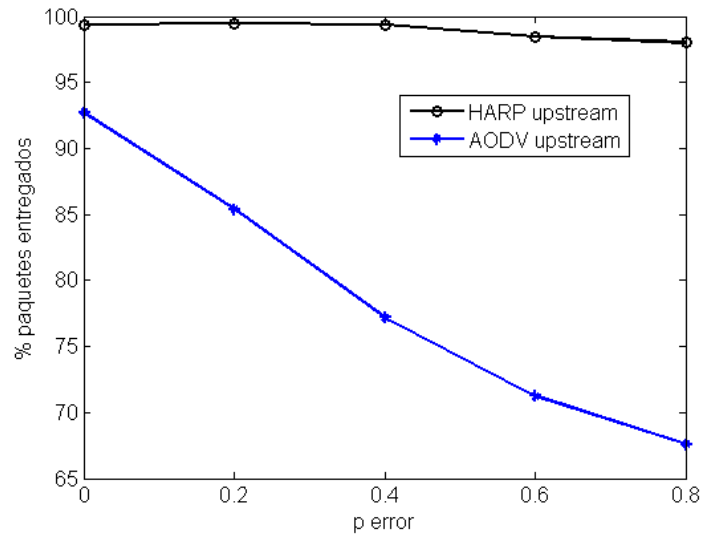
- El hecho de introducir menor sobrecarga, causado por el mayor tiempo de vida de las rutas en HARP, hace que proporcione mucho menor retardo que AODV para probabilidades de error bajas y algo menor cuando ésta se incrementa, debido a las nuevas asociaciones producidas dentro del proceso de recuperación local de rutas que evitan el inicio de procedimientos de *broadcast* pero introducen cierto retardo en la entrega de paquetes.

7.3.2.3. Resumen

Las simulaciones realizadas demuestran que HARP proporciona mayor eficiencia que AODV en términos de encaminamiento en esta estructura de red heterogénea, con diferentes tipos de nodos y diferentes enlaces entre los mismos. El protocolo propuesto hace uso de mecanismos definidos para reconfigurar la estructura de rutas y adaptarse a las circunstancias cambiantes del entorno y reducir de



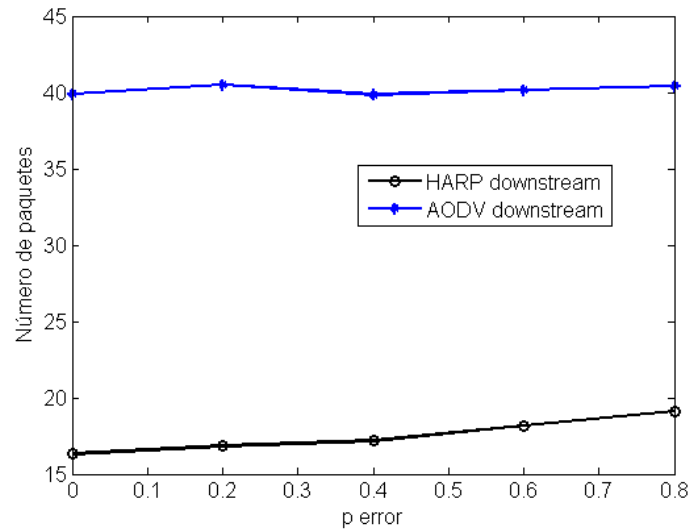
(a)



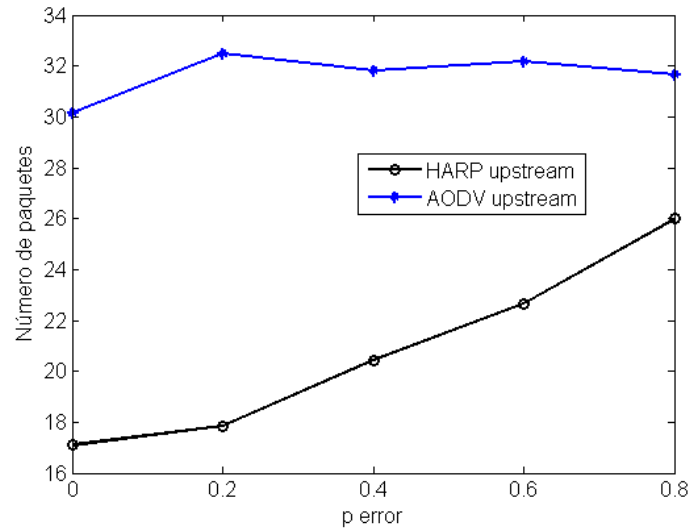
(b)

Figura 7.20: Porcentaje de paquetes de datos entregado con éxito antes del inicio de nuevos descubrimientos de ruta: a) downstream; b) upstream.

esta forma la cantidad de tráfico de control adicional necesario para la comunicación entre el TMC y los nodos móviles, ajustándose a los requisitos impuestos por la plataforma de transporte utilizada en GUIADE y proporcionando un control del tráfico más inteligente.

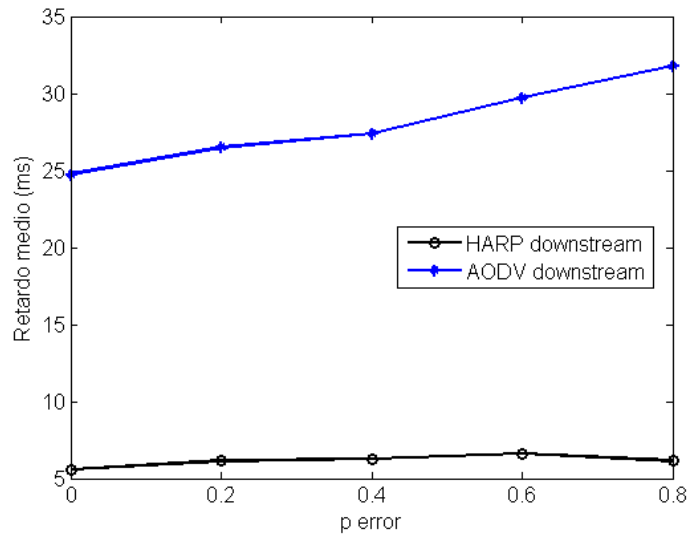


(a)

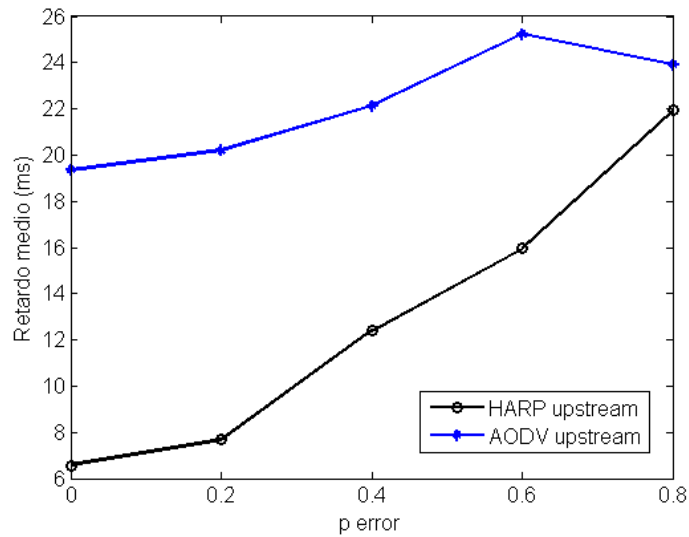


(b)

Figura 7.21: Paquetes totales (control+datos) por paquete de datos y por nodo móvil en función de la probabilidad de error de los enlaces con nodos móviles: a) downstream; b) upstream



(a)



(b)

Figura 7.22: Retardo medio extremo a extremo de los paquetes de datos entregados al destino en función de la probabilidad de error de los enlaces con nodos móviles: a) downstream; b) upstream.

7.4. Conclusiones

En este capítulo se han evaluado las prestaciones de HARP en WSN y se ha demostrado que el protocolo propuesto proporciona un encaminamiento eficiente en este tipo de redes desde el punto de vista energético, así como mecanismos que permiten transmitir distintos tipos de tráfico en la red, lo que favorece su aplicabilidad en un abanico muy amplio de aplicaciones. La estructura de rutas implementada en HARP es dinámica, tolerante a fallos de red y adaptable a los cambios producidos en el entorno, ofreciendo ahorro energético a los nodos de la red sin necesidad de tener que conocer su posición dentro de la misma.

La evaluación realizada de las prestaciones del protocolo propuesto pone en evidencia esta eficiencia energética, proporcionando un funcionamiento estable y uniforme de los nodos de la red, minimizando la sobrecarga del sistema con respecto a otros protocolos que utilizan difusión o inundación de paquetes para alcanzar el destino requerido. La estructura de la trama TDMA definida permite gestionar de forma efectiva las transmisiones en la red, proporcionando además procedimientos de detección temprana y subsanación de fallos dentro de cada estructura jerárquica construida.

Además de WSN, se han realizado simulaciones en otras aplicaciones en las que HARP ha sido comparado con uno de los protocolos más representativos de las redes MANET como es AODV. Los resultados obtenidos demuestran que HARP consigue mejorar el encaminamiento en la red prolongando el tiempo de vida efectivo de las rutas entre los nodos a través de su estructura de rutas y de los mecanismos de recuperación local de fallos definidos en el protocolo.

Capítulo 8

Conclusiones, publicaciones y líneas futuras

Una conclusión es el lugar donde llegaste cansado de pensar.

Anónimo.

8.1. Conclusiones y aportaciones

Sobre la base del estudio de la situación actual del encaminamiento en WSN, de los criterios de diseño de un protocolo del nivel de red en este escenario particular y de los objetivos descritos en el Capítulo 1, esta Tesis propone una nueva arquitectura de protocolo de encaminamiento denominado HARP (*Hierarchical Adaptive Routing Protocol*), que permite minimizar y balancear el consumo energético de una WSN a través de la rotación de roles y de la creación de estructuras multi-salto que ahorran energía en las comunicaciones. HARP es un protocolo jerárquico en el que los nodos se distribuyen en *clusters*, siendo elegidos en cada ronda de operación un número determinado de ellos para desempeñar la función de CH. En este sentido, la propuesta de s-HARP incluye un nuevo método de selección efi-

ciente y distribuida de CH. Cada CH planifica las transmisiones dentro su *cluster* a través del uso de TDMA para evitar las colisiones. Esta técnica de acceso al medio puede ser llevada a cabo a través de un nuevo método propuesto en este trabajo basado en una planificación distribuida y que, junto con la planificación centralizada, establecen las dos opciones para la asignación de franjas temporales de transmisión a cada nodo del cluster. Teniendo en cuenta la problemática particular de las redes inalámbricas, cuyas nuevas modalidades de comunicación (como las comunicaciones cooperativas o distribuídas) no son del todo bien gestionadas por arquitecturas inflexibles basadas en el diseño clásico, horizontal, e independiente por capas, la utilización de un diseño *cross-layer* de cooperación entre los niveles de acceso al medio y de red del modelo OSI permite aprovechar mejor las dependencias entre ellos para obtener mejores rendimientos.

Teniendo en cuenta las restricciones energéticas de los nodos sensores y tomando como uno de los criterios de diseño principales el ahorro energético de los nodos, HARP crea estructuras jerárquicas eficientes tanto en el interior de los *clusters* (árboles *intra-cluster*) como entre los distintos CH y el nodo sumidero (árbol *inter-cluster*) con objeto de minimizar el consumo derivado de las comunicaciones. Al permitirse además a los nodos entrar en modos de funcionamiento de baja energía, los resultados de simulación demuestran que HARP consigue optimizar el funcionamiento de los nodos sensores, aprovechar mucho mejor el ancho de banda disponible y prolongar el tiempo de vida útil de la red.

