



**Escola Politècnica Superior
de Castelldefels**

UNIVERSITAT POLITÈCNICA DE CATALUNYA

PROYECTO FINAL DE CARRERA

TÍTULO DEL PFC: Evaluación de métricas de encaminamiento basadas en LQI para un protocolo de encaminamiento en redes IEEE802.15.4.

TITULACIÓN: Ingeniería de Telecomunicaciones (segundo ciclo)

**AUTORES: Omar García Bermejo
Tomás García Sotelo**

DIRECTOR: Carles Gómez Montenegro

FECHA: 31 de Octubre de 2008

Título: Evaluación de métricas de encaminamiento basadas en LQI para un protocolo de encaminamiento en redes IEEE802.15.4.

Autores: Omar García Bermejo
Tomás García Sotelo

Director: Carles Gómez Montenegro

Fecha: 31 de Octubre de 2008

Resumen

Uno de los campos de investigación más activos actualmente es el de las redes inalámbricas ad-hoc. De origen militar, durante más de 30 años se ha trabajado en este tipo de redes, pero ha sido en los últimos años cuando el auge de la tecnología inalámbrica y su bajo coste han provocado el mayor avance en el desarrollo de la investigación en este campo.

Las redes ad-hoc no requieren infraestructura ni administración centralizada. Los nodos hacen a la vez las funciones de terminal y router, ya que todos deberán ser capaces de retransmitir la información de la que no sean destinatarios hacia otros nodos hasta que se alcance el destino.

El desarrollo de protocolos y arquitecturas para el diseño de redes ad-hoc, y en concreto de redes de sensores, ha ocurrido paralelamente a la aparición de nuevos estándares de comunicación inalámbricos como el IEEE 802.15.4, orientado a redes inalámbricas de baja velocidad, bajo coste y bajo consumo de potencia.

Una de las topologías que define IEEE 802.15.4 es la denominada topología mesh que requiere un protocolo de encaminamiento para la transmisión de datos de un origen a un destino.

Pese a que existen estudios que caracterizan la calidad de un enlace IEEE 802.15.4 en base a parámetros proporcionados por las capas físicas y MAC, existe escasa literatura acerca de su aplicación al encaminamiento en redes IEEE 802.15.4 con topología mesh. Este documento presenta un estudio de métricas de encaminamiento alternativas a la métrica clásica denominada "número de saltos" para el protocolo de encaminamiento nst-AODV, diseñado para redes de sensores IEEE 802.15.4 con topología mesh. En particular, se han implementado y evaluado tres métricas: las métricas PDR, ZigBee y ETX que tienen en cuenta la calidad de los enlaces. Para ello, se ha empleado el parámetro LQI, proporcionado por el estándar 802.15.4.

Title: Routing metrics evaluation based on LQI for a routing protocol in IEEE 802.15.4 networks.

Authors: Omar García Bermejo
Tomás García Sotelo

Director: Carles Gómez Montenegro

Date: October, 31th 2008

Overview

One of the most active fields of research at present is the wireless ad-hoc networks. Of military origin, for more than 30 years, investigators have been working in this type of networks, but has been in the last years when the proliferation of the wireless technology and its low cost have provoked the biggest advance in the development of the research in this field.

Ad-hoc networks do not require any type of fixed infrastructure and centralized administration neither, where the nodes do at the same time the functions of terminal and router, since they all will have to be capable of broadcasting the information of which they are not the destination towards other nodes until the destination is reached.

The development of protocols and architectures for ad-hoc networks design, and specifically of sensor networks have been developed closely to the new wireless communication standards like IEEE 802.15.4, oriented to low rate, low cost and low power wireless networks.

One of the topologies defined by IEEE 802.15.4 is the named mesh topology that requires a routing protocol for data transmission.

Although there are studies that remark the quality of a IEEE 802.15.4 link, based on parameters which are provided by the physical and MAC layers, there is few literatura related to the application of the routing of IEEE 802.15.4 networks with mesh topology. This document presents a study of alternative routing metrics to the classical metrics named "hops" for the nst-AODV routing protocol, designed to sensors IEEE 802.15.4 networks with mesh topology. Particularly, it has been implemented three metrics: PDR, ZigBee and ETX that take the quality of the links into account. For that, the parameter LQI, provided by the standard 802.15.4, has been used.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1. TECNOLOGÍA RADIO IEEE 802.15.4	3
1.1. Introducción	3
1.2. Dispositivos y topologías existentes en 802.15.4.....	4
1.3. Capa Física.....	6
1.4. Capa de Control de Acceso al Medio	8
1.4.1. Modo de Funcionamiento	8
1.5. Mecanismos de robustez del 802.15.4	9
1.6. Implementación utilizada en el proyecto	9
CAPÍTULO 2. REDES INALÁMBRICAS MULTISALTO.....	10
2.1. Introducción	10
2.2. Definición de conceptos y clasificación de redes	10
2.3. Redes MANET	10
2.4. Redes LoWPAN.....	11
2.4.1. Características y problemáticas de las redes LoWPAN.....	11
2.4.2. Iniciativas de estandarización.....	12
CAPÍTULO 3. ENCAMINAMIENTO EN REDES INALÁMBRICAS MULTISALTO	13
3.1. Introducción.....	13
3.2. Encaminamiento en redes MANET	13
3.2.1. Clasificación protocolos de encaminamiento	13
3.2.1.1. Protocolos proactivos.....	13
3.2.1.2. Protocolos Reactivos.....	14
3.2.1.3. Híbridos	15
3.2.1.4. Multicast	15
3.3. Protocolo AODV, TinyAODV e implementación nst-AODV.....	15
3.3.1. AODV.....	15
3.3.2. TinyAODV	16
3.3.3. nst-AODV.....	16
3.4. Encaminamiento en redes LoWPAN	17
3.4.1. Propuestas estándar	17
3.5. Estado del arte de métricas de encaminamiento en redes multisalto.....	18
3.5.1. Métricas evaluadas.....	18

CAPÍTULO 4. TECNOLOGIAS UTILIZADAS	21
4.1. Introducción	21
4.2. Plataforma Hardware.....	21
4.2.1. Telosb	22
4.3. Plataforma Software	24
4.3.1. TinyOS.....	24
4.3.2. NesC.....	26
4.3.3. Protocolo nst-AODV	27
4.3.4. Interface USBShark.....	34
CAPÍTULO 5. ESTUDIO DEL PARAMETRO LQI.....	36
5.1. Cálculo de LQI y RSSI con el chipcon CC2420.	36
5.2. Escenario de pruebas	36
5.3. Estudio del impacto de la orientación y la distancia de los motes.....	38
5.3.1. Caracterización de una fila con los motes sin orientación fija.....	38
5.3.2. Caracterización de la antena de los motes Telosb	39
5.3.2.1. Plano horizontal.....	39
5.3.2.2. Plano vertical.....	41
5.3.3. Caracterización de una fila con orientación fija de los motes.	43
5.4. Estudio del impacto de interferencias en los motes	45
5.5. Relación LQI-LDR	47
CAPÍTULO 6. EVALUACIÓN DE LAS MÉTRICAS DE ENCAMINAMIENTO	49
6.1. Implementación de las distintas métricas.	49
6.1.1. Métrica HOPS.....	49
6.1.2. Métrica PDR.	49
6.1.3. Métrica ZigBee.	52
6.1.4. Métrica ETX.....	53
6.1.5. Otras implementaciones.....	55
6.2. Pruebas y resultados.	56
6.2.1. Escenario de las pruebas.....	56
6.2.2. Resultados de las pruebas.	58
6.2.2.1. Resultados generales de las pruebas con reconocimientos.....	58
6.2.2.2. Resultados específicos caminos largos con reconocimientos.....	61
6.2.2.3. Resultados específicos caminos cortos con reconocimientos.....	65
6.2.2.4. Resultados generales de las pruebas sin reconocimientos.....	69
6.2.2.5. Resultados específicos caminos largos sin reconocimientos.....	70
6.2.2.6. Resultados específicos caminos cortos sin reconocimientos.....	75
CAPÍTULO 7. CONCLUSIONES Y LÍNEAS FUTURAS	79
7.1. Conclusiones	79
7.2. Líneas futuras	80

IMPLICACIONES MEDIOAMBIENTALES	81
BIBLIOGRAFÍA	82
ANEXO A. ESTRUCTURAS DE TRAMAS DEL IEEE 802.15.4	87
A.1. Campos comunes en todas las tramas.....	89
ANEXO B. CONFIGURACIÓN PARÁMETROS CHIP CC2420	90
B.1. Archivo de configuración AM.h	90
B.2. Archivo de configuración CC2420Const.h	91
ANEXO C. AD HOC ON-DEMAND VECTOR ROUTING PROTOCOL (RFC 3561)	92
C.1. Introducción	92
C.2. Proceso de descubrimiento de rutas	92
C.3. Mantenimiento de rutas	93
ANEXO D. TINYOS	95
D.1. Instalación TinyOs para Windows	95
D.2. Directorio TinyOS	96
D.3. Compilación de aplicaciones en TinyOS	98

INTRODUCCIÓN

Uno de los campos de investigación más activos actualmente es el de las redes inalámbricas ad-hoc. De origen militar, durante más de 30 años se ha venido trabajando en este tipo de redes, pero ha sido en los últimos años cuando el auge de la tecnología inalámbrica y su bajo coste han provocado el mayor avance en el desarrollo de la investigación en este campo.

Las redes ad-hoc no requieren ningún tipo de infraestructura fija ni administración centralizada, en donde los nodos hacen a la vez las funciones de terminal y router, ya que todos deberán ser capaces de retransmitir la información de la que no sean destinatarios hacia otros nodos hasta que se alcance el destino.

La principal ventaja de estas redes es la flexibilidad, pero hay que tener en cuenta sus características y problemas, puesto que en redes inalámbricas nos encontramos con enlaces de calidades variables, nodos con movilidad... Así pues los protocolos usados deberían ser completamente adaptativos para ser adecuados para el entorno inalámbrico. Por este motivo, el desarrollo de protocolos y arquitecturas para el diseño de redes de ad-hoc, y en concreto de redes de sensores, se ha convertido en un campo de investigación muy importante en los últimos años. A su vez, se han ido desarrollando paralelamente nuevos estándares de comunicación inalámbricos como el IEEE 802.15.4, orientado a redes inalámbricas de baja velocidad, bajo coste y bajo consumo de potencia.

Una de las topologías que define IEEE 802.15.4 es la denominada topología mesh que requiere un protocolo de encaminamiento para la transmisión de datos de un origen a un destino. Existen estudios que sugieren la optimización del rendimiento del protocolo de encaminamiento empleando métricas alternativas al clásico número de saltos, ya que dadas las características del medio radio no tiene por qué ser la ideal en estos entornos.

El objetivo de este Proyecto Final de Carrera, es la optimización del protocolo de encaminamiento nst-AODV (implementación propia de la UPC) para la implementación, estudio y evaluación de las métricas de encaminamiento número de saltos (HOPS), Path delivery ratio (PDR), ZigBee y Expected Transmission Count (ETX) en redes de sensores IEEE 802.15.4.

Para alcanzar dicho objetivo, este PFC se ha estructurado en seis capítulos que se detallan a continuación:

El primer capítulo está dedicado a la introducción de la tecnología IEEE 802.15.4 utilizada en los dispositivos inalámbricos Telosb de este PFC.

El segundo capítulo está dedicado a definir las clasificaciones, características y problemáticas de las redes inalámbricas multisalto y más en concreto del tipo de redes con las que se va a trabajar en el proyecto.

En el tercer capítulo se trata en profundidad el encaminamiento en redes inalámbricas multisalto, citando los estándares y propuestas más destacadas. En este capítulo también se realiza un estudio del estado del arte de las métricas de encaminamiento en este tipo de redes y se explica en detalle el protocolo nst-AODV utilizado para la realización de las pruebas.

En el cuarto capítulo se enumeran las tecnologías utilizadas describiendo las plataformas hardware y software empleadas en este proyecto.

En el capítulo número cinco se realiza un completo estudio del parámetro LQI para conocer su comportamiento y ser utilizado para la implementación de las distintas métricas evaluadas.

Por último, en el capítulo seis, se presenta una comparativa experimental del rendimiento de las cuatro métricas en el protocolo nst-AODV.

En la parte final de la memoria, se describen las conclusiones extraídas así como un breve apartado con las posibles líneas futuras e implicaciones medioambientales.

CAPÍTULO 1. TECNOLOGÍA RADIO IEEE 802.15.4

En este capítulo se realiza una breve introducción al estándar IEEE 802.15.4. Se describe las características más relevantes como la capa física y de control de acceso al medio que define los distintos modos de funcionamiento de que dispone y se hace un brevemente inciso en el tipo de transceptor utilizado en el entorno de pruebas de este proyecto.

1.1. Introducción

El estándar IEEE 802.15.4 [1] fue creado para cubrir la necesidad del mercado de estándares inalámbricos de baja tasa para aplicaciones en redes de sensores. Los estándares existentes hasta el momento en el mercado estaban destinados a aplicaciones con mayores requisitos en cuanto a ancho de banda se refiere, como pueden ser videoconferencias o redes domésticas. Los ejemplos más representativos de estas tendencias son el IEEE 802.11, también conocido como WIFI, y el IEEE 802.15.1, Bluetooth. Los inconvenientes que surgían al utilizar cualquiera de éstos, eran su gran consumo de energía y ancho de banda frente a la baja tasa y bajos requisitos de energía necesaria para las redes de sensores. A continuación se puede observar una pequeña comparativa entre el estándar 802.15.4 y otros estándares como son Bluetooth y wi-fi:

Tabla 1.1. Comparativa de estándares inalámbricos

Estándar	Ancho de banda	Consumo de potencia	Ventajas	Aplicaciones
Wi-fi	Hasta 54 Mbps	400 mA transmitiendo 20 mA en reposo	Gran ancho de banda	Navegar por Internet, redes de ordenadores, transferencia de ficheros
Bluetooth	1 Mbps	40 mA transmitiendo, 0.2mA en reposo	Interoperatividad, sustituto del cable	Wireless USB, móviles, informática doméstica
802.15.4	250 Kbps	1,8 mA transmitiendo, 5,1µA en reposo	Batería de larga duración, bajo coste	Control remoto, productos dependientes de la batería, sensores etc

La necesidad de este nuevo estándar impulsó la creación de distintos grupos de trabajo, como por ejemplo, el IEEE 802.15 WPAN Task Group 4 [2], con el objetivo de investigar en comunicaciones con baja tasa de transmisión y el cual

publicó en octubre de 2003 las especificaciones para el estándar IEEE 802.15.4, y seis meses después una agrupación de compañías para el desarrollo de aplicaciones sobre este tipo de redes, llamada ZigBee Alliance [3] publicó la especificación ZigBee v1.0 indicando recomendaciones para la creación de aplicaciones y definiendo una arquitectura completa funcionando sobre dicho estándar.

De esta manera, el estándar 802.15.4 define la capa física y la subcapa MAC para las redes LR-WPAN (Low Rate Wireless Personal Area Network), mientras que ZigBee Alliance, aprovechando y basándose en las capas inferiores definidas en el estándar 802.15.4, trabaja en las capas superiores y define la capa de aplicación, la capa de red y los servicios de seguridad.

La tecnología inalámbrica 802.15.4 permite comunicaciones de corto alcance, con distancias comprendidas de hasta 75 m, y bajo consumo. Puede funcionar en la banda de 2,4 GHz a una tasa de 250 Kbps, en la de 868 MHz a 40 Kbps y en la de 915 MHz a 20 Kbps, aunque la mayoría de fabricantes optarán por la elección de la primera ya que puede ser usada en todo el mundo, mientras que las dos últimas sólo se pueden usar en Europa y EEUU, respectivamente.

El objetivo es que un sensor equipado con esta tecnología pueda ser alimentado con dos pilas AA un periodo de entre seis meses y dos años, dependiendo del tipo de aplicación.

En conclusión, IEEE 802.15.4 resulta ideal para redes de sensores, escalables, con muchos dispositivos, pocos requisitos de ancho de banda y dónde se requiera una duración muy prolongada de la batería. Por lo tanto, se puede afirmar que en ciertas condiciones y para determinadas aplicaciones puede ser una buena alternativa a otras tecnologías inalámbricas ya consolidadas en el mercado.

1.2. Dispositivos y topologías existentes en 802.15.4

Dentro del estándar 802.15.4 se pueden definir dos tipos de nodos según su funcionamiento y la topología utilizada en la red sensorial, a continuación definimos los dos tipos de nodos:

- *FFD (Full Function Device)* son dispositivos capaces de organizar y coordinar el acceso al medio de otros dispositivos de la misma red. Estos dispositivos se suelen utilizar en redes donde se necesita un nodo central, como puede ser en redes con topología en estrella, y suelen requerir un consumo de energía superior a otros nodos por lo que se suelen conectar a la red eléctrica.
- *RFD (Reduced Function Device)*: son dispositivos con un bajo consumo de energía y de un bajo coste y simplicidad. Disponen de menores capacidades a las de los FFD y se suelen utilizar en cualquier tipo de red.

Dependiendo de la aplicación que se esté desarrollando, se pueden configurar tres tipos de topología: en estrella (Star), peer to peer y cluster-tree. A continuación definiremos cada una:

En la topología en estrella la comunicación se establece entre los nodos (RFD o FFD) y el nodo central llamado PAN coordinator. Una vez se conectan los nodos en una red en estrella, se elige cuál va a ser el nodo coordinador de dicha red. El nodo elegido proporciona un identificador de red que no puede ser igual al identificador de otra red dentro del radio de acción de este nodo coordinador (área de cobertura o huella). El nodo coordinador será el que autorice la transmisión a los demás nodos debido a que será este el controlador de la red.

Las aplicaciones más comunes que utilizan este tipo de topología son la conexión entre el ordenador personal y los periféricos, domótica o juguetes.

A continuación se puede observar la estructura que tendría una red con esta topología.

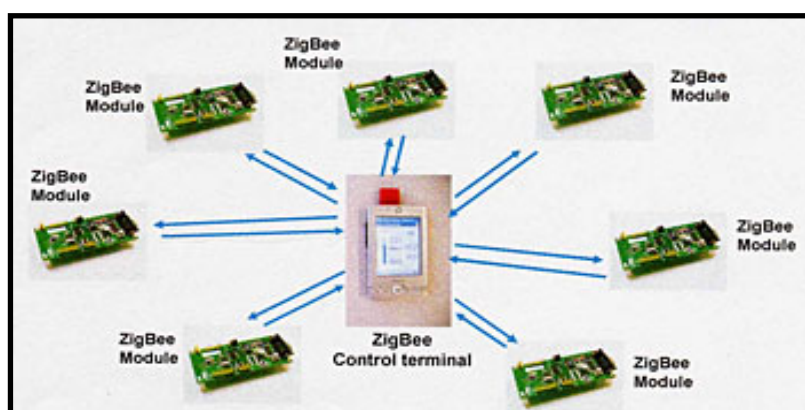


Fig. 1.1. Topología en estrella

En la topología Peer to peer o Mesh, también existe el papel del dispositivo PAN coordinator pero no tiene las mismas funciones revelantes. En contraste con la topología en estrella, cualquier dispositivo puede comunicarse con cualquier otro mientras ambos estén en la misma área de cobertura o bien utilizando otros nodos para llegar al destino (topología Mesh) debido a que tienen la misma prioridad a la hora de transmitir. Este tipo de topología es utilizada en redes ad hoc y se implementa en distintas aplicaciones como pueden ser control industrial, redes de sensores inalámbricas o aplicaciones de inventario. Esta topología permite múltiples saltos entre el nodo origen y destino con lo que requiere la utilización de protocolos de enrutamiento como el nst-AODV utilizado en este proyecto. A continuación podemos observar la estructura que tendría una red con esta topología.

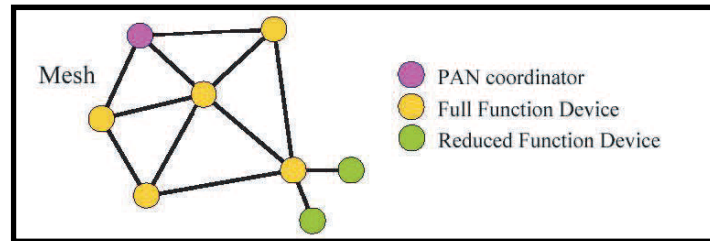


Fig. 1.2. Topología punto a punto o Mesh

La topología Cluster tree es un caso especial de la topología peer-to-peer donde se conectan varios dispositivos FFD y RFD entre sí, formando una jerarquía de árbol. En este tipo de topologías existen varios nodos coordinadores en una determinada zona y luego existe el papel del PAN coordinator que es el coordinador de toda la red situado a nivel superior como se puede observar en la siguiente figura:

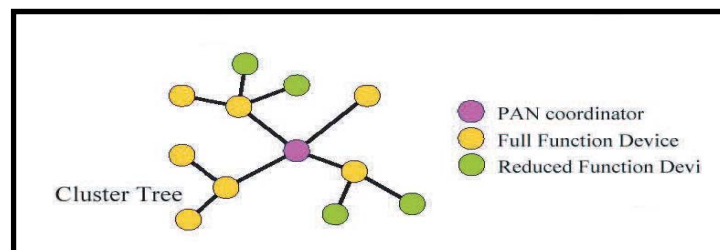


Fig. 1.3. Topología Cluster Tree

1.3. Capa Física

La capa física del 802.15.4 está separada en dos subcapas: PHY data service y PHY management que son las encargadas de transmitir y recibir mensajes a través del medio radio.

Algunas características globales de la capa física son el control del transceptor radio, calidad del enlace (LQI), selección de canal, detector de energía(ED), detección de portadora (CCA) para su uso en CSMA-CA a nivel MAC, etc...

El estándar define dos opciones de transmisión según la banda de frecuencia utilizada (868/915 MHz y 2450 MHz), ambos están basados en el DSSS¹ (direct sequence spread spectrum o espectro ensanchado). Como se puede observar en la tabla 1.2., se dispone de distintas velocidades de transmisión dependiendo la frecuencia que se utilice, con lo que conlleva que a mayor frecuencia mayor velocidad pero menor área de cobertura debido a la atenuación de la señal a frecuencias elevadas.

¹ DSSS es una técnica de modulación que utiliza un código de pseudoruido para modular directamente una portadora, de tal forma que aumente el ancho de banda de la transmisión y reduzca la densidad de potencia espectral. La señal resultante tiene un espectro muy parecido al del ruido, de tal forma que a todos los radiorreceptores les parecerá ruido menos al que va dirigida la señal.

Por otro lado, se puede observar los distintos tipos de modulación que se utilizan dependiendo de la frecuencia utilizada (BPSK y O-QPSK) (véase [4]).

Tabla 1.2. Parámetros establecidos en función de la banda utilizada.

PHY (MHz)	Frequency (MHz)	Spreading parameters		Data Parameters		
		Chip Rate (Kchip/s)	Modulation	Kbps	Ksymbols/s	2 ⁿ
868/915	868-868,6	300	BPSK	20	20	1
	902-928	600	BPSK	40	40	1
2450	2400-2483,5	2000	O-QPSK	250	62.5	4

En la banda de 2,4GHz, que es la más eficiente en cuanto al uso del ancho de banda, disponemos de un total de 16 bandas o canales de 2 MHz con una distancia entre canales de 5MHz para minimizar interferencia por canal adyacente tal y como se muestra en la fig. 1.4.

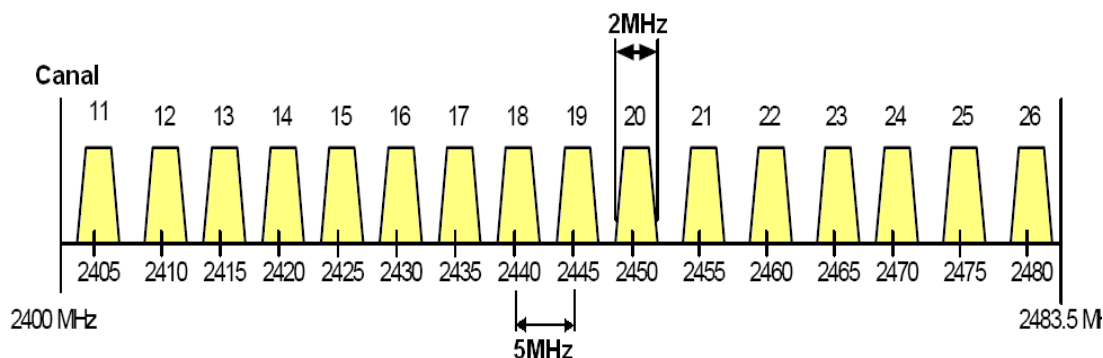


Fig. 1.4. División espectral de canales en la banda de 2.4GHz

La unidad de datos a nivel físico (PPDU) que se ilustra en la fig. 1.5. está compuesta por tres partes:

- SHR, permite a un dispositivo receptor sincronizarse para poder leer bien la información contenida en la PPDU, también indica el final de trama ya que la trama puede tener una longitud variable.
- PHR indica la longitud de información ya que esta puede ser variable como hemos comentado anteriormente.
- PSDU es la información de la trama a nivel físico.

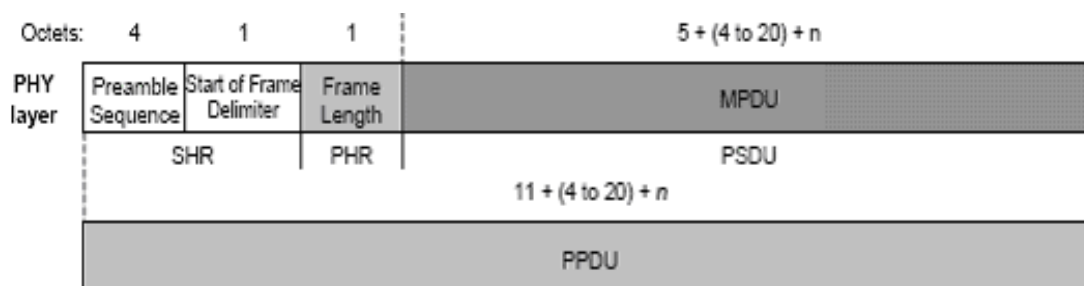


Fig. 1.5. Estructura paquete PPDU.

1.4. Capa de Control de Acceso al Medio

La capa de control de acceso al medio (MAC) proporciona dos servicios: MAC data service y MAC management service. Ambos servicios interactúan en la capa MAC y permiten la transmisión y recepción de unidades de datos, MPDU, a través del servicio de datos PHY. Las funciones más relevantes son:

- Generar beacons en el caso de un PAN coordinator y que el resto de nodos se sincronicen al ritmo de los beacons.
- Mecanismo de acceso al medio CSMA-CA.
- Asociación o desasociación a una PAN.
- Funciones de seguridad (encriptación AES).
- Mecanismos de fiabilidad entre nodos (ACK's).

La ventaja de este nivel MAC respecto al de otros estándares es que tan solo se dispone de 21 primitivas de servicio o comandos, lo que redundará en un hardware más sencillo y más barato de fabricar.

1.4.1. Modo de Funcionamiento

La capa MAC define dos modos de funcionamiento a la hora de transmitir datos: modo beacon y en modo beaconless.

La transmisión en modo beacon está orientada a redes donde existe el papel del coordinador (por ejemplo cuando estamos delante de una topología en estrella), el PAN coordinator se encarga de transmitir beacons cada cierto tiempo para que los dispositivos dentro de su red se puedan sincronizar, permitiendo en caso necesario una latencia mínima para aquellos dispositivos que necesiten tener este parámetro garantizado (transmisión con beacons y un tiempo de acceso garantizado). Como mecanismo de acceso al medio utilizamos CSMA-CA ranurado.

La transmisión en modo beaconless está orientada a redes peer to peer donde todos los nodos se comunican entre sí sin la intervención de un coordinador. El mecanismo de acceso al medio es el CSMA-CA no ranurado de modo que cada dispositivo transmite en el momento que es necesario sin esperar la pauta (beacon) de un PAN coordinator.

Esta última es la configuración que nos encontramos por defecto en los dispositivos utilizados en el transcurso de las pruebas. El funcionamiento consiste en que cada vez que un dispositivo desea transmitir datos o comandos MAC tiene que esperarse un tiempo aleatorio (backoff). Si encuentra el canal libre pasado este tiempo intenta transmitir. Si el canal estuviera ocupado después del periodo de backoff volvería a esperar otro tiempo aleatorio así

sucesivamente hasta llegar a un número máximo de intentos definidos por la variable BE dentro del algoritmo CSMA-CA.

Existen cuatro tipos de tramas dentro del estándar 802.15.4: la trama beacon, la trama de datos, la trama ack y por último la trama de comando MAC. Para observar sus respectivas estructuras véase anexo A.

1.5. Mecanismos de robustez del 802.15.4

El estándar IEEE 802.15.4 ha desarrollado para entornos hostiles y medios compartidos por lo que se han definido una serie de mecanismos para que sea robusto en estos entornos:

- CSMA-CA: sistema anteriormente comentado basado en la detección de portadora evitando colisiones. Su esquema de funcionamiento lo hace excelente compartiendo el medio.
- Tramas con confirmación (ACK): cuando enviamos tramas, se nos devuelve una trama ACK confirmando que la trama de datos o cualquier otra ha sido recibida correctamente.
- Verificación de los datos (CRC): mediante un polinomio generador de grado 16 se obtiene la redundancia y se puede comparar el CRC enviado con el calculado en destino para verificar los datos.
- Restricciones de consumo: IEEE 802.15.4 está pensado para aplicaciones que utilicen una batería o una unidad de energía agotable. Estas aplicaciones transmitirán información de forma muy esporádica por lo que la cantidad de energía que consume un nodo cuando escucha el canal es ultra baja.
- Seguridad: implementa seguridad de clave simétrica mediante el estándar de encriptación AES. El manejo y gestión de las claves es derivado a capas superiores.

1.6. Implementación utilizada en el proyecto

El transceptor radio que ha sido utilizado en el proyecto es el modelo CC2420 perteneciente a la casa Chipcon [5], que implementa el estándar IEEE 802.15.4.

Este chipset cumple con todas las características mencionadas en los apartados anteriores, destacando que funciona por defecto en el modo IEEE 802.15.4 sin beacons y utiliza el mecanismo de acceso al medio CSMA-CA con la única modificación de que sólo realiza dos intentos de transmisión, es decir, realiza un backoff para intentar transmitir y si ve el medio ocupado realiza otro backoff de mayor duración e intenta transmitir. Si no lo consigue, desiste de la transmisión.

Para consultar alguno de sus parámetros por defecto véase anexo B.

CAPÍTULO 2. REDES INALÁMBRICAS MULTISALTO

2.1. Introducción

Este capítulo está dedicado a la definición y aclaración de conceptos sobre redes inalámbricas multisalto y en concreto sobre las redes 802.15.4. Se exponen los diferentes tipos de redes inalámbricas multisalto existentes y la problemática relativa al encaminamiento, tema central de este PFC.

