

CARLOS HERNÁN TOBAR ARTEAGA



DISEÑO DE UN MOTOR DE APRENDIZAJE EN REDES DE SENSORES
INALÁMBRICAS COGNITIVAS

Tesis de Maestría

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Maestría en Electrónica y Telecomunicaciones

Popayán
2012

CARLOS HERNÁN TOBAR ARTEAGA

DISEÑO DE UN MOTOR DE APRENDIZAJE EN REDES DE SENSORES
INALÁMBRICAS COGNITIVAS

Tesis presentada a la Facultad de Ingeniería Electrónica y Telecomunicaciones de la
Universidad del Cauca para la obtención del título de

Magister en
Electrónica y Telecomunicaciones

Director
Magister Guefry Leider Agredo Méndez

Popayán
2012

Agradecimientos

Al Magister Guefry Leider Agredo Méndez, docente del departamento de Telecomunicaciones de la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca, por su contribución en el desarrollo del presente trabajo de investigación.

Al Departamento de Telecomunicaciones, y al grupo de I+D en Nuevas Tecnologías en Telecomunicaciones de la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca, por su apoyo durante el desarrollo del presente trabajo de investigación.

Resumen

En una red de sensores inalámbrica la comunicación entre los nodos se realiza mediante canales de radiofrecuencia, por lo general de poco ancho de banda, y asignados en bandas de frecuencia compartidas con otras tecnologías y estándares de comunicación. Esto hace que los enlaces de radio varíen en tiempo y espacio, donde el ruido e interferencia pueden comprometer el desempeño en la transferencia de los datos, razón por la cual se requiere el establecimiento de mecanismos de cognición, que les permitan ser conscientes del ambiente de radio cercano, tomar decisiones sobre cambios en sus parámetros de operación y comunicarlos a los demás nodos de la red, con el objetivo de lograr un mejor desempeño en la transmisión de los datos sobre los canales de radio.

En esta tesis se presenta un modelo de redes de sensores inalámbricas cognitivas que incorpore las funciones de sensado, aprendizaje y razonamiento, y propone un motor de aprendizaje que lo implemente; el cual es soportado por una red neuronal con alimentación hacia delante.

Palabras clave: redes de sensores inalámbricas, radio cognitivo, redes neuronales.

Abstract

The communication between nodes in a wireless sensor network is done through radiofrequency channels. They have low bandwidth and are allocated in frequency bands shared with others technologies and communication standards. This implies that the radio links fluctuate in time and space, and the performance on transfer data is degraded by noise and interference. Therefore it is required to incorporate novel cognition mechanisms to the nodes of the sensor network. It can allow awareness of close radio environment, to make decisions about changes on operating parameters and send messages for signaling and coordinating to other nodes of the network. The goal is to get an improved performance.

This work presents a model of cognitive wireless sensor network that incorporates functions of sensing, learning and reasoning, and proposes a learning engine for its implementation; it is supported through the use of a feed forward neural network.

Keywords: Wireless sensor networks, cognitive radio, neural networks.

Contenido

	Pág.
Lista de tablas	x
Lista de figuras	xi
Lista de abreviaturas	xiii
Introducción.....	1
1.1 Justificación de la tesis	3
1.2 Contribución de la tesis.....	4
1.3 Organización de la tesis.....	4
Marco teórico.....	7
2.1 Redes de sensores inalámbricas	7
2.2 Radio cognitivo.....	13
2.3 Técnicas de inteligencia artificial.....	16
2.3.1 Redes neuronales artificiales.....	16
2.3.2 Computación evolutiva.....	21
2.3.3 Lógica difusa.....	21
2.4 Conclusiones.....	24
Modelo de redes de sensores inalámbricas cognitivas	27
3.1 Modelo conceptual.....	27

3.2 Arquitectura de red.....	29
3.3 Arquitectura de los nodos.....	31
Mecanismo de aprendizaje en redes de sensores inalámbricas cognitivas	35
4.1 Sensado	35
4.2 Aprendizaje	38
4.2.1 Método de aprendizaje red neuronal.....	39
4.2.2 Algoritmo de cálculo red neuronal.....	43
4.3 Control.....	44
4.4 Persistencia de datos	47
Escenarios de operación del motor de aprendizaje y métricas de desempeño	51
5.1 Selección dinámica de canal.....	53
5.1.1 CWSN en estrella.....	53
5.1.2 CWSN en árbol de <i>clusters</i>	61
5.2 Selección de capa física.....	65
5.2.1 CWSN en estrella.....	65
5.2.2 CWSN en árbol de clusters.....	74
5.3 Métricas de desempeño	78
Conclusiones y recomendaciones	81
Análisis cuantitativo de las capas físicas de IEEE 802.15.4.....	85
A.1 Capa física IEEE 802.15.4 en 2400 MHz.....	86
A.2 Capa PHY BPSK en 868/915 MHz	89
A.3 Capa PHY ASK en 868/915 MHz.....	89
A.4 Capa PHY O-QPSK en 868/915 MHz.....	90
A.5 Análisis de coexistencia	92
A.5.1 Desempeño de coexistencia en la banda de 2400 MHz.....	92
Simulador de red NS-2 para el análisis de desempeño de IEEE 802.15.4.....	95
El método Levenberg-Marquardt para problemas de aproximación de curva utilizando mínimos cuadrados no lineales	99