A diferencia de los protocolos tradicionales de WSN, cuya aplicabilidad se reduce exclusivamente a aplicaciones de captura de datos o *data gathering*, HARP está diseñado para encaminar cualquier tipo de flujo de datos que se produzca en una WSN, incluyendo la diseminación de paquetes desde el sumidero a los nodos (*data dissemination*), modelos de funcionamiento basado en peticiones lanzadas por el sumidero (*query-based*) o flujo de paquetes entre cualquier par de nodos sensores de la red (*peer-to-peer*).

HARP incorpora mecanismos de tolerancia a fallos de red que permiten una gestión eficiente desde el punto de vista energético, limitando la sobrecarga producida en estos procesos. El mecanismo de recuperación local de rutas puede utilizarse en cualquier tipo de red inalámbrica, dentro de la arquitectura general propuesta del protocolo. Específicamente para WSN, estos mecanismos se complementan con otras propuestas como son el *slot de recuperación* y el *nodo sustituto*, los cuales proporcionan funcionalidades que permiten la adaptación dinámica del protocolo ante cambios en la topología de la red. A la vista de nuevas aplicaciones en las cuales los sensores pueden necesitar capacidad de movimiento, otros aspectos importantes como son la movilidad de los nodos y la adición de nuevos nodos a la red también está contemplados dentro de la funcionalidad del protocolo.

Esta Tesis establece además un marco teórico que posibilita el análisis de las prestaciones de HARP y su comparación con otros protocolos existentes basándose en el estudio probabilístico del tiempo de duración de las rutas entre cualquier par de nodos de la red. Este parámetro está directamente relacionado con el número de veces que se inicia el proceso de descubrimiento de rutas, el cual introduce sobrecarga de paquetes en la red y penaliza la eficiencia del protocolo de encaminamiento utilizado. Este estudio propone una expresión general para el cálculo de la *cdf* de dicho tiempo de vida y se complementa con un nuevo algoritmo que permite calcular para una topología de red dada la probabilidad de existencia de rutas disjuntas entre dos nodos. En cuanto al proceso de recuperación local de rutas, se analiza su estructura y se propone una formulación para calcular su probabilidad de éxito, en función de los flujos de paquetes que circulan por la ruta.

Este análisis teórico se valida en un nuevo simulador denominado WiNS y desarrollado para este tipo de redes y que sirve, a su vez, para comparar mediante simulaciones la eficiencia de HARP con los protocolos más representativos existentes en la actualidad. WiNS se ha diseñado siguiendo el modelo de simulación por eventos discretos y utilizando un lenguaje orientado a objetos como Java. WiNS se

articula en torno a la cola de eventos del sistema, que es la que procesa y organiza todos los eventos (transmisiones) que se producen en el sistema, implementando métodos y funcionalidades de propósito general, como el acceso al medio o los procesos de transmisión, propagación y recepción de paquetes, así como otras específicas y particulares para cada uno de los protocolos simulados.

Los resultados muestran que HARP y s-HARP obtienen mucho mejores prestaciones que otros protocolos representativos en WSN, tanto en redes homogéneas como en heterogéneas (en las que un porcentaje de los nodos tiene mayor capacidad energética). Además de las estructuras multi-salto creadas en HARP y que ahorran energía, s-HARP es capaz de equilibrar de forma muy notable el gasto energético en la red aprovechando al máximo las capacidades de cada nodo, lo que se traduce en un aumento considerable del tiempo de vida útil de la red y del número de paquetes de datos transmitidos.

Por último, se proponen y analizan otras aplicaciones del protocolo y se compara su eficiencia en redes distintas a las WSN, como son las redes inalámbricas de sensores y actuadores (WSAN) o las redes ad-hoc inalámbricas, las cuales han dado lugar a propuestas que han sido aceptadas y presentadas en diferentes congresos internacionales.

8.2. Publicaciones científicas

Los resultados de investigación obtenidos en esta Tesis Doctoral han dado lugar a los siguientes documentos científicos para su publicación en conferencias internacionales:

- F.J. Atero, J.J. Vinagre, F.J. Simó, M.R. Wilby, “*Adaptive Routing Protocol in a Communications Platform for Autonomous Public Transport Vehicles and Intelligent Traffic Management*”. Annual International Conference on

Control, Automation and Robotics (CAR 2011). Febrero de 2011. Singapur. Concedido el “**Best Paper Award**”.

- Fco. Javier Atero, Juan Jose Vinagre, Inmaculada Mora, Mark R. Wilby, “*Hierarchical Adaptive and Reliable Routing Protocol for Efficient Actor-Actor Coordination in Wireless Sensor and Actor ZigBee Networks*”. International Conference on Signal Processing, Communications and Networking (ICSPCN 2011) organizado por World Academy of Science, Engineering and Technology (WASET). Del 27 al 29 de abril de 2011. Venecia, Italia.
- F.J. Atero, J.J. Vinagre, E. Morgado, M.R. Wilby, “*A Low Energy and Adaptive Architecture for Efficient Routing and Robust Mobility Management in Wireless Sensor Networks*”. Fourth International Workshop on Sensor Networks (SN 2011) dentro del IEEE 31st International Conference on Distributed Computing System (ICDCS 2011). Del 20 al 24 de Junio de 2011. Minneapolis, Minnesota, EEUU. Aceptado para su presentación.
- Fco. Javier Atero, Juan Jose Vinagre, Julio Ramiro, Mark Wilby, “*A Low Energy and Adaptive Routing Architecture for Efficient Field Monitoring in Heterogeneous Wireless Sensor Networks*”. Workshop on Optimization Issues in Energy Efficient Distributed Systems (OPTIM 2011) como parte del IEEE International Conference on High Performance Computing and Simulation (HPCS 2011). Del 4 al 8 de Julio de 2011. Estambul, Turquía. Aceptado para su presentación.

8.3. Líneas futuras

El trabajo realizado en esta Tesis aporta una solución novedosa para el problema del encaminamiento eficiente en redes de sensores inalámbricos. A su vez, se desarrolla la arquitectura completa de un simulador por eventos discretos cuyo

uso puede ser generalizado a cualquier tipo de ser inalámbrica. Sin embargo, a lo largo del desarrollo de la misma, el trabajo realizado ha dado lugar a nuevos retos y posibilidades de ampliación que pueden ser interesantes de abordar. A continuación se enumeran algunas líneas futuras de investigación que se desprenden del trabajo llevado a cabo.

Programación de HARP en otros simuladores y comparativa con otros protocolos de WSN

En este sentido, el siguiente paso puede ser la implementación del protocolo propuesto en el simulador ns-2, ampliamente utilizado por grupos de investigación que trabajan en la línea de las redes inalámbricas, así como valorar su integración en otros simuladores de amplia aceptación. ns-2 permitiría la reutilización del código existente en dicho simulador relacionado con los diferentes niveles OSI, lo que concentra el trabajo a desarrollar exclusivamente en el nivel de red. Este nuevo desarrollo e integración en ns-2 facilitaría de forma inmediata la posibilidad de comparación de HARP con otros protocolos de encaminamiento propuestos por la comunidad científica cuyo código se encuentra ya incluido en el sistema.

Optimización del simulador WiNS

Una de las tareas siguientes que se puede abordar es la de modelar en el simulador de forma más real los aspectos relativos a la detección de eventos y toma de valores de los parámetros del entorno por parte de los nodos sensores. En la actualidad este proceso se ha enmascarado y simplificado en WiNS y puede ser interesante introducir nuevas funcionalidades que permitan el modelado mucho más real de la labor de monitorización de los nodos sensores, a través del diseño de mecanismos de generación de eventos y de asignación de valores a parámetros

(temperatura, humedad, etc.) en función de modelos empíricos. De esta forma se puede discriminar la lectura de esos parámetros por parte de los nodos en función de los dispositivos sensores que lleven integrados.

Otro de los aspectos que se pueden mejorar en el simulador desarrollado es la posibilidad de creación de una interfaz gráfica amigable con el usuario (GUI, *Graphical User Interface*) que facilite las labores de configuración de la red y que permita la creación rápida e intuitiva de distintas topologías de nodos, el lanzamiento de simulaciones, la visualización final de los resultados y la posible exportación de los más relevantes a otros formatos.

Con el objetivo de dotar al simulador de mayor capacidad para reproducir las condiciones reales sería beneficioso la introducción del modelo de propagación denominado *Shadowing Model*. A diferencia de los modelos de propagación en espacio libre y de 2 rayos, que predicen la potencia de recepción como una función determinista de la distancia y representan el radio de comunicación como un círculo ideal, este modelo más general representa más fielmente los fenómenos de reflexión reales y asume que la potencia de recepción a cierta distancia es una variable aleatoria debido a efectos de propagación multicamino.

Por último, como mejora sería interesante valorar la posibilidad de creación de un “modo emulación”, aunque ésta no sea propiamente una funcionalidad que se deba esperar de un simulador de red. Un emulador permite que dispositivos reales interaccionen entre sí a través la red simulada, siendo de esta forma posible evaluar las prestaciones del protocolo con toma de datos y funcionamiento real de dispositivos sensores.

Comparación de HARP con otros protocolos específicos de movilidad en WSN

En diferentes capítulos de esta Tesis se han descrito los procedimientos definidos en el protocolo propuesto para gestionar la movilidad de los nodos de la red y, en general, los métodos de tolerancia ante fallos de nodos y enlaces. Estos procesos tienen como objetivo final la adaptación dinámica del protocolo a los cambios en la topología de la red, siempre con la vista puesta en optimizar la eficiencia energética de los nodos. Ante la propuesta y relativamente reciente aparición de nuevos protocolos de encaminamiento diseñados específicamente para redes de sensores con movilidad, se plantea como futuro trabajo derivado de esta Tesis la comparación de las prestaciones de HARP con los más representativos dentro de este grupo para su utilización en aplicaciones en las cuales la movilidad tanto del nodo sumidero como de los sensores sea un factor importante, una vez hayan sido todos ellos incluidos en alguna plataforma de simulación como ns-2.

Implementación de un *testbed* con nodos reales

El desarrollo de un prototipo de implementación de HARP en una WSN con sensores reales y el estudio de los aspectos experimentales de un despliegue de este tipo es un buen método para refinar el diseño del protocolo y de los mecanismos en él definidos, así como para validar firmemente los resultados de simulación obtenidos. Estos aspectos experimentales pueden servir también para facilitar discusiones y propuestas relativas a aspectos de funcionamiento real no resueltos en su totalidad por la comunidad científica, así como a replantear o refinar asunciones establecidas, ya sea en el diseño hardware, software o relativas a los modelos empleados.

Interoperabilidad de HARP con otras redes

La integración de las WSN con otro tipo de redes, en particular las redes IP, es fundamental a la hora de extender su uso en aplicaciones reales. Este aspecto es además un tema que despierta actualmente gran interés en la comunidad científica. En consecuencia, la realización de experimentos reales de HARP como pasarela de encaminamiento que permita la interoperabilidad entre diferentes plataformas es un reto interesante que puede ser abordado en próximos trabajos.

Introducción de mecanismos de seguridad

Un aspecto no contemplado en la presente Tesis es el relacionado con los mecanismos de seguridad del protocolo de encaminamiento para evitar posibles amenazas de seguridad externas, como puede ser la integración en el funcionamiento de la red de nodos “intrusos” que logren extraer o analizar la información transmitida. La introducción de este tipo de mecanismos de seguridad que no perjudiquen las capacidades y prestaciones del protocolo se plantea como un tema con suficiente entidad y que merece un estudio adicional que puede ser interesante abordar en un futuro cercano.