2.2. Definición de conceptos y clasificación de redes

Se puede considerar a las redes inalámbricas de sensores como un caso particular de redes inalámbricas multisalto. En este tipo de redes los nodos están conectados mediante enlaces sin hilos actuando como encaminadores de paquetes de otros nodos que no están en conexión directa con sus destinos. Dentro de estas redes se pueden diferenciar dos grandes grupos: redes con infraestructura y redes sin infraestructura.

En las redes inalámbricas multisalto con infraestructura existen dos tipos de nodos, los activos o encaminadores, capaces de enrutar la información hacia su destino y los pasivos que únicamente se limitan a enviar y recibir paquetes.

Por otro lado, en las redes sin infraestructura los nodos móviles tienen las mismas prestaciones y se conectan entre sí de manera arbitraria cooperando para formar una red, es decir, todos los nodos de la red son nodos activos capaces de saber y elegir la mejor ruta hacia el destino de todos los caminos posibles. Éstas son las denominadas redes ad-hoc o MANET (Mobile Ad-hoc Network).

Este proyecto final de carrera, se va a centrar en las redes inalámbricas multisalto sin infraestructura y más concretamente en el caso particular de las redes de sensores, también conocidas como LoWPANs (Low-power Wireless Personal Area Networks).

2.3. Redes MANET

Las redes MANET o ad-hoc aparecen a principios de los años 70. Como en muchos otros ámbitos de las telecomunicaciones este tipo de redes tienen un origen militar. Concretamente nacen gracias al proyecto Packet Radio Networks (PRNETs). A causa al gran interés despertado por las redes inalámbricas y la progresiva reducción del tamaño de los dispositivos, la investigación de estos tipos de redes ha pasado de ser terreno exclusivamente militar a estar presentes también en ámbitos académicos y científicos con finalidades civiles

Las características más relevantes de las redes ad-hoc, son:

- *Topología variable*: en este tipo de redes se asume que los nodos son móviles, pudiéndose desplazar libremente, desapareciendo de la zona de cobertura de unos dispositivos y formando nuevos enlaces con nuevos nodos siendo capaces de redescubrir nuevas rutas.
- *Dispositivos con capacidad limitada*: debido a la naturaleza móvil de este tipo de redes, los dispositivos que la forman, han de ser portátiles y ligeros, características que causan restricciones a nivel hardware y software
- *Enlaces inalámbricos*: la comunicación en este tipo de redes se lleva a cabo de manera inalámbrica, caracterizada por anchos de banda reducidos y más propensos a errores que los enlaces cableados.
- *Baterías limitadas*: debido a la movilidad de los nodos y sus escenarios de aplicación, éstos no se pueden tener enchufados a la red eléctrica por lo que su autonomía viene marcada por baterías limitando el tiempo de vida del dispositivo al tiempo que dure la batería.
- *Ausencia de infraestructura*: no existe ninguna entidad centralizada ni infraestructura de algún tipo. Cada nodo puede desempeñar tanto la función de terminal como la de encaminador.

Entre los puntos de interés de este tipo de redes encontramos su facilidad de despliegue, auto-configuración. Entre sus desafíos principales está la creación de rutas optimizadas para la comunicación en cada situación.

2.4. Redes LoWPAN

2.4.1. Características y problemáticas de las redes LoWPAN

Una LoWPAN es una red de comunicaciones de bajo coste que permite conectividad sin hilos a elementos de funcionalidades reducidas en aplicaciones sin grandes requerimientos de velocidad ni capacidad. Estas redes cumplen el estándar IEEE 802.15.4, del cual se ha hablado en el capítulo anterior. Este tipo de redes están orientadas a dispositivos como los sensores: de bajo coste, con capacidades de procesamiento, memoria de almacenamiento y batería limitados.

Las características que se definen en el RFC 4919 sobre las LoWPAN son las siguientes:

- *El reducido tamaño de los paquetes*: el máximo tamaño de paquete en capa física es de 127 bytes, y en capa MAC de 102 octetos.
- Ofrecen soporte para direcciones de 16 o 64 bits.

- Anchos de banda limitados.
- *Topologías*: las redes han de soportar topologías mesh y de estrella.
- Bajo consumo y bajo coste.
- *Gestión y configuración limitada*: puede darse el caso en el que dada la localización de algunos dispositivos éstos sean difíciles de acceder. Por este motivo, la configuración de los protocolos debería ser mínima.
- *Descubrimiento de servicios*: las LoWPANs requieren protocolos simples de descubrimiento para descubrir, controlar y mantener los servicios que proporcionan los dispositivos.
- *Seguridad*: debe considerarse dependiendo de las necesidades de cada aplicación. IEEE 802.15.4 proporciona seguridad a nivel de enlace mediante AES (Advanced Encryption Standard). Debido a la naturaleza de los dispositivos y sus limitaciones, deberían descartarse soluciones que impliquen un excesivo ancho de banda o nivel de cálculo.

2.4.2. Iniciativas de estandarización

La estandarización en el ámbito de las redes LoWPAN es llevada a cabo por el grupo de trabajo 6LoWPAN del IETF cuyo objetivo principal es intentar proporcionar los mecanismos necesarios con el fin que una red 802.15.4 pueda ser vista como un enlace IPv6. Para la consecución de este fin se requiere aspectos como la fragmentación de los paquetes para que puedan ser encapsulados en tramas 802.15.4, el enrutamiento a través de la red de sensores, etc.

Hasta la fecha han conseguido la publicación de 2 RFC's uno que trata sobre la problemática que se presentan en estos tipos de redes (RFC 4919) y otro sobre la transmisión de paquetes IPv6 sobre redes IEEE 802.15.4 (RFC 4944).

En cuanto al encaminamiento en este tipo de redes ver punto 3.4.1.

CAPÍTULO 3. ENCAMINAMIENTO EN REDES INALÁMBRICAS MULTISALTO

3.1. Introducción

El encaminamiento es siempre un punto crítico en toda red. Los protocolos de encaminamiento para redes inalámbricas deberán tener en cuenta las características y problemas de este tipo de redes como ancho de banda, limitación de baterías, propagación radio, etc. Desde mediados de los años 90, el encaminamiento en redes ad-hoc ha sido uno de los temas centrales de investigación.

En este capítulo se analiza el encaminamiento en redes MANET citando las estandarizaciones y propuestas más destacadas en este ámbito así como en las redes LoWPAN. Se hace especial inciso en el protocolo AODV y TinyAODV dado que son la base del protocolo de encaminamiento nst-AODV utilizado en este proyecto. Finalmente se realiza un estudio del estado del arte en cuanto a métricas de encaminamiento para redes de sensores.

3.2. Encaminamiento en redes MANET

El grupo MANET del IETF [6] tiene como objetivo estandarizar protocolos de encaminamiento apropiados para redes inalámbricas con topologías tanto estáticas como dinámicas. Las estandarizaciones pueden dividirse en dos líneas, una para protocolos proactivos (PMP, Proactive Manet Protocol) y otra para reactivos (RMP, Reactive Manet Protocol).

A continuación se realiza una clasificación de protocolos de encaminamiento y se citan los estándares de MANET así como los protocolos que están en proceso de estandarización.

3.2.1. Clasificación protocolos de encaminamiento

De las múltiples clasificaciones que pueden hacerse sobre los protocolos de encaminamiento, una primera aproximación es la división en dos grupos, reactivos y proactivos, y en un tercero derivado de ellos, los protocolos híbridos.

3.2.1.1. Protocolos proactivos

Los protocolos de encaminamiento proactivos tratan de mantener actualizada continuamente la información necesaria para el enrutamiento, de manera que cada nodo tiene siempre una ruta hacia cada destino. Los cambios en la red implican un intercambio de mensajes entre los nodos para la actualización de las tablas y por tanto un overhead en la red. Por otra parte los protocolos proactivos exhiben retardo extremo a extremo reducido ya que antes de

mandar un paquete, un nodo siempre tiene una ruta disponible hacia un destino.

Los principales protocolos proactivos estandarizados por MANET son OLSR (Optimized Link State Routing Protocol) [7] y TBRPF (Topology dissemination Based on Reverse-Path Forwarding) [8], ambos publicados como RFC Experimental. OLSR se basa en la filosofía del estado del enlace. Existe un intercambio de mensajes entre todos los nodos para formar una visión completa de toda la red para poder definir el encaminamiento. Además de este mecanismo existe un otro adicional, de paquetes “hello”, para el descubrimiento de vecinos. TBRPF también es un protocolo de estado del enlace que se basa en el cálculo de un árbol cuya raíz es un nodo origen que genera caminos a todos los nodos alcanzables. Si no hay cambios en la red se minimiza el overhead puesto que para el descubrimiento de vecinos se envían mensajes diferenciales.

También es destacable la evolución de OLSR, llamada OLSRv2 que se encuentra disponible en versión draft [9] y que será sometido para su publicación como RFC Standards Track.

3.2.1.2. Protocolos Reactivos

A diferencia de los protocolos proactivos, los protocolos reactivos, únicamente crean una ruta hacia el destino cuando es necesario enviar un paquete de información. El nodo emisor inicia el proceso de descubrimiento de rutas y este termina cuando se ha logrado alcanzar el destino de la forma más óptima (según la métrica implementada) o cuando examinadas todas las alternativas ninguna lleva al destino final. En comparación con los proactivos, los protocolos reactivos dan lugar a un notable aumento de la latencia, puesto que se debe considerar el tiempo de descubrimiento de la ruta, pero por otra parte, no hay necesidad de generar mensajes de control constantemente para actualizar rutas.

El principal protocolo reactivo estándar es el AODV (Ad-hoc On demand Distance Vector) [10] que dado que es la base del protocolo de encaminamiento utilizado en este proyecto se comenta en el apartado 3.3 con más detalle. AODV no es el único protocolo de MANET que ha alcanzado el estatus de RFC, ya que DSR (Dynamic Source Routing) [11] también fue aprobado como RFC Experimental. La característica fundamental de DSR es que el encaminamiento se realiza desde el origen incluyendo una cabecera (almacenada durante el descubrimiento de ruta) en los paquetes indicando los nodos que éste deberá atravesar para llegar a su destino.

DYMO (Dynamic Manet On demand routing) [12] es otro protocolo reactivo que hereda características de AODV y DSR. Se encuentra disponible de momento en versión draft pero en el futuro alcanzará el estatus de RFC Standard Track.

3.2.1.3. Híbridos

Por su parte, los protocolos de encaminamiento híbridos combinan características de los proactivos y de los reactivos, según el protocolo, de manera que la adaptación a ciertos entornos sea la más apropiada posible. Existen algunos protocolos híbridos en los que se mantiene una filosofía proactiva en un ámbito local y reactiva a nivel más global, como el protocolo ZRP (Zone Routing Protocol).

3.2.1.4. Multicast

Dentro de las aportaciones del grupo MANET, finalmente se destaca el protocolo SMF (Simplified Multicast Forwarding) [13], que proporciona una funcionalidad de reenvío de paquetes multicast. Su objetivo es adaptar de una manera eficiente los diseños de inundación en entornos MANET, como por ejemplo los utilizados en OLSR y TBRPF, y aplicar esos mecanismos al reenvío de paquetes IP multicast.

3.3. Protocolo AODV, TinyAODV e implementación nst-AODV

En este apartado se explica el funcionamiento de los protocolos AODV y TinyAODV que son la base del protocolo utilizado en este proyecto. Finalmente se explica en profundidad el funcionamiento del protocolo nst-AODV.

3.3.1. AODV

En AODV cuando un nodo necesita comunicarse con otro que no está en su tabla inicia un proceso de descubrimiento de ruta. Para ello envía en modo broadcast un mensaje de descubrimiento de ruta RREQ (Route Request). Si un nodo que no es el destino solicitado recibe este paquete, comprueba mediante los números de secuencia si lo ha recibido previamente. Si no lo había recibido lo reenvía incrementando el número de saltos y creando la ruta inversa por la que luego volverá la respuesta. La respuesta de establecimiento de ruta, RREP (Route Reply) la podrán generar los nodos intermedios que ya dispongan de una ruta hacia el destino (mecanismo *indirect reply*) o el mismo destino una vez el mensaje RREQ le haya llegado. Los nodos envían estas respuestas RREP a los nodos de los que habían recibido el mensaje RREQ. La ruta se confirma una vez el RREP llega al nodo que había iniciado el descubrimiento de ruta.

Cuando se establece una ruta, ésta sigue siendo válida durante un período de tiempo. Esto es porqué en escenarios dinámicos no se puede asegurar que un nodo no vaya a moverse. Así pues, para el mantenimiento de rutas se definen varios mecanismos. Uno de ellos es el envío periódico de mensajes *hello* para indicar la presencia de un nodo en su vecindario.

Para invalidar una ruta se hace uso de los mensajes RERR (Route Error). Cuando un nodo detecta la ausencia de un vecino mediante los mensajes *hello* o un error en el envío de un mensaje, origina un mensaje RERR. Este mensaje

llegará hasta el origen de la ruta para que el nodo empiece un proceso de descubrimiento si es necesario.

Para más información sobre el protocolo AODV, el lector puede consultar el RFC 3561[12].

3.3.2. TinyAODV

TinyAODV revisión 3 es una versión reducida del protocolo AODV (véase anexo C). El protocolo se diferencia al RFC 3561 en los siguientes aspectos:

- los mensajes RREP son únicamente generados por el nodo destino.
- Las rutas no expiran nunca.
- No se realiza mantenimiento de rutas, pero si un nodo detecta la rotura del enlace mediante los *acknowledgments* a nivel de enlace se envía un RERR hacía el origen de la misma anulando las rutas hacía el nodo destino.

También destacar que con TinyAODV el primer paquete para el que se deba descubrir ruta hacia el destino se perderá. Por ello será necesario incorporar una fiabilidad adicional para no perder los paquetes que se envían.

3.3.3. nst-AODV

Nst-AODV (Not So Tiny – AODV) [14] fue desarrollado por el grupo de redes móviles sin hilos del departamento de ingeniería Telemática de la UPC. Fue implementado por Pere Salvatella. El objetivo de este protocolo es el de proporcionar una solución intermedia entre una solución minimalista como TinyAODV y la versión completa de AODV a la vez que aportar algunas características que puedan proporcionar beneficios en algunos escenarios, especialmente en redes cuya topología es muy dinámica. Como se ha observado TinyAODV no proporciona fiabilidad en los envíos de paquetes, ya que los primeros paquetes de datos que necesitan realizar el descubrimiento de ruta se pierden. Además, los paquetes que no se han podido enviar se descartan. La implementación de nst-AODV, más orientada a escenarios dinámicos, ha añadido un nivel más alto de fiabilidad siendo mediante buffers en los nodos y múltiples reintentos de envío.

El nst-AODV proporciona un mecanismo LLN activado por defecto e incorpora las funciones de respuesta indirecta y local repair que define AODV y que TinyAODV no implementa. Los paquetes de datos que impliquen un descubrimiento de ruta no se descartan sino que se retienen hasta que se haya creado la ruta para poder ser enviados. Al enviar un paquete se dispone de hasta tres intentos; si tras estos intentos el paquete no puede enviarse, se almacena en un buffer y se envía si se crea una nueva ruta. Si no, se descarta

el paquete. Para todas estas funciones, los nodos disponen de dos buffers FIFO, uno de recepción y otro de transmisión. Igual que TinyAODV, nst-AODV usa la métrica del mínimo número de saltos.

3.4. Encaminamiento en redes LoWPAN

Dadas las características y limitaciones de las redes LoWPAN en cuanto a duración de baterías, memoria y capacidad de procesamiento de los dispositivos, así como los frecuentes cambios en la topología, el encaminamiento en este tipo de redes es un punto crítico. Los protocolos deberán adaptarse a estas características ya que de otro modo el encaminamiento no será eficiente ni adecuado.

3.4.1. Propuestas estándar

En el seno del 6LoWPAN WG, existe un documento Internet Draft [6] que especifica los requisitos para los protocolos de routing en entornos LoWPAN.

Como por ejemplo:

- El diseño de un protocolo de encaminamiento para redes LoWPAN debe minimizar los requerimientos computacionales y complejidad algorítmica.
- Los estados de encaminamiento deben de ser minimizados.
- Los procedimientos de reparación y mantenimiento de rutas así como el descubrimiento de nodos vecino, etc., no debe aumentar el consumo de energía.
- Deben existir mecanismos de sincronización e hibernación para ahorrar energía.
- Se han de soportar flujos punto a punto, punto a multipunto y multipunto a punto.
- Han de aceptar cambios de topología dinámicos.
- Escalable.
- Han de implementar mecanismos de seguridad basados en AES.
- Evitar la fragmentación de las tramas a nivel físico.
- Hacer uso de mecanismos de capas inferiores para detectar roturas de enlace con los vecinos evitando envío de tramas HELLO con el correspondiente ahorro de energía.
- Soporte para direcciones de 16 bits y 64 bits entre otros.

- Minimizar el consumo utilizando caminos que lo optimizan, evitando retransmisiones. Utilizando indicadores como el LQI para encontrar rutas más fiables.

En el pasado, han existido propuestas de adaptación del protocolo AODV que no han prosperado. En la actualidad el grupo de trabajo ROLL de IETF está trabajando en el área de encaminamiento en redes de sensores en general. Por el momento solo han publicado documentos borradores de requerimientos en diferentes ámbitos como la domestica, industrial, urbano...

3.5. Estado del arte de métricas de encaminamiento en redes multisalto

Existen multitud de propuestas en cuanto a estrategias de protocolos de encaminamiento, pero otro aspecto a tener en cuenta es la métrica usada en dichos protocolos. Tradicionalmente la métrica más usada ha sido la del mínimo número de saltos hacia un destino. Pero en redes inalámbricas, esta métrica puede no ser la más adecuada para este tipo de entornos. Dicho de otra manera, el camino más corto no tiene por qué ser siempre ni el más rápido ni el más fiable para llegar a un destino teniendo en cuenta que en las redes inalámbricas podemos encontrarnos con enlaces de diferente calidad. Así que en estos casos podría ser más conveniente escoger otro camino con un mayor número de saltos pero de más calidad o bien el de mayor calidad en el caso de que existan varias opciones con el mismo número de saltos. Además, hay que tener en cuenta que una misma métrica no tiene por qué ser la mejor para todo tipo de entornos y aplicaciones.

3.5.1. Métricas evaluadas

A continuación se describe el funcionamiento de las métricas evaluadas en este proyecto:

- Métrica Hops:

Esta métrica se basa en la distancia, medida en número de saltos, entre origen y destino escogiendo en cada momento el camino más corto, es decir, el que tiene un menor número de saltos para alcanzar el destino correctamente.

Entre las principales ventajas de la métrica Hops se destaca la fácil implementación de ésta en cualquier protocolo de enrutamiento. Destacar que a igualdad de calidades de enlace, esta métrica proporciona la ruta con menor retardo, menor número de roturas de camino y mayor tasa de entrega

Sin embargo, la elección del camino más corto no necesariamente implica ser el mejor camino o el más rápido puesto que únicamente toma en cuenta el

número de saltos obviando otros criterios importantes como la calidad del enlace.

➤ Métrica PDR:

Esta métrica intenta maximizar el path delivery ratio de la ruta. Para obtener este parámetro primero se ha de realizar una predicción del Link Delivery Ratio (LDR) en cada salto del camino. A continuación se observa el cálculo que se realiza para la obtención de los diferentes costes en cada ruta:

$$p^* = \arg \max \prod_{i \in L_p} ESTLDR(i) \quad \begin{array}{l} P: \text{caminos potenciales} \\ L_p: \text{conjunto de enlaces} \\ \text{donde } p \in P \end{array} \quad (3.1)$$

La principal ventaja de esta métrica es que elige los caminos de mayor calidad para la comunicación, lo que conlleva a una mayor tasa de entrega y un menor número de cortes en dichos enlaces respecto a Hops.

➤ Métrica Expected Transmission Count (ETX)

El objetivo de esta métrica es elegir el camino que minimiza el número de transmisiones de capa de enlace necesarias a lo largo de un camino entre el nodo origen y destino, incluidas retransmisiones. En el caso de que dos rutas hacia un mismo destino tengan la misma calidad se seleccionará la ruta que tenga un menor número de saltos.

El cálculo del ETX de una ruta es la suma de los ETX's de cada enlace de esa ruta como se observa a continuación:

$$ETX_i = \frac{1}{(1 - P_{fi})(1 - P_{ri})} \quad \begin{array}{l} ETX_i : \text{Coste ETX del enlace } i. \\ P_{fi} : \text{Probabilidad de error enlace ida.} \\ P_{ri} : \text{Probabilidad de error enlace vuelta.} \end{array} \quad (3.2)$$

$$ETX_P = \sum_{i \in P} ETX_i \quad ETX_P : \text{Coste ETX de la ruta } P. \quad (3.3)$$

Esta métrica está basada en la tasa de entrega de cada enlace y penaliza rutas con un número de saltos elevado debido a que en cada salto como mínimo se hace una transmisión que se contabiliza en el coste total del enlace.

La principal ventaja de este tipo de métrica es que rebaja el consumo de energía en los nodos debido a que minimiza el número de transmisiones en la comunicación. Este tipo de métricas es ideal para redes inalámbricas donde los nodos se alimentan de baterías autónomas.

➤ Métrica ZigBee

Esta métrica la propone la ZigBee Alliance para redes 802.15.4. El coste cada se obtiene según la siguiente fórmula:

$$C\{l\} = \begin{cases} 7, \\ \min\left(7, \text{round}\left(\frac{1}{P_l^4}\right)\right) \end{cases} \quad \begin{array}{l} C\{l\} : \text{Coste de enlace.} \\ P_l : \text{Probabilidad de entrega en enlace l.} \end{array} \quad (3.4)$$

El coste de una ruta es la suma de los costes de los enlaces que la componen.

$$C\{P\} = \sum_{l \in P} C\{l\} \quad C\{P\} : \text{Coste de la ruta.} \quad (3.5)$$

En la especificación realizada por ZigBee Alliance, se deja libre al implementador la definición de la variable P_l aunque recomiendan que esté basada en el parámetro de calidad de enlace LQI proporcionado por IEEE 802.15.4-2003 MAC y PHY.

CAPÍTULO 4. TECNOLOGÍAS UTILIZADAS

4.1. Introducción

Este capítulo presenta las características de la plataforma hardware y del software utilizados para la realización de este proyecto.

4.2. Plataforma Hardware

La plataforma hardware utilizada como nodo de red IEEE 802.15.4 es un dispositivo desarrollado y distribuido por la University of California at Berkeley y Crossbow Technology Inc, denominada Telos revision B [17].

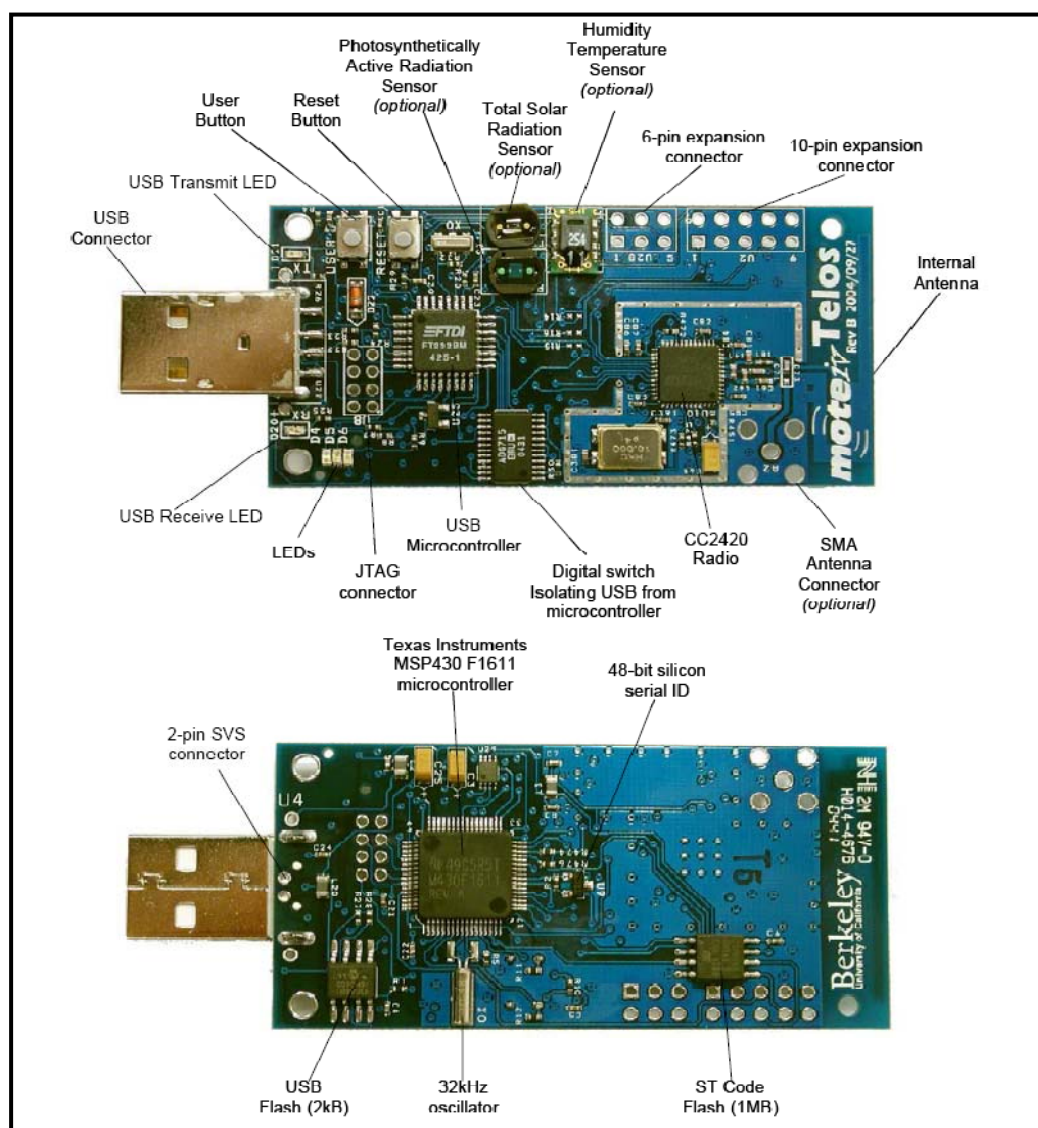


Fig. 4.1. Vista frontal y trasera de un mote telos rev.B [datasheet telosB]

Las características principales de este dispositivo, también denominado mote son:

- Transceptor IEEE 802.15.4.
- Compatibilidad global con las bandas ISM (2.4 hasta 2.4835 GHz).
- Velocidad de 250 kbps.
- Antena integrada en la placa base.
- Microcontrolador MSP430 de 8 MHz con 10 kB de RAM.
- Bajo consumo.
- Memoria flash externa de 1MB.
- Programación y recolección de datos por USB.
- Sensores varios integrados. Luz, temperatura y humedad.
- Opera con TinyOS 1.1.11 o superior.

La elección de esta plataforma frente a otras, ha sido principalmente su interfaz USB, que nos facilita la recopilación de datos a través de un bus conectado a un PC. Otro de los motivos es su sistema operativo TinyOS de libre distribución y diseñado para ser utilizado en dispositivos de bajo coste.

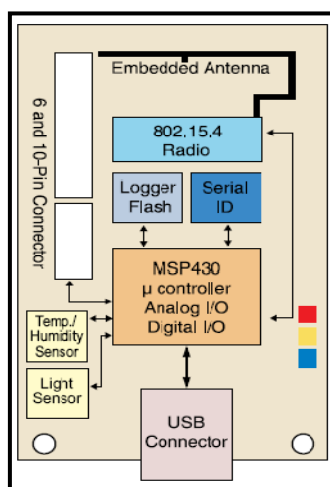
4.2.1. Telosb

La familia de motes Telos emplea el chip radio Chipcon CC2420, ya que cumple con la normativa IEEE 802.15.4. A continuación se muestra una tabla que incluye las especificaciones relevantes para este proyecto, proporcionadas por el fabricante.

Tabla 4.1. Especificaciones del mote Telos rev. B.

Specifications	TPR2420CA	Remarks
Module		
Processor Performance	16-bit RISC	
Program Flash Memory	48 Kbytes	
Measurement Serial Flash	1024 Kbytes	
RAM	10 Kbytes	
Configuration EEPROM	16 Kbytes	
Serial Communications	UART	0-3V transmission levels
Analog to Digital Converter	12 bit ADC	8 channels, 0-3V input
Digital to Analog Converter	12 bit DAC	2 ports
Other Interfaces	Digital I/O, I2C, SPI	
Current Draw	1,8 mA	Active mode
	5,1 μ A	Sleep mode
RF Transceiver		
Frecuency band	2400 MHz to 2483,5 MHz	ISM band
Transmit (TX) data rate	250 Kbps	
RF Power	-25 dBm to 0 dBm	
Receive Sensitivity	-90 dBm (min), -94 dBm (typ)	
Outdoor Range	75 m to 100 m	Inverted-F antenna
Indoor Range	20 m to 30 m	Inverted-F antenna
Current Draw	23 mA	Receive mode
	21 μ A	Idle mode
	1 μ A	sleep mode

La plataforma utilizada por los motes Telosb es la TPR2420CA cuyo diagrama de bloques se muestra en la Fig. 4.2.

**Fig. 4.2.** Diagrama de bloques TPR2420CA

Como transceptor RF se usa el chip CC2420, que utiliza la banda de 2.4 GHz con velocidades de 250 kbps, empleando una transmisión DSSS a 2 Mchips con una modulación O-QPSK. De los dos mecanismos de acceso al medio, CC2420 bajo TinyOS utiliza el modo beaconless por defecto. De esta manera, no se necesita sincronización entre los diferentes modos, haciendo que la configuración sea más simple.

4.3. Plataforma Software

En este apartado se describen las principales características de software utilizado en este proyecto: el sistema operativo TinyOS [18], el lenguaje NesC [19], el protocolo nst-AODV y la interfaz USBShark [20] utilizada para capturar los paquetes 802.15.4.

4.3.1. TinyOS

TinyOS es un sistema operativo de código abierto diseñado para redes inalámbricas de sistemas embebidos [véase anexo D]. Es un sistema operativo basado en componentes, tareas y orientado a eventos todo ello con el uso de poca memoria.

La dinámica de los programas realizados bajo TinyOS está conducida por tareas, las cuales pueden ser interrumpidas para atender eventos ya que éstos disponen de una prioridad mayor para ser atendidos [véase fig. 4.3].

Para entender TinyOS, debemos de tener presentes 3 clases de abstracción que son la esencia para su entendimiento y posterior programación:

- Los comandos son las llamadas a otros componentes de capas inferiores.
- Los eventos conllevan el efecto inverso que los comandos, es decir, una llamada hacia capas superiores. Un componente de bajo nivel avisa a uno de alto nivel de que ha sucedido un evento como, por ejemplo, la recepción de un paquete. Si sucede un evento, éste tiene mayor prioridad que cualquier tarea y pasaría a ejecutarse.
- Las tareas son porciones de código que se ejecutan de forma asíncrona siempre y cuando la CPU no tenga que ejecutar ningún evento. Éstas se ejecutan por orden de llamada (disciplina FIFO) y en caso de que la cola estuviese vacía el procesador entraría en standby hasta que un nuevo evento hiciese despertar al procesador.