Apéndices

Apéndice A

Ejemplo de aplicación de los algoritmos propuestos para el cálculo de probabilidad de rutas disjuntas

En la Sección 5.1 se implementó el Algoritmo 1, que realiza llamadas a los Algoritmos 2, 3 y 4, para proporcionar el valor de la probabilidad de existencia de un número determinado de rutas disjuntas $P[M_i = m_i]$ para cualquier distribución de nodos en la red, con objeto de calcular la *pdf* final del tiempo de vida de las rutas entre dos nodos de la misma. En este Apéndice se presenta un ejemplo de aplicación del mismo para el caso particular de la estructura de red mostrada en la Figura A.1, con 3 niveles intermedios y 5 nodos que proporcionan diferentes caminos de comunicación entre el origen S y el destino D.

Al vista de la Figura A.1 anterior se tiene que $n_2 = n_3 = n_4 = 5$. Se asume que $\Omega_i = 1, i = 1, 2, 3, 4$.

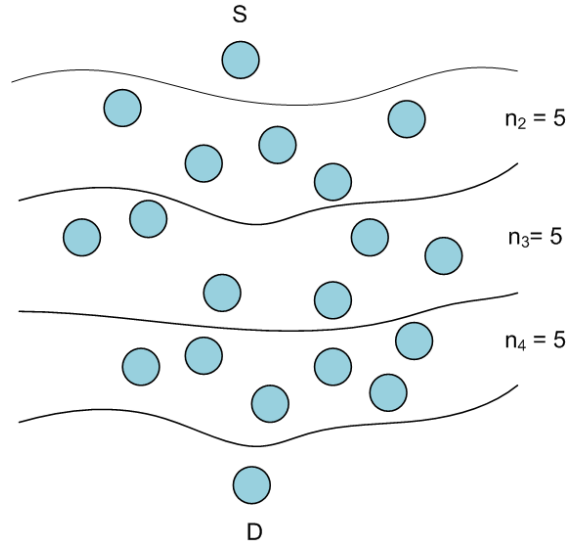


Figura A.1: Estructura de niveles y nodos utilizada en la aplicación de los Algoritmos 1, 2, 3 y 4 propuestos en esta Tesis.

A.1. Primer nivel intermedio

El primer nivel intermedio tiene en cuenta los nodos disponibles en el resto de niveles para la formación de rutas entre S y D. Para simplificar la formulación se definen los siguientes parámetros:

$$\begin{aligned}
 N_1^1 &= \left(\frac{n_2 - 1}{n_2} \cdot \frac{n_3 - 1}{n_3} \cdot \frac{n_4 - 1}{n_4} \right) \\
 N_2^1 &= \left(\frac{n_2 - 2}{n_2} \cdot \frac{n_3 - 2}{n_3} \cdot \frac{n_4 - 2}{n_4} \right) \\
 N_3^1 &= \left(\frac{n_2 - 3}{n_2} \cdot \frac{n_3 - 3}{n_3} \cdot \frac{n_4 - 3}{n_4} \right) \\
 N_4^1 &= \left(\frac{n_2 - 4}{n_2} \cdot \frac{n_3 - 4}{n_3} \cdot \frac{n_4 - 4}{n_4} \right)
 \end{aligned} \tag{A.1}$$

Con esa notación las variables asociadas a los primeros términos de los valores de probabilidad serían las siguientes:

$$\begin{aligned}
T_{21}(1) &= \prod_{m=0}^0 \left(\prod_{l=2}^4 \frac{(n_l - m) \cdot \Omega_1}{n_l} \right) = 1 \\
T_{21}(2) &= \prod_{m=0}^1 \left(\prod_{l=2}^4 \frac{(n_l - m) \cdot \Omega_1}{n_l} \right) = N_1^1 \\
T_{21}(3) &= \prod_{m=0}^2 \left(\prod_{l=2}^4 \frac{(n_l - m) \cdot \Omega_1}{n_l} \right) = N_1^1 \cdot N_2^1 \\
T_{21}(4) &= \prod_{m=0}^3 \left(\prod_{l=2}^4 \frac{(n_l - m) \cdot \Omega_1}{n_l} \right) = N_1^1 \cdot N_2^1 \cdot N_3^1 \\
T_{21}(5) &= \prod_{m=0}^4 \left(\prod_{l=2}^4 \frac{(n_l - m) \cdot \Omega_1}{n_l} \right) = N_1^1 \cdot N_2^1 \cdot N_3^1 \cdot N_4^1
\end{aligned} \tag{A.2}$$

Análogamente se puede calcular el segundo término utilizado para calcular cada uno de los valores de probabilidad en este nivel, de la forma siguiente:

$$\begin{aligned}
T_{22}(1) &= \sum_{h=1}^1 \prod_{j=1}^4 \left(1 - \prod_{l=2}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^1)^4 \\
T_{22}(2) &= \sum_{h=1}^4 \prod_{j=1}^3 \left(1 - \prod_{l=2}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^1)^3 + \\
&\quad + (1 - N_1^1)^2 \cdot (1 - N_2^1) + \\
&\quad + (1 - N_1^1) \cdot (1 - N_2^1)^2 + (1 - N_2^1)^3 \\
T_{22}(3) &= \sum_{h=1}^6 \prod_{j=1}^2 \left(1 - \prod_{l=2}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^1)^2 + \\
&\quad + (1 - N_1^1) \cdot (1 - N_2^1) + (1 - N_1^1) \cdot (1 - N_3^1) + \\
&\quad + (1 - N_2^1)^2 + (1 - N_2^1) \cdot (1 - N_3^1) + (1 - N_3^1)^2 \\
T_{22}(4) &= \sum_{h=1}^4 \prod_{j=1}^1 \left(1 - \prod_{l=2}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^1) + \\
&\quad + (1 - N_2^1) + (1 - N_3^1) + (1 - N_4^1) \\
T_{22}(5) &= 1
\end{aligned} \tag{A.3}$$

Finalmente, las probabilidades de encontrar un número determinado de rutas disjuntas que parten del origen y utilizan como primer salto los nodos de este primer nivel intermedio se calculan a través de las variables obtenidas anteriormente, de la forma siguiente:

$$\begin{aligned}P(M_1 = 0) &= 0 \\P(M_1 = 1) &= T_{11}(1) \cdot T_{12}(1) \\P(M_1 = 2) &= T_{11}(2) \cdot T_{12}(2) \\P(M_1 = 3) &= T_{11}(3) \cdot T_{12}(3) \\P(M_1 = 4) &= T_{11}(4) \cdot T_{12}(4) \\P(M_1 = 5) &= T_{11}(5) \cdot T_{12}(5)\end{aligned}\tag{A.4}$$

Y finalmente, para comprobar los resultados obtenidos, su suma deberá ser igual a uno:

$$\begin{aligned}P(M_1) &= P(M_1 = 1) + P(M_1 = 2) + P(M_1 = 3) + P(M_1 = 4) + P(M_1 = 5) = \\&= 1\end{aligned}\tag{A.5}$$

A.2. Segundo nivel intermedio

De forma análoga al primer nivel, se procede inicialmente a definir las variables que se utilizarán en las operaciones posteriores que, en este caso, son las siguientes:

$$\begin{aligned}
 N_1^2 &= \left(\frac{n_3 - 1}{n_3} \cdot \frac{n_4 - 1}{n_4} \right) \\
 N_2^2 &= \left(\frac{n_3 - 2}{n_3} \cdot \frac{n_4 - 2}{n_4} \right) \\
 N_3^2 &= \left(\frac{n_3 - 3}{n_3} \cdot \frac{n_4 - 3}{n_4} \right) \\
 N_4^2 &= \left(\frac{n_3 - 4}{n_3} \cdot \frac{n_4 - 4}{n_4} \right)
 \end{aligned} \tag{A.6}$$

$$P(M_2 = 0)$$

Los primeros términos para el cálculo de probabilidades son todos iguales a uno:

$$\begin{aligned}
 T_{21}(0, 1) &= 1 \\
 T_{21}(0, 2) &= 1 \\
 T_{21}(0, 3) &= 1 \\
 T_{21}(0, 4) &= 1 \\
 T_{21}(0, 5) &= 1
 \end{aligned} \tag{A.7}$$

Por su parte, los segundos términos se calculan de la siguiente forma:

$$\begin{aligned}
T_{22}(0, 1) &= \sum_{h=1}^1 \prod_{j=1}^4 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^2)^4 \\
T_{22}(0, 2) &= \sum_{h=1}^1 \prod_{j=1}^3 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_2^2)^3 \\
T_{22}(0, 3) &= \sum_{h=1}^1 \prod_{j=1}^2 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_3^2)^2 \\
T_{22}(0, 4) &= \sum_{h=1}^1 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_4^2) \\
T_{22}(0, 5) &= 1
\end{aligned} \tag{A.8}$$

Siendo $M_{T_2} = M_1$ el número de las rutas disjuntas creadas que utilizan nodos del nivel primero, las probabilidades condicionadas serán las siguientes:

$$\begin{aligned}
P(M_2 = 0 \mid M_{T_2} = 1) &= T_{21}(0, 1) \cdot T_{22}(0, 1) \\
P(M_2 = 0 \mid M_{T_2} = 2) &= T_{21}(0, 2) \cdot T_{22}(0, 2) \\
P(M_2 = 0 \mid M_{T_2} = 3) &= T_{21}(0, 3) \cdot T_{22}(0, 3) \\
P(M_2 = 0 \mid M_{T_2} = 4) &= T_{21}(0, 4) \cdot T_{22}(0, 4) \\
P(M_2 = 0 \mid M_{T_2} = 5) &= T_{21}(0, 5) \cdot T_{22}(0, 5)
\end{aligned} \tag{A.9}$$

Finalmente, teniendo en cuenta las probabilidades calculadas para el primer nivel se obtiene la probabilidad final de que no existan rutas desde este segundo

nivel, cuya expresión general tendrá la siguiente forma:

$$\begin{aligned}
 P(M_2 = 0) &= P(M_2 = 0 \mid M_{T_2} = 1) \cdot P(M_{T_2} = 1) + \\
 &+ P(M_2 = 0 \mid M_{T_2} = 2) \cdot P(M_{T_2} = 2) + \\
 &+ P(M_2 = 0 \mid M_{T_2} = 3) \cdot P(M_{T_2} = 3) + \\
 &+ P(M_2 = 0 \mid M_{T_2} = 4) \cdot P(M_{T_2} = 4) + \\
 &+ P(M_2 = 0 \mid M_{T_2} = 5) \cdot P(M_{T_2} = 5)
 \end{aligned} \tag{A.10}$$

$$P(M_2 = 1)$$

Los primeros términos para el cálculo de probabilidades en este nivel y con este número de rutas son los siguientes:

$$\begin{aligned}
 T_{21}(1, 1) &= N_1^2 \\
 T_{21}(1, 2) &= N_2^2 \\
 T_{21}(1, 3) &= N_3^2 \\
 T_{21}(1, 4) &= N_4^2 \\
 T_{21}(1, 5) &= 1
 \end{aligned} \tag{A.11}$$

Siendo los segundos de la forma:

$$\begin{aligned}
T_{22}(1, 1) &= \sum_{h=1}^4 \prod_{j=1}^3 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^2)^3 + \\
&\quad + (1 - N_1^2)^2 \cdot (1 - N_2^2) + \\
&\quad + (1 - N_1^2) \cdot (1 - N_2^2)^2 + (1 - N_1^2)^2 \\
T_{22}(1, 2) &= \sum_{h=1}^3 \prod_{j=1}^2 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_2^2)^2 + \\
&\quad + (1 - N_2^2) \cdot (1 - N_3^2) + (1 - N_3^2)^2 \\
T_{22}(1, 3) &= \sum_{h=1}^2 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_3^2) + (1 - N_4^2) \\
T_{22}(1, 4) &= 1 \\
T_{22}(1, 5) &= 0
\end{aligned} \tag{A.12}$$