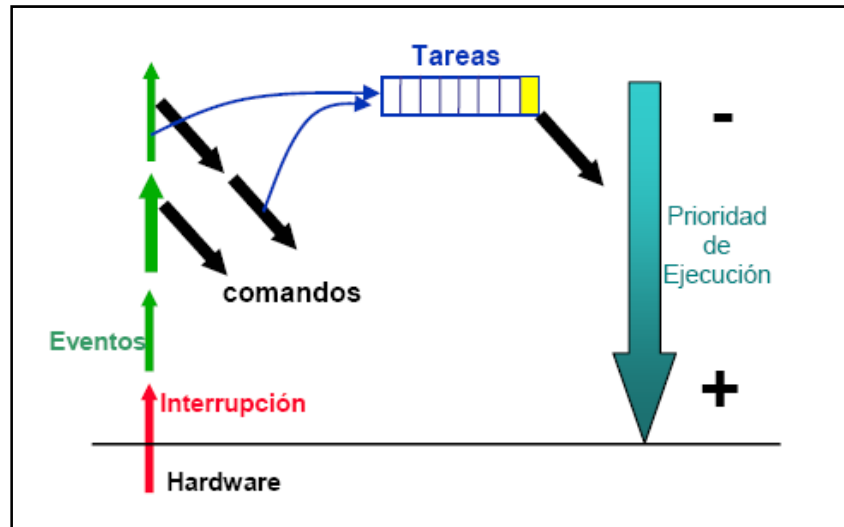


Fig. 4.3. Esquema de funcionamiento de TinyOS

TinyOS crea un nivel intermedio entre las capas inferiores PHY y MAC proporcionadas por el 802.15.4 y las capas superiores de encaminamiento tal y como muestra la siguiente figura:

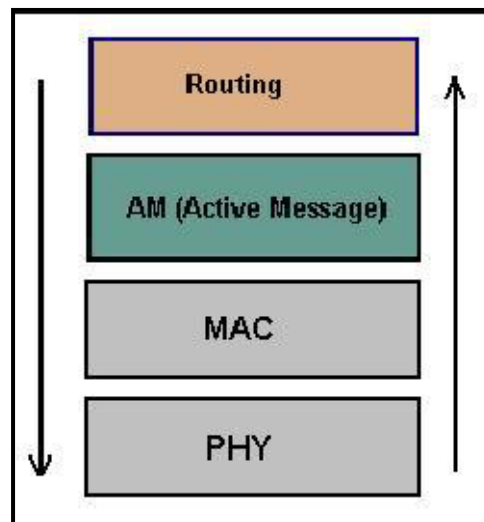


Fig. 4.4. Interacción de capas.

La mencionada capa intermedia es la AM (Active Message). Esta capa actúa como puente entre la capa de routing y la capa MAC permitiendo de esta forma la comunicación entre ellas de manera sencilla. La siguiente figura muestra el formato de los mensajes AM así como su encapsulado dentro de los paquetes 802.15.4:

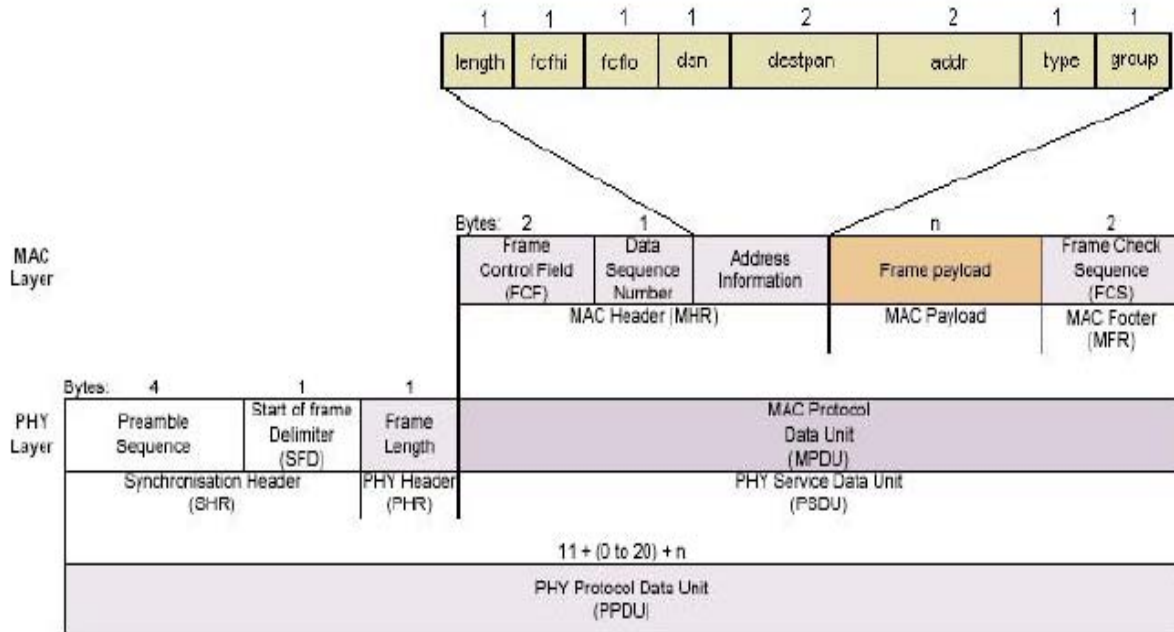


Fig. 4.5. Encapsulado del mensaje AM en la trama 802.15.4

Para programar mediante TinyOS se utiliza una extensión del lenguaje C llamada NesC, la cual permite el uso de interfaces que van conectadas a componentes.

Para crear un programa en NesC, únicamente se tiene que elegir los componentes (las partes funcionales) previamente programadas y después relacionarlas mediante otro fichero de configuración a las interfaces que serán utilizadas como código en nuestra futura aplicación.

4.3.2. NesC

NesC es un lenguaje de programación que se utiliza para crear aplicaciones que serán ejecutadas en sensores con el sistema operativo TinyOS. Por tanto, dicho lenguaje, proporciona características necesarias para poder realizar aplicaciones de una forma más cómoda para el programador.

La característica principal de este lenguaje de programación es que combina las ventajas de una programación orientada a objetos (POO) y de una programación orientada a eventos (POE). La POO permite programar los diferentes componentes de los sensores de forma genérica.

En NesC se definen dos clases de componentes: los módulos y las configuraciones. Los módulos contienen el código en sí del componente mediante interfaces. Por otra parte, las configuraciones enlazan o relacionan las interfaces de los componentes que se van a necesitar en nuestro módulo. Este fichero es crucial porque si no está bien configurado, la aplicación no compilará. Tanto módulos como configuraciones utilizan la misma extensión *.nc. La diferencia entre ambos radica en la sintaxis que utilizan y es por ello

que siguen un código de letras para diferenciar unos de otros. Por ejemplo, cuando trabajamos con un programa “aplicación”, aplicaciónM.nc sería el módulo y aplicación.nc sería el fichero configuración.

4.3.3. Protocolo nst-AODV

El protocolo nst-AODV es el protocolo utilizado para implementar y evaluar las métricas detalladas en el capítulo anterior. En este apartado se explica su principio de funcionamiento y sus características más relevantes.

A continuación se detalla la arquitectura empleada y las funcionalidades que incluye cada submódulo del nst-AODV. En la Fig. 3.1 podemos ver los submódulos que enlazan el protocolo con las capas inferiores de comunicaciones.

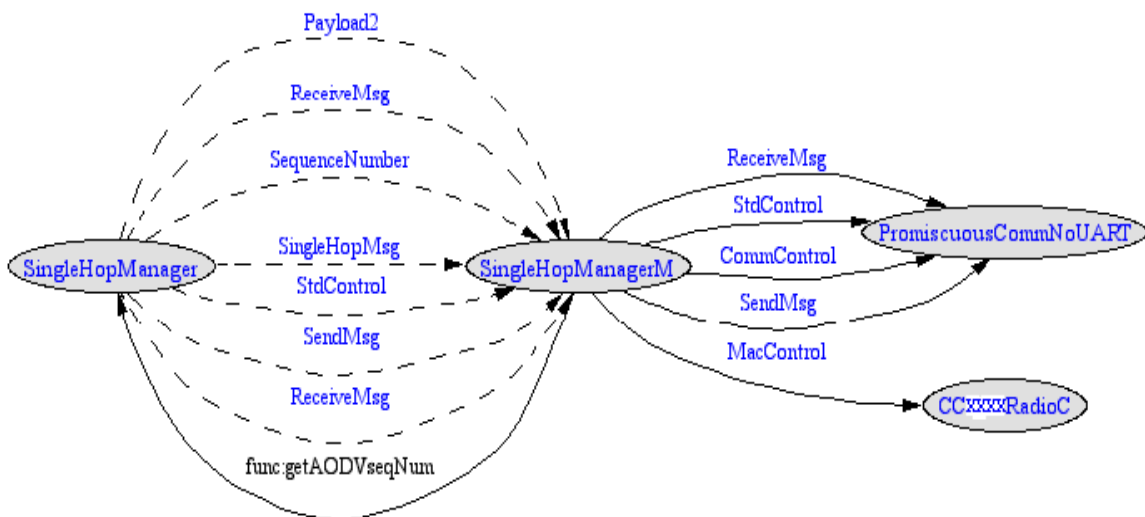


Fig. 4.6. Arquitectura del submódulo que enlaza el protocolo con las capas inferiores.

La funcionalidad del submódulo SingleHopManager consiste en ofrecer comunicaciones en salto simple directo entre nodos. Y añade una cabecera de 2 bytes adicional a la cabecera AM que identifica el nodo emisor del mensaje y un número de secuencia que determina el número de mensaje emitido por este módulo.

En la siguiente figura se observa el conjunto de módulos que actúan propiamente en el funcionamiento del protocolo de encaminamiento nst-AODV.

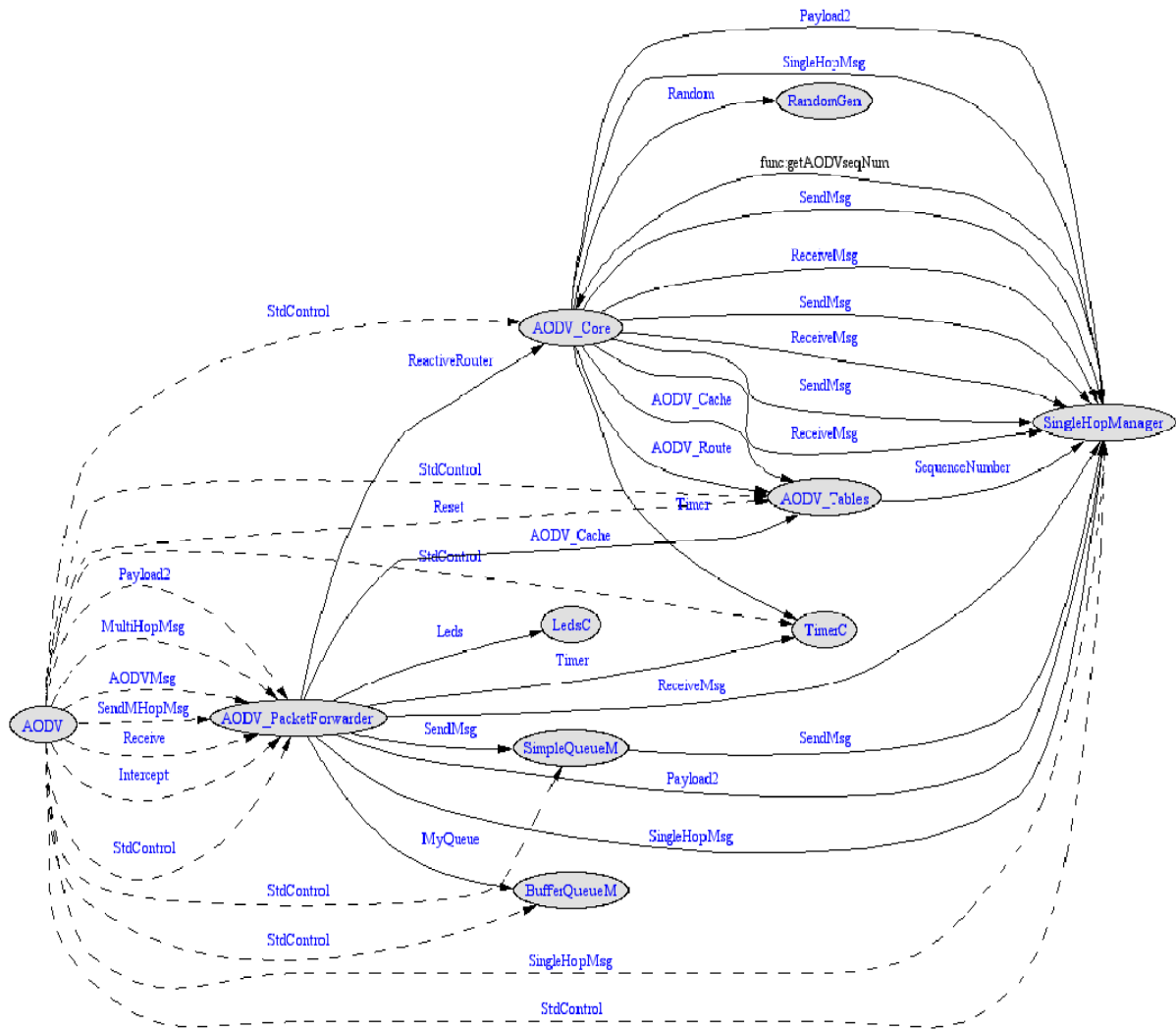


Fig. 4.7. Arquitectura de los módulos del nst-AODV.

A continuación se detalla el funcionamiento de cada uno de los módulos que forman el protocolo de encaminamiento nst-AODV:

➤ **Módulo AODV_Core.nc**

Este módulo proporciona el conjunto de acciones de control del protocolo. Es el responsable de la creación de rutas, reparación de las mismas e indicar cuando estas ya no están disponibles.

Mantiene el número de secuencia propio del nodo y se comunica con el módulo AODV_Tables para almacenar mensajes recibidos y rutas creadas.

➤ **Módulo AODV_PacketForwarder.nc**

Este módulo se encarga de enviar los mensajes, consultando la existencia de ruta y ejecutando las tareas necesarias en otros módulos para poder realizar el envío del mensaje.

➤ Módulo AODV_Tables.nc

Este módulo almacena la información relativa a los mensajes recibidos y a las diferentes rutas creadas.

➤ Módulos BufferQueue.nc y SimpleQueue.nc

Estos módulos actúan como colas de espera, y permiten evitar la pérdida de mensajes ante situaciones de fallo de ruta. La primera es la cola de entrada al servicio de envío proporcionado por AODV_PacketForwarder. La segunda sirve para realizar los reintentos múltiples en el envío de los paquetes en salto simple cuando se llama al submódulo SingleHopManager.

Como en el protocolo AODV, el nst-AODV utiliza tres mensajes de control y uno de datos. Los mensajes de control son: RREQ, RREP y el RERR).

El mensaje de datos emplea un formato formado por una cabecera de 9 bytes. De estos, 2 bytes son de cabecera de salto simple, 4 son de cabecera de salto múltiple y 3 son de cabecera AODV. Más detalle sobre los campos en la Fig. 4.8.

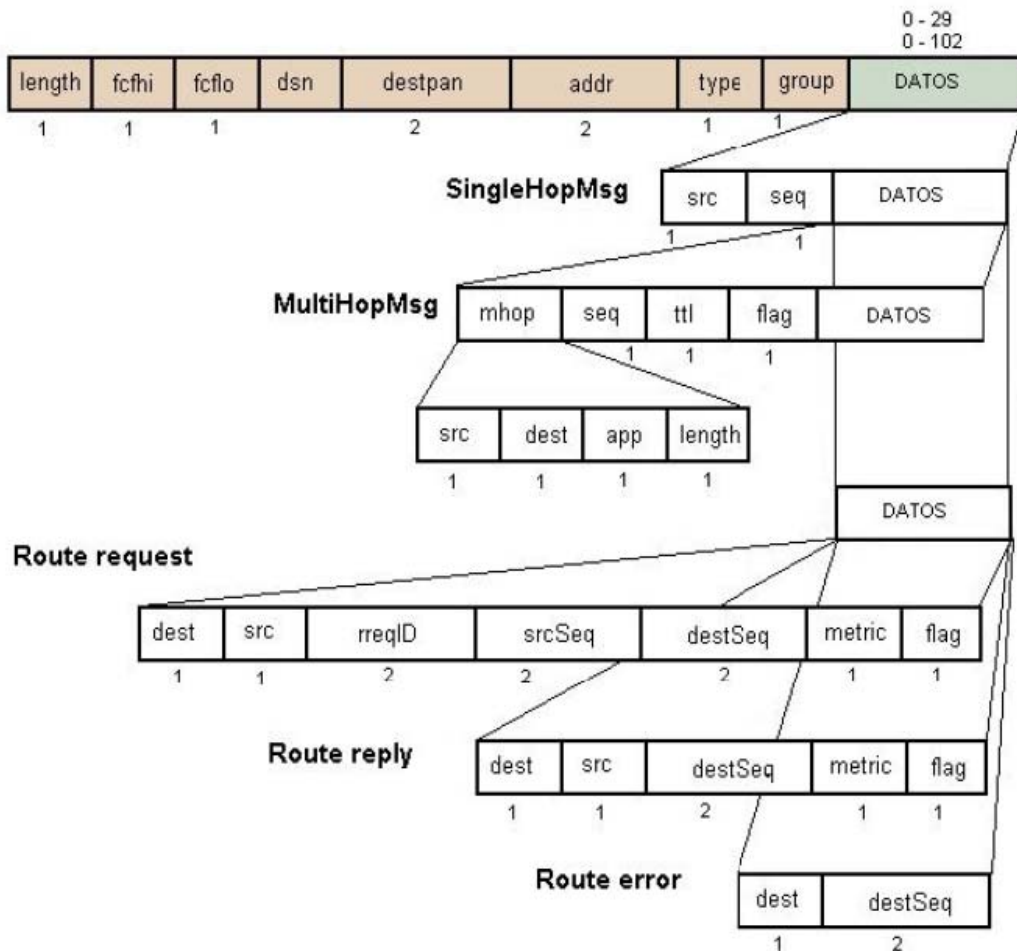


Fig. 4.8. Formatos de los Mensajes NST-AODV

El proceso de envío de un mensaje es el siguiente:

Para enviar un mensaje se llama a la función `SendMHopMsg` indicando el destino deseado. Esta función del módulo `AODV_PacketForwarder`, pasa el mensaje a la cola de `BufferQueue`. En caso de no existir tareas que ocupen el módulo gestor, se sirve el mensaje. En caso contrario este permanecerá en dicha cola hasta que el paquete previo haya sido enviado.

En el siguiente paso, el módulo `AODV_PacketForwarder` consulta la existencia de ruta mediante un interfaz hacia el módulo `AODV_Core`. Este consulta el módulo `AODV_tables` y si no encuentra ruta comunica el evento a `AODV_PacketForwarder` e inicia un proceso de descubrimiento de ruta.

Como en el protocolo AODV, el proceso de descubrimiento de ruta consiste en el envío de un mensaje `RREQ` en modo broadcast haciendo una inundación en la red controlada por número de saltos. Cuando el mensaje `RREQ` llega al nodo destino responde mediante un mensaje `RREP` (si activamos el uso de `RREP` indirectos podrá responder cualquier nodo intermedio que tenga ruta al nodo destino). Cuando el nodo origen recibe el mensaje `RREP` se indica al módulo `AODV_PacketForwarder` que hay ruta hacia al nodo destino y por lo tanto se puede enviar el mensaje. Si en algún momento llega una opción de ruta más favorable (según número de saltos o identificador de mensaje más actual), ésta sustituirá a la anterior.

El proceso de envío consiste en pasar dicho mensaje al módulo `SimpleQueue`, que intentará un número máximo de reintentos de envío del mensaje de forma correcta. En caso de que el envío sea satisfactorio, se indicará el evento `SendDone()` mediante el interfaz de envío `SendMHopMsg`.

En caso contrario, es decir, si el envío es incorrecto, éste es detectado en el módulo `AODV_PacketForwarder`. Si el nodo que detecta error es el nodo origen, inicia un nuevo proceso de descubrimiento de ruta invalidando previamente la anterior. Si se encuentra ruta se sustituye la ruta anterior por la nueva ruta, en caso contrario, se elimina la ruta mediante el envío de un mensaje `RERR` (ver Fig. 4.9).

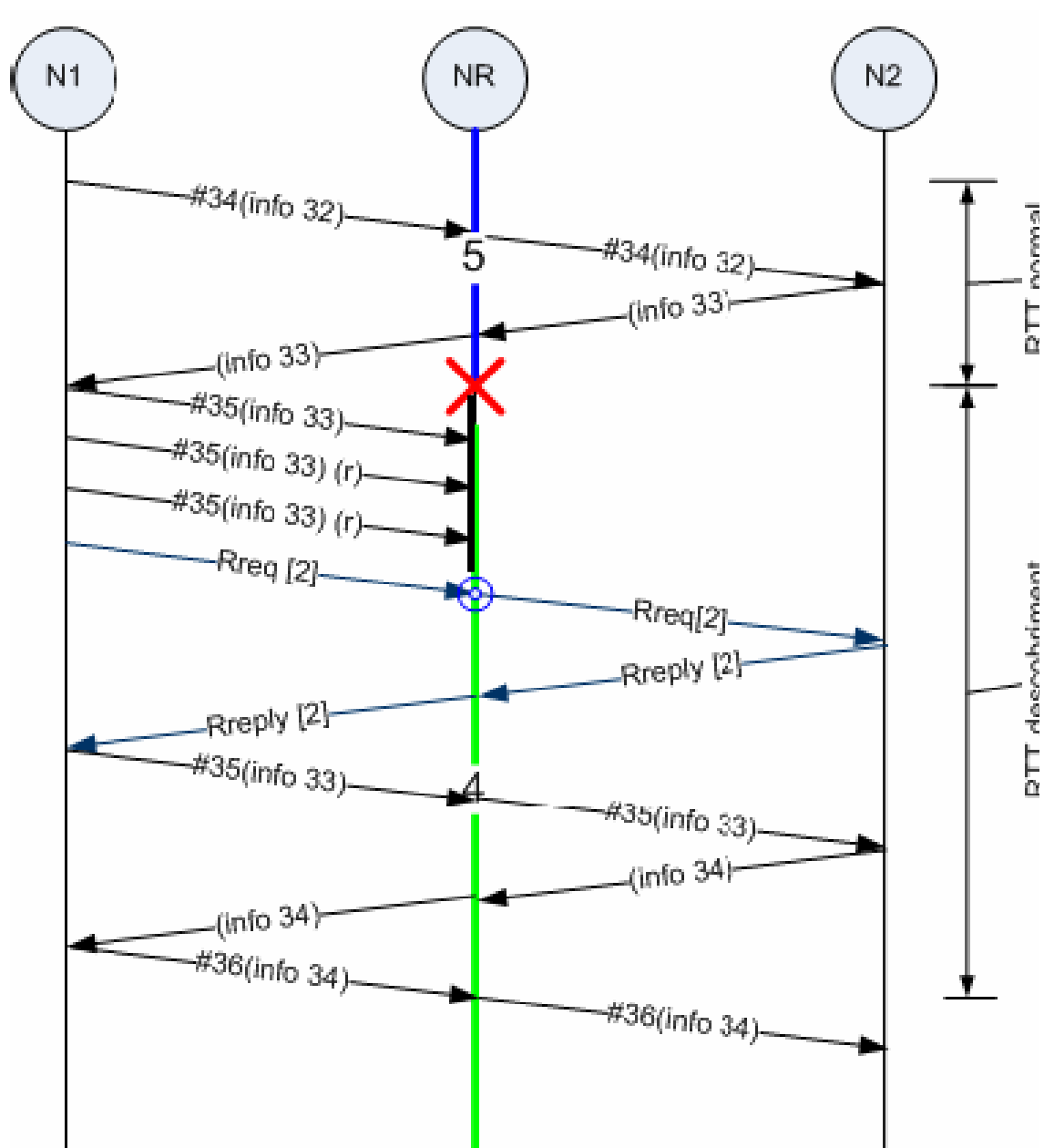


Fig. 4.9. Proceso de reparación de una ruta en el nodo inicial.

En el caso de que un nodo intermedio detecte que el envío es incorrecto, éste empezará un proceso de *local repair* que consiste en buscar una nueva ruta hacia el destino final desde el nodo intermedio. El RREP de vuelta será reenviado hasta el origen inicial de la ruta, de modo que todos los nodos actualizarán la ruta (numero de saltos, numero de secuencia) como se observa en la siguiente figura.

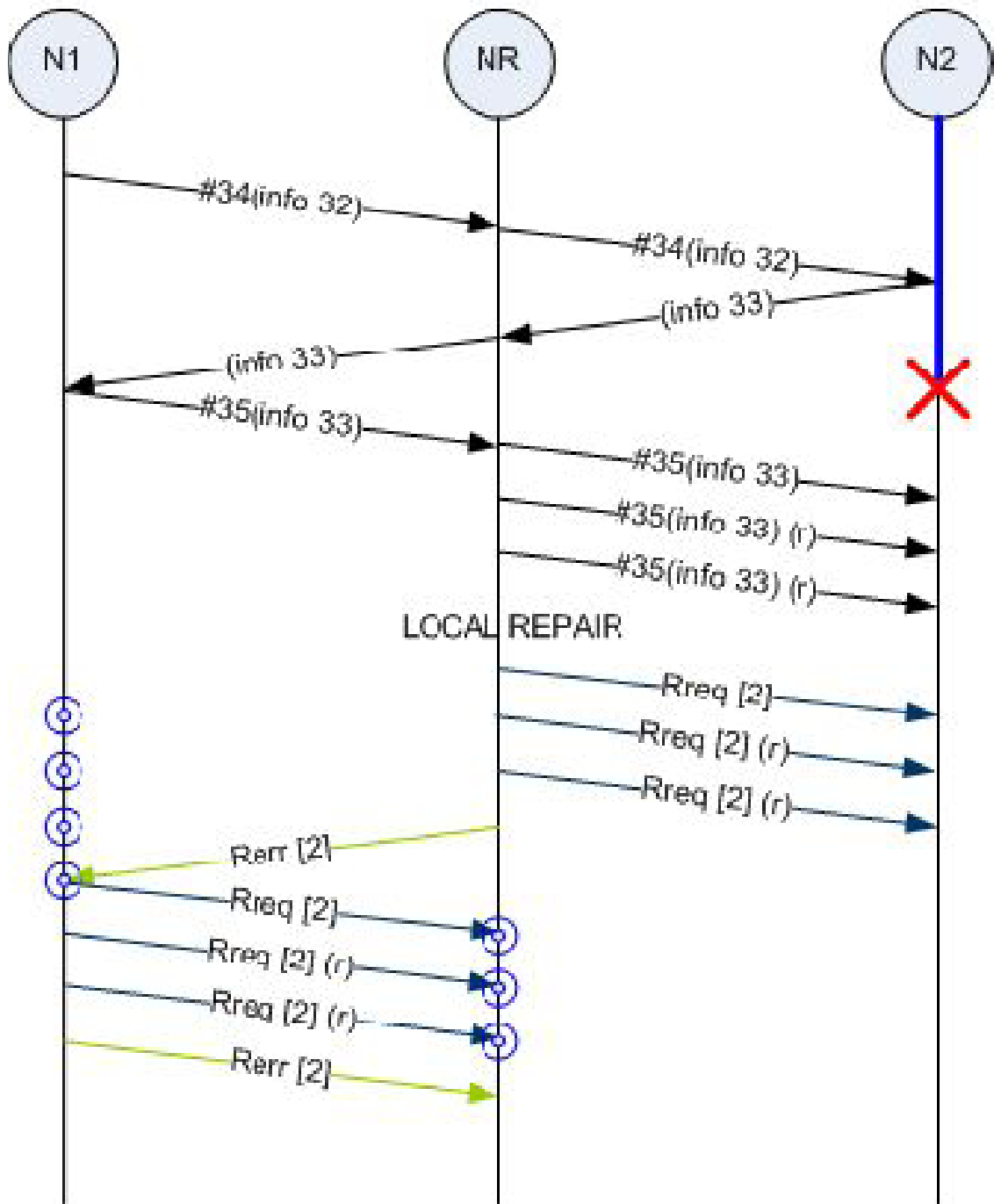


Fig. 4.10. Proceso de recuperación insatisfactorio en nodo intermedio.

Si el proceso de *local repair* no consigue reparar la ruta, ésta será eliminada. Esta operación se realiza mediante el envío de un mensaje RERR indicando el destino inalcanzable mediante inundación controlada. Los nodos que reciban el mensaje de error solamente reenviarán el mensaje si disponen de una ruta con destino a dicho nodo en su propia tabla de rutas o el nodo al que no se puede llegar se encuentra en el siguiente salto hacia otra ruta.

Cabe destacar que el protocolo nst-AODV permite diferenciar el tipo de nodo mediante *flags*. Este parámetro permite diferenciar los nodos, otorgándoles distinto rango según sea su nivel de autonomía y movilidad, permitiendo priorizar las rutas pasando por nodos identificados como nodos de baja movilidad (generalmente estáticos) haciendo aumentar la fiabilidad de las rutas creadas. También se priorizan rutas con nodos que tienen conexión a una fuente de alimentación, ya que afectarán de menor forma a la autonomía general del sistema.

Esta última opción se presenta interesante cuando tenemos nodos conectados con conexión a una toma de corriente, y cuya autonomía no se ve afectada por el número de mensajes enviados.

Por último se presentan las interfaces de conexión disponibles por el módulo de configuración AODV.nc. A través de estas interfaces se puede controlar o manipular el protocolo de encaminamiento desde el nivel superior.

- Control (StdControl): controla el funcionamiento general de protocolo. Permite inicializar, iniciar y parar el funcionamiento del mismo.
- TimerControl (StdControl): controla el funcionamiento de los Timers empleados en el protocolo. Esta separación es necesaria puesto que si parásemos el protocolo de comunicaciones (si se quiere anular las comunicaciones en un nodo) también pararíamos el funcionamiento global de los Timers utilizados en la aplicación.
- Receive: permite la recepción de mensajes. Incorpora identificador de puerto según aplicación indicada en el campo mhop.app de la cabecera de multisalto.
- Intercept: permite interceptar mensajes no dirigidos al nodo local. Esta función es adecuada a entornos donde se quiere monitorizar los nodos colindantes.
- SendMHopMsg: permite el envío de mensajes unicast hacia distintos nodos y en modo inundación hacía el conjunto de la red mediante control por número de saltos.
- SingleHopMsg, MultiHopMsg, AODVMSg: permiten acceder a los distintos campos de las cabeceras del mensaje.
- Payload: permite acceder a la posición de los datos dentro del mensaje con la finalidad de insertar la información necesaria.
- Reset: permite inicializar la tabla de rutas. Implementado para testear el funcionamiento del descubrimiento de rutas.

4.3.4. Interface USBShark

La interfaz USBShark es una aplicación realizada por Raúl Giménez, proyectista y compañero de laboratorio que nos facilitó la manera de capturar y visualizar el tráfico 802.15.4 entre nodos de nuestra red. Dicha aplicación se realizó de forma que fuese fácil realizar modificaciones en ella y de esta manera adaptarse perfectamente a las necesidades de otros proyectos como es nuestro caso particular.

A continuación se describen las principales características de esta aplicación para una mejor comprensión de su funcionamiento.

En la siguiente figura se puede observar el diagrama de bloques que nos detalla la relación entre clases utilizadas para implementar este software.

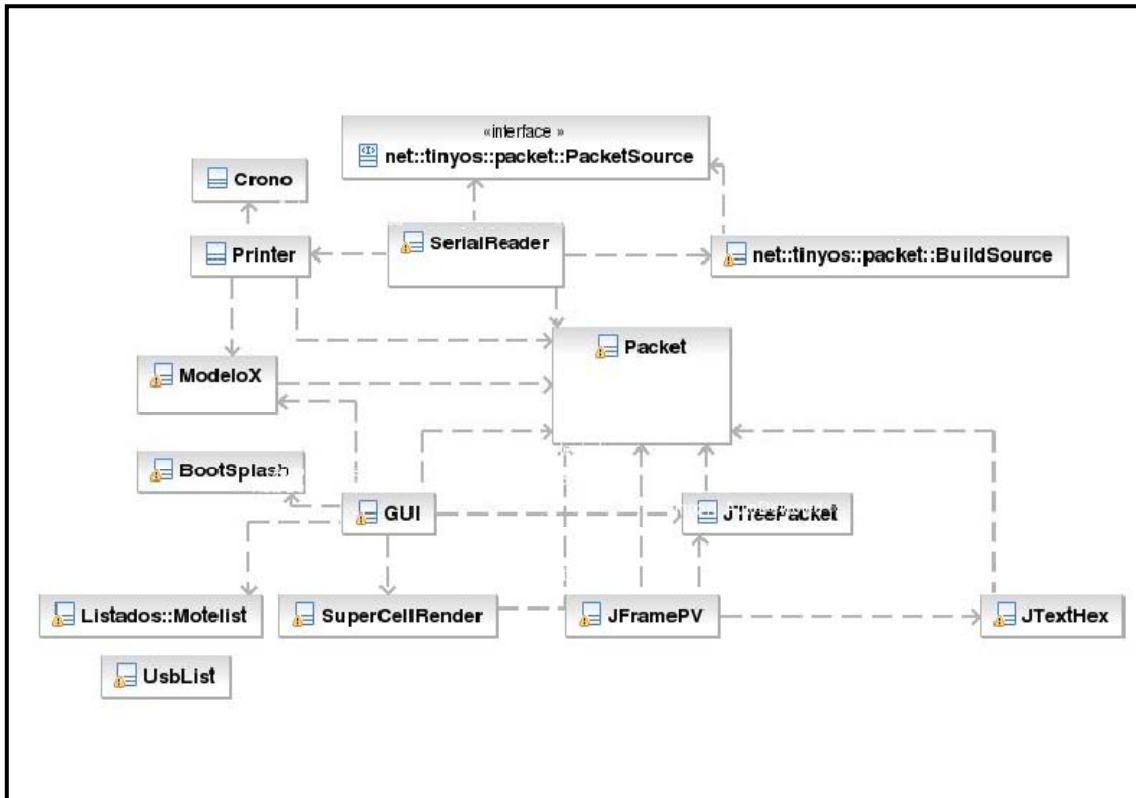


Fig. 4.11. Diagrama de bloques de dependencia entre clases

Como se puede observar, la clase principal, GUI, hace uso tanto de las diferentes vistas que crean la interfaz de usuario como son las JtreePacket, SuperCellRender y ModeloX, como de las clases necesarias para comunicarse con las interfaces Serial para listar todos puertos USB en uso. También se aprecia como el objeto Packet es usado por todas estas clases para extraer la información a mostrar.

En cuanto a la interfaz de usuario comentar que es muy amigable e intuitiva como se muestra en la siguiente figura.

The screenshot shows the main interface of the USBShark application. At the top, there are menu options (File, Devices, Graphs) and control buttons like 'Stop Capture' and 'Clear Table'. Below that, there are 'Filter' and 'Clear filter' buttons. A section for 'nstAODV Options' includes checkboxes for 'Route Request', 'Route Reply', 'Route Error', 'Data', and 'No nstAODV Packet'. The main part of the interface is a table with the following columns: 'Nº Packet', 'Time', 'Device Port', and 'Packet Description...'. The table contains 24 rows of data, all from the device port '/dev/ttyUSB0' and with a description of '[TEST HOPS] Saltos: X | PDR: Y | Seq Z'. Below the table, there is a detailed view of a 'Zigbee Frame 802.15.4 :: UNKNOWN TYPE', showing a 'FRAME :: TOS_Msg -> mote with a CC2420 radio, MicaZ, Telos rev.A, Te os rev.B, ...' and 'nstAODV SingleHop Details'.

Nº Packet	Time	Device Port	Packet Description...
121	84.435	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 120
122	84.883	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 66 Seq 121
123	85.383	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 66 Seq 122
124	87.425	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 123
125	87.809	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 124
126	88.305	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 125
127	88.783	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 126
128	89.636	/dev/ttyUSB0	[TEST HOPS] Saltos: 7 PDR: 100 Seq 127
129	89.775	/dev/ttyUSB0	[TEST HOPS] Saltos: 3 PDR: 100 Seq 128
130	90.272	/dev/ttyUSB0	[TEST HOPS] Saltos: 3 PDR: 100 Seq 129
131	91.039	/dev/ttyUSB0	[TEST HOPS] Saltos: 5 PDR: 66 Seq 130
132	91.871	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 66 Seq 131
133	92.708	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 132
134	93.183	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 133
135	93.683	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 134
136	94.158	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 135
137	94.654	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 66 Seq 136
138	95.134	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 66 Seq 137
139	95.632	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 138
140	96.128	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 139
141	96.619	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 140
142	97.104	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 50 Seq 141
143	97.6	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 100 Seq 142
144	98.083	/dev/ttyUSB0	[TEST HOPS] Saltos: 2 PDR: 66 Seq 143

Fig. 4.12. Pantalla principal de la aplicación USBShark

Como podemos observar, sigue bastante la estética y la forma de presentar los datos al conocido programa Ethereal proporcionando la siguiente información.

- *Nº Packet*: número de secuencia de paquete referenciando siempre al primero recibido.
- *Time*: la diferencia de tiempo con respecto a la recepción del primer paquete.
- *Device Port*: el puerto del que hemos capturado dicha información.
- *Packet Description*; como su nombre indica, se trata de una breve descripción del paquete. Este campo es modificable pudiéndose mostrar en él el/los bytes del paquete que se desee.

En cuanto al tema del almacenamiento de las estadísticas comentar que cada una de las pruebas realizadas ha sido exportada a un archivo CVS (*Comma Separated Values*). Con este método se puede utilizar los datos obtenidos desde cualquier hoja de cálculo como puede ser *MS Excel*, simplemente hemos de guardar la información que queramos separar por columnas entre comas y cada salto de línea indica el fin de una fila.

CAPÍTULO 5. ESTUDIO DEL PARAMETRO LQI

A tenor de lo expuesto en el capítulo 3, la calidad del enlace tiene una gran importancia en las propuestas de encaminamiento de redes multisalto. Por este motivo, se ha realizado un exhaustivo estudio para caracterizar el parámetro LQI, mediante el cual se calcula la calidad de un enlace, y estudiar su uso en métricas de encaminamiento.

5.1. Cálculo de LQI y RSSI con el chipcon CC2420.

Valores como el LQI, que indica la calidad de los paquetes recibidos, y RSSI, que indica la intensidad de señal recibida, los proporcionan las capas PHY (física) y MAC del IEEE 802.15.4, y son proporcionados por el chip radio CC2420 de los motes Telosb. La estructura de los paquetes 802.15.4 contiene dos bytes de FCS² que se comprueban en el extremo receptor. Cuando el paquete pasa a la capa MAC del receptor, este campo de dos bytes es reemplazado. El RSSI, medido sobre los primeros ocho símbolos que siguen al SFD³, ocupará el primer byte del FCS. El bit más significativo del segundo byte corresponde al CRC, que indica si la trama recibida es correcta. Y por último, los 7 bits menos significativos del segundo byte del FCS se reemplazan con el LQI, que se basa en un muestreo de la tasa de error de los primeros ocho símbolos de la cabecera de la trama recibida de la capa PHY.

El rango de valores de LQI es de 50 a 110, donde el valor 50 va asociado a la mínima calidad y 110 a la máxima. Los valores de RSSI obtenidos corresponden al valor del registro RSSI_VAL. Este valor contiene signo y por tanto al valor obtenido se deberá restar 256, puesto que las potencias recibidas siempre tienen signo negativo ya que el rango de potencias de transmisión en Telosb va de -25 dBm a 0 dBm. Para obtener el valor real de potencia recibida, se debe utilizar la fórmula $P_{RX} = RSSI_VAL + RSSI_OFFSET$ [dBm], donde el manual del chip CC2420 indica usar un valor de offset de -45.

5.2. Escenario de pruebas

Para caracterizar el parámetro LQI se han realizado una serie de pruebas con el fin de conocer el impacto de diferentes factores como la distancia, las interferencias, la orientación de los nodos, etc sobre este parámetro y de esta manera conocer en detalle su compartamiento.

Para la realización de estas pruebas el escenario consta de 60 dispositivos Telosb ubicados en una cuadrícula de madera a 2,5 metros del suelo, a la cual a partir de ahora llamaremos grid, con una separación entre dispositivo de 90 cm y orientación al azar. Cada nodo está conectado mediante una red USB a un PC con el software especificado en el apartado 4.3. de este documento.

² FCS: Frame Check Sequence.

³ FSD: Frame Start Delimiter.

En la siguiente figura se observa la representación del grid de pruebas.

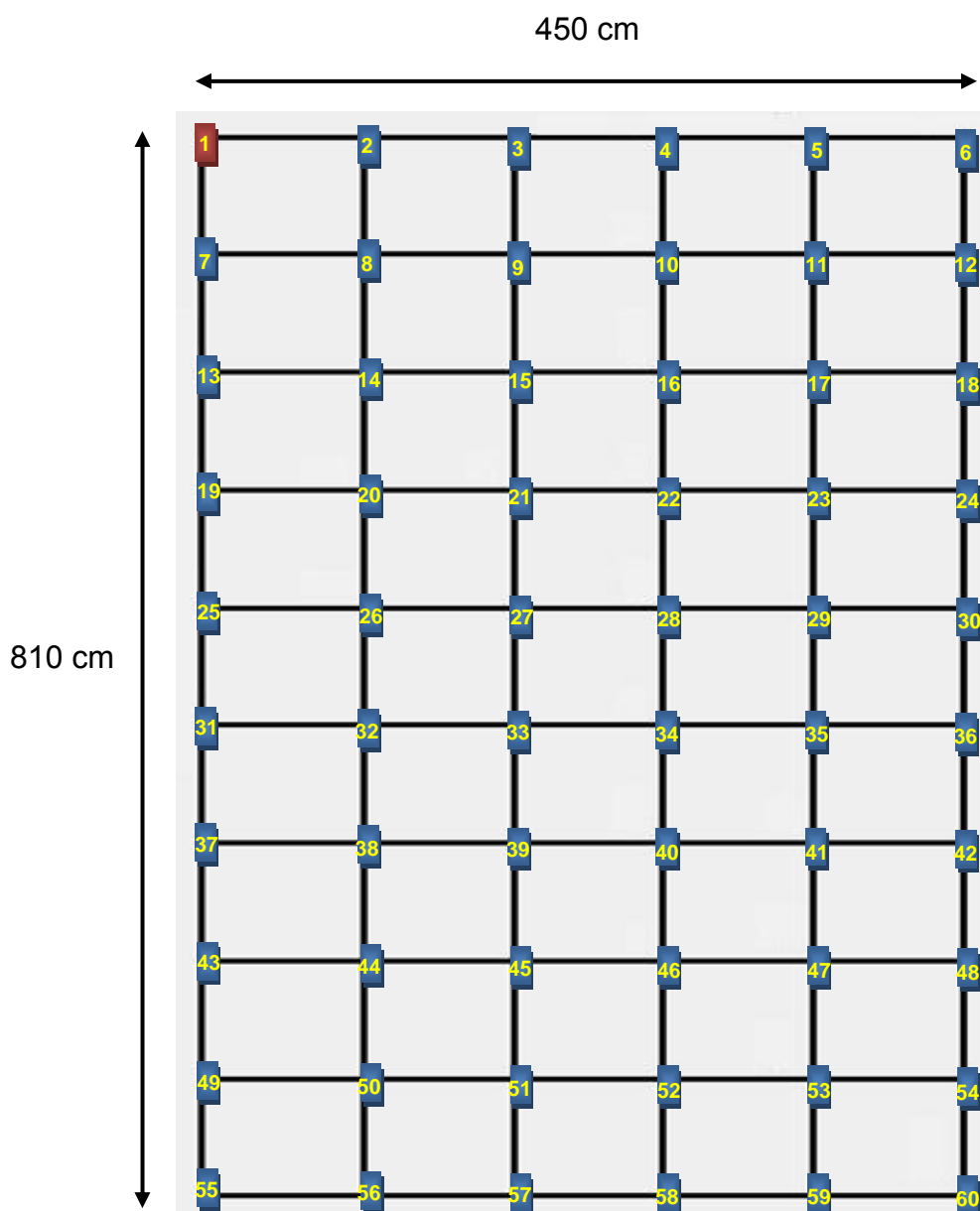


Fig. 5.1. Representación del grid.

Cabe comentar también que las pruebas se han realizado en su totalidad por la tarde en días laborables, en la franja de las 16:00-21:00 y con personas en el laboratorio, factores que han podido influir en las medidas tomadas.

Una vez detallado el escenario de las pruebas, a continuación se pasa a explicar con detalle las medidas obtenidas en distintas pruebas realizadas para caracterizar y conocer el comportamiento del LQI como indicador de la calidad de un enlace.

Cabe comentar que todos los experimentos constan de 1000 transmisiones a partir de las cuales se extraen resultados estadísticos.

5.3. Estudio del impacto de la orientación y la distancia de los motes

En este apartado se muestran las pruebas cuyo objetivo es conocer el impacto causado tanto por la orientación de la antena de los motes telosb como por la distancia.

5.3.1. Caracterización de una fila con los motes sin orientación fija

Para realizar esta prueba el emisor se sitúa en la esquina superior izquierda, (véase fig. 5.1), y los receptores, a lo largo de su misma fila⁴ a distancias múltiples de 90 cm ejecutando una aplicación tipo “ping” a potencia 0x03 en el canal 26.

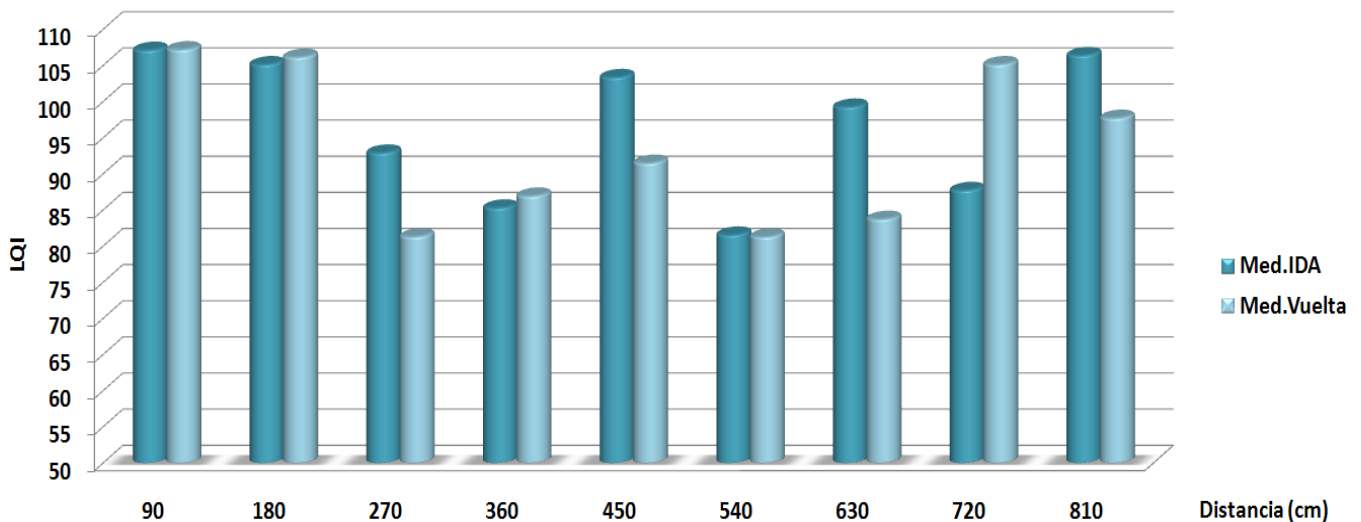


Fig. 5.2. Prueba fila 1

Como se observa en la fig. 5.2 no existe tendencia creciente o decreciente clara del LQI en ninguno de los dos sentidos de los enlaces respecto a la distancia. Otro aspecto destacable que se aprecia es el fenómeno conocido como asimetría.

Viendo este resultado y barajando diferentes explicaciones para este fenómeno se apostó por estudiar el diagrama de radiación de la antena que incorpora los motes Telosb pensando que la orientación de éstos sería un factor muy importante a tener en cuenta para la realización de futuras pruebas.

Por dicha razón la siguiente prueba realizada es el estudio del posible impacto que puede tener la orientación de los motes sobre la calidad de los enlaces.

⁴ Se entiende como fila el lado del grid que mide 810 cm y que comprende por ejemplo la línea de dispositivos formada entre los motes 1 y 55.

5.3.2. Caracterización de la antena de los motes Telosb

Como se ha comentado anteriormente es importante conocer con detalle el comportamiento de la antena que incorporan los motes Telosb utilizados. Para ello, se ha realizado una serie de pruebas para conocer el diagrama de radiación de la antena F-invertida [21], que es el tipo de antena de que disponen los motes. Para ello se ha tenido en cuenta ambos planos, el horizontal y el vertical⁵ para extraer un diagrama completo. Para realizar las pruebas de ambos planos se ha colocado un transmisor ubicado en el centro del grid y siempre en la misma posición, enviando 1000 paquetes con una tasa de 3 paq/s y potencia 0x02 y un receptor situado a aproximadamente 127 cm. el cual se iba variando su posición 45° respecto el transmisor.

5.3.2.1. Plano horizontal

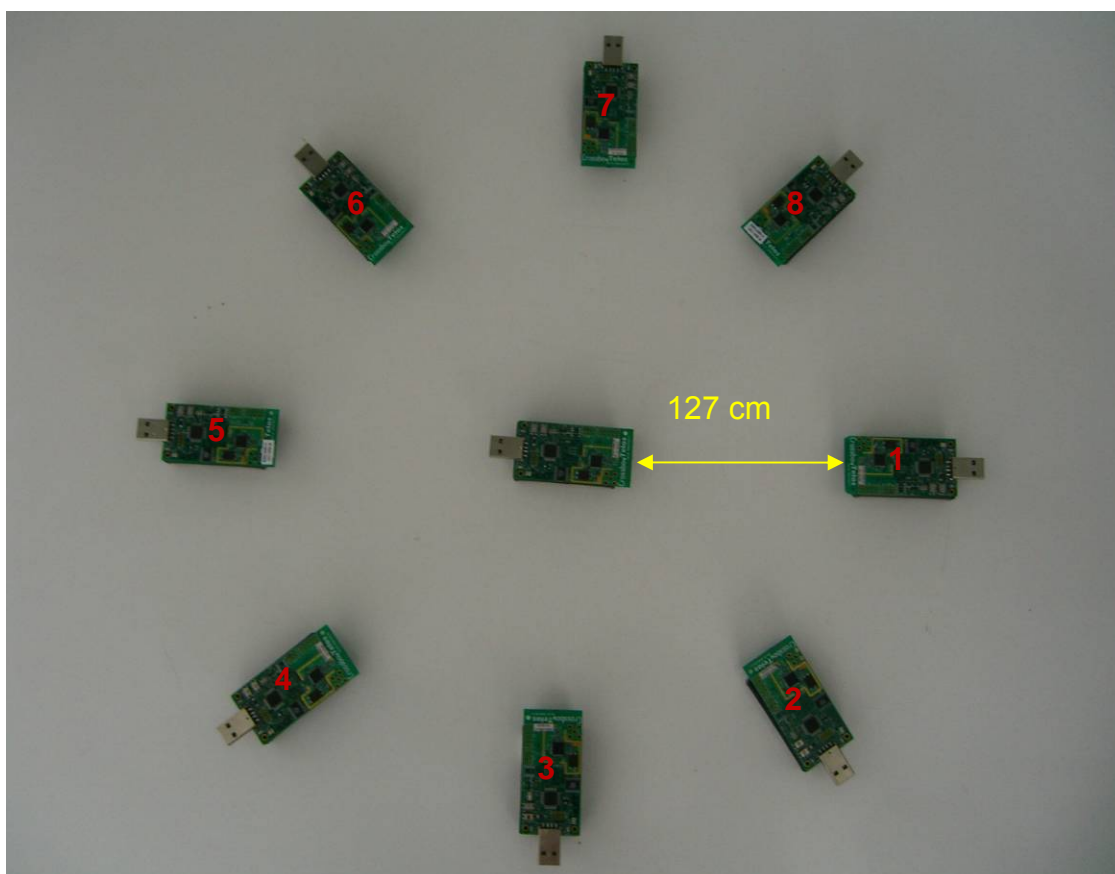


Fig. 5.3. Distribución motes, prueba radiación plano horizontal.

Los resultados obtenidos son los siguientes:

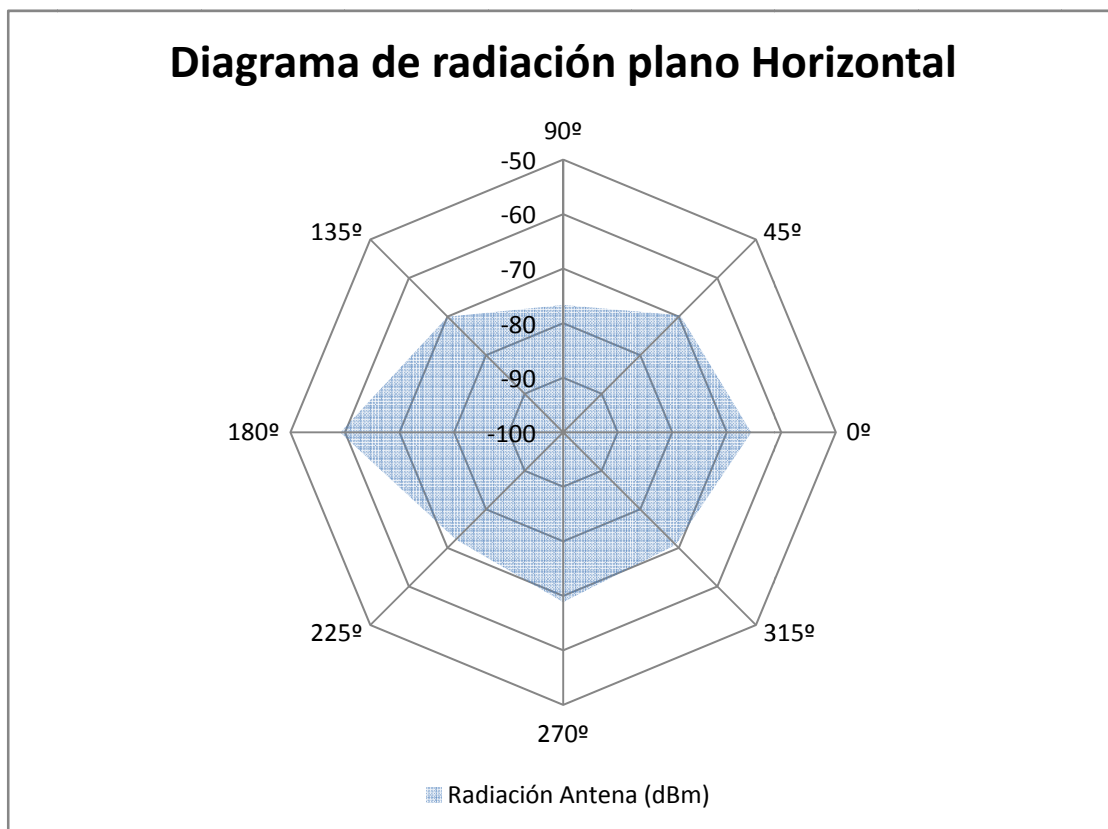
⁵ Horizontal y vertical, en este proyecto, se definen como muestran las Fig. 5.3 y Fig. 5.6.

Tabla 5.1. Resultados de radiación plano horizontal.

HORIZONTAL	nº prueba	Ángulo(º)	LQI _{medio}	RSSI _{medio} (dBm)	Paquetes perdidos
	1	0	106,91	-65,5	0
	2	315	102,69	-70,85	1
	3	270	105,55	-68,86	0
	4	225	105,98	-72,22	0
	5	180	107,13	-59,13	0
	6	135	106,81	-70	0
	7	90	103,55	-76,62	1
	8	45	105,73	-69,65	0
Media			105,54375	-69,10375	0,25

Como se observa en la tabla anterior, el número de prueba en el que mayor intensidad de señal se recibe es en la número 5, en la cual se recibe una potencia de señal de -59.13 dBm y un LQI medio de 107.13. Por otra parte, se comprueba que, en efecto, el diagrama no corresponde al de una antena omnidireccional.

A partir de los resultados obtenidos se ha podido realizar el diagrama de radiación experimental de la antena de los motes. En la siguiente figura se puede observar el plano horizontal de dicho diagrama.

**Fig. 5.4.** Diagrama de radiación del plano horizontal.

Por lo tanto, observando la tabla de resultados y el diagrama de radiación de la antena, se considera que la mejor posición para colocar los motes en el plano horizontal es la que se observa en la fig. 5.5.

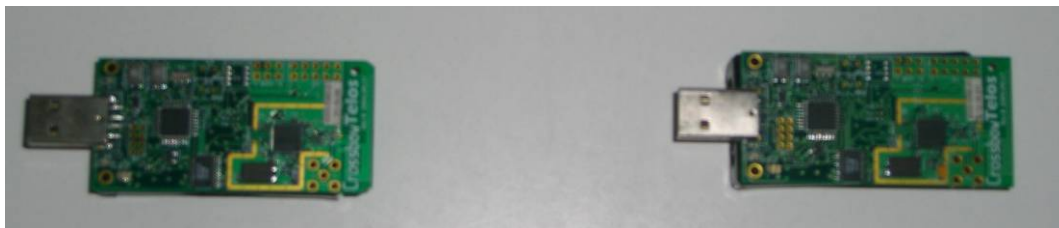


Fig. 5.5. Orientación óptima plano horizontal.

5.3.2.2. Plano vertical

En la siguiente figura se muestra el escenario para realizar la prueba en el plano vertical.

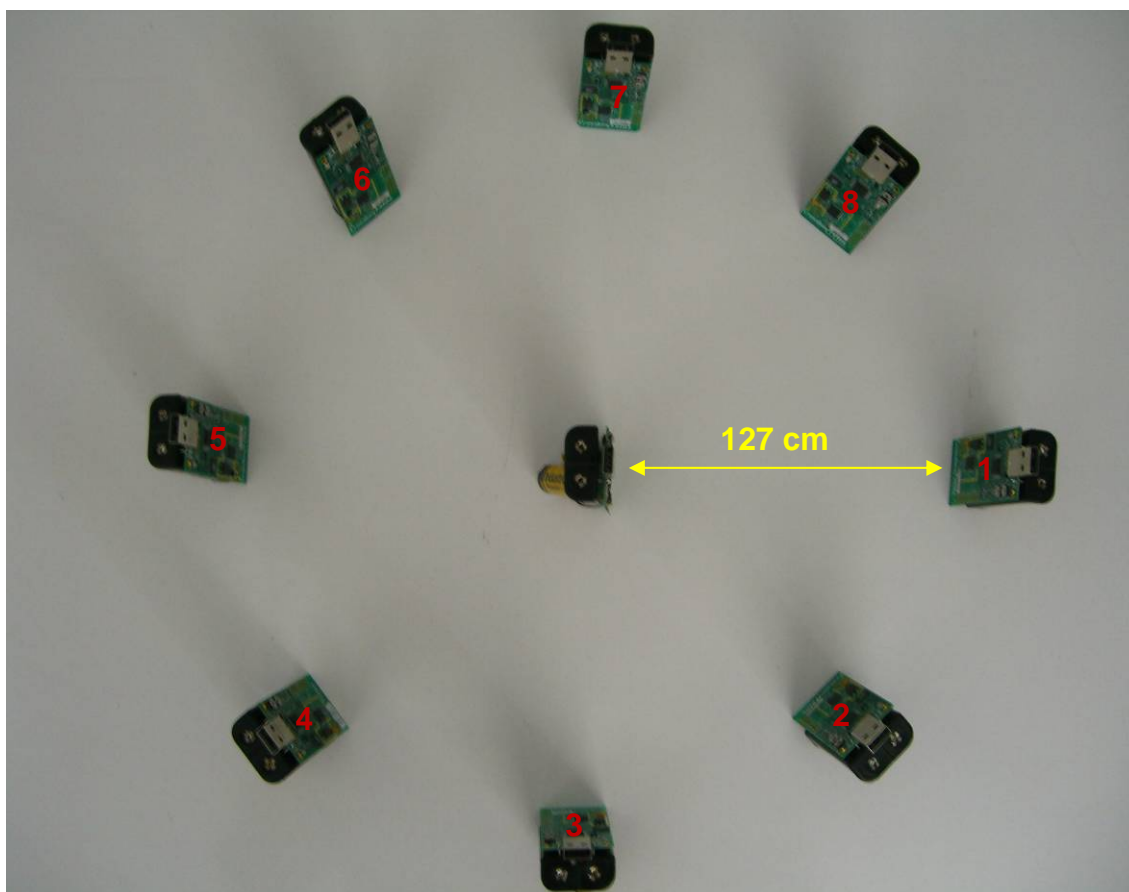


Fig. 5.6. Distribución motes, prueba radiación plano vertical.

Los resultados obtenidos se presentan en la siguiente tabla.

Tabla 5.2. Resultados de radiación plano vertical.