Las probabilidades condicionadas al número de rutas disponibles en el nivel anterior serían las siguientes:

$$\begin{aligned}
P(M_2 = 1 \mid M_{T_2} = 1) &= T_{21}(1, 1) \cdot T_{22}(1, 1) \\
P(M_2 = 1 \mid M_{T_2} = 2) &= T_{21}(1, 2) \cdot T_{22}(1, 2) \\
P(M_2 = 1 \mid M_{T_2} = 3) &= T_{21}(1, 3) \cdot T_{22}(1, 3) \\
P(M_2 = 1 \mid M_{T_2} = 4) &= T_{21}(1, 4) \cdot T_{22}(1, 4) \\
P(M_2 = 1 \mid M_{T_2} = 5) &= T_{21}(1, 5) \cdot T_{22}(1, 5)
\end{aligned} \tag{A.13}$$

Finalmente, teniendo en cuenta las probabilidades en el primer nivel, se obtiene la probabilidad de encontrar una única ruta que parta desde algún nodo de este segundo nivel:

$$\begin{aligned}
P(M_2 = 1) &= P(M_2 = 1 \mid M_{T_2} = 1) \cdot P(M_{T_2} = 1) + \\
&+ P(M_2 = 1 \mid M_{T_2} = 2) \cdot P(M_{T_2} = 2) + \\
&+ P(M_2 = 1 \mid M_{T_2} = 3) \cdot P(M_{T_2} = 3) + \\
&+ P(M_2 = 1 \mid M_{T_2} = 4) \cdot P(M_{T_2} = 4) + \\
&+ P(M_2 = 1 \mid M_{T_2} = 5) \cdot P(M_{T_2} = 5)
\end{aligned} \tag{A.14}$$

$$P(M_2 = 2)$$

Los primeros términos se representan en este caso de la forma siguiente:

$$\begin{aligned}
T_{21}(2, 1) &= N_1^2 \cdot N_2^2 \\
T_{21}(2, 2) &= N_2^2 \cdot N_3^2 \\
T_{21}(2, 3) &= N_3^2 \cdot N_4^2 \\
T_{21}(2, 4) &= 1 \\
T_{21}(2, 5) &= 0
\end{aligned} \tag{A.15}$$

Los segundos términos se simplifican conforme aumenta el número de rutas, resultando para este valor del número de rutas como sigue:

$$\begin{aligned}
T_{22}(2, 1) &= \sum_{h=1}^6 \prod_{j=1}^2 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^2)^2 + \\
&+ (1 - N_2^2)^2 (1 - N_3^2)^2 + (1 - N_1^2) \cdot (1 - N_2^2) + \\
&+ (1 - N_1^2) \cdot (1 - N_3^2) + (1 - N_2^2) \cdot (1 - N_3^2) \\
T_{22}(2, 2) &= \sum_{h=1}^2 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_2^2) + \\
&+ (1 - N_3^2) + (1 - N_4^2) \\
T_{22}(2, 3) &= 1 \\
T_{22}(2, 4) &= 0 \\
T_{22}(2, 5) &= 0
\end{aligned} \tag{A.16}$$

Las probabilidades condicionadas son las siguientes:

$$\begin{aligned}
P(M_2 = 2 \mid M_{T_2} = 1) &= T_{21}(2, 1) \cdot T_{22}(2, 1) \\
P(M_2 = 2 \mid M_{T_2} = 2) &= T_{21}(2, 2) \cdot T_{22}(2, 2) \\
P(M_2 = 2 \mid M_{T_2} = 3) &= T_{21}(2, 3) \cdot T_{22}(2, 3) \\
P(M_2 = 2 \mid M_{T_2} = 4) &= T_{21}(2, 4) \cdot T_{22}(2, 4) \\
P(M_2 = 2 \mid M_{T_2} = 5) &= T_{21}(2, 5) \cdot T_{22}(2, 5)
\end{aligned} \tag{A.17}$$

Finalmente, de igual forma que en los casos anteriores, la expresión general de la probabilidad queda como sigue:

$$\begin{aligned}
P(M_2 = 2) &= P(M_2 = 2 \mid M_{T_2} = 1) \cdot P(M_{T_2} = 1) + \\
&+ P(M_2 = 2 \mid M_{T_2} = 2) \cdot P(M_{T_2} = 2) + \\
&+ P(M_2 = 2 \mid M_{T_2} = 3) \cdot P(M_{T_2} = 3) + \\
&+ P(M_2 = 2 \mid M_{T_2} = 4) \cdot P(M_{T_2} = 4) + \\
&+ P(M_2 = 2 \mid M_{T_2} = 5) \cdot P(M_{T_2} = 5)
\end{aligned} \tag{A.18}$$

$$P(M_2 = 3)$$

Los primeros términos se corresponden con las siguientes expresiones:

$$\begin{aligned}
T_{21}(3, 1) &= N_1^2 \cdot N_2^2 \cdot N_3^2 \\
T_{21}(3, 2) &= N_2^2 \cdot N_3^2 \cdot N_4^2 \\
T_{21}(3, 3) &= 1 \\
T_{21}(3, 4) &= 1 \\
T_{21}(3, 5) &= 0
\end{aligned} \tag{A.19}$$

Los segundos términos se expresan ahora de la forma:

$$\begin{aligned}
T_{22}(3, 1) &= \sum_{h=1}^4 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^2) + \\
&+ (1 - N_2^2) + (1 - N_3^2) + (1 - N_4^2) \\
T_{22}(3, 2) &= 1 \\
T_{22}(3, 3) &= 0 \\
T_{22}(3, 4) &= 0 \\
T_{22}(3, 5) &= 0
\end{aligned} \tag{A.20}$$

Las probabilidades condicionadas se obtienen de forma análoga a los casos anteriores:

$$\begin{aligned}
P(M_2 = 3 \mid M_{T_2} = 1) &= T_{21}(3, 1) \cdot T_{22}(3, 1) \\
P(M_2 = 3 \mid M_{T_2} = 2) &= T_{21}(3, 2) \cdot T_{22}(3, 2) \\
P(M_2 = 3 \mid M_{T_2} = 3) &= T_{21}(3, 3) \cdot T_{22}(3, 3) \\
P(M_2 = 3 \mid M_{T_2} = 4) &= T_{21}(3, 4) \cdot T_{22}(3, 4) \\
P(M_2 = 3 \mid M_{T_2} = 5) &= T_{21}(3, 5) \cdot T_{22}(3, 5)
\end{aligned} \tag{A.21}$$

Finalmente, teniendo en cuenta A.20, los tres últimos términos de la expresión general siguiente son nulos:

$$\begin{aligned}
P(M_2 = 3) &= P(M_2 = 3 \mid M_{T_2} = 1) \cdot P(M_{T_2} = 1) + \\
&+ P(M_2 = 3 \mid M_{T_2} = 2) \cdot P(M_{T_2} = 2) + \\
&+ P(M_2 = 3 \mid M_{T_2} = 3) \cdot P(M_{T_2} = 3) + \\
&+ P(M_2 = 3 \mid M_{T_2} = 4) \cdot P(M_{T_2} = 4) + \\
&+ P(M_2 = 3 \mid M_{T_2} = 5) \cdot P(M_{T_2} = 5)
\end{aligned} \tag{A.22}$$

$$P(M_2 = 4)$$

Análogamente a los casos anteriores, se tiene que:

$$\begin{aligned}
T_{21}(4, 1) &= N_1^2 \cdot N_2^2 \cdot N_3^2 \cdot N_4^2 \\
T_{21}(4, 2) &= 1 \\
T_{21}(4, 3) &= 1 \\
T_{21}(4, 4) &= 1 \\
T_{21}(4, 5) &= 0
\end{aligned} \tag{A.23}$$

En cuanto a los segundos términos, todos son nulos excepto el primero:

$$\begin{aligned}
 T_{22}(4, 1) &= 1 \\
 T_{22}(4, 2) &= 0 \\
 T_{22}(4, 3) &= 0 \\
 T_{22}(4, 4) &= 0 \\
 T_{22}(4, 5) &= 0
 \end{aligned}
 \tag{A.24}$$

El valor de la probabilidad será en este caso:

$$\begin{aligned}
 P(M_2 = 4 \mid M_{T_2} = 1) &= T_{21}(4, 1) \cdot T_{22}(4, 1) \\
 P(M_2 = 4 \mid M_{T_2} = 2) &= T_{21}(4, 2) \cdot T_{22}(4, 2) \\
 P(M_2 = 4 \mid M_{T_2} = 3) &= T_{21}(4, 4) \cdot T_{22}(4, 4) \\
 P(M_2 = 4 \mid M_{T_2} = 4) &= T_{21}(4, 4) \cdot T_{22}(4, 4) \\
 P(M_2 = 4 \mid M_{T_2} = 5) &= T_{21}(4, 5) \cdot T_{22}(4, 5)
 \end{aligned}
 \tag{A.25}$$

A la vista de los segundos términos de estas probabilidades en A.24, únicamente será no nulo el primer sumando de la expresión general de la probabilidad incluida en la siguiente fórmula:

$$\begin{aligned}
 P(M_2 = 4) &= P(M_2 = 4 \mid M_{T_2} = 1) \cdot P(M_{T_2} = 1) + \\
 &+ P(M_2 = 4 \mid M_{T_2} = 2) \cdot P(M_{T_2} = 2) + \\
 &+ P(M_2 = 4 \mid M_{T_2} = 3) \cdot P(M_{T_2} = 3) + \\
 &+ P(M_2 = 4 \mid M_{T_2} = 4) \cdot P(M_{T_2} = 4) + \\
 &+ P(M_2 = 4 \mid M_{T_2} = 5) \cdot P(M_{T_2} = 5)
 \end{aligned}
 \tag{A.26}$$

$$P(M_2 = 5)$$

$P(M_2 = 5) = 0$, ya que por las asunciones realizadas siempre existirá una ruta como mínimo desde S hasta D.

Cálculo de $P(M_2)$

Se puede comprobar la validez de los resultados obtenidos sabiendo que:

$$\begin{aligned} P(M_2) &= P(M_2 = 1) + P(M_2 = 2) + P(M_2 = 3) + P(M_2 = 4) + P(M_2 = 5) = \\ &= 1 \end{aligned} \tag{A.27}$$

A.3. Tercer nivel intermedio

Siguiendo los mismos pasos que para los niveles primero y segundo, se definen en primer lugar las siguientes variables:

$$\begin{aligned} N_1^3 &= \frac{n_4 - 1}{n_4} \\ N_2^3 &= \frac{n_4 - 2}{n_4} \\ N_3^3 &= \frac{n_4 - 3}{n_4} \\ N_4^3 &= \frac{n_4 - 4}{n_4} \end{aligned} \tag{A.28}$$

$$P(M_T)$$

El primer paso a ejecutar en este tercer nivel consiste en el cálculo de la probabilidad de rutas disjuntas existentes en los niveles anteriores al tercero, de tal forma que se obtienen las siguientes combinaciones de casos posibles en función

de cómo se repartan las rutas entre los niveles intermedio primero y segundo:

$$\begin{aligned}
P(M_T = 1) &= P(M_2 = 0 \mid M_1 = 1) \cdot P(M_1 = 1) \\
P(M_T = 2) &= P(M_2 = 0 \mid M_1 = 2) \cdot P(M_1 = 2) + \\
&\quad + P(M_2 = 1 \mid M_1 = 1) \cdot P(M_1 = 1) \\
P(M_T = 3) &= P(M_2 = 0 \mid M_1 = 3) \cdot P(M_1 = 3) + \\
&\quad + P(M_2 = 1 \mid M_1 = 2) \cdot P(M_1 = 2) + \\
&\quad + P(M_2 = 2 \mid M_1 = 1) \cdot P(M_1 = 1) \\
P(M_T = 4) &= P(M_2 = 0 \mid M_1 = 4) \cdot P(M_1 = 4) + \\
&\quad + P(M_2 = 1 \mid M_1 = 3) \cdot P(M_1 = 3) + \\
&\quad + P(M_2 = 2 \mid M_1 = 2) \cdot P(M_1 = 2) + \\
&\quad + P(M_2 = 3 \mid M_1 = 1) \cdot P(M_1 = 1) \\
P(M_T = 5) &= P(M_2 = 0 \mid M_1 = 5) \cdot P(M_1 = 5) + \\
&\quad + P(M_2 = 1 \mid M_1 = 4) \cdot P(M_1 = 4) + \\
&\quad + P(M_2 = 2 \mid M_1 = 3) \cdot P(M_1 = 3) + \\
&\quad + P(M_2 = 3 \mid M_1 = 2) \cdot P(M_1 = 2) + \\
&\quad + P(M_2 = 4 \mid M_1 = 1) \cdot P(M_1 = 1)
\end{aligned} \tag{A.29}$$