VERTICAL	nº prueba	Ángulo(º)	LQI _{medio}	RSSI _{medio} (dBm)	Paquetes perdidos
	1	0	106,6	-67,7	0
	2	315	92,02	-75,04	2
	3	270	102,92	-69,89	1
	4	225	106,83	-67,92	4
	5	180	102,43	-76,79	11
	6	135	105,21	-66,74	4
	7	90	106,73	-62,15	1
	8	45	104,48	-66,74	4
Media			103,4025	-69,12125	3,375

En este caso el número de prueba con mejores resultados es la prueba 7 en la que se ha obtenido un RSSI de -62,15 y un LQI de 106,73.

En el plano vertical el diagrama de la antena es el siguiente:

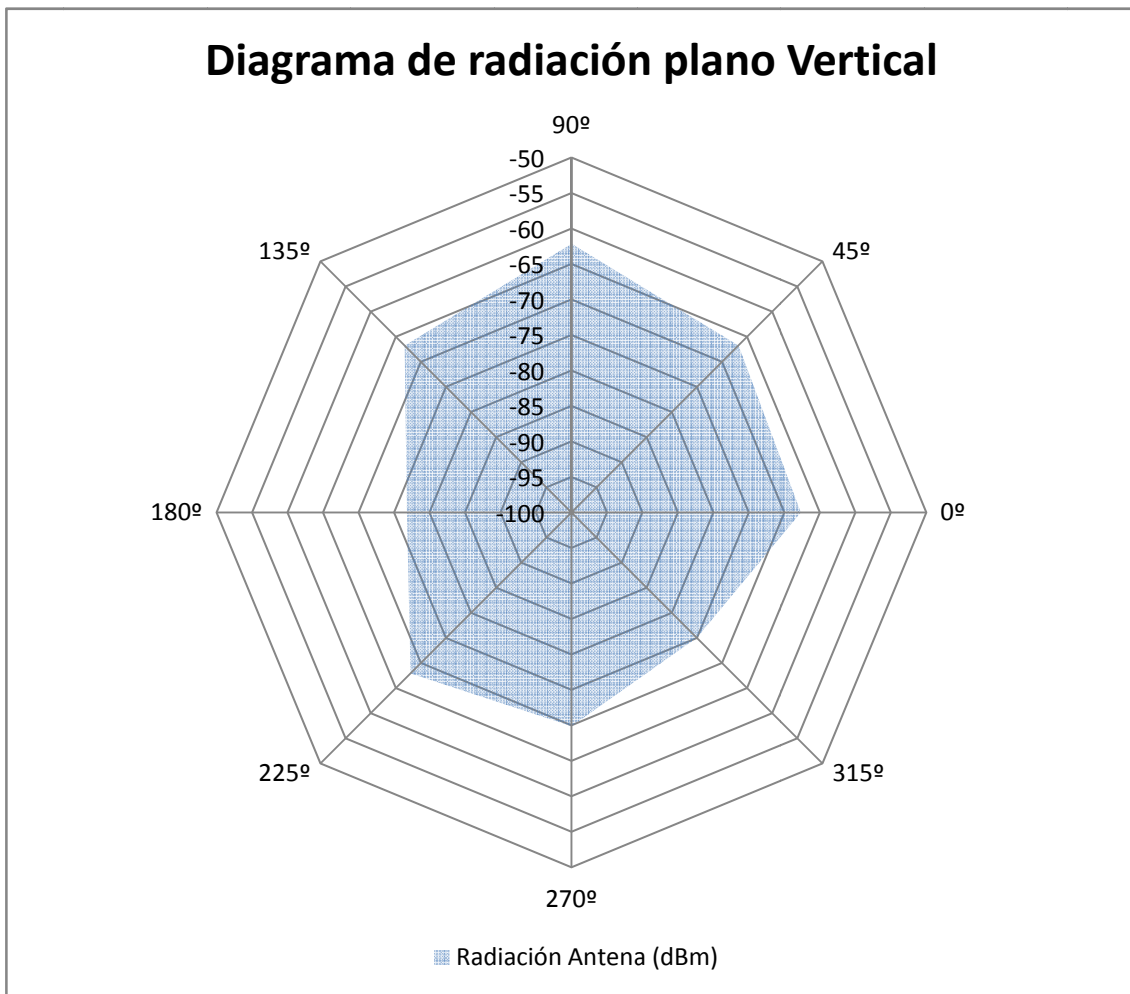


Fig. 5.7. Diagrama de radiación del plano vertical.

En este caso la mejor posición de los motes es la siguiente.

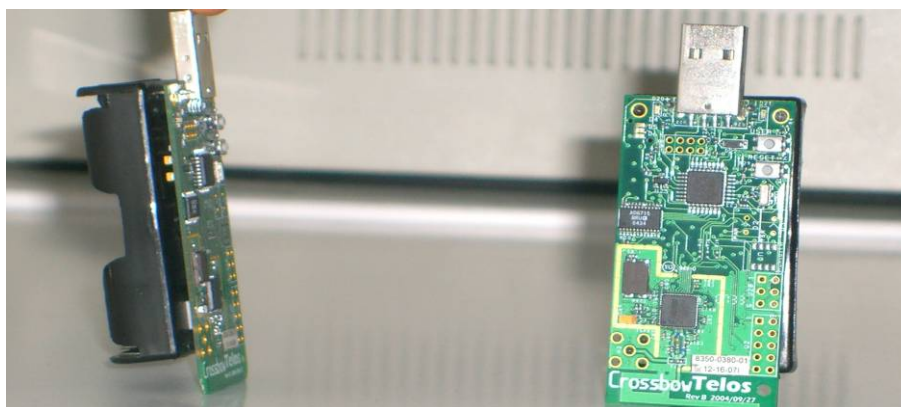


Fig. 5.8. Orientación óptima plano vertical.

Por lo tanto una vez visto los resultados obtenidos en las pruebas anteriormente detalladas y teniendo en cuenta la geometría del grid, y las posibles posiciones en que se podían fijar los motes, se concluyó que la mejor posición para colocarlos era situarlos en el plano horizontal encarando la antena de uno de ellos hacia el conector USB del otro tal y como se ha mostrado anteriormente en la Fig. 5.5.

5.3.3. Caracterización de una fila con orientación fija de los motes.

Una vez determinada la orientación de los motes, se dispuso a fijar toda la fila de tal manera que los motes de ésta tuviesen siempre la misma posición y orientación.

Una vez realizada esta labor se hizo una prueba (véase fig. 5.9) con la misma metodología que la ejecutada en el apartado 5.3.1., con el fin de comprobar que realmente el impacto de la orientación en los motes es el esperado.

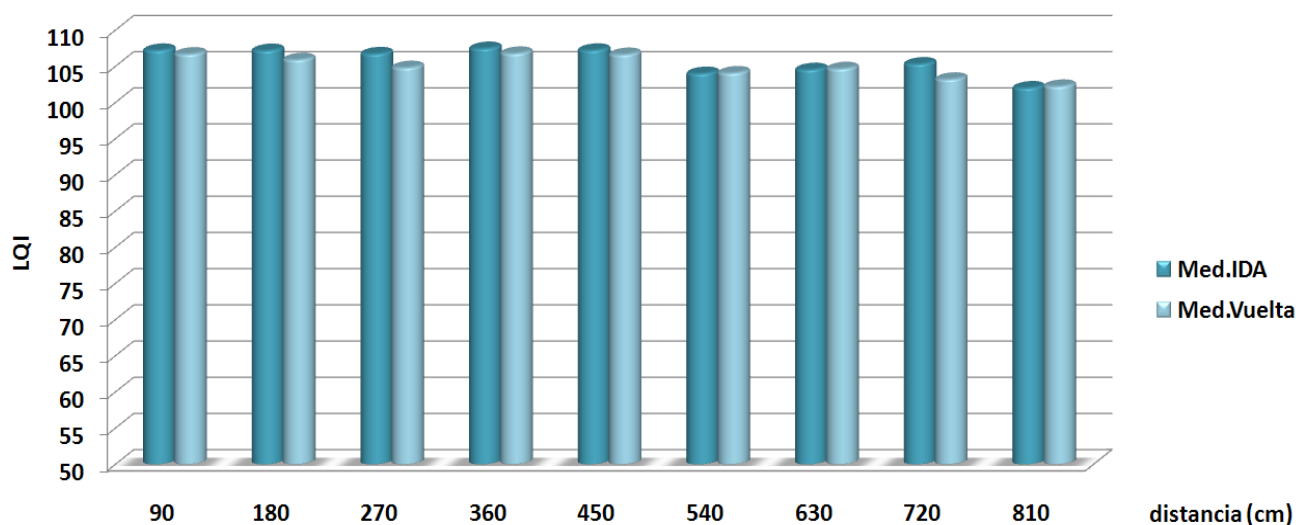


Fig. 5.9. Prueba fila 1 con orientación fija.

Efectivamente, viendo los resultados y comparándolos con los obtenidos en el apartado 5.3.1. donde los motes estaban sueltos sin una orientación fija, se observa una notable mejora tanto en los valores del LQI como en la simetría de los sentidos de los enlaces. Cabe comentar que el LQI muestra una tendencia decreciente a medida que se aumenta la distancia entre motes, aunque se aprecie puntos donde no se sigue dicha tendencia, posiblemente por causas de la propagación radio como el multicamino u otros efectos que se escapan de nuestro control.

Verificada la clara mejora de los resultados se dispuso a fijar los motes de todo el grid de tal manera que hubiese el mayor número de éstos en dicha posición y que los restantes no estuvieran colocados de forma inadecuada, es decir, en una posición poco óptima, para de esta manera fijar los motes de forma regular y obtener, por lo tanto, resultados coherentes en pruebas futuras.

Se optó por la siguiente distribución de los motes en el grid:

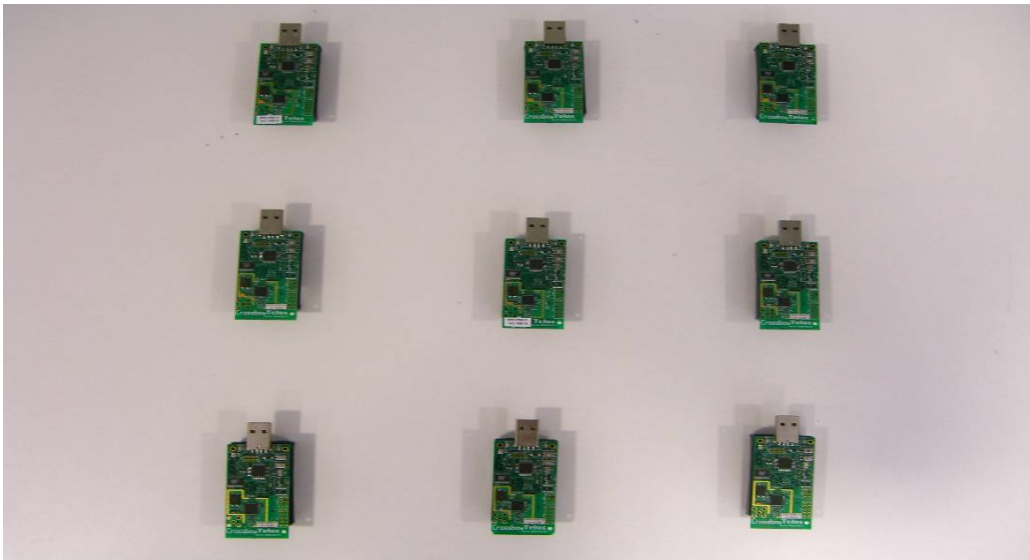


Fig. 5.10. Distribución de los motes en el grid.

Una vez fijados todos los motes en el grid, se realizaron pruebas para caracterizarlo. Estas pruebas consistían en transmitir flujos de datos entre motes alineados en una fila (entre los motes 1-55) y motes alineados en una columna (entre los motes 1-6), con el fin de caracterizar el comportamiento de las transmisiones en ambas direcciones del plano (fila y columna). En la siguiente tabla se resumen los resultados obtenidos:

Tabla 5.3. Resultados fila-columna entre motes fijados y sueltos.

	FLUJOS	LQI	Tasa entrega (%)
1--6	SUELTO	99,7	89,1
	FIJO	105,6	92,12
1--55	SUELTO	92,5	65,7
	FIJO	104,2	81,6

Como se puede comprobar en la tabla anterior, al fijar los motes como en la fig. 5.10 se consigue una mejora en el LQI respecto los resultados obtenidos sin orientación fija.

A lo largo de una fila, con la potencia de transmisión empleada, se obtiene un 81.6% de tasa de entrega entre emisor y receptor a 8,1 m. de distancia, mientras que a lo largo de una columna, la tasa de entrega con transmisor y receptor a 4,5 m. de distancia es de 92.12%.

5.4. Estudio del impacto de interferencias en los motes

En este apartado se estudia el impacto de las interferencias en la calidad de los enlaces. Para ello, se dispusieron diversos motes interferentes situados a una distancia suficiente para que la señal emitida por éstos no compitiera por el medio con el mote emisor, sino que fuese percibido como ruido por el mote receptor.

Para ello, se empezó situando los motes interferentes lo más alejado posible del mote transmisor, pero se observó que no había consecuencias notables en los resultados obtenidos, por lo que se fueron aproximando para observar posibles efectos. A continuación se describen algunas de las pruebas realizadas que resultan significativas:

La primera prueba se ha realizado poniendo un mote transmisor emitiendo a una tasa de 3 paq/seg y potencia de transmisión de -20dBm al lado de un mote receptor y 3 motes interferentes lo más alejado posible de éstos, por la razón anteriormente expuesta, emitiendo a una tasa mucho más elevada, en concreto a 142 paq/seg a la misma potencia que el emisor, para en caso de tener resultados, que fuesen significativos.

TX	RX				
					INTER
				INTER	INTER

Fig. 5.11. Distribución de los motes en la primera prueba de interferencias.

Para realizar esta prueba se transmitieron 1000 paquetes desde el emisor, los 300 primeros sin ninguna interferencia en el grid y los restantes con los motes interferentes activados. Esta metodología ha sido la seguida en todas las pruebas de este apartado.

Los resultados obtenidos son los siguientes:

Tabla 5.4. Resultados primera prueba interferencias.

	LQI	RSSI
Sin interferencias	106,07	-71,03
Con interferencias	105,96	-71,28

Como se observa en la tabla la diferencia entre no tener y tener interferencias mientras se transmite es poco significativa. Al comprobar los resultados obtenidos se aumentó el número de motes interferentes a 9 dando lugar a la siguiente prueba cuyo escenario es el que se muestra a continuación.

TX	RX				
			INTER	INTER	INTER
			INTER	INTER	INTER
			INTER	INTER	INTER

Fig. 5.12. Distribución de los motes en la segunda prueba de interferencias.

Cabe comentar que los parámetros de configuración de los motes son los mismos que en la prueba anteriormente comentada.

En esta prueba los resultados obtenidos son los siguientes:

Tabla 5.5. Resultados segunda prueba interferencias.

	LQI	RSSI
Sin interferencias	105,76	-72,53
Con interferencias	105,65	-72,71

Observando estos resultados se puede afirmar que, como en el caso anterior, la diferencia entre las medidas sin interferencias y con ellas son prácticamente negligibles.

Por ello se decidió aproximar los motes interferentes al enlace entre emisor y receptor para estudiar el efecto de la interferencia en los enlaces de comunicación con la disposición que se muestra a continuación.

TX	RX			INTER	
				INTER	
				INTER	
				INTER	
INTER	INTER	INTER	INTER	INTER	

Fig. 5.13. Distribución de los motes en la tercera prueba de interferencias.

Los resultados en este caso son los que se muestran en la siguiente tabla:

Tabla 5.6. Resultados tercera prueba interferencias.

	LQI	RSSI
Sin interferencias	105,93	-71,47
Con interferencias	105,54	-72,36

Viendo los resultados obtenidos en esta prueba se evidencia que, de nuevo, la interferencia no afecta de una forma clara a la calidad de los enlaces.

Observando en global los resultados obtenidos se puede afirmar que la interferencia entre motes TelosB no degrada la calidad de los enlaces. En consecuencia, el uso del parámetro LQI en métricas de encaminamiento no se vería afectado por la presencia de flujos disjuntos simultáneos.

5.5. Relación LQI-LDR

La relación entre LQI-LDR se obtuvo del trabajo fin de carrera de Raúl Giménez “Evaluación de un protocolo de encaminamiento en redes multsalto IEEE 802.15.4” [20], puesto que el escenario utilizado en este proyecto es el mismo

que el usado por este proyectista en su TFC (hardware, software, laboratorio, etc..).

La relación entre LQI y LDR es la siguiente:

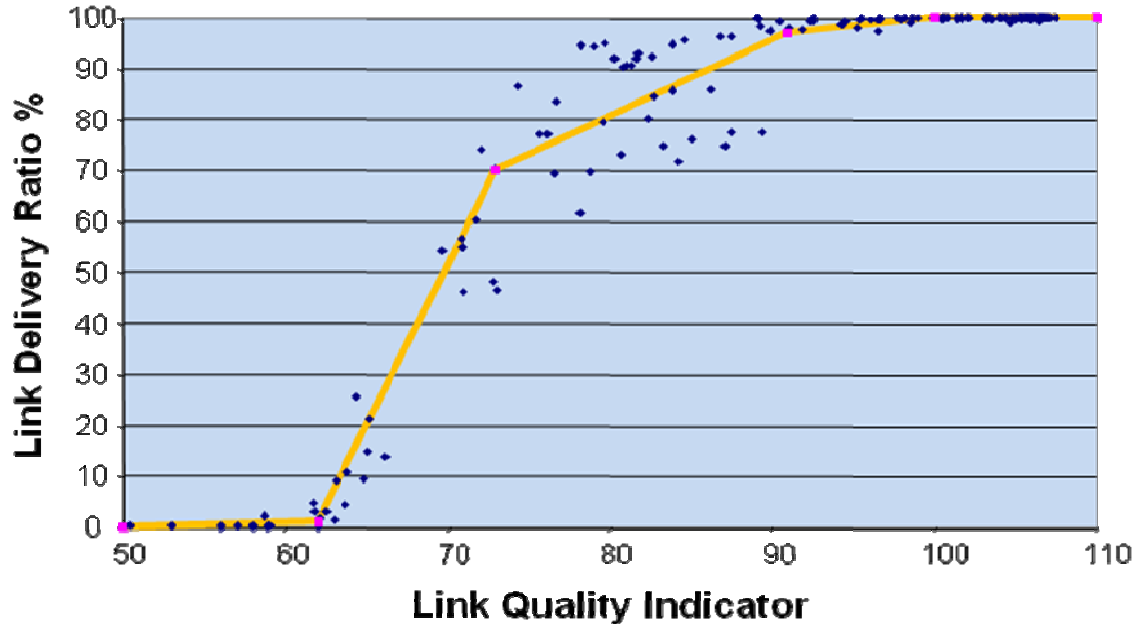


Fig. 5.14. Aproximación entre LQI y LDR.

Se puede extraer una aproximación lineal por partes del LDR de un enlace a partir del LQI de ese enlace. Esta aproximación tendrá mayor error para valores de LQI de la franja intermedia comprendida en el rango de 62 a 90, debido a su variabilidad con el tiempo, puesto que en dicha franja, el parámetro LQI presenta un comportamiento inestable.

La aproximación del LDR a partir del LQI se efectúa mediante cinco rectas cuyas ecuaciones son las siguientes:

$$y = 1 \quad x \in [50, 62] \quad (5.1)$$

$$y = \frac{70}{11}x - \frac{4340}{11} \quad x \in [63, 73] \quad (5.2)$$

$$y = \frac{3}{2}x - \frac{79}{2} \quad x \in [74, 91] \quad (5.3)$$

$$y = \frac{1}{3}x + \frac{200}{3} \quad x \in [92, 100] \quad (5.4)$$

$$y = 100 \quad x \in [101, 110] \quad (5.5)$$

Esta aproximación es utilizada para la implementación y el estudio de las distintas métricas evaluadas en el siguiente capítulo.

CAPÍTULO 6. EVALUACIÓN DE LAS MÉTRICAS DE ENCAMINAMIENTO

En este capítulo se presenta la implementación de las diferentes métricas de encaminamiento que se han programado sobre nst-AODV así como una exhaustiva evaluación de tales métricas.

Las métricas evaluadas son la del mínimo número de saltos (HOPS), la métrica PATH-DR (PDR), una métrica propuesta por el grupo ZigBee Alliance (ZigBee) y por último la métrica Expected Transmission Count (ETX), todas ellas explicadas en el tercer capítulo del proyecto.

6.1. Implementación de las distintas métricas.

En este apartado se detallan los cambios realizados y la manera como se ha implementado cada una de las métricas evaluadas en el protocolo nst-AODV.

6.1.1. Métrica HOPS.

Esta métrica viene implementada por defecto en el protocolo nst-AODV, por lo que no fue necesario realizar ninguna modificación del mismo.

6.1.2. Métrica PDR.

Para implementar esta métrica se han realizado diferentes cambios en diversos módulos del protocolo nst-AODV. En el modulo AODV_Core.nc se ha creado la función *convertLqiToLdr*, que se encarga de convertir el valor obtenido de LQI, en el enlace, a LDR según la aproximación de correlación descrita en el apartado 5.5.

```
command uint8_t convertLqiToLdr (uint8_t lqi){
    float ldrDec=0;
    uint8_t ldr;
    if ((lqi < 50) || (lqi > 110)){
        ldr = 0; } /*invalid ldr*/
    else{
        if ((lqi >= 50) && (lqi <= 62)){
            ldrDec = 1;}
        else if ((lqi > 62) && (lqi <= 73)){
            ldrDec = ((70*lqi)-4340)*1/11;}
        else if ((lqi > 73) && (lqi <= 91)){
            ldrDec = ((3*lqi)-79)*1/3;}
        else if ((lqi > 91) && (lqi <= 100)){
            ldrDec = (lqi+200)*1/3;}
        else if ((lqi > 100) && (lqi <= 110)){
            ldrDec = 100;}
        ldr = floor(ldrDec);
    }
    return ldr;
}
```

Fig. 6.1. Función *convertLqiToLdr* del modulo AODV_Core.nc.

Esta función verifica que el parámetro de LQI esté dentro del rango válido de 50 a 110 y devuelve el valor de LDR correspondiente al enlace en cuestión. Cabe comentar que la función `convertLqiToLdr` ha sido empleada en todas las métricas estudiadas exceptuando la métrica HOPS.

Otro cambio implementado en esta métrica es el realizado en el archivo `AODV_Msg`, donde se ha añadido un campo tanto al paquete RREQ como al paquete de respuesta RREP.

```
typedef struct AODV_Rreq_Msg{

    uint8_t dest;           // the destination desired for the RREQ
    uint8_t src;           // identifies the RREQ with the rreqID
    uint16_t rreqID;       // seq# identifies the RREQ with the src
    uint16_t srcSeq;       // seq# used for storing entries back to
                           // the source
    uint16_t destSeq;      // seq# last received from the destination
                           // by the source
    uint8_t metric[1];     // the number of hops from the Originator
                           // IP address
    uint8_t metric_pdr;   // Pdr value from the source.

    uint8_t flag;
    uint8_t data[1];
} AODV_Rreq_Msg;
```

Fig. 6.2. Estructura del mensaje RREQ.

El campo añadido `metric_pdr` contiene el valor acumulado del coste de cada uno de los enlaces por los que ha pasado el paquete RREQ en busca del nodo destino. Este valor se inicializa a 100 en el nodo origen y se recalcula en cada nodo intermedio, multiplicándolo por el valor de LDR del enlace en cuestión y dividiéndolo por 100 hasta llegar al nodo destino, tal y como se observa en la siguiente figura:

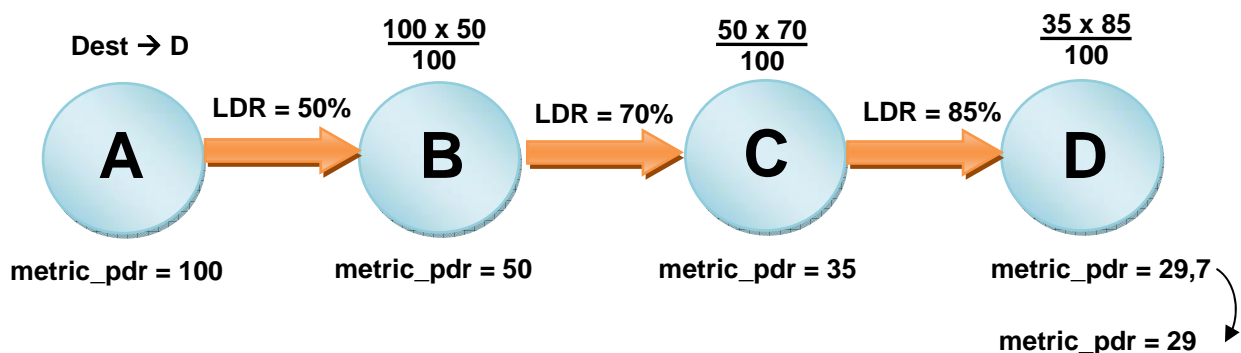


Fig. 6.3. Ejemplo de cálculo del campo `metric_pdr`.

En el caso del paquete RREP, se ha creado un campo en el mensaje llamado `metric_pdr` en el que se guarda el valor del coste total del camino entre nodo origen y nodo destino.

```

typedef AODV_Rreq_Msg* AODV_Rreq_MsgPtr;

typedef struct AODV_Rreply_Msg{
    uint8_t dest;           // the destination of the route supplied
    uint8_t src;           // the source of the route supplied
    uint16_t destSeq;      // seq# associated with the route
    uint8_t metric[1];     // The number of hops from the src to the dest

    uint8_t metric_pdr;   // Pdr value of the route.

    uint8_t flag;
    uint8_t data[1];
} AODV_Rreply_Msg;

```

Fig. 6.4. Estructura del mensaje RREP.

Este campo no se recalcula por ningún nodo intermedio, solo indica el coste del camino, valor a partir del cual el nodo origen decide cual es la mejor ruta hasta el destino.

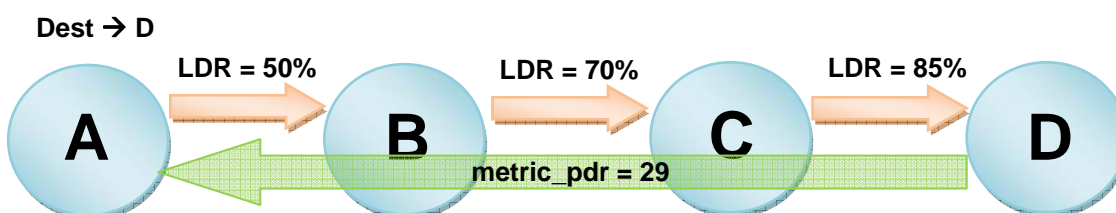


Fig. 6.5. Ejemplo del campo `metric_pdr` en el mensaje RREP.

Al añadir un campo en la estructura del mensaje RREQ y RREP, es necesario modificar el módulo `AODV_Table` añadiéndole el nuevo campo en las tablas y modificar el algoritmo de elección de ruta ya que al implementar la métrica PDR se elige como ruta la que mayor PDR tenga.

La estructura de las tablas queda de la siguiente forma:

```

typedef struct {
    uint8_t dest;
    uint8_t nextHop;
    uint16_t destSeq;
    uint8_t numHops;

    uint8_t metric_pdr;

    uint16_t lifetimeSeq;
    uint8_t flag;
} AODV Route Table;

```

```

typedef struct {
    uint8_t dest;
    uint8_t src;
    uint8_t nextHop;
    uint16_t rreqID;
    uint16_t destSeq;
    uint8_t numHops;

    uint8_t metric_pdr;
} AODV_Route_Cache

```

Fig. 6.6. Estructura de la tabla de rutas y cache.

El algoritmo de elección de ruta implementado para la métrica PDR, está ubicado en el módulo `AODV_Table` dentro de la función `AODV_Route.updateRouteInfo`. A continuación se muestra únicamente la parte de la función donde se realiza esta elección:

```

i = getRTableIndex(dest);

if ((routeTable[i].destSeq==INVALID_INDEX)&&
    (routeTable[i].metric_pdr==INVALID_INDEX)){

//the route is incorrect

}else if ((routeTable[i].destSeq >= destSeq) || ((routeTable[i].destSeq ==
destSeq) && (metric_pdr > routeTable[i].metric_pdr))){

//new route with more PDR

}else
// it is not a correct route, we don't need to update data.
    return SUCCESS;
}

//we add the new route.

routeTable[i].dest          = dest;
routeTable[i].destSeq      = destSeq;
routeTable[i].nextHop      = nextHop;
routeTable[i].numHops      = numHops;

routeTable[i].metric_pdr = metric_pdr;

routeTable[i].lifetimeSeq = mySeq;
routeTable[i].flag        = flag;

```

Fig. 6.7. Algoritmo de elección de ruta.

Este algoritmo consiste en comprobar la información contenida en la tabla de ruta y compararla con la información recibida en cada paquete RREP. Si ésta es mejor, la ruta es sustituida, en caso contrario, se descarta el paquete.

Hay que comentar que todas las funcionalidades del protocolo nst-AODV como pueden ser borrar ruta, buscar ruta, actualizar tabla, etc siguen el mismo procedimiento que con la métrica HOPS pero empleando el nuevo campo añadido como criterio.

6.1.3. Métrica ZigBee.

Para implementar esta métrica se han realizado cambios similares a los expuestos anteriormente en la métrica PDR, es decir, añadir el campo `metric_ZigBee` en los paquetes de control, modificar las tablas de rutas e implementar el algoritmo de elección de ruta, muy similar al de la métrica PDR, por lo que la explicación se centra únicamente en los aspectos novedosos de esta métrica.

La implementación más significativa es la de convertir el LDR proporcionado por la función *convertLqiToLdr* del modulo *AODV_Core.nc* vista en el apartado anterior al coste ZigBee definido mediante la fórmula 3.3.

```

command uint8_t convertLdrToZig (uint8_t ldr){
    uint8_t zig;

    if (ldr > 84){
        zig = 1;
    }
    else if (ldr > 77){
        zig = 2;
    }
    else if (ldr > 70){
        zig = 3;
    }
    else if (ldr > 67){
        zig = 4;
    }
    else if (ldr > 64){
        zig = 5;
    }
    else if (ldr > 62){
        zig = 6;
    }
    else{
        zig = 7;
    }

    return zig;
}

```

Fig. 6.8. Implementación métrica ZigBee.

El comando *convertLdrToZig* está implementado en el módulo *AODV_Core.nc*, consiguiendo de esta forma el coste ZigBee relacionado al LDR del enlace en cuestión. Esta métrica se puede tomar como una cuantificación de 7 niveles de la métrica PDR para un enlace en cuestión.

6.1.4. Métrica ETX.

Igual que en el apartado anterior los cambios realizados para implementar esta métrica son muy parecidos a los comentados con anterioridad, por esta razón, a continuación, solo se detalla la manera de implementar la métrica ETX propiamente a partir de la fórmula 3.2.