Como en los casos anteriores, la suma de todas las probabilidades será igual a la unidad:

$$\begin{aligned}
P(M_T) &= P(M_T = 1) + P(M_T = 2) + P(M_T = 3) + P(M_T = 4) + P(M_T = 5) = \\
&= 1
\end{aligned} \tag{A.30}$$

$P(M_3 = 0)$

Primeros términos:

$$\begin{aligned}
 T_{31}(0, 1) &= 1 \\
 T_{31}(0, 2) &= 1 \\
 T_{31}(0, 3) &= 1 \\
 T_{31}(0, 4) &= 1 \\
 T_{31}(0, 5) &= 1
 \end{aligned} \tag{A.31}$$

Segundos términos:

$$\begin{aligned}
 T_{32}(0, 1) &= \sum_{h=1}^1 \prod_{j=1}^4 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_1^3)^4 \\
 T_{32}(0, 2) &= \sum_{h=1}^1 \prod_{j=1}^3 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_2^3)^3 \\
 T_{32}(0, 3) &= \sum_{h=1}^1 \prod_{j=1}^2 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_3^3)^2 \\
 T_{32}(0, 4) &= \sum_{h=1}^1 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = (1 - N_4^3) \\
 T_{32}(0, 5) &= 1
 \end{aligned} \tag{A.32}$$

Las probabilidades condicionadas serían las siguientes:

$$\begin{aligned}
 P(M_3 = 0 \mid M_T = 1) &= T_{31}(0, 1) \cdot T_{32}(0, 1) \\
 P(M_3 = 0 \mid M_T = 2) &= T_{31}(0, 2) \cdot T_{32}(0, 2) \\
 P(M_3 = 0 \mid M_T = 3) &= T_{31}(0, 3) \cdot T_{32}(0, 3) \\
 P(M_3 = 0 \mid M_T = 4) &= T_{31}(0, 4) \cdot T_{32}(0, 4) \\
 P(M_3 = 0 \mid M_T = 5) &= T_{31}(0, 5) \cdot T_{32}(0, 5)
 \end{aligned} \tag{A.33}$$

Finalmente, se aplica la expresión general de la probabilidad:

$$\begin{aligned}
 P(M_3 = 0) &= P(M_3 = 0 \mid M_T = 1) \cdot P(M_T = 1) + \\
 &+ P(M_3 = 0 \mid M_T = 2) \cdot P(M_T = 2) + \\
 &+ P(M_3 = 0 \mid M_T = 3) \cdot P(M_T = 3) + \\
 &+ P(M_3 = 0 \mid M_T = 4) \cdot P(M_T = 4) + \\
 &+ P(M_3 = 0 \mid M_T = 5) \cdot P(M_T = 5)
 \end{aligned} \tag{A.34}$$

$$P(M_3 = 1)$$

Primeros términos:

$$\begin{aligned}
 T_{31}(1, 1) &= N_1^3 \\
 T_{31}(1, 2) &= N_2^3 \\
 T_{31}(1, 3) &= N_3^3 \\
 T_{31}(1, 4) &= N_4^3 \\
 T_{31}(1, 5) &= 1
 \end{aligned} \tag{A.35}$$

Segundos términos:

$$\begin{aligned}
T_{32}(1, 1) &= \sum_{h=1}^4 \prod_{j=1}^3 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = \\
&= (1 - N_1^3)^3 + (1 - N_1^3)^2 \cdot (1 - N_2^3) + \\
&+ (1 - N_1^3) \cdot (1 - N_2^3)^2 + (1 - N_2^3)^2 \\
T_{32}(1, 2) &= \sum_{h=1}^3 \prod_{j=1}^2 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = \\
&= (1 - N_2^3)^2 + (1 - N_2^3) \cdot (1 - N_3^3) + \\
&+ (1 - N_3^3)^2 \\
T_{32}(1, 3) &= \sum_{h=1}^2 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = \\
&= (1 - N_3^3) + (1 - N_4^3) \\
T_{32}(1, 4) &= 1 \\
T_{32}(1, 5) &= 0
\end{aligned} \tag{A.36}$$

Según los algoritmos, las probabilidades condicionadas tendrán en este caso las expresiones siguientes:

$$\begin{aligned}
P(M_3 = 1 \mid M_T = 1) &= T_{31}(1, 1) \cdot T_{32}(1, 1) \\
P(M_3 = 1 \mid M_T = 2) &= T_{31}(1, 2) \cdot T_{32}(1, 2) \\
P(M_3 = 1 \mid M_T = 3) &= T_{31}(1, 3) \cdot T_{32}(1, 3) \\
P(M_3 = 1 \mid M_T = 4) &= T_{31}(1, 4) \cdot T_{32}(1, 4) \\
P(M_3 = 1 \mid M_T = 5) &= T_{31}(1, 5) \cdot T_{32}(1, 5)
\end{aligned} \tag{A.37}$$

Finalmente:

$$\begin{aligned}
 P(M_3 = 1) &= P(M_3 = 1 \mid M_T = 1) \cdot P(M_T = 1) + \\
 &+ P(M_3 = 1 \mid M_T = 2) \cdot P(M_T = 2) + \\
 &+ P(M_3 = 1 \mid M_T = 3) \cdot P(M_T = 3) + \\
 &+ P(M_3 = 1 \mid M_T = 4) \cdot P(M_T = 4) + \\
 &+ P(M_3 = 1 \mid M_T = 5) \cdot P(M_T = 5)
 \end{aligned} \tag{A.38}$$

$$P(M_3 = 2)$$

Primeros términos:

$$\begin{aligned}
 T_{31}(2, 1) &= N_1^3 \cdot N_2^3 \\
 T_{31}(2, 2) &= N_2^3 \cdot N_3^3 \\
 T_{31}(2, 3) &= N_3^3 \cdot N_4^3 \\
 T_{31}(2, 4) &= 1 \\
 T_{31}(2, 5) &= 0
 \end{aligned} \tag{A.39}$$

Los segundos términos serán:

$$\begin{aligned}
T_{32}(2, 1) &= \sum_{h=1}^6 \prod_{j=1}^2 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = \\
&= (1 - N_1^3)^2 + (1 - N_2^3)^2 (1 - N_3^3)^2 + \\
&+ (1 - N_1^3) \cdot (1 - N_2^3) + (1 - N_1^3) \cdot (1 - N_3^3) + \\
&+ (1 - N_2^3) \cdot (1 - N_3^3) \\
T_{32}(2, 2) &= \sum_{h=1}^2 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = \\
&= (1 - N_2^3) + (1 - N_3^3) + (1 - N_4^3) \\
T_{32}(2, 3) &= 1 \\
T_{32}(2, 4) &= 0 \\
T_{32}(2, 5) &= 0
\end{aligned} \tag{A.40}$$

Las probabilidades se pueden expresar en función de los primeros y segundos términos anteriores de la forma:

$$\begin{aligned}
P(M_3 = 2 \mid M_T = 1) &= T_{31}(2, 1) \cdot T_{32}(3, 1) \\
P(M_3 = 2 \mid M_T = 2) &= T_{31}(2, 2) \cdot T_{32}(3, 2) \\
P(M_3 = 2 \mid M_T = 3) &= T_{31}(2, 3) \cdot T_{32}(3, 3) \\
P(M_3 = 2 \mid M_T = 4) &= T_{31}(2, 4) \cdot T_{32}(3, 4) \\
P(M_3 = 2 \mid M_T = 5) &= T_{31}(2, 5) \cdot T_{32}(3, 5)
\end{aligned} \tag{A.41}$$

Finalmente, teniendo en cuenta las probabilidades de rutas disjuntas calculadas para los niveles anteriores:

$$\begin{aligned}
P(M_3 = 2) &= P(M_3 = 2 \mid M_T = 1) \cdot P(M_T = 1) + \\
&+ P(M_3 = 2 \mid M_T = 2) \cdot P(M_T = 2) + \\
&+ P(M_3 = 2 \mid M_T = 3) \cdot P(M_T = 3) + \\
&+ P(M_3 = 2 \mid M_T = 4) \cdot P(M_T = 4) + \\
&+ P(M_3 = 2 \mid M_T = 5) \cdot P(M_T = 5)
\end{aligned} \tag{A.42}$$

$$P(M_3 = 3)$$

Los primeros términos tendrán las expresiones siguientes:

$$\begin{aligned}
T_{31}(3, 1) &= N_1^3 \cdot N_2^3 \cdot N_3^3 \\
T_{31}(3, 2) &= N_2^3 \cdot N_3^3 \cdot N_4^3 \\
T_{31}(3, 3) &= 1 \\
T_{31}(3, 4) &= 1 \\
T_{31}(3, 5) &= 0
\end{aligned} \tag{A.43}$$

Y los segundos términos:

$$\begin{aligned}
T_{32}(3, 1) &= \sum_{h=1}^4 \prod_{j=1}^1 \left(1 - \prod_{l=3}^4 \frac{(n_l - u_{hj}) \cdot \Omega_1}{n_l} \right) = \\
&= (1 - N_1^3) + (1 - N_2^3) + (1 - N_3^3) + \\
&+ (1 - N_4^3) \\
T_{32}(3, 2) &= 1 \\
T_{32}(3, 3) &= 0 \\
T_{32}(3, 4) &= 0 \\
T_{32}(3, 5) &= 0
\end{aligned} \tag{A.44}$$

Las probabilidades condicionadas son las siguientes:

$$\begin{aligned}
P(M_3 = 3 \mid M_T = 1) &= T_{31}(3, 1) \cdot T_{32}(3, 1) \\
P(M_3 = 3 \mid M_T = 2) &= T_{31}(3, 2) \cdot T_{32}(3, 2) \\
P(M_3 = 3 \mid M_T = 3) &= T_{31}(3, 3) \cdot T_{32}(3, 3) \\
P(M_3 = 3 \mid M_T = 4) &= T_{31}(3, 4) \cdot T_{32}(3, 4) \\
P(M_3 = 3 \mid M_T = 5) &= T_{31}(3, 5) \cdot T_{32}(3, 5)
\end{aligned} \tag{A.45}$$

Finalmente, la fórmula general es la indicada a continuación:

$$\begin{aligned}
P(M_3 = 3) &= P(M_3 = 3 \mid M_T = 1) \cdot P(M_T = 1) + \\
&+ P(M_3 = 3 \mid M_T = 2) \cdot P(M_T = 2) + \\
&+ P(M_3 = 3 \mid M_T = 3) \cdot P(M_T = 3) + \\
&+ P(M_3 = 3 \mid M_T = 4) \cdot P(M_T = 4) + \\
&+ P(M_3 = 3 \mid M_T = 5) \cdot P(M_T = 5)
\end{aligned} \tag{A.46}$$

$$(M_3 = 4)$$

Primeros términos para el cálculo de la probabilidad:

$$\begin{aligned}
 T_{31}(4, 1) &= N_1^3 \cdot N_2^3 \cdot N_3^3 \cdot N_4^3 \\
 T_{31}(4, 2) &= 1 \\
 T_{31}(4, 3) &= 1 \\
 T_{31}(4, 4) &= 1 \\
 T_{31}(4, 5) &= 0
 \end{aligned}
 \tag{A.47}$$