Pero antes de mostrar el código se explica la manera como se ha llegado a la fórmula final mostrada en la siguiente figura.

Recuérdese la fórmula de la métrica ETX:

$$ETX_i = \frac{1}{(1 - P_{fi})(1 - P_{ri})} \quad \begin{array}{l} ETX_i : \text{Coste ETX del enlace } i. \\ P_{fi} : \text{Probabilidad de error enlace ida.} \\ P_{ri} : \text{Probabilidad de error enlace vuelta.} \end{array} \quad (6.1)$$

Se estima la probabilidad de error, tanto del camino de ida como del camino de vuelta de la siguiente forma:

$$P_x = (1 - LDR_x) \quad LDR_x : \text{Tasa de entrega enlace } x. \quad (6.2)$$

Por lo tanto la fórmula general ya simplificada es la siguiente:

$$ETX_i = \frac{1}{LDR_{fi} \times LDR_{ri}} \quad \begin{array}{l} LDR_f : \text{Tasa de entrega enlace ida.} \\ LDR_r : \text{Tasa de entrega enlace vuelta.} \end{array} \quad (6.3)$$

Si ahora suponemos enlaces simétricos, es decir, enlaces con la misma calidad tanto en la ida como en la vuelta, la fórmula 6.3 anteriormente vista queda de la siguiente manera:

$$ETX_i = \frac{1}{LDR_i^2} \quad (6.4)$$

Considerando las limitaciones computacionales de los motes se decidió omitir el cuadrado del denominador para agilizar los cálculos, puesto que sin él, la elección de las rutas y las estadísticas extraídas de las pruebas son exactamente iguales que considerándolo. La única diferencia reside en el hecho que el coste ETX es mayor con el cuadrado que sin él.

Por lo tanto la fórmula final implementada es la siguiente:

$$ETX_i = \frac{1}{LDR_i^2} \quad \longrightarrow \quad ETX_i = \frac{1}{LDR_i} \quad (6.5)$$

Una vez visto cómo se calcula la métrica ETX para un enlace se muestra su implementación dentro del protocolo nst-AODV.

```
command uint8_t convertLdrToEtx (uint8_t ldr){
    float metric_zig;
    uint16_t metric_Zig;

    metric_zig = 1000/ldr;
    metric_Zig = floor(metric_zig);
    return metric_Zig;
}
```

Fig. 6.9. Implementación métrica ETX.

6.1.5. Otras implementaciones

Aparte de los cambios expuestos anteriormente, que eran cambios específicos de cada una de las métricas, se han realizado cambios en el funcionamiento del protocolo nst-AODV, con el fin de que éste trabaje con mecanismos de reconocimiento y retransmisiones, y sin ellos para la realización de las pruebas finales.

Por defecto el protocolo nst-AODV asume el uso de reconocimientos de capa 2 y realiza hasta 3 intentos de transmisión por enlace. Para desactivar los ack's se han modificado los módulos SingleHopManagerM.nc de la librería AODV y el módulo RadioControlM.nc de la librería CC2420 del chip radio.

Los cambios realizados son los siguientes:

```
call MacControl.disableAck();
```

Fig. 6.10. Deshabilitación de ack's a nivel de la capa MAC.

Esta instrucción le indica a la capa MAC que deshabilite el mecanismo de confirmación de tramas y, por lo tanto, que no se envíen ni se esperen ack's.

```
event result_t RadioSend.sendDone[uint8_t id](TOS_MsgPtr msg, result_t
success) {

    result_t ret = success;
    msg->ack=1;
    seq++;

    if ((msg->addr == TOS_BCAST_ADDR)&&(success==SUCCESS)){
        msg->ack=1;
        return signal SendMsg.sendDone[id](msg, SUCCESS);
    }

    if ((msg->ack == 1) && (success == SUCCESS)){
        msg->ack = 1;
        ret = signal SendMsg.sendDone[id](msg, SUCCESS);
    }else{
        ret = signal SendMsg.sendDone[id](msg, success);
    }
    return SUCCESS;
}
```

Fig. 6.11. Deshabilitación de ack's en el protocolo.

Con este cambio se consigue que el protocolo nst-AODV no espere un paquete de reconocimiento al transmitir por el medio radio, puesto que con esta instrucción se indica que se ha recibido el reconocimiento correctamente.

Cabe destacar también los cambios realizados en la herramienta USBShark para adaptarla a las necesidades de este proyecto. Las modificaciones

consisten en adaptar el paquete para que muestre por la interfaz gráfica los campos deseados.

6.2. Pruebas y resultados.

En este apartado se muestran las pruebas realizadas con el objetivo de comparar y evaluar el rendimiento del protocolo de encaminamiento NST-AODV con cuatro métricas diferentes: la métrica por defecto (HOPS), y las métricas basadas en la calidad del enlace PDR, ZigBee y ETX programadas sobre el NST-AODV existente.

6.2.1. Escenario de las pruebas

Las pruebas se han realizado en su totalidad en el grid instalado en el laboratorio 325 de la EPSC, cuyo esquema de distribución de nodos se puede observar en el apartado 5.2. A continuación se muestra una foto representativa de dicho grid.

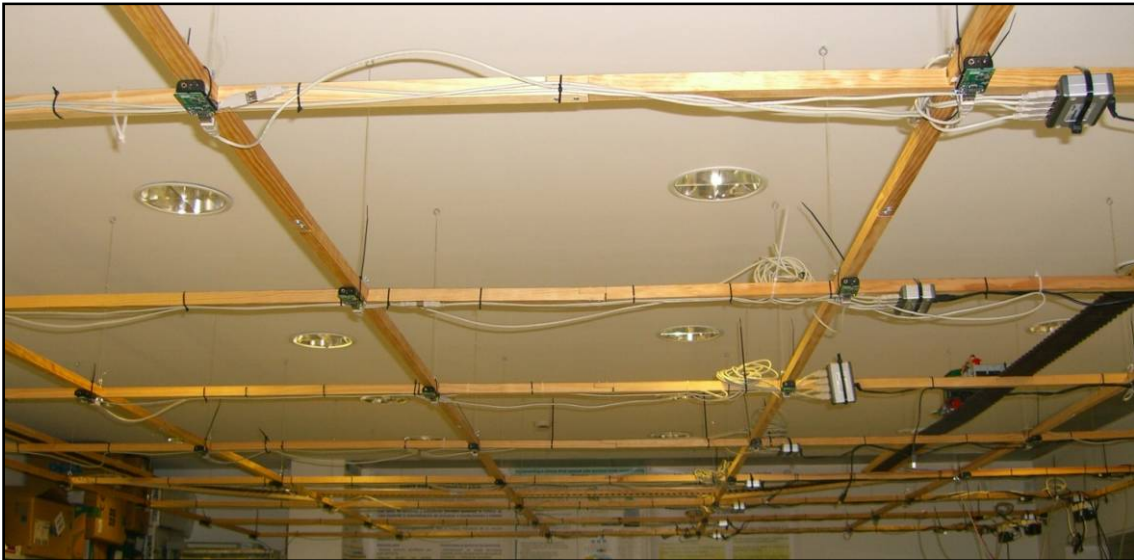


Fig. 6.12. Banco de pruebas del laboratorio 325.

En cuanto a la configuración de los dispositivos se refiere, todos los nodos del grid se han configurado para transmitir a potencia 0x02 que equivale aproximadamente a unos -20dBm en el canal 26 de la banda de 2.4 GHz. En esta banda de frecuencias trabajan también otras tecnologías inalámbricas como es el caso del 802.11x (WiFi) presente en el laboratorio de pruebas. Se ha elegido dicho canal puesto que es el más alejado respecto a estas tecnologías y por lo tanto no le afectan posibles interferencias provenientes de ellas como se observa en la siguiente figura.

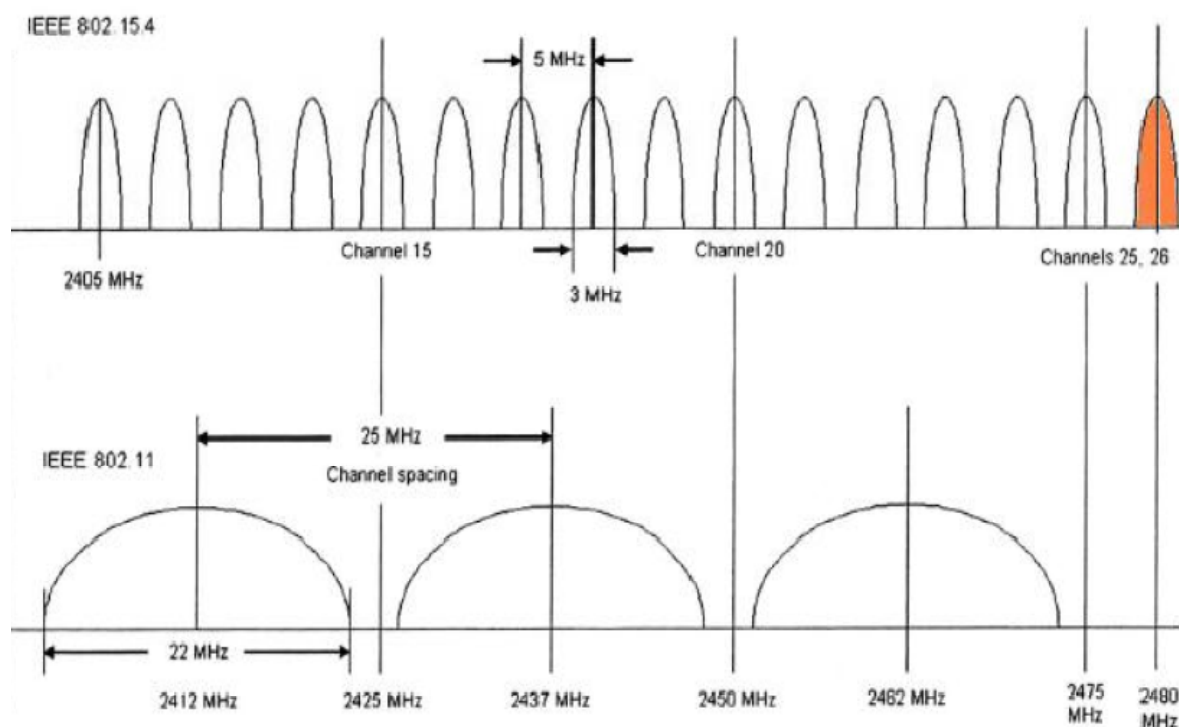


Fig. 6.13. Espectro de frecuencias 802.15.4 vs. 802.11.

En lo referente a la metodología de las pruebas, su mecánica es siempre la misma: el nodo emisor transmite un total de 1000 paquetes a una tasa de 3 paq/seg hacia el receptor, el cual va variando su posición en el escenario como se muestra posteriormente en la fig.6.11 y donde se recogen las estadísticas de la prueba almacenando la tasa de entrega obtenida, el pdr estimado por el protocolo nst-AODV, el número de saltos realizado hasta alcanzar el destino y el coste de la métrica en el caso de ZigBee y ETX. Con estos datos posteriormente se realiza el estudio, evaluación y comparación de los cuatro protocolos.

Importante comentar también que se diferencian dos escenarios distintos en las pruebas, el escenario de flujos largos, compuesto por 26 flujos y el escenario de flujos cortos, compuesto por 24 flujos.

En el primer caso, los nodos receptores se encuentran a una distancia física tal que se espera que el número de saltos requerido esté entre 1 y 4 aproximadamente. En el segundo, se espera que sean necesarios entre 3 y 8 saltos, aproximadamente. También se han realizado pruebas del nst-AODV con y sin reconocimiento. Se ha efectuado este estudio con el objetivo de conocer el comportamiento del protocolo con las diferentes configuraciones de nst-AODV lo más minuciosamente posible y, de esta manera, poder obtener conclusiones precisas y significativas.

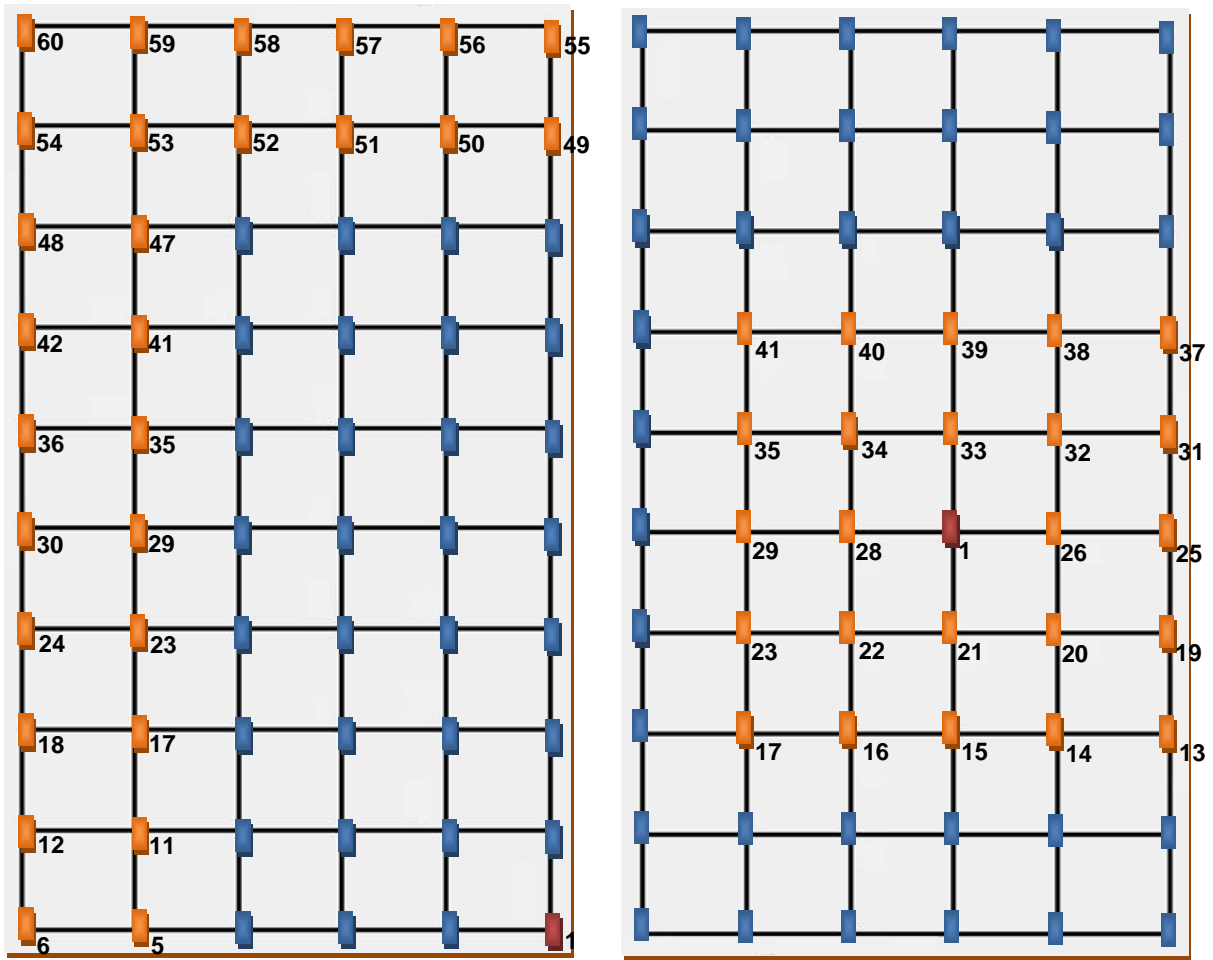


Fig. 6.14. De izquierda a derecha, escenario de flujos largos y escenario de flujos cortos respectivamente.

Como se observa en la figura anterior, el nodo rojo es el nodo transmisor, los nodos en azul son los nodos intermedios y los nodos en naranja representan la posición que va tomando el receptor en cada una de las pruebas.

Cabe comentar que las pruebas se han realizado por separado, es decir, con un único flujo simultáneo en toda la red y con condiciones externas siempre similares, es decir, el mismo número de personas aproximadamente en sus respectivos puestos de trabajo en horas comprendidas dentro de la franja horaria de 15:00 a 21:00 horas.

6.2.2. Resultados de las pruebas.

6.2.2.1. Resultados generales de las pruebas con reconocimientos.

A continuación se muestran los resultados globales de las pruebas realizadas con el protocolo nst-AODV por defecto, es decir, con confirmaciones de capa 2 y hasta 3 intentos de transmisión por enlace.

Tabla 6.1. Resultados globales con nst-AODV por defecto.

	HOPS			PDR			Zigbee			ETX		
	Largos	Cortos	Total	Largos	Cortos	Total	Largos	Cortos	Total	Largos	Cortos	Total
Tasa entrega (%)	68,28	89,01	77,85	74,48	95	83,95	59,27	93,66	75,14	65,01	93,98	78,38
PDR estimado	54,05	78,36	65,27	68,52	85,37	76,3	49,51	81,58	64,31	62,91	81,23	71,37
Salto	4,24	2,26	3,33	5,56	3,14	4,44	5,41	3,27	4,42	4,29	2,37	3,40
Nº redescubrimientos	5,73	6,33	6,01	5,5	3,58	4,61	5,21	3	4,19	4,68	2,67	3,75

En la tabla 6.1 se muestra los resultados obtenidos al implementar las cuatro métricas estudiadas en este proyecto. Cada métrica ha sido evaluada según tasa de entrega de paquetes, el PDR estimado⁶, el número de saltos medio en las transmisiones y por el número de redescubrimientos medio realizado. Cabe comentar que estos resultados están expresados como la media de los resultados obtenidos en cada uno de los flujos tanto largos como cortos y el resultado total ha sido ponderado según el número de flujos que hay en cada grupo.

La métrica que presenta un mejor comportamiento, en lo que se refiere a tasa de entrega, es la métrica PDR con un 83,95% de paquetes entregados seguida por las métricas ETX, HOPS y ZigBee con un 78,38%, un 77,85% y un 75,14% respectivamente. Estos resultados son los esperados ya que la métrica PDR pretende optimizar la tasa de entrega. Las otras métricas tienen valores muy similares de tasa de entrega, siendo la métrica ZigBee la que proporciona una tasa de entrega menor con un 75,14%. Se esperaba que tanto la métrica ETX como la ZigBee estuvieran por encima de la métrica HOPS debido que las dos primeras basan el cálculo del coste del enlace en la calidad del mismo pero, como se puede observar en la tabla, la métrica HOPS presenta unos valores de entrega muy superiores en los caminos largos a los proporcionados por la métrica ZigBee en estos mismos caminos, por lo que repercute que la métrica ZigBee sea la que menor tasa de entrega ofrece entre las métricas evaluadas. Esto es debido a la cuantificación del PDR que supone la métrica ZigBee, ya que hace que se acumule el error a lo largo del camino, en la medida en que éste es largo (para ZigBee tiene el mismo coste un enlace de LDR igual al 84% que uno de 100%).

Como se observa, la tasa de entrega es superior en los caminos cortos respecto a los largos en todas las métricas evaluadas debido al menor número de saltos realizados. Cuanto más larga es una ruta, más enlaces intervienen y por lo tanto existe mayor probabilidad de degradación. El PDR estimado en el caso de utilizar mecanismos de confirmación nos da una idea de la calidad del camino seleccionado hacia el receptor. Debe tenerse en cuenta que si el protocolo tiene habilitado el mecanismo de confirmación (ack's), puede hacer

⁶ Consiste en el promedio de los PDR extraídos de los paquetes RREQ de cada descubrimiento de ruta que tenga lugar con éxito en una misma prueba.

un máximo de tres intentos de transmisión de paquetes de datos si no recibe los ack's correspondientes, por lo que la probabilidad de entrega es superior a la indicada por este parámetro. Por esto mismo, se puede ver como el PDR estimado para cada métrica tienen un valor inferior a la tasa de entrega correspondiente.

Aun así, la métrica PDR es la que tiene un índice de PDR mayor con un 76,3 ya que está basada en el propio parámetro PDR. Igual que en la tasa de entrega, las otras métricas siguen la misma distribución, es decir, la métrica ETX es la que presenta el segundo índice mayor de PDR y la métrica HOPS, que no se basa en la calidad del enlace, presenta un índice de PDR superior a la métrica ZigBee. Esto es debido a que en los enlaces largos tiene un índice de PDR muy por debajo de lo esperado por la razón indicada anteriormente y repercute de mayor forma en el total al haber más flujos en los caminos largos que en los cortos.

En lo que a número de saltos se refiere, se observa que la métrica que menor número de saltos hace entre origen y destino es la métrica HOPS con 3,33 saltos de media. Esto es lo que se esperaba ya que la métrica HOPS está diseñada para elegir la ruta más corta hacia el destino. La métrica ETX tiene un comportamiento muy similar a la métrica HOPS ya que realiza 3,4 saltos de media. Esta similitud se justifica en que la métrica ETX se basa en minimizar el número de transmisiones por lo que opta por caminos cortos y calidades de enlaces elevadas. Las métricas PDR y ZigBee se encuentran por encima con valores de 4,44 y 4,42 respectivamente.

Estos valores tienen sentido ya que estas métricas se basan solamente en la calidad del enlace y no siempre el camino más corto es el de mayor calidad. Cabe destacar que la diferencia a nivel de saltos entre una métrica que se basa en el número de saltos y otra que se basa en la calidad del enlace no es muy grande debido a que las métricas que se basan en la calidad de enlace sufren una reducción brusca de la calidad del camino por cada salto que realiza debido al cálculo de la propia métrica (véase fig. 6.5).

Por último, destacamos que la métrica que menor número de redescubrimientos realiza a lo largo de la transmisión de los 1000 paquetes es la métrica ETX con un número medio de redescubrimientos de 3,75. Este resultado coincide con el previsto teóricamente, puesto que ETX intenta elegir la ruta más estable y que da lugar a menos retransmisiones. Por el contrario, la métrica que más redescubrimientos provoca es la HOPS con un número medio de 6,01 redescubrimientos debido a que no se basa en la calidad del enlace. Por ello, es más propicia a tener más errores y realizar más redescubrimientos. Cabe destacar que al realizar menos redescubrimientos se introduce menor *overhead* del protocolo de encaminamiento ya que se introducen menos paquetes de control en la red.

6.2.2.2. Resultados específicos caminos largos con reconocimientos

En este apartado se presentan los resultados obtenidos de cada flujo en los caminos largos representados en el plano de la página 63, donde cada posición del grid es representada por una tabla que engloba los resultados obtenidos para el mote receptor que ocupa dicha posición para las cuatro métricas.

Como se observa en el plano, se percibe una tendencia, en lo que se refiere a tasa de entrega, habiendo un mayor porcentaje de éxito en los nodos que se encuentran aproximadamente en línea recta con el transmisor, ya sea fila (motes 1 – 55) o columna (motes 1 – 6), y experimentando una degradación en la tasa de entrega a medida que la posición del receptor se acerca al vértice opuesto del transmisor (mote 60). Esto es debido a dos factores, por un lado se encuentra la orientación de la antena (véase fig. 5.4.), coincidiendo esta orientación con un ángulo de menor radiación y por otro lado el factor distancia puesto que a mayor distancia, rutas más largas y por tanto más probabilidad de error dentro de una misma métrica.

Matizar que dentro de esta tendencia se encuentran casos particulares que rompen con este comportamiento, como es el caso del mote 29 con métrica HOPS que presenta una tasa de entrega de un 26,55% siendo ésta significativamente más baja que la de sus vecinos estando más cerca del transmisor que ellos. Estos casos pueden estar motivados por la baja calidad de uno o más enlaces de la ruta seleccionada en el momento de realizar la prueba, ya que se observa que el pdr estimado de ese flujo, con valor 33,65, demuestra que la ruta es de baja calidad y propicia a bajas tasas de entrega.

Cabe comentar que hay determinadas posiciones en el grid donde no se reciben paquetes con ninguna de las cuatro métricas evaluadas (motes 60 y 54), esto puede ser debido a que por factores que se escapan a nuestro control, el número de paquetes de descubrimiento de ruta que llegan al destino son escasos y los paquetes de confirmación de ruta no alcanzan el nodo transmisor y por lo tanto nunca se encuentra ruta hacia éste. Esta situación parece anómala por el hecho que el siguiente nodo de la misma fila (mote 48) tiene una tasa de entrega de entre 60% - 80% según la métrica.

En lo que se refiere a los parámetros evaluados, destacar que la ruta más larga observada en la pruebas corresponde al mote 53, con la métrica ZigBee, con una media de saltos de 12,07 seguido por el mote 41, con métrica PDR, con una media de 9,72. Por el contrario, las rutas más cortas corresponden a los motes 17 y 11 ambos con métrica HOPS y con una media de saltos de 2,12 y 2,53, respectivamente.

La tendencia creciente en número de saltos hasta el receptor, a medida que éste se aleja del transmisor, no es monótona. En cuanto al PDR estimado se aprecia, en muchos casos, una alta correlación con el PDR obtenido.

A continuación se observa la ruta que más ha prevalecido en la prueba del mote 53:

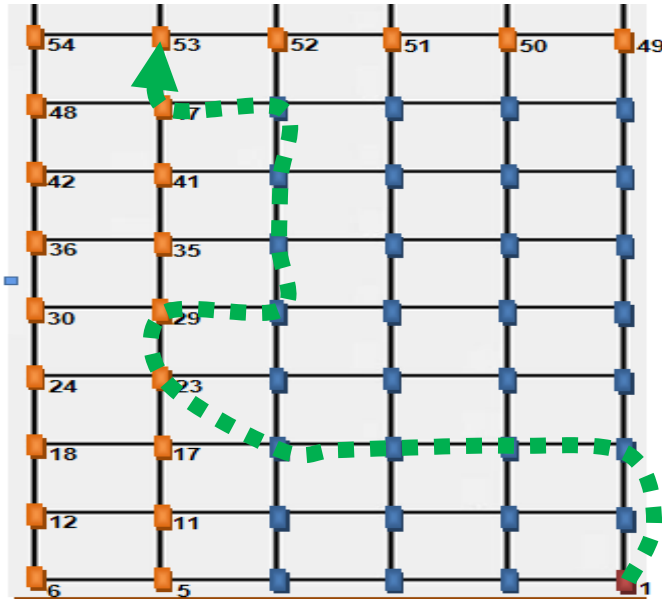


Fig. 6.15. Camino al mote 53 con métrica ZigBee.

A continuación se expone la función de distribución acumulativa (CDF) de la tasa de entrega medida sobre los 28 flujos para las cuatro métricas.

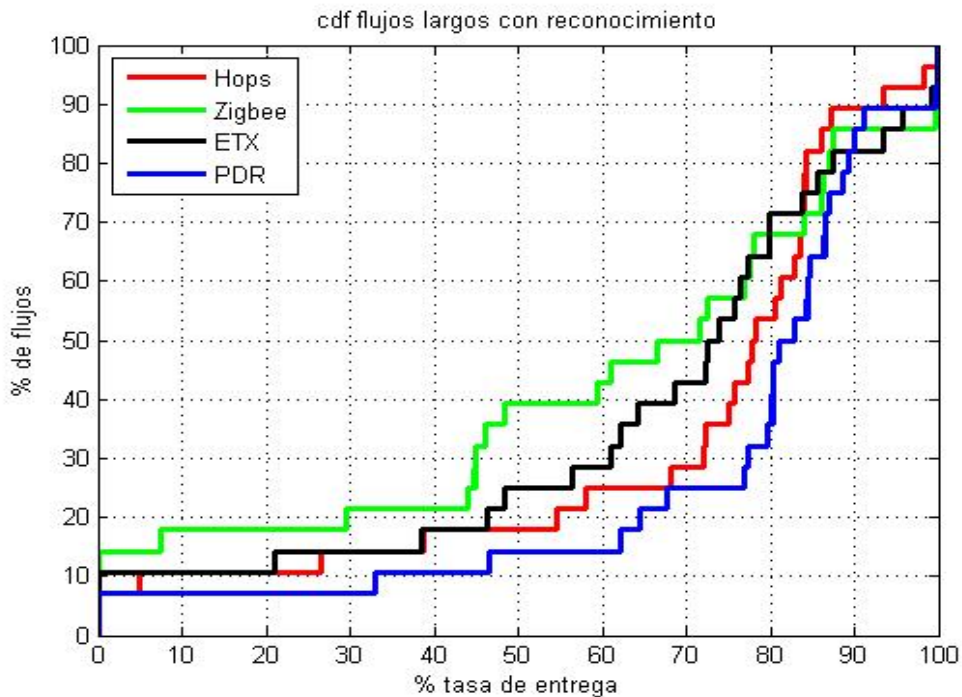


Fig. 6.16. CDF caminos largos con ack's.

Como se observa, la métrica que presenta un mejor comportamiento es la PDR ya que el 60% de los flujos evaluados entregan una tasa superior o igual al 80%, seguido de la métrica HOPS que entrega aproximadamente más de un 75%, la métrica ETX con más de un 65% y finalmente la que presenta un peor rendimiento en este escenario, tal y como se había comentado anteriormente, es la métrica ZigBee con una tasa de entrega superior o igual al 48% en el 60% de los flujos estudiados.

6.2.2.3. Resultados específicos caminos cortos con reconocimientos

A diferencia de los caminos largos, en los caminos cortos, mostrados en el plano de la página 67, los receptores rodean al transmisor formando dos coronas de 8 y 16 motes, creando caminos de poca distancia hasta el receptor en todas las direcciones alrededor del mote transmisor. Como se observa en el plano, la tasa de entrega es alta en ambas coronas siendo la segunda las que registra valores ligeramente más bajos debido a la distancia.

Los motes de la primera corona presentan una gran similitud en lo que a tasa de entrega se refiere entre las cuatro métricas evaluadas en un mismo receptor. Esta igualdad en la tasa de entrega entre métricas se debe a que la distancia es muy pequeña entre el mote transmisor y el mote receptor, lo que hace que no se note el factor de la métrica en los flujos. Cabe comentar que el mote 33 es el único presenta una tasa de entrega del 100% y una ruta de 1 salto con las cuatro métricas evaluadas. También comentar el caso particular del mote 28, situado en la primera corona, que presenta un camino de 4,85 saltos con la métrica ZigBee siendo el camino más largo en la primera corona. A continuación se observa la ruta que más ha prevalecido en la prueba del mote 28:

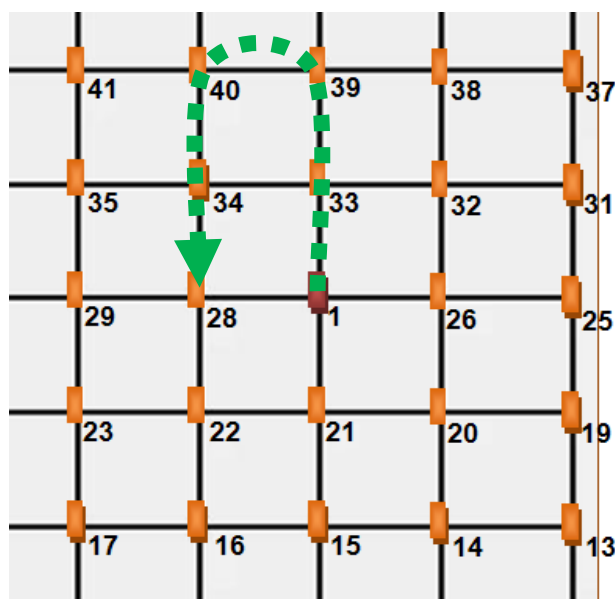


Fig. 6.17. Camino al mote 28 métrica ZigBee.

En la segunda corona se observa que para un mismo mote las tasas de entrega varían dependiendo la métrica utilizada debido a que la distancia entre transmisor y receptor es mayor y se debe efectuar más de un salto en más ocasiones, por lo tanto el efecto de la métrica se hace notorio.

Al igual que en los enlaces largos, hay casos particulares que no presentan un buen rendimiento en términos de tasa de entrega como es el caso del mote 23, con métrica HOPS, que presenta una tasa de entrega del 50,04% siendo ésta significativamente más baja que las de sus vecinos con misma métrica y muy inferior si la comparamos con las otras métricas ya que tanto la métrica ETX, ZigBee como PDR presentan valores superiores al 75% en el mismo mote.

Estos casos pueden estar motivados por la baja calidad de los enlaces de la ruta seleccionada en el momento de realizar la prueba, ya que se observa que el pdr estimado de esos flujo indica que la ruta es de baja calidad y propicia a bajas tasas de entrega. No obstante, ambas tasas se obtienen con la métrica HOPS, que no se basa en la calidad del camino.

Finalmente, en lo que se refiere a los parámetros evaluados, destacar que la ruta más larga corresponde al mote 15, con métrica ZigBee, con una media de saltos de 6,26 seguido por el mote 16, con métrica ZigBee, con 6,05 saltos de media. Por el contrario, las rutas más cortas se localizan en varios motes, sobretodo de la primera corona, a una distancia 1 salto debido a que la distancia es muy pequeña.

A continuación se muestra la función de distribución de a tasa de entrega de los flujos de este escenario:

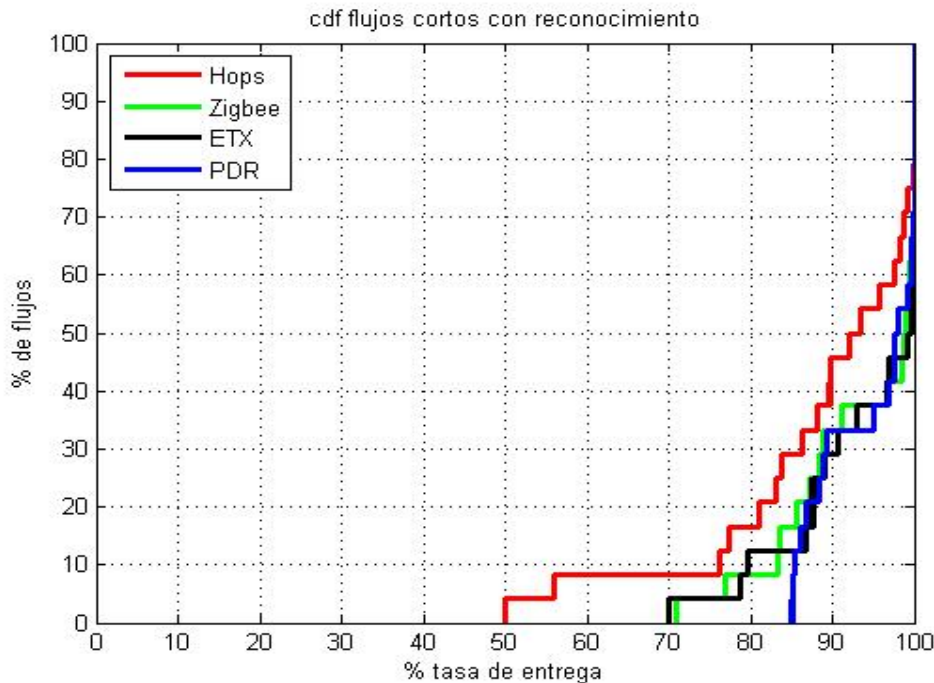


Fig. 6.18. CDF caminos cortos con ack's.

Como se observa en la gráfica las cuatro métricas presentan buenas tasas de entrega debido a la corta distancia entre el transmisor y los distintos receptores. Las métricas que se basan en la calidad del enlace presentan un mejor comportamiento que la métrica HOPS que no tiene en cuenta este parámetro, por lo que se puede ver como las rectas de las métricas PDR, ETX y ZigBee están prácticamente superpuestas, con ligero mejor comportamiento para la métrica PDR.

Los resultados que se obtienen de esta gráfica es que la métrica PDR proporciona una tasa de entrega superior al 85% en el 92% de los flujos, seguida por la métrica ETX con más de un 79% de tasa de entrega, la métrica ZigBee con más de un 77% y finalmente la métrica HOPS con más de un 56% de tasa de entrega en el 92% de sus flujos.

6.2.2.4. Resultados generales de las pruebas sin reconocimientos.

A continuación se muestran los resultados globales de las pruebas realizadas con el protocolo nst-AODV trabajando sin mecanismos de confirmación y retransmisión.

Tabla 6.2. Resultados globales sin ack's.

	HOPS			PDR			Zigbee			ETX		
	Largos	Cortos	Total	Largos	Cortos	Total	Largos	Cortos	Total	Largos	Cortos	Total
Tasa entrega (%)	37,61	68,5	51,87	56,07	70,78	62,86	46,36	65,42	55,16	46,28	69,46	56,98
PDR estimado	60,75	73,25	66,52	72,48	83,38	77,51	52,64	71,17	61,19	60,13	77,29	68,05
Salto	3,8	2,96	3,412	5,4	3,46	4,505	5,6	3,75	4,746	4,35	3,04	3,745

Cada métrica ha sido evaluada según tasa de entrega de paquetes, el PDR estimado y por el número de saltos medio en las transmisiones. Al utilizar el protocolo sin mecanismos de reconocimiento, no hay redescubrimientos en la red ya que no hay manera de detectar un fallo en la misma, por lo que no se evalúa éste parámetro.

La métrica que presenta un mejor comportamiento, en lo que se refiere a tasa de entrega, es la métrica PDR con un 62,86% de paquetes entregados seguida por las métricas ETX, ZigBee y HOPS con un 56,98%, un 55,16% y un 51,87% respectivamente. Como era de prever, estos valores son inferiores a los obtenidos con mecanismos de reconocimiento y retransmisiones ya que, en este caso, si se produce una caída de enlace en la ruta, no hay forma de detectarlo y los paquetes se siguen enviando como si la ruta estuviera operativa. No obstante, a diferencia del caso de utilizar ack's, se observa que las métricas que se basan en la calidad del enlace (PDR, ETX y ZigBee) presentan mejor resultado que la métrica que no se basa en ese parámetro (HOPS). Esta diferencia se debe a que al no tener habilitado los ack's por parte del protocolo, no se producen reintentos de envío de paquetes y la probabilidad de que un paquete llegue con éxito repercute aun más si cabe en la elección del camino por parte de la métrica, por lo que las métricas que eligen rutas de mejor calidad tienen tasas de entrega superiores.

Como se observa, existe poca diferencia entre las tasas de entrega de las diferentes métricas para los caminos cortos, pero ésta aumenta en el caso de los caminos largos.

A diferencia que en el caso de tener los ack's activados, el PDR estimado en el caso de no utilizar mecanismos de confirmación sí se puede considerar una estimación de la calidad de la ruta ya que el protocolo solamente realiza un intento de transmisión de paquetes de datos en cada enlace. Como se observa

en la tabla 6.2, el pdr estimado es superior a la tasa de entrega. Esta diferencia es debida a que no se tiene en cuenta la calidad del camino durante la transmisión, es decir, la estimación de la calidad de la ruta solamente se realiza en el momento de búsqueda de ruta. Una vez seleccionada la ruta se transmiten los paquetes de datos sin comprobar la calidad del camino. Como se ha visto en el apartado 5.5, un enlace sufre variaciones de calidad ya que ésta se basa en el parámetro LQI, el cual es muy variable sobretodo en enlaces de media y baja calidad. Por tanto el PDR estimado se obtiene para una ruta cuyas condiciones posteriormente pueden cambiar, dando lugar a mala calidad de enlaces y pérdidas de paquetes.

No obstante, se puede observar que existe una correlación elevada entre el PDR estimado y la tasa de entrega medida para cada métrica.

En lo que a número de saltos se refiere, la métrica HOPS es la que menor número de saltos realiza con 3,41 saltos de media seguida por las métricas ETX, PDR y ZigBee con 3,74, 4,50 y 4,74 respectivamente. Estos resultados son similares a los obtenidos con los reconocimientos ya que el tener activado ack's o no, no afecta al número de saltos de las rutas.

6.2.2.5. Resultados específicos caminos largos sin reconocimientos

A continuación se comentan los resultados obtenidos representados en el plano mostrado posteriormente [véase página 73].

Igual que en el caso de caminos largos con reconocimientos y retransmisiones, se percibe una tendencia, en lo que a tasa de entrega se refiere, habiendo mayor tasa de entrega en los nodos próximos a los extremos del grid (nodos 49/55 y 5/6) y disminuyendo ésta a medida que el receptor se sitúa en el vértice opuesto del transmisor (nodo 60).

En este escenario se encuentran casos particulares que rompen con esta tendencia, como es el caso del mote 11 con métrica ZigBee que presenta una tasa de entrega de un 27,91% siendo ésta mucho más baja que la de los motes de su alrededor en la misma métrica, o el mote 29 que tiene una tasa de entrega de 16,73 con la métrica ETX. Cabe comentar que el número de posiciones del grid donde no se reciben paquetes aumenta en este caso respecto al caso con reconocimientos, siendo la métrica HOPS la que presenta un número mayor de motes inalcanzables, concretamente 7 motes en los que hay 0% de paquetes entregados.

En lo que se refiere a los parámetros evaluados, comentar que la ruta más larga corresponde al mote 18, con la métrica ZigBee, con un número de saltos de 14 seguido por el mote 52, con métrica PDR, con 12 saltos, y el mote 58, con métrica ZigBee, con el mismo número de saltos. A continuación se observa la ruta seleccionada por la métrica ZigBee hacia el mote 18.

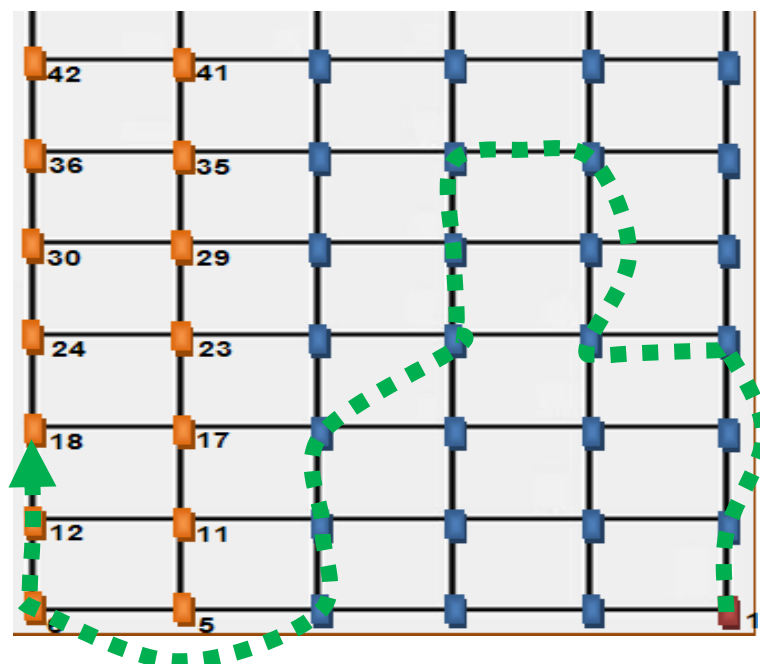


Fig. 6.19. Camino al mote 18 métrica ZigBee.

Por el contrario, las rutas más cortas corresponden a los motes 11, 17 y 23 todos con métrica HOPS y con una media de 2 saltos.

A continuación se muestra la grafica obtenida al realizar la función de distribución de la tasa de entrega de los flujos largos sin reconocimientos.

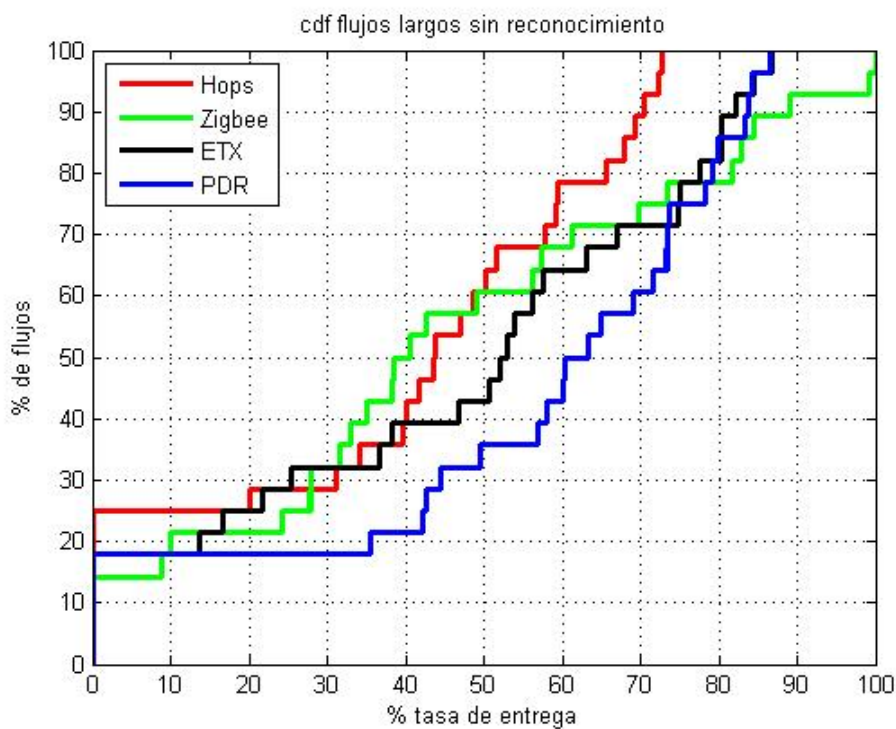


Fig. 6.20. CDF caminos largos sin ack's.

Como se observa en la gráfica, la tasa de entrega es inferior en este escenario a la del escenario con reconocimientos activados.

En este caso, para un 50% de los flujos, la métrica que presenta una mejor tasa de entrega es la PDR con más de un 60% de paquetes entregados seguida por la métrica ETX con más de un 52%, la métrica HOPS con más de un 43% y la que peor rendimiento ofrece, en este punto, es la métrica ZigBee con una tasa de entrega superior al 38%.

Las tasas de entrega individuales más elevadas obtenidas en este escenario han sido proporcionadas por la métrica ZigBee, ya que dos de sus flujos entregan una tasa del 100% y el 99,2% respectivamente. Destacar que en media, el comportamiento es el expuesto en el caso de caminos largos con reconocimiento y retransmisión, donde la métrica PDR es la que ofrece mejor rendimiento, seguida por las métricas ETX y ZigBee con una tasa de entrega media muy similar y por último el peor comportamiento es el ofrecido por la métrica HOPS.

PLANO LARGOS SIN ACK

6.2.2.6. Resultados específicos caminos cortos sin reconocimientos

En este apartado se observa los resultados obtenidos en los flujos de los enlaces cortos, a partir de los cuales se comentan curiosidades surgidas en estas pruebas.

Como se observa en el plano representado en la página 77, la tasa de entrega en ambas coronas es inferior a la obtenida en las pruebas con reconocimiento, siendo esta diferencia de un 20% aproximadamente igual que en el caso de los caminos largos.

Como era de esperar, la primera corona presenta tasas de entrega superiores a los de la segunda, manteniendo la tendencia mostrada en el caso de caminos cortos con ack's.

Al igual que en los caminos largos, hay casos particulares que no presentan un buen rendimiento en términos de tasa de entrega como es el caso de los motes 23 y 17 cuyas tasas de entrega son significativamente bajas, con valores del 10% para diversas métricas, hecho que no se esperaba por la corta distancia de los enlaces. Otro caso a destacar, es el del mote 38, cuya tasa de entrega en la métrica ZigBee es de aproximadamente 90% y su pdr estimado es de 1, esta curiosidad se debe a que en el instante de descubrir la ruta, ésta presentaba una muy baja calidad, la cual, observando la tasa de entrega, experimentó un indudable aumento durante el transcurso de la prueba ofreciendo una tasa de entrega muy superior al pdr estimado para ese flujo.

La ruta más larga corresponde al mote 32, con métrica PDR y con un número de saltos igual a 9 como se puede ver en la siguiente figura.

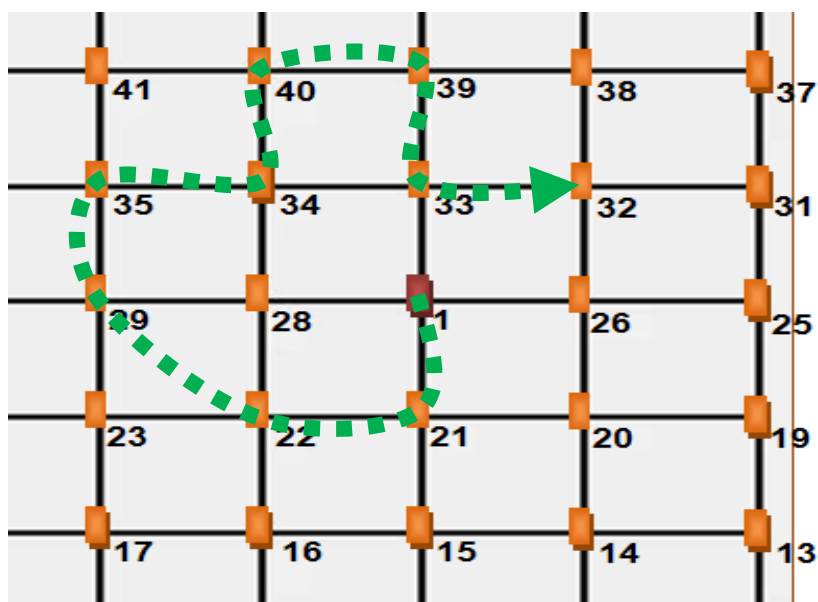


Fig. 6.21. Camino al mote 32 métrica ZigBee.

Por el contrario, las rutas más cortas se localizan en varios nódulos, sobretodo de la primera corona, a una distancia 1 salto debido a que la distancia es muy pequeña.

A continuación se presenta la función de distribución de la tasa de entrega de los flujos cortos sin reconocimientos:

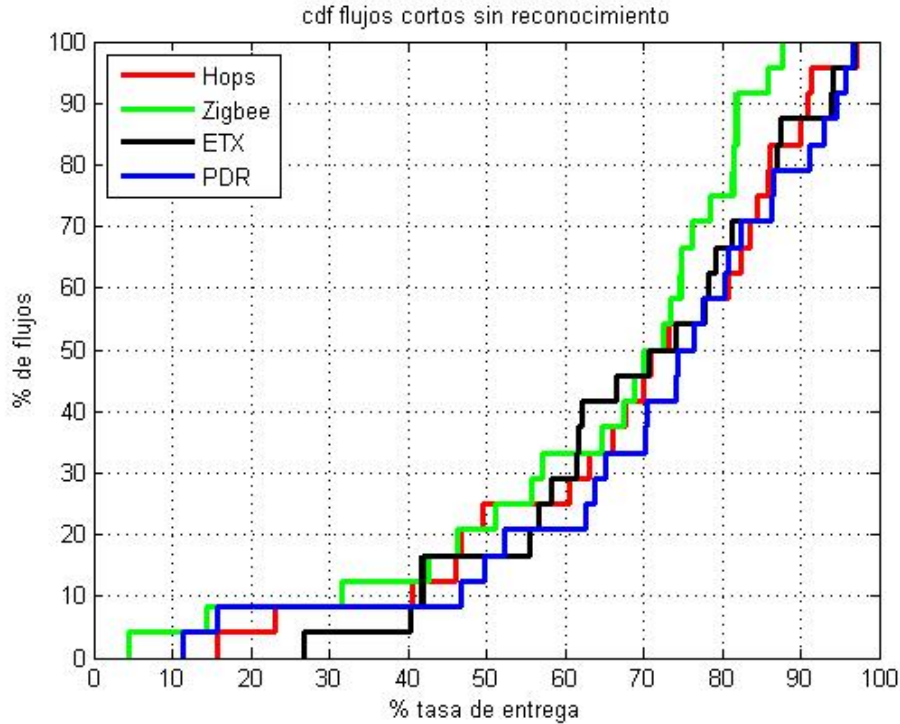


Fig. 6.22. CDF caminos cortos sin ack's.

Como se observa, las cuatro métricas tienen un comportamiento muy similar. En este escenario no se detectan cortes ni se realizan descubrimientos en la ruta. Por lo que la aportación de la métrica es menor ya que solo es aplicada en el descubrimiento.

Cabe comentar, que a diferencia de los flujos largos, la métrica ZigBee es la que presenta la menor tasa de entrega en los máximos de cada métrica ofreciendo como mayor tasa individual un 87,59% de paquetes entregados. Las demás métricas tienen tasas de entrega máximas similares alrededor de un 96% como se observa en la gráfica.

Plano corto sin ack

CAPÍTULO 7. CONCLUSIONES Y LÍNEAS FUTURAS

Este capítulo presenta las principales conclusiones y líneas futuras que se derivan del proyecto.

7.1. Conclusiones

En este proyecto se ha realizado un estudio en el campo del encaminamiento en redes inalámbricas multisalto IEEE 802.15.4. Pese a que existen estudios que caracterizan la calidad de un enlace IEEE 802.15.4 en base a parámetros proporcionados por las capas físicas y MAC, existe escasa literatura acerca de su aplicación al encaminamiento en redes IEEE 802.15.4 con topología mesh. Este documento presenta un estudio de métricas de encaminamiento alternativas a la métrica clásica basada en número de saltos (HOPS) para el protocolo de encaminamiento nst-AODV. En particular, además de la métrica HOPS, las métricas evaluadas han sido PATH-DR, ZigBee y ETX que tienen en cuenta la calidad de los enlaces y persiguen objetivos distintos.

Para la implementación de estas métricas, se ha realizado un estudio previo sobre el parámetro LQI proporcionado por la capa MAC de la tecnología IEEE 802.15.4. Se ha observado que en enlaces de media-baja calidad este parámetro es muy variable, pudiendo provocar errores de estimación de la calidad de un enlace en cuestión. Por el contrario, en enlaces de alta calidad este parámetro proporciona valores estables constituyendo un buen indicador de calidad.

Para llevar a cabo la evaluación, cuyo escenario de pruebas ha sido el grid de motes del laboratorio 325 de la EPSC, se realizó en primer lugar un estudio sobre el comportamiento del parámetro LQI respecto la orientación de la antena de los motes. En éste, se observó que el impacto de la orientación sobre el LQI es relevante y se posicionó los dispositivos en el escenario de pruebas de forma regular. Por otra parte, se estudió el efecto que producen las interferencias causadas por transmisiones de otros motes en la calidad de los enlaces sin apreciar que éstas tengan un efecto importante en la misma.

Una vez conocido con detalle el comportamiento del parámetro LQI, se implementaron las métricas en el protocolo de encaminamiento nst-AODV y se procedió a su evaluación en dos escenarios diferentes, con un total de 52 flujos y con dos configuraciones distintas de nst-AODV: la empleada por defecto que usa reconocimiento de capa 2 y hasta 3 intentos de retransmisión, y también sin tales reconocimientos y con un solo intento de transmisión por enlace.

En el caso de tener el mecanismo de reconocimiento y retransmisión activado, la métrica que presenta un mejor comportamiento, en lo que se refiere a tasa de entrega, es la métrica PDR con un 83,95% de paquetes entregados seguida por las métricas ETX, HOPS y ZigBee con un 78,38%, un 77,85% y un 75,14% respectivamente. Estos resultados son los esperados, ya que la métrica PDR pretende elegir la ruta con mayor tasa de entrega. Las otras métricas tienen

valores muy similares de tasa de entrega, siendo la métrica ZigBee la que proporciona una tasa de entrega menor con un 75,14%.

En el caso de no tener activado el mecanismo de reconocimiento y retransmisión, la métrica que presenta un mejor comportamiento es la métrica PDR con un 62,86% de paquetes entregados seguida por las métricas ETX, ZigBee y HOPS con un 56,98%, un 55,16% y un 51,87% respectivamente. Como era de prever, estos valores son inferiores a los obtenidos con mecanismos de reconocimiento y retransmisión, a causa de la ausencia de detección de rutas no operativas y, pese a que se degrade la calidad de una ruta, los paquetes se siguen enviando como si la ruta estuviera operativa.

Como se ha visto, en el caso de no tener reconocimientos activados, las métricas que basan su elección en la calidad del enlace presentan un mejor comportamiento que la métrica HOPS, debido a que al no tener habilitado los ack's por parte del protocolo, no se producen reintentos de envío de paquetes y la probabilidad de que un paquete llegue con éxito repercute aún más si cabe en la elección del camino por parte de la métrica, por lo que las métricas que eligen rutas de mejor calidad tienen tasas de entrega superiores.

La métrica HOPS, como era de esperar, minimiza el número de saltos, seguida por la métrica ETX. La métrica PDR proporciona caminos de mayor número de saltos, pero con mayor tasa de entrega.

Por otra parte, la métrica ETX, cuyo objetivo es elegir los caminos con menor número de retransmisiones en sus enlaces, proporciona los caminos más estables, dando lugar al menor número de redescubrimientos obtenidos. Se aprecia que la métrica HOPS proporciona los caminos más cortos pero más inestables.

7.2. Líneas futuras

Como líneas futuras se pueden destacar puntos como la combinación del parámetro LQI con el parámetro de señal de potencia recibida RSSI para implementarla en las métricas y evaluar su comportamiento, o la realización de las pruebas en otros entornos como puede ser el campo abierto para tener una visión más global del rendimiento de estas métricas.

En cualquier caso, se deberán seguir las evoluciones del trabajo realizado por 6LoWPAN, el grupo de trabajo de la IETF que se centra en las redes IEEE 802.15.4.

IMPLICACIONES MEDIOAMBIENTALES

El desarrollo sostenible es uno de los grandes retos del siglo XXI y actualmente cualquier proyecto tecnológico ha de tener en cuenta el posible impacto que puede causar al medioambiente. A continuación se realiza un análisis de diferentes aspectos medioambientales y su relación con el proyecto.

Los sensores utilizados en el proyecto están basados en el estándar IEEE 802.15.4, en el cual uno de los aspectos más importantes es el bajo consumo de energía. El consumo energético es un aspecto muy importante a tener en cuenta, puesto que los sensores hacen uso de baterías y puede darse el caso que haya dispositivos cuyo emplazamiento no sea fácilmente accesible. Por tanto cuanto más se minimice el consumo menor batería consumirán. Si las baterías son desechables éstas deberán reciclarse ya que sino con el paso del tiempo pueden expulsar compuestos tóxicos y dañinos para el medioambiente.

Si una vez gastadas las baterías se tiran directamente a la basura, algunos materiales como el mercurio y el cadmio, nocivos para el medioambiente y la salud, pueden liberarse al ciclo del agua a través de la lluvia o del gas producido al quemar las basuras en los vertederos.

Conscientes de esto, los fabricantes de pilas han conseguido reducir notablemente los componentes peligrosos como el mercurio. Además, han comenzado a desarrollar tecnologías para recuperar los componentes de las pilas cuando éstas se agotan e incluso se han inventado las denominadas pilas verdes o biopilas, que son pilas que destacan por la ausencia de los metales pesados en ellas y por tanto por la poca contaminación que causa su consumo en el medioambiente.

A parte de estos aspectos se considera que este proyecto es respetuoso con el medioambiente.

BIBLIOGRAFÍA

- [1] ZigBee/IEEE 802.15.4 Summary,
<http://www.eecs.berkeley.edu/~csinem/academic/publications/ZigBee.pdf>
- [2] IEEE 802.15 WPAN Task Group 4,
<http://www.ieee802.org/15/pub/TG4.html>
- [3] ZigBee Alliance,
<http://www.ZigBee.org>
- [4] Tipos de modulaciones en el 802.15.4
<http://es.wikipedia.org/wiki/DSSS>
- [5] RF Transmisores
<http://www.chipcon.com/>
- [6] Web grupo MANET
<http://www.ietf.org/html.charters/manetcharter.html>
- [7] Clausen T., Jacquet P., RFC 3626 “Optimized Link State Routing Protocol (OLSR)”.
- [8] Ogier R., Templin F., Lewis M., RFC 3684 “Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)”.
- [9] Draft OLSR-V2
- [10] Perkins C., Belding-Royer E., Das S., RFC 3561 “3561 Ad hoc On-Demand Distance Vector (AODV) Routing”.
- [11] Jonson D., Maltz D., Hu Y., “The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)” IETF Internet-Draft draft-ietf-manet-dsr-10.txt.
- [12] Chakeres I., Perkins C., “Dynamic MANET On-demand (DYMO) Routing” IETF Internet-Draft draft-ietf-manet-dymo-01.txt.
- [13] Macker J., “Simplified Multicast Forwarding for MANET”, IETF Internet-Draft draft-ietf-manet-smf-01.txt.
- [14] C. Gomez, P. Salvatella, O. Alonso, J. Paradells, 'Adapting AODV for IEEE 802.15.4 Mesh Sensor Networks: Theoretical Discussion and Performance Evaluation in a Real Environment', 7th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM 2006), Niagara Falls, USA.

[15] Kim K., Montenegro G., Daniel Park S., Chakeres I., Yoo S., “Dynamic MANET On-demand for 6LoWPAN (DYMO-low) Routing”, IETF Internet-Draft draft-montenegro-6lowpan-dymo-low-routing-00.txt.

[16] Kim K., Montenegro G., Daniel Park S., Yoo S., Kushalnagar N., “6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD)”, IETF Internet-Draft draft-daniel-6lowpan-load-adhoc-routing-02.txt.

[17] Datasheet notes TelosB,
http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/TelosB_Datasheet.pdf

[18] Web TinyOS
<http://www.tinyos.net>

[19] Web NesC
<http://nesc.sourceforge.net/>

[20] Raúl Giménez Valladares “*Evaluación de un protocolo de encaminamiento en redes multisalto IEEE 802.15.4*” trabajo de final de carrera, Julio 2008.