Segundos términos:

$$\begin{aligned}
 T_{32}(4, 1) &= 1 \\
 T_{32}(4, 2) &= 0 \\
 T_{32}(4, 3) &= 0 \\
 T_{32}(4, 4) &= 0 \\
 T_{32}(4, 5) &= 0
 \end{aligned}
 \tag{A.48}$$

Las probabilidades están condicionadas por la existencia de rutas en los niveles anteriores, según se expresa en las siguientes ecuaciones:

$$\begin{aligned}
 P(M_3 = 4 \mid M_T = 1) &= T_{31}(4, 1) \cdot T_{32}(4, 1) \\
 P(M_3 = 4 \mid M_T = 2) &= T_{31}(4, 2) \cdot T_{32}(4, 2) \\
 P(M_3 = 4 \mid M_T = 3) &= T_{31}(4, 3) \cdot T_{32}(4, 3) \\
 P(M_3 = 4 \mid M_T = 4) &= T_{31}(4, 4) \cdot T_{32}(4, 4) \\
 P(M_3 = 4 \mid M_T = 5) &= T_{31}(4, 5) \cdot T_{32}(4, 5)
 \end{aligned}
 \tag{A.49}$$

Finalmente, teniendo en cuenta A.48 el único término no nulo de la fórmula general siguiente será el primero de ellos:

$$\begin{aligned}
P(M_3 = 4) &= P(M_3 = 4 \mid M_T = 1) \cdot P(M_T = 1) + \\
&+ P(M_3 = 4 \mid M_T = 2) \cdot P(M_T = 2) + \\
&+ P(M_3 = 4 \mid M_T = 3) \cdot P(M_T = 3) + \\
&+ P(M_3 = 4 \mid M_T = 4) \cdot P(M_T = 4) + \\
&+ P(M_3 = 4 \mid M_T = 5) \cdot P(M_T = 5)
\end{aligned} \tag{A.50}$$

$$P(M_3 = 5)$$

Por último, análogamente al caso del segundo nivel intermedio, $P(M_3 = 5) = 0$ debido a que siempre existirán rutas que parten de S y utilizan nodos de los niveles anteriores a éste.

Cálculo de $P(M_3)$

La suma de los valores de probabilidad en este nivel también debe ser igual a uno:

$$\begin{aligned}
P(M_3) &= P(M_3 = 1) + P(M_3 = 2) + P(M_3 = 3) + P(M_3 = 4) + P(M_3 = 5) = \\
&= 1
\end{aligned} \tag{A.51}$$

A.4. Resultados

Aplicando la formulación anterior con los valores $n_2 = n_3 = n_4 = 5$ se obtienen los resultados numéricos indicados en la Tabla A.1. En ellos puede observarse que la configuración más probable de rutas disjuntas entre S y D es la siguiente:

- Dos rutas disjuntas que, partiendo de S, utilizan nodos del primer nivel intermedio hacia D.
- Dos rutas que parten de cada uno de los nodos del primer nivel utilizados en las rutas anteriores y desde ahí son disjuntas con todas las demás.
- Una ruta principal que parte de uno de los nodos del segundo nivel utilizados en las rutas anteriores y alcanza el destino sin compartir enlaces con otras rutas.

Por último, una vez obtenidos los valores de probabilidad de rutas, aplicando estos resultados probabilísticos a la Ecuación (5.53) se obtiene la expresión de la *pdf* del tiempo de vida de las rutas desde S hasta D para este caso particular de aplicación de HARP. De forma análoga puede obtenerse la *pdf* de la estructura *single-path* y de la estructura de N rutas disjuntas desde el origen mediante la aplicación del teorema de la probabilidad total a las Ecuaciones (5.5) y (5.8), derivando después con respecto a la variable tiempo la expresión de la *cdf* obtenida para obtener la *pdf*. La Figura A.2 muestra la representación gráfica de la *pdf* de estas tres configuraciones para un valor del tiempo de vida medio de un enlace inalámbrico de $l = 10s$, de la que se deduce que para esta configuración de red, al estar su *pdf* más desplazada hacia la derecha que las otras dos, la estructura de rutas disjuntas creada por HARP siempre presentará mayor probabilidad de prolongar el tiempo de vida de las rutas respecto de las otras dos estructuras. Esta *pdf* es un límite inferior para las prestaciones de HARP, ya que aquí no está incluida la contribución de las rutas no disjuntas a la prolongación del tiempo de vida total.

Número de rutas disjuntas	Probabilidad
<i>Primer nivel intermedio</i>	
$M_1 = 0$	0
$M_1 = 1$	0.0567
$M_1 = 2$	0.5554
$M_1 = 3$	0.3652
$M_1 = 4$	0.0226
$M_1 = 5$	0.00005
<i>Segundo nivel intermedio</i>	
$M_2 = 0$	0.4260
$M_2 = 1$	0.4561
$M_2 = 2$	0.1106
$M_2 = 3$	0.0071
$M_2 = 4$	0.00008
$M_2 = 5$	0
<i>Tercer nivel intermedio</i>	
$M_3 = 0$	0.4065
$M_3 = 1$	0.4638
$M_3 = 2$	0.1212
$M_3 = 3$	0.0082
$M_3 = 4$	0.00003
$M_3 = 5$	0

Tabla A.1: Valores numéricos de probabilidad de existencia de rutas disjuntas con origen en cada uno de los tres niveles intermedios obtenidos tras la aplicación de los Algoritmos 1, 2, 3 y 4.

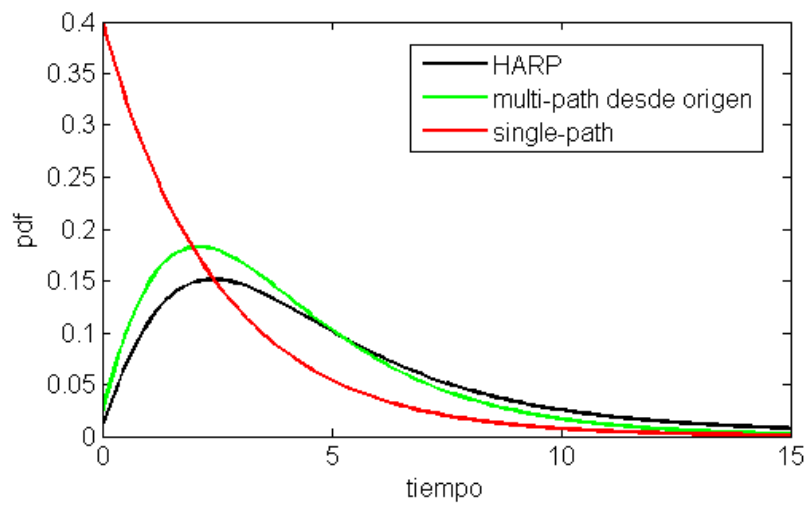


Figura A.2: Representación gráfica de las pdf de los tiempos de vida de las rutas entre S y D para el ejemplo de topología de este apéndice, calculadas tras aplicar los algoritmos de cálculo de probabilidad de rutas disjuntas propuestos en esta Tesis.

Bibliografía

- [1] Ambient intelligence. [Online]. Available: <http://www.ambientintelligence.org>
- [2] Castalia: a simulator for wsn. [Online]. Available: <http://castalia.npc.nicta.com.au/>
- [3] Chipcon cc2420 2.4 ghz ieee 802.15.4 / zigbee-ready rf transceiver, datasheet 2010. [Online]. Available: <http://focus.ti.com/lit/ds/symlink/cc2420.pdf>
- [4] Crossbow sensor devices. [Online]. Available: www.xbow.com
- [5] Entorno de desarrollo java netbeans. [Online]. Available: <http://netbeans.org/>
- [6] Herramienta matlab. [Online]. Available: <http://www.mathworks.com/products/matlab/>
- [7] Jist/swans: Java in simulation time/scalable wireless ad hoc network simulator. [Online]. Available: <http://jist.ece.cornell.edu/>
- [8] Lenguaje java. [Online]. Available: <http://java.about.com/>
- [9] Opnet. [Online]. Available: <http://www.opnet.com/>
- [10] Scalable network technologies. [Online]. Available: <http://www.scalable-networks.com/>
- [11] Tossim. [Online]. Available: <http://docs.tinyos.net/index.php/TOSSIM>
- [12] Visualsense: Visual modeling for wireless and sensor network systems. [Online]. Available: <http://ptolemy.eecs.berkeley.edu/papers/05/visualsense/>
- [13] Wsnet: wireless network event-driven simulator. [Online]. Available: <http://wsnet.gforge.inria.fr/index.html>
- [14] (2008) Healthcare. [Online]. Available: <http://onworld.com/research/healthcare/vip/>

- [15] (2010) Proyecto guiade. [Online]. Available: <http://www.proyectoguiade.com/>
- [16] (October 2009) Smart buildings. [Online]. Available: <http://onworld.com/research/smbldgs/vip/>
- [17] M. Abbas and T. Abbasi, “An analytical framework for disjoint multipath routing in mobile ad hoc networks,” in *IFIP International Conference on Wireless and Optical Communications Networks*, 2006.
- [18] M. Abbas and B. N. Jain, “Path diminution in disjoint multipath routing for mobile ad hoc networks,” in *Proceedings of 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 1, Sep. 2004, pp. 130–134.
- [19] M. Abbas and B. Jain, “An analytical framework for route failure time of multiple node-disjoint paths in mobile ad hoc networks,” in *Proceedings of 12th IEEE International Telecommunications Network Strategy and Planning Symposium (NETWORKS)*, New Delhi, India, Nov. 2006, pp. 1–6.
- [20] J. Abdulai, M. Ould-Khaoua, and L. Mackenzie, “Neighbour coverage: A dynamic probabilistic route discovery for mobile ad hoc networks,” in *International Symposium on Performance Evaluation of Computer and Telecommunication Systems SPECTS*, 2008.
- [21] F. Adelstein, S. Gupta, G. Richard, and L. Schwievert, *Fundamentals of Mobile and Pervasive Computing*. McGraw-Hill Professional Engineering, 2005.
- [22] I. Akyldiz and I. Kasimoglu, “Wireless sensor and actor networks: Research challenges,” *Ad Hoc Networks (Elsevier)*, 2004.
- [23] J. Al-Karaki and A. Kamal, “Routing techniques in wireless sensor networks: a survey,” *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–28, Dec. 2004.
- [24] S. Alizadeh and A. Ghaffari, “An energy-efficient hierarchical clustering protocol in wireless sensor networks,” in *3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT)*, 2010.
- [25] J. Altes and X. Hesselbach, *Análisis de redes y sistemas de comunicaciones*. Ediciones UPC, 1ª ed., pp. 86–88, 2002.
- [26] F. and A. Chen and S. Liu and L. Zhang, “A scalable solution to minimum cost forwarding in large sensor networks,” in *Proceedings of the tenth International Conference on Computer Communications and Networks (ICCCN)*, 2001.

- [27] Y. and D. Estrin and R. Govindan, “Geographical and energy-aware routing: A recursive data dissemination protocol for wireless sensor networks,” in *UCLA Computer Science Department Technical Report*, 2001.
- [28] S. and G. Barrenechea, “Constrained random walks on random graphs: routing algorithms for large scale wsns,” in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pp. 12–21, USA,, 2002.
- [29] T. and H. Ishibuchi, “Performance evaluation of genetic algorithms for flows-hop scheduling problems,” in *Proceedings of the 1st IEEE Conf. Evolutionary Computation*, 1994.
- [30] Y. and J. Heidemann and D. Estrin, “Geography-informed energy conservation for ad-hoc routing,” in *Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking*, 2001.
- [31] I. Aron and K. Gupta, “Analytical comparison of local and end-to-end error recovery in reactive routing protocols for mobile ad hoc networks,” in *Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, 2000, pp. 69–76.
- [32] F. Bai and N. Sadagopan, “Modeling path duration distributions in manets and their impact on reactive routing protocols,” *IEEE Journal On Selected Areas In Ccommunications*, vol. 22, no. 7, pp. 1357– 1372, Sep. 2004.
- [33] M. Bani-Yassein, M. Ould-Khaoua, and S. Papanastasiou, “Performance analysis of adjusted probabilistic broadcasting in mobile ad hoc networks,” *International Journal of Wireless Information Networks*, pp. 1–14, 2006.
- [34] R. Barr, Z. J. Haas, and R. van Renesse, “Jist: An efficient approach to simulation using virtual machines,” in *Software-Practice and Experience*. John Wiley & Sons, Ltd., 2004.
- [35] A. Bhattacharyya, A. Seetharam, S. Dasgupta, D. Das, S. Mitra, and M. Nas- kar, “Lifetime enhancement in wireless sensor networks through selective data handover,” in *Third International Conference on Wireless Communi- cation and Sensor Networks (WCSN)*, 2007.
- [36] Q. Bian, Y. Zhang, and Y. Zhao, “Research on clustering routing algo- rithms in wireless sensor networks,” in *International Conference on Inte- lligent Computation Technology and Automation*, 2010.
- [37] R. Billinton and R. N. Allan, *Reliability Evaluation of Engineering Systems*. Plenum Press, New York, Second Edition, 1992.

- [38] A. Boukerche, M. Z. Ahmad, B. Turgut, and D. Turgut, *A Taxonomy of Routing Protocols in Sensor Networks*. Algorithms and Protocols for Wireless Sensor Networks. Wiley-IEEE Press., 2008.
- [39] D. Braginsky and D. Estrin, “Rumor routing algorithm for sensor networks,” in *Proceedings of the First Workshop on Sensor Networks and Applications (WSNA)*, 2002.
- [40] O. Buyanjargal and Y. Kwon, “Adaptive and energy efficient clustering algorithm for event-driven application in wireless sensor networks (aeec),” *Journal of Networks*, vol. 5, no. 8, pp. 904–911, Aug. 2010.
- [41] D. Cai and D. Zhu, “Research and simulation of energy efficient protocol for wireless sensor network,” in *2nd International Conference on Computer Engineering and Technology (ICCET)*, 2010.
- [42] J. Capella, A. Bonastre, R. Ors, and S. Climent, “A new energy-efficient, scalable and robust architecture for wireless sensor networks,” in *3rd International Conference on New Technologies, Mobility and Security (NTMS)*, 2009.
- [43] J. Chen, R. Lin, and Y. Sun, “Development of simulation environment wnsim for wireless sensor networks,” in *Chinese Journal of Sensors and Actuators*, 19(2), pp. 1–6, 2005. [Online]. Available: <http://www.winsim.com/>
- [44] W. Choi, S. K. Das, J. Cao, and A. K. Datta, “Randomized dynamic route maintenance for adaptive routing in multihop mobile ad hoc networks,” *Journal of Parallel and Distributed Computing*, vol. 65, pp. 107–123, 2005.
- [45] I. Cidon, R. Rom, and Y. Shavitt, “Analysis of multi-path routing,” *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 885–896, 1999.
- [46] E. Commission, *Doing more with less: Green Paper on energy efficiency*. Office for official publications of the European Communities, 2005.
- [47] X. Cui, “Research and improvement of leach protocol in wireless sensor networks,” in *IEEE International Symposium on Microwave, Antenna, Propagation, and EMC Technologies For Wireless Communications*, 2007.
- [48] J. Elson and D. Estrin, *Sensor networks: a bridge to the physical world*. Kluwer Academic Publishers. Norwell, MA, USA, pp. 3-20, 2004.
- [49] R. Ennaji and M. Boulmalf, “Routing in wireless sensor networks,” in *International Conference on Multimedia Computing and Systems (ICMCS)*, 2009.

- [50] M. O. Farooq, A. B. Dogar, and G. A. Shah, “Mr-leach: Multi-hop routing with low energy adaptive clustering hierarchy,” in *Fourth International Conference on Sensor Technologies and Applications*, 2010.
- [51] J. Faruque and A. Helmy, “Tabs: Link loss tolerant data routing protocol for multi-hop wireless sensor networks,” in *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, 2010.
- [52] F. Cavallo, M. Aquilano, L. Odetti, M. Arvati, and M. C. Carrozza, “A first step toward a pervasive and smart zigbee sensor system for assistance and rehabilitation,” in *EEE 11th International Conference on Rehabilitation Robotics. Japan*, 2009.
- [53] R. Fujimoto, “Parallel discrete event simulation,” in *Proceedings of the 21st conference on Winter simulation (WSC)*, 1989.
- [54] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, “Highly-resilient, energy-efficient multipath routing in wireless sensor networks,” *ACM Mobile Computing and Communications Review*, vol. 1, no. 2, p. 1–13, 2002.
- [55] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker, *Complex Behavior at Scale: An Experimental Study of Low-Power Wireless Sensor Networks*, UCLA, LA, UCLA CS Technical Report UCLA/CSD-TR 02–0013 Std., 2002.
- [56] D. Gay, P. Levis, R. von Behren, M. Welsh, E. Brewer, and D. Culler, “The nesc language: A holistic approach to networked embedded systems,” in *Proceedings of Programming Language Design and Implementation (PLDI)*, 2003.
- [57] M. Gerharz, C. de Waal, M. Frank, and P. Martini, “Link stability in mobile wireless ad hoc networks,” in *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks (LCN)*, 2002.
- [58] N. W. Group, *RFC 2501. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*, The Internet Society. 1999 Std., Rev. 1999, 1998.
- [59] L. Gu and J. A. Stankovic, “Radio-triggered wake-up for wireless sensor networks,” *ACM Real Time Systems*, vol. 29, no. 2, 2005.
- [60] J. Ha, H. Park, S. Choi, and W. Kwo, “Ehrp: Enhanced hierarchical routing protocol for zigbee mesh networks,” *IEEE Communication Letters*, pp. 1028–1031, 2007.
- [61] D. Hall, *Mathematical Techniques in Multisensor Data Fusion*. Artech House, Boston, MA, 1992.

- [62] D.-M. Han and J.-H. Lim, "Design and implementation of smart home energy management systems based on zigbee," *IEEE Transactions on Consumer Electronics*, Aug. 2010.
- [63] S. Hedetniemi and A. Liestman, "A survey of gossiping and broadcasting in communication networks," *Networks*, vol. 18, no. 4, 1988.
- [64] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy efficient communication protocol for wireless microsensor networks," in *Proc. of the Hawaii International Conference on System Sciences*, 2000.
- [65] W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of 5th ACM/IEEE Mobicom Conference*, 1999.
- [66] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [67] R. Hekmat and P. V. Miegheem, *Connectivity in Wireless Ad-hoc Networks with a Log-normal Radio Model*. Mobile Networks and Applications, vol. 11, pags. 351–360, 2006.
- [68] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, *System Architecture Directions for Networked Sensors*. Architectural Support for Programming Languages and Operating Systems, pp. 93–104, 2000. [Online]. Available: <http://webs.cs.berkeley.edu>
- [69] J. W. Hooper, "Strategy-related characteristics of discrete-event languages and models," in *Simulation, April 1986, vol. 46, num. 4, pp. 153-159*, 1986.
- [70] W.-W. Huang, M. Yu, L.-Q. Xiong, and J. Wen, "Energy-efficient hierarchical routing protocol for wireless sensor networks," in *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, 2008.
- [71] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking (MobiCom)*, 2000.
- [72] Y. Jia, L. Zhao, and B. Ma, "Clustering-based expanding-ring routing protocol applied in wireless sensor networks," in *Proceedings of 6th International Conference on ITS Telecommunications*, 2006.
- [73] P. Jiang, Y. Wen, J. Wang, X. Shen, and A. Xue, "A study of routing protocols in wireless sensor networks," in *Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China*, 2006.

- [74] G. JingMin, Z. Zhiliang, and Gaoyang, “Research and improvement of routing protocol for wireless sensor network,” in *International Conference on Computer, Mechatronics, Control and Electronic Engineering (CMCE)*, 2010.
- [75] J. Kim, Q. Zhang, and D. Agrawal, “Probabilistic broadcasting based on coverage area and neighbor confirmation in mobile ad hoc networks,” in *Proceedings of IEEE Workshop on Wireless Ad Hoc and Sensor Networks*, 2004.
- [76] K.-H. Kim and K. G. Shin, “On accurate and asymmetry-aware measurement of link quality in wireless mesh networks,” *IEEE/ACM Transactions on Networking*, vol. 17, no. 4, pp. 1172–1185, Aug. 2009.
- [77] L. Klein, *Sensor and Data Fusion Concepts and Applications*. SPIE Optical Engr Press, WA, 1993.
- [78] L. Kleinrock, *Queueing Systems*. New York Wiley, vol.2, 1976.
- [79] D. Krahl and J. S. Lamperti, “A message-based discrete event simulation architecture,” in *Proceedings of the 29th conference on Winter simulation (WSC)*, 1997.
- [80] K. Kuladinithi, M. Becker, C. Gorg, and S. Das, “Radio disjoint multipath routing in manet,” in *Center of Excellence in Wireless and Information Technology 2005 Conference (CEWIT 2005)*, Stony Brook, 2005.
- [81] R. J. La and Y. Han, “Distribution of path durations in mobile ad hoc networks and path selection,” *IEEE/ACM Transactions On Networking*, vol. 15, no. 5, pp. 993–1006, Oct. 2007.
- [82] S. Lee, J. Lee, H. Sin, S. Yoo, Y. Lee, and S. Kim, “An energy efficient distributed unequal clustering for wireless sensor networks,” in *Proc. of Int. Conf. on Communication Software and Networks, Macau, China*, 2009.
- [83] C. Li, M. Ye, G. Chen, and J. Wu, “An energy-efficient unequal clustering mechanism for wireless sensor networks,” in *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, 2005.
- [84] H. Li, X. Shunjie, L. Shurong, Z. Weixia, and Z. Zheng, “Novel method for optimal number of cluster heads in leach,” in *(International Conference on Information Engineering (WASE))*, 2009.
- [85] X. Li, B. Hu, Y. Jin, and S. Chen, “An analytical evaluation on the performance of heterogeneous ad hoc networks,” in *International Conference on Wireless Communications, Networking and Mobile Computing*, 2005.