[21] C. Soras, M. Karaboikis, G. Tsachtsiris, V. Makios. “ Analysis and Design of an Inverted-F Antenna for the 2.4 GHz ISM Band”



Escola Politècnica Superior
de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

ANEXOS

TÍTULO DEL PFC: Evaluación de métricas de encaminamiento basadas en LQI para un protocolo de encaminamiento en redes IEEE 802.15.4

TITULACIÓN: Ingeniería de Telecomunicaciones (segundo ciclo)

**AUTORES: Omar García Bermejo
Tomás García Sotelo**

DIRECTOR: Carles Gómez Montenegro

FECHA: 31 de Octubre de 2008

ANEXO A. ESTRUCTURAS DE TRAMAS DEL IEEE 802.15.4

En este anexo se realizará una descripción breve sobre el tipo de tramas que contiene el estándar 802.15.4 y su significado.

El estándar 802.15.4 define 4 tipos de trama: trama de beacon, trama de datos, trama ack's y Trama de comandos MAC.

- Trama beacon.

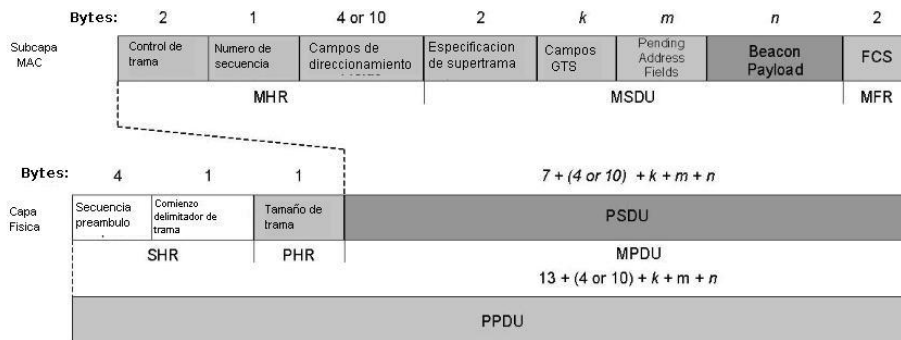


Fig. A.1 Trama beacon

Especificación de supertrama: Este campo especifica diversos parámetros referentes a la supertrama, tales como tiempo de supertrama, quien transmite el beacon o si permite la asociación de nuevos nodos. Nos da información para que el resto de nodos sepan cuando pueden transmitir.

Campos GTS: Aquí se especifica que nodo quiere latencia mínima y slot que ocupará dentro de la trama CFP.

Pending Address Fields: A modo informativo nos dice desde que direcciones se va a transmitir o recibir tramas.

Beacon payload: Este campo se utiliza para llevar información de capas superiores o información encriptada en caso de usar el algoritmo de encriptación AES⁷ establecido en el estándar.

- Trama de datos.

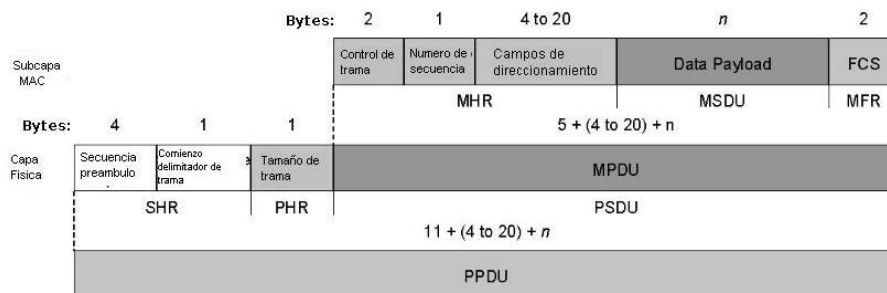


Fig. A.2. Trama de Datos

⁷ AES es un esquema de cifrado por bloque adoptado como un estándar de encriptación por el gobierno de los Estados Unidos, y se espera que sea usado en el mundo entero.

Data payload: Únicamente contendrá la información indicada por capas superiores.

- Trama de ack's

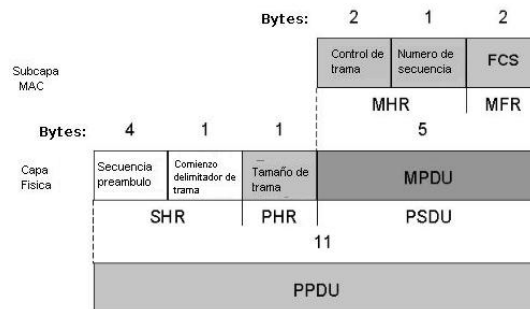


Fig. A.3. Trama de ack's

La trama de ack's no llevan ningún campo específico, únicamente quien envía una trama de datos o una trama de comandos MAC espera recibir, desde la dirección emisora, una trama ACK para confirmar la llegada.

- Trama de comando MAC.

La trama de comandos MAC nos permite llevar a cabo todas las funcionalidades primitivas que define el estándar 802.15.4 tales como asociación, desasociación, resincronización con PAN's coordinators, solicitudes de latencia con QoS (GTS), etc...

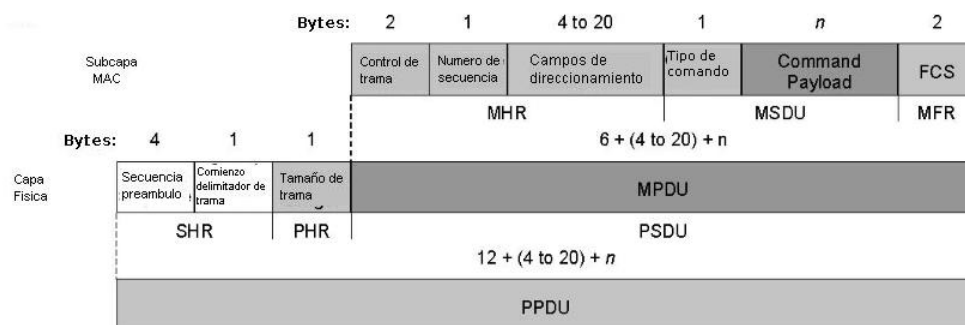


Fig. A.4. Trama de comandos MAC

Tipo de comando: Indica que tipo de comando solicita entre un total de 9 distintos.

Command payload: Su longitud es variable en función del comando utilizado.

A.1. Campos comunes en todas las tramas

Como podemos ver, cada trama tiene una función distinta muy localizada, excepto la trama de comandos MAC que puede llevar informaciones muy diversas sobre primitivas de servicio.

Los campos comunes en la cabecera MAC (MHR) son:

Control de trama: Nos permite escoger entre las 4 clases de tramas descritas.

Numero de secuencia: Este campo irá en función del paquete de origen y se incrementará para cada clase de trama que sea enviada.

Campos de direccionamiento: En función de si usamos direcciones cortas (16bit) o largas (48bits) de origen y de destino, en las direcciones largas los 20 primeros bits identifican el fabricante del dispositivo y los 28 restantes identifican el dispositivo, las direcciones cortas las asigna el PAN coordinator una vez ha recibido al dispositivo con la dirección larga y se realiza este cambio para tener más espacio en la trama para transmitir datos útiles.

ANEXO B. Configuración parámetros chip CC2420

A continuación se muestran y explican los parámetros más importantes de los dos archivos de configuración básicos que definen las variables del chip CC2420

B.1. Archivo de configuración AM.h

En este apartado se pueden ver las sentencias más importantes definidas en el archivo de configuración AM.h con sus valores por defecto.

```
#define DEF_TOS_AM_GROUP 0x7d //define el valor del campo grupo. Por defecto 125

#define TOSH_AM_LENGTH 1 //define el valor del campo grupo. Por defecto toma valor 1

typedef struct TOS_Msg //Estructura que define los campos del mensaje TOS
{
```

Los siguientes campos son los que se transmiten y/o se reciben por medio radio

```
uint8_t length; //Este campo indica cuantos bytes enviamos en el campo datos.
```

```
uint8_t fcfhi; //Los campos fcfhi y fcflo son un registro de 16 bits (parte hi y lo) donde cada grupo de bits indican una característica de la transmisión (utilización ack o el tipo de trama utilizada, etc).
uint8_t fcflo;
```

```
uint8_t dsn; //Secuencia de paquetes enviados, en cada nuevo paquete se incrementa en 1, este campo es rellenado por el sistema TinyOS.
```

```
uint16_t destpan; //Indica la dirección de la red de área local a la que va destinado el paquete.
```

```
uint16_t addr; //Indica la dirección del mote al que va enviado el paquete.
```

```
uint8_t type; //Indica el tipo de paquete que enviamos.
```

```
uint8_t group; //Indica a que grupo pertenece el mote receptor
```

```
int8_t data[TOSH_DATA_LENGTH]; // Campo donde metemos la información útil que queremos transmitir
```

Los siguientes campos no se transmiten y/o reciben por medio radio.

```
uint8_t strength; //Este campo nos indica la potencia de recepción.
```

```
uint8_t lqi; //Indica la calidad del enlace por el cual ha recibido un determinado paquete.
```

```
bool crc; //Este campo nos indica si el crc es correcto.
```

```
uint8_t ack; //Indica si recibimos reconocimiento de una determinada trama.
```

```
uint16_t time; //Referencia interna del procesador.
```

```
} TOS_Msg;
```

Estos campos no son modificables una vez nos encontramos en medio de una comunicación por lo que toman el valor definido en este archivo de configuración.

B.2. Archivo de configuración CC2420Const.h

Otro archivo muy importante es el CC2420Const.h, donde por ejemplo decidimos en que canal queremos transmitir. En nuestro caso decidimos trabajar en el canal 26 ya que es en el que menos interferencias se producen con otras tecnologías inalámbricas que también transmiten a la frecuencia de 2,4 GHz. Otra variable muy importante es la que define la potencia a la que queremos que transmitan los sensores.

A continuación mostramos los campos definidos en este archivo que nosotros consideramos importantes para nuestro trabajo.

```
// Tiempos para el CC2420 en microsegundos
enum {
CC2420_TIME_BIT = 4,
CC2420_TIME_BYTE = CC2420_TIME_BIT << 3,
CC2420_TIME_SYMBOL = 16
};

//En este campo definimos la potencia de recepción de la antena
CC2420. En este caso seleccionamos 0x1f que es 0 dBm, la potencia
máxima.
#define CC2420_DEF_RFPOWER 0x1f

//Definimos el canal por el que vamos a transmitir o recibir.
#define CC2420_DEF_CHANNEL 26

//Definimos la banda de frecuencia que vamos a utilizar, en este caso
seleccionamos la banda libre (2.405 GHz).
#define CC2420_DEF_PRESET 2405

//Definimos el campo de control de trama donde podemos controlar
cuestiones como el tipo de trama o si queremos o no el envío de ack.
#define CC2420_DEF_FCF_LO 0x08
#define CC2420_DEF_FCF_HI 0x01 // sin ACK
#define CC2420_DEF_FCF_HI_ACK 0x21 // con ACK
#define CC2420_DEF_FCF_TYPE_BEACON 0x00
#define CC2420_DEF_FCF_TYPE_DATA 0x01
#define CC2420_DEF_FCF_TYPE_ACK 0x02
```

ANEXO C.AD HOC ON-DEMAND VECTOR ROUTING PROTOCOL (RFC 3561)

C.1. Introducción

AODV es una de las propuestas adoptadas por el grupo MANET como protocolo de enrutamiento reactivo para redes ad hoc con movilidad. Además es una de las opciones generalmente más utilizadas y múltiples implementaciones del mismo han sido desarrolladas como la KERNEL-AODV NIST, AODV-UU Uppsala o UoB-JAdhoc AODV.

Una de las principales características que definen el funcionamiento de AODV es la utilización de tablas de enrutamiento para evitar la utilización de cabeceras grandes en los paquetes enviados.

Así cada entrada de la tabla de rutas además del destino y siguiente salto a utilizar incorpora un número de secuencia y un temporizador. La función del número de secuencia es diferenciar la información nueva de la antigua, de manera que evitamos la creación de bucles y utilización de rutas desconocidas.

En el caso del temporizador su utilización se requiere para evitar utilizar rutas de las que no se conoce su estado desde hace mucho tiempo.

Otra característica es que AODV no mantiene rutas hacia cada uno de los nodos de la red, estas rutas se descubren a medida que son requeridas. Este descubrimiento bajo demanda se realiza mediante una gestión petición/respuesta. AODV permite además transmisiones unicast, multicast y broadcast. La transmisión unicast permite el envío de datos de un nodo hacia otro nodo de la red, la transmisión multicast permite el envío de datos hacia un conjunto de nodos y la transmisión broadcast consiste en enviar información a toda la red.

C.2. Proceso de descubrimiento de rutas

Cuando un nodo requiere enviar datos a otro nodo, primero comprueba su tabla de rutas para buscar una entrada hacia el destino deseado. Si encuentra una entrada activa envía los datos por la ruta disponible, en caso contrario se inicia el descubrimiento de una nueva ruta.

Las peticiones para la creación de una ruta se envían mediante un mensaje específico denominado RREQ (*Route Request*) que contiene información del destino solicitado. Cada paquete se identifica mediante un identificador (*rreqid*) propio en cada nodo solicitante y que se incrementa a cada nueva solicitud de ruta. Esto permite a los nodos intermedios conocer cuando deben retransmitir una solicitud o cuando deberán rechazarla. Cuando un nodo reenvía un RREQ

incorpora una ruta inversa en su tabla de rutas apuntando al origen del RREQ, para devolver la información relativa a la ruta al origen de la petición.

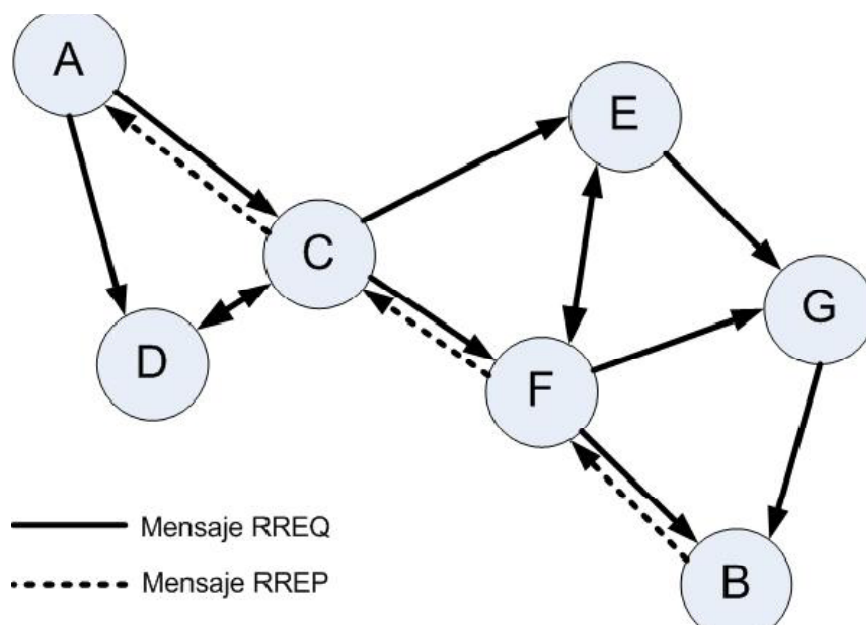


Fig. C.1. Proceso de descubrimiento de una ruta entre A y B.

El mensaje RREQ será respondido por cualquier nodo que conozca de una ruta hacia el destino solicitado. Esta respuesta se envía mediante otro mensaje específico, el RREP (*Route Reply*), por la ruta inversa a la que ha sido solicitada. En este proceso de vuelta se actualizan las tablas de los nodos intermedios para introducir la entrada hacia el destino.

Una vez el destino recibe un RREP, este almacena la petición en la cache para futuros envíos al mismo destino. Entonces las otras respuestas recibidas sobrescribirán esta entrada de cache si estas son mejores que la almacenada, esto se decide mediante una elección basada en el número de saltos de la ruta empleada.

C.3. Mantenimiento de rutas

Cuando obtenemos una ruta entre dos nodos, se considera ruta válida durante un periodo de tiempo limitado. Esto es necesario en escenarios con movilidad donde no se puede asegurar la posición de un nodo y la ausencia de interferencias. Ante estas situaciones una ruta puede dejar de ser la óptima o incluso no ser válida. Como primer mecanismo se emplea un protocolo por el que cada nodo que forma parte de una ruta inválida la misma si se mueve de su posición inicial, de la misma forma como en ocasiones no es posible conocer el movimiento del mismo se emplea la utilización de mensajes HELLO.

La utilización de estos mensajes consiste en enviar periódicamente un mensaje indicando la presencia del nodo en el escenario. Cada nodo que recibe el mensaje añade información del nodo en una tabla de vecinos y actualiza la información relativa al temporizador. Cuando un temporizador acaba, se invalidan todas las rutas pasando por el nodo del que no tenemos confirmación de presencia.

La invalidación de una ruta se realiza mediante el envío de un mensaje RERR (*route error*). Este mensaje se origina en cualquier nodo de una ruta que detecte un error en el envío de un mensaje o la desaparición de un nodo mediante los mensajes de HELLO. Entonces el RERR se envía a través de la ruta del mensaje hasta el origen, y este nodo deberá originar una nueva búsqueda si lo considera necesario.

ANEXO D.TINYOS

D.1. Instalación TinyOs para Windows

Requisitos para la instalación:

- El CD-ROM de CrossBow que contiene las herramientas de soporte de TinyOS.
- Ordenador con SO Microsoft Windows (XP, 2000, NT), con 1 GB de espacio libre en la unidad de disco de destino de la instalación.
- Tener los programas: WinZip, Adobe Acrobat y Programmers Notepad (disponible gratuitamente en <http://www.pnotepad.org/>).

La instalación de TinyOS para Windows es sencilla. En primer lugar se tiene que instalar el archivo ejecutable *tinyos-1.1.0-1is.exe*, el cual nos instalará la plataforma básica de desarrollo Cygwin, las librerías nesC de TinyOS y el compilador nesC.

El entorno de instalación de este ejecutable es muy intuitivo, guiado en todos sus pasos. Este archivo se puede encontrar en:

- El CD de CrossBow, dentro de la carpeta *TinyOS Install*.
- En el servidor de descargas de la plataforma Windows de www.tinyos.net que se encuentra en la siguiente página de internet <http://www.tinyos.net/dist-1.1.0/tinyos/windows/>

Después de la instalación de la plataforma TinyOS, la consola Cygwin y compilador de nesC, concentrados en el archivo *tinyos-1.1.0-1is.exe*, procederemos a la instalación del paquete rpm que nos actualizará a la última versión del compilador nesC. El nombre del paquete es el *nesc-1.1.2a-1.cygwin.i386.rpm* y se puede encontrar en el mismo servidor de descargas anterior (<http://www.tinyos.net/dist-1.1.0/tinyos/windows/>).

La instalación de este paquete se tiene que efectuar desde la consola Cygwin, dentro del directorio donde se encuentre el paquete rpm. La secuencia a escribir es la siguiente:

```
rpm --force --ignoreos -Uvh nesc-1.1.2a-1.cygwin.i386.rpm
```

Finalmente se instalará la actualización de TinyOS que se desee, como requisito necesita tener instalado el anterior paquete. La actualización más reciente es la 1.1.15 de diciembre de 2005, y para descargarla sólo se tiene que ir al servidor de descargas de Windows ya indicado por duplicado anteriormente. El link a presionar es el [tinyos-1.1.15Dec2005cvs-1.cygwin.noarch.rpm](http://www.tinyos.net/dist-1.1.15Dec2005cvs-1.cygwin.noarch.rpm).

La instalación de este segundo paquete rpm se instalará de semejante forma al anterior paquete:

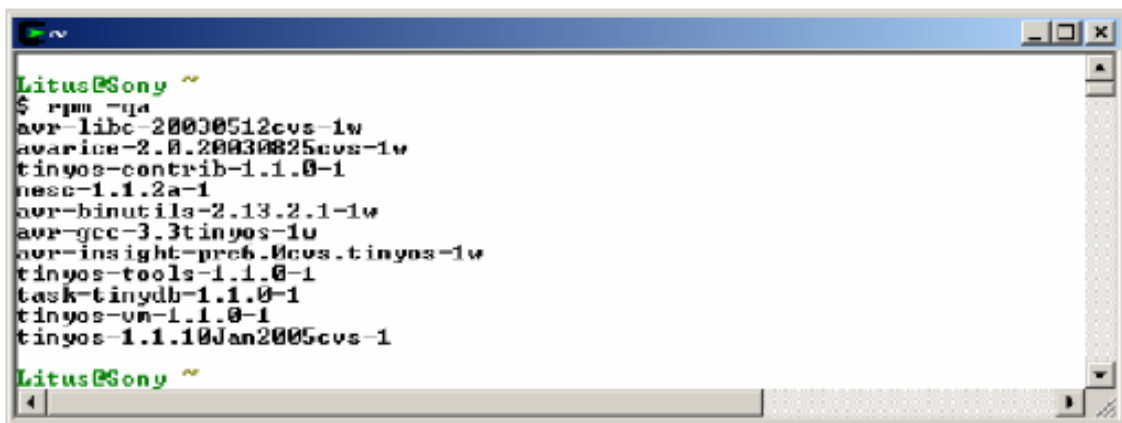
```
rpm --force --ignoreos -Uvh tinyos-1.1.15Dec2005cvs-1.cygwin.noarch.rpm
```

La aplicación para los motes TelosB realizadas en este trabajo han sido realizadas con las versiones de TinyOS:

- [tinyos-1.1.7July2004cvs-2.cygwin.noarch.rpm](#)
- [tinyos-1.1.10Jan2005cvs-1.cygwin.noarch.rpm](#)

La carpeta xbow del CD de CrossBow dentro de la carpeta *TinyOS Updates* se copiará en la carpeta de nuestro disco C:\tinyos\cygwin\opt\tinyos-1.x\contrib. De esta forma se podrá comenzar a trabajar desde el directorio C:\tinyos\cygwin\opt\tinyos-1.x\contrib\xbow\apps que se ha copiado, programando nuestras aplicaciones en nesC.

Con el comando **rpm -qa** en la consola Cygwin se puede comprobar la correcta instalación de TinyOS y los paquetes de actualización.



```
Litus@Sony ~
$ rpm -qa
avr-libc-20030512cvs-1w
avarice-2.0.20030825cvs-1w
tinyos-contrib-1.1.0-1
nesc-1.1.2a-1
avr-binutils-2.13.2.1-1w
avr-gcc-3.3tinyos-1w
avr-insight-preh.0cvs.tinyos-1w
tinyos-tools-1.1.0-1
task-tinydb-1.1.0-1
tinyos-un-1.1.0-1
tinyos-1.1.10Jan2005cvs-1
Litus@Sony ~
```

Fig. D.1 Ejemplo del comando de comprobación “rpm -qa”

D.2. Directorio TinyOS

Para trabajar con el directorio de TinyOS se utilizará la consola Cygwin que con los comandos de linux nos permite explorar el directorio.

El directorio de carpetas de TinyOS está compuesto de numerosas carpetas y archivos. Este apartado pretende mostrar el contenido de las carpetas más importantes de TinyOS y su estructura.

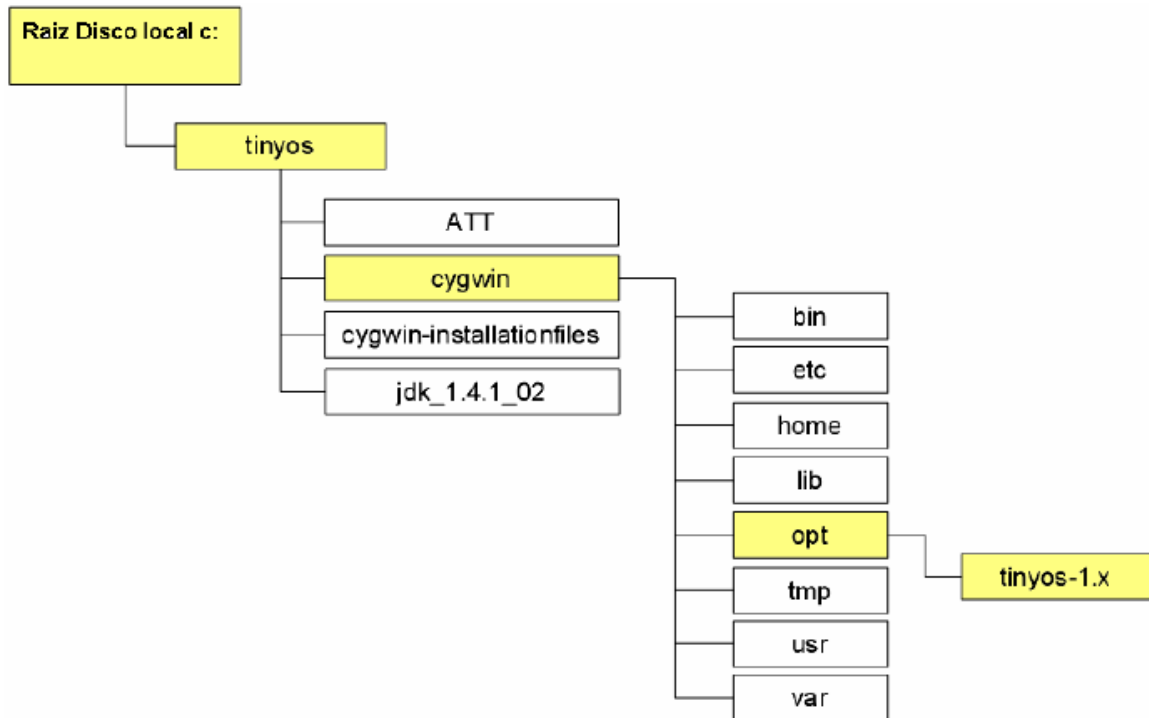


Fig. D.2 Estructura principal desde la raíz del directorio TinyOS

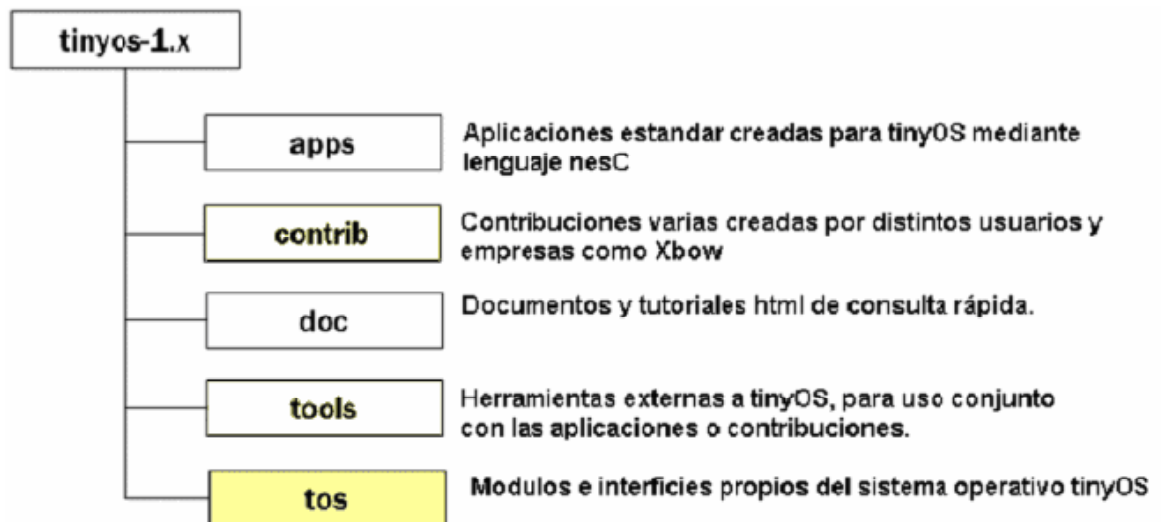


Fig. D.3 Estructura y contenido del directorio tinyos-1.x

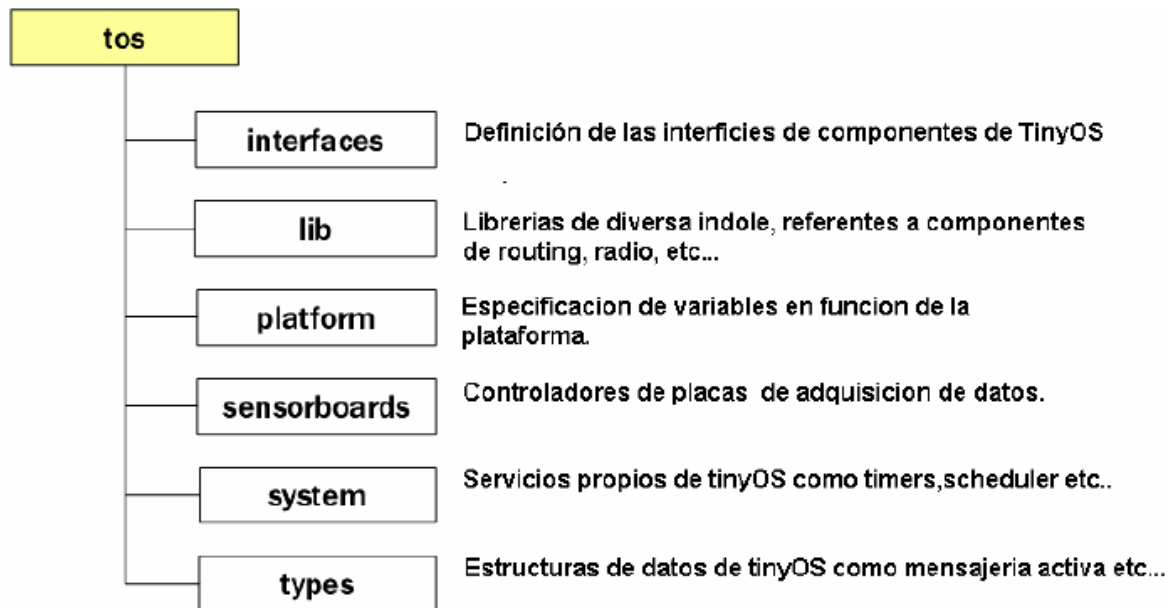


Fig. D.4 Estructura y contenido del directorio tos dentro de la carpeta tinyos-1.x

D.3. Compilación de aplicaciones en TinyOS

La sintaxis que se utiliza en la consola Cygwin para compilar las aplicaciones es la siguiente:

```
make <plataforma> install <dirección dispositivo> <programador>,<puerto>
o
make <plataforma> reinstall <programador>,<puerto>
```

donde:

- plataforma es el nombre de la plataforma, en este caso "telosb".
- dirección dispositivo es un número que es proporcionado mediante UART protocol. Se utiliza el comando "motelist" para saber este número.
- programador es el nombre del microcontrolador, en este caso msp-430-bsl "bsl", cuyas siglas quieren decir Boot Strap Loader.
- puerto es el "COM-1", donde COM es la dirección del dispositivo.

La diferencia entre install o reinstall está detallada a continuación.

- Install.<n>: Compila la aplicación para la plataforma seleccionada, puede configurar el Identificador de nodo (<n>) del mote y lo programa.
- Reinstall.<n>: Puede también configurar el Identificador de nodo y instala el programa precompilado, no recompila. Esta opción es sustancialmente más rápida.