- [86] S. Lindsey and C. Raghavendra, “Pegasis: Power-efficient gathering in sensor information systems,” in *IEEE Aerospace Conference Proceedings*, 2002.
- [87] T. Linjun and W. Hanxing, “Performance analysis of route maintenance of manet,” in *IET Conference on Wireless, Mobile and Sensor Networks 2007 (CCWMSN07)*, 2007, p. 133–137.
- [88] J. Liu, L. Perrone, D. Nicol, M. Liljenstam, D. Pearson, and C. Elliott, “Simulator for wireless ad hoc networks,” in *Proceedings of European Simulation Interoperability Workshop (EURO-SIW)*, 2001.
- [89] M. Liu, Y. Zheng, J. Cao, G. Chen, L. Chen, and H. Gong, “An energy-aware protocol for data gathering applications in wireless sensor networks,” in *Proceedings of ICC*, 2007.
- [90] R. P. Liu, Z. Rosberg, I. B. Collings, C. Wilson, A. Y. Dong, and S. Jha, “Energy efficient reliable data collection in wireless sensor networks with asymmetric links,” *International Journal of Wireless Information Networks*, vol. 16, no. 3, pp. 131–141, Aug. 2009.
- [91] S. Madden, M. Franklin, J. M. Hellerstein, and W. Hong, “The design of an acquisitional query processor for sensor networks,” in *ACM SIGMOD*, 2003.
- [92] A. Manjeshwar and D. P. Agarwal, “Teen: a routing protocol for enhanced efficiency in wireless sensor networks,” in *Proc. International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, IEEE Computer Society San Francisco, USA*, 2001.
- [93] A. Manjeshwar and D. Agarwal, “Apten: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks,” in *Proceedings International Parallel and Distributed Processing Symposium, IPDPS*, 2002.
- [94] S. Mao, D. Bushmitch, S. Narayanan, and S. Panwar, “Mrtp: A multi-flow realtime transport protocol for ad hoc networks,” in *Proceedings of IEEE VTC 2003*, 2003, p. 1736–1745.
- [95] A. McDonald and T. Znabi, “A path availability model for wireless ad hoc networks,” in *Proceedings of IEEE WCNC, New Orlean, LA*, 1999.
- [96] P. D. Mil, E. D. Poorter, B. Latré, and I. Moerman, “Definition and evaluation of local path recovery mechanisms in wireless sensor and actuator networks,” in *Third International Conference on Sensor Technologies and Applications*, 2009.

- [97] S. Mueller, R. Tsang, and D. Ghosal, "Multipath routing in mobile ad hoc networks: Issues and challenges," in *International Conference on Performance Tools and Applications to Networked Systems*, 2004.
- [98] A. Nasipuri, R. Castañeda, and S. Das, *Performance of Multipath Routing for On-Demand Protocols in Mobile Ad Hoc networks*. Kluwer Academic Publishers, Mobile Networks and Applications, 6(4), pp. 339-349, 2001.
- [99] T. Nguyen and A. Zakhor, "Path diversity with forward error correction (pdf) system for packet switched networks," in *Proceedings of IEEE INFOCOM*, 2003.
- [100] (2010) The ns-3 network simulator. [Online]. Available: <http://www.nsnam.org/>
- [101] S. Park, A. Savvides, and M. Srivastava, "Sensorsim: A simulation framework for sensor networks," in *Proceedings of MSWiM*, 2000.
- [102] Z. Peng and X. Li, "The improvement and simulation of leach protocol for wsns," in *IEEE International Conference on Software Engineering and Service Sciences (ICSESS)*, 2010.
- [103] C. Perkins, E. Belding-Royer, and S. Das, *Ad hoc On-demand Distance-Vector (AODV) routing*, Internet-Draft, IETF. Std., 2002.
- [104] P. Pham and S. Perreau, "Performance analysis of reactive shortest path and multi-path routing mechanism with load balance," in *IEEE INFOCOM*, 2003.
- [105] G. Pottie and W. Kaiser, "Wireless integrated network sensors," *Communications of the ACM*, vol. 43, no. 5, pp. 51-58, 2000.
- [106] D. M. Qiangfeng Jiang, "Routing protocols for sensor networks," Tech. Rep., 2004.
- [107] J. Rathi and G. Rajendran, "An enhanced leach protocol using fuzzy logic for wireless sensor networks," in *International Journal of Computer Science and Information Security (IJCSIS)*, Vol. 8, No. 7, 2010.
- [108] L. Reddy and S. Raghavan, *SMORT: Scalable multipath on-demand routing for mobile ad hoc networks*. Elsevier ScienceDirect, Ad Hoc Networks, 5, pp. 162-188, 2007.
- [109] R. D. Ringeisen and F. S. Roberts, *Applications of Discrete Mathematics*. Siam, 1988.

- [110] R. Rönngren and M. Liljenstam, “On event ordering in parallel discrete event simulation,” in *13th Workshop on Parallel and Distributed Simulation (PADS’99)*, 1999.
- [111] V. Rodoplu and T. Ming, “Minimum energy mobile wireless networks,” *IEEE Journal of Selected Areas in Communications*, vol. 17, pp. 1333–1344, 1999.
- [112] A. M. Ross, *Useful Bounds on the Expected Maximum of Correlated Normal Variables*, 2003.
- [113] S. Ross, *Introduction to Probability Models*. Academic Press, Ninth Edition, 2007.
- [114] E. Ruiz-Ibarra, L. Villaseñor-Gonzalez, and R. Aquino, “Design issues and considerations for coordination mechanisms in wireless sensor and actuator networks,” in *Fourth Congress of Electronics, Robotics and Automotive Mechanics*, 2007.
- [115] P. Samar and S. Wicker, “Characterizing the communication links of a node in a mobile ad hoc network,” in *International Conference on Communications. Vol. 7, pp. 4092-4096.*, 2004.
- [116] M. Saraogi, “Security in wireless sensor networks,” in *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2004.
- [117] T. J. Schriber, “Inside discrete-event simulation software: How it works and why it matters,” in *Proceedings of the 2006 Winter Simulation Conference*, 2006.
- [118] C. Schurgers and M. Srivastava, “Energy efficient routing in wireless sensor networks,” in *MILCOM Proceedings on Communications for Network-Centric Operations: Creating the Information Force*, 2001.
- [119] S. Selvakennedy and S. Sinnappan, “An adaptive data dissemination strategy for wireless sensor network,” *International Journal of Distributed Sensor Networks*, vol. 3, p. 23–40, 2007.
- [120] R. Shah and J. Rabaey, “Energy aware routing for low energy ad hoc sensor networks,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, 2002.
- [121] V. Shah and S. Krishnamurthy, “Handling asymmetry in power heterogeneous ad hoc networks: A cross layer approach,” in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICSCS)*, 2005.

- [122] D. R. Shier, *Network Reliability and Algebraic Structures*. Oxford Science Publications, 1991.
- [123] G. Smaragdakis, I. Matta, and A. Bestavros, “Sep: A stable election protocol for clustered heterogeneous wireless sensor networks,” in *Proceedings of 2nd International Workshop on Sensor and Actor Network Protocols and Applications (SANPA)*, 2004, pp. 251–261.
- [124] A. Sobeih, J. Hou, L. Kung, N. Li, H. Zhang, W. Chen, H. Tyan, and H. Lim, “J-sim: a simulation and emulation environment for wireless sensor networks,” in *IEEE Wireless Communications*, 13(4), pp 104-119, 2006. [Online]. Available: <http://jvasim.codehaus.org/>
- [125] W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Communications Magazine*, vol. 50, no. 8, pp. 102–114, 2002.
- [126] L. Subramanian and R. H. Katz, “An architecture for building self configurable systems,” in *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking and computing*, pp. 63-73, 2001.
- [127] L. Tao, Z. Qing-Xin, and Z. Luqiao, “An improvement for leach algorithm in wireless sensor network,” in *5th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2010.
- [128] F. Tashtarian, A. Haghghat, M. T. Honary, and H. Shokrzade, “An energy-efficient hierarchical clustering protocols in wireless sensor networks,” in *IEEE/ACS International Conference on Computer Systems and Applications, (AICCSA)*, 2008.
- [129] F. Tashtarian, M. T. Honary, A. Haghghat, and J. Chitizadeh, “A new energy-efficient level-based clustering algorithm for wireless sensor networks,” in *15th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2007.
- [130] Y. Tia, Y. Wang, and S.-F. Zhang, “A novel cluster-chain channel adaptive routing protocol in wireless sensor networks,” in *International Conference on Wireless Communications, Networking and Mobile Computing (WiCom)*, 2007.
- [131] Y. Tseng, S. Ni, and E. Shih, “Adaptive approaches to relieving broadcast storms in a wireless multihop mobile ad hoc networks,” *IEEE Transactions on Computers*, vol. 52, pp. 545–557, 2003.
- [132] Y.-C. Tseng, Y.-F. Li, and Y.-C. Chang, “On route lifetime in multihop mobile ad hoc networks,” *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, 2003.

- [133] A. Tsirigos and Z. Haas, "Analysis of multipath routing - part i: The effect on the packet delivery ratio," *IEEE Transactions on Wireless Communications*, vol. 3, no. 1, p. 138–146, 2004.
- [134] A. Varga, "The omnet++ discrete event simulation system," in *Proceedings of the European Simulation Multiconference (ESM)*, 2001.
- [135] J. J. Vinagre, "Teoría del encaminamiento en redes ad hoc inalámbricas," in *Tesis Doctoral. Universidad Carlos III, Madrid*, 2007.
- [136] (2009) Network simulator - ns (versión 2.34). VINT project, SAMAN, CONSER. [Online]. Available: http://nslam.isi.edu/nslam/index.php/Main_Page
- [137] J. Wan, D. Yuan, and X. Xu, "A review of routing protocols in wireless sensor networks," in *4th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, 2008.
- [138] L. Wang, J. Liu, and W. Wang, "An improvement and simulation of leach protocol for wireless sensor network," in *First International Conference on Pervasive Computing, Signal Processing and Applications*, 2010.
- [139] W. Weichao, D. Fei, and X. Qijian, "An improvement of leach routing protocol based on trust for wireless sensor networks," in *5th International Conference on Wireless Communications, Networking and Mobile Computing, (WICOM)*, 2009.
- [140] G. C. W.G. Cochran, *Experimental Designs*. John Wiley and Sons Inc., Segunda Edición, 1992.
- [141] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking computing, MOBIHOC*, 2002.
- [142] J. Xibei, Z. Huazhong, and Z. Jingchen, "Research of data aggregation routing protocol in wsn data-related applications," in *3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT)*, 2010.
- [143] Q. Xuegong and C. Yan, "A control algorithm based on double cluster-head for heterogeneous wireless sensor network," in *2nd International Conference on Industrial and Information Systems*, 2010.
- [144] H. Yang and B. Sikdar, "Performance analysis of polling based tdma mac protocols with sleep and wakeup cycles," in *IEEE International Conference on Communications (ICC'07)*, 2007.

- [145] J. Yang and D. Zhang, “An energy balancing unequal clustering protocol for wireless sensor networks,” *Journal of Information Technologies*, vol. 8, no. 1, pp. 57–63, 2009.
- [146] P. Yang and B. Huang, “Multi-path routing protocol for mobile ad hoc network,” in *International Conference on Computer Science and Software Engineering*, 2008.
- [147] J. Ying-ying and Z. Jian-wu, “The improvement of routing protocols in wireless sensor networks,” in *IET Conference on Wireless, Mobile and Sensor Networks, (CCWMSN07)*, 2007.
- [148] H. Yu, L. Wei, and K. Zhenhua, “Study on energy efficient hierarchical routing protocols of wireless sensor network,” in *International Conference on Information Engineering (WASE)*, 2009.
- [149] J. Yu, W. Liu, J. Song, and B. Cao, “Eemr: An energy-efficient multi-hop routing protocol for wireless sensor networks,” in *IEEE/ACS International Conference on Computer Systems and Applications (AICCSA)*, 2008.
- [150] X. Zeng, R. Bagrodia, and M. Gerla, “Glomosim: a library for the parallel network simulation environment,” in *Proceedings of the 12th Workshop on Parallel and Distributed Systems*, 1998.
- [151] Z. Zhen-chuan and Z. Xin-xiu, “Research of improved clustering routing algorithm based on load balance in wireless sensor networks,” in *IET International Communication Conference on Wireless Mobile and Computing (CCWMC), Shanghai, China*, 2009.
- [152] Home automation. Zigbee Alliance. [Online]. Available: <http://www.zigbee.org/Markets/ZigBeeHomeAutomation/Features.asp>
- [153] (2007) Zigbee specification q42007. ZigBee Alliance. [Online]. Available: <http://www.zigbee.org/Products/DownloadZigBeeTechnicalDocuments.aspx>