C.1 El método de gradiente descendente	100
C.2 El método Gauss-Newton.....	101
C.3 El método Levenberg-Marquardt.....	102
C.4 Implementación numérica	103
C.5 Análisis de error.....	103
Bibliografía	105

Lista de tablas

	Pág.
Tabla 2.1. Capas físicas IEEE 802.15.4	11
Tabla 5.2. Pesos y umbrales obtenidos para la red neuronal cuando no se presenta interferencia.....	59
Tabla 5.3. Pesos y umbrales obtenidos para la red neuronal cuando se presenta interferencia.....	61
Tabla 5.4. Pesos y umbrales obtenidos para la red neuronal PHY BPSK.....	71
Tabla 5.5. Pesos y umbrales obtenidos para la red neuronal PHY ASK	71
Tabla A.1. Máscara espectral de potencia en 802.11b.....	93
Tabla B.1. Desempeño de la tasa de entrega de paquetes en una red IEEE 802.15.4	98

Lista de figuras

	Pág.
Figura 2.1. WSN para el monitoreo de incendios en zonas forestales	8
Figura 2.2. Nodos y topologías de red	10
Figura 2.3. Estructura de supertrama en IEEE 802.15.4.....	12
Figura 2.4. Ciclo de cognición	14
Figura 2.5. Concepto de hueco de espectro	15
Figura 2.6. Modelo de una neurona	18
Figura 2.7. Ejemplos de funciones de activación	19
Figura 3.1. Modelo conceptual de una CWSN	29
Figura 3.2. CWSN en estrella.....	30
Figura 3.3. CWSN nodal.....	30
Figura 3.4. Arquitectura nodo sensor	31
Figura 3.5. Arquitectura nodo cabecera	32
Figura 3.6. Arquitectura nodo coordinador	33
Figura 4.1. Diagrama de flujo función de sensado	37
Figura 4.2. Diagrama de flujo función de aprendizaje	38
Figura 4.3. Diagrama de flujo función de control para selección de canal	45
Figura 4.4. Diagrama de interacción función de control	46
Figura 4.5. Motor de aprendizaje en CWSN.....	47
Figura 4.6. Red neuronal para caracterización de desempeño.....	40
Figura 4.7. Función de activación: a) tangsig, b) lineal	40
Figura 5.1. Escenarios de operación motor de aprendizaje	52

Figura 5.2. Interacciones función de sensado	56
Figura 5.3. Interacciones función de aprendizaje	57
Figura 5.4. Interacciones función de razonamiento	57
Figura 5.5. Caracterización del desempeño canales sin interferencia.....	58
Figura 5.6. Caracterización del desempeño en presencia de una fuente 802.11b interferente	60
Figura 5.7. Interacciones función de sensado	63
Figura 5.8. Interacciones función de aprendizaje	64
Figura 5.9. Interacciones función de razonamiento	64
Figura 5.10. Interacciones función de sensado	68
Figura 5.11. Interacciones función de aprendizaje	69
Figura 5.12. Interacciones función de razonamiento	69
Figura 5.13. Caracterización del desempeño sin interferencia PHY ASK	70
Figura 5.14. Caracterización del desempeño sin interferencia PHY BPSK	71
Figura 5.15. Caracterización del desempeño PHY ASK en presencia de interferencia	73
Figura 5.16. Caracterización del desempeño PHY BPSK en presencia de interferencia.....	74
Figura 5.17. Interacciones función de sensado	76
Figura 5.18. Interacciones función de aprendizaje	77
Figura 5.19. Interacciones función de razonamiento	77
Figura 5.20. Desempeño algoritmo de aprendizaje en el escenario selección de canal.....	79
Figura 5.21. Desempeño algoritmo de aprendizaje en para la capa PHY ASK.....	79
Figura 5.22. Desempeño algoritmo de aprendizaje en para la capa PHY BPSK.....	80
Figura A.1. Modelo de simulación IEEE 802.15.4 capa física DSSS-OQPSK.....	86
Figura A.2. Desempeño capa física O-QPSK IEEE 802.15.4 en la banda de 2400 MHz	88
Figura A.3. Modelo funcional de referencia capa PHY BPSK.....	89
Figura A.4. Modelo funcional de referencia capa PHY ASK.....	90
Figura A.5. Desempeño de error capas PHY IEEE 802.15.4 en la banda de 915 MHz	91
Figura A.6. Resultados de simulación desempeño de coexistencia	93
Figura B.1. Proceso de simulación IEEE 802.15.4/ZigBee	96

Lista de abreviaturas

ANN:	Red Neuronal Artificial, <i>Artificial Neural Network</i> .
AWGN:	Ruido Gaussiano Aditivo Blanco, <i>Additive White Gaussian Noise</i> .
ASK:	Modulación por Desplazamiento de Amplitud, <i>Amplitude Shift Keying</i> .
BER:	Tasa de Error de Bit, <i>Bit Error Rate</i> .
BPSK:	Modulación por Desplazamiento de Fase Binaria, <i>Binary Phase Shift Keying</i> .
CAP:	Periodo de Acceso por Contención, <i>Contention Access Period</i> .
CFP:	Periodo Libre de Contención, <i>Contention-Free Period</i> .
CR:	Radio Cognitivo, <i>Cognitive Radio</i> .
CRAHN:	Red de Radio Cognitivo Ad Hoc, <i>Cognitive Radio Ad Hoc Networks</i> .
CSMA-CA:	Acceso Múltiple por Sensado de Portadora con Evasión de Colisión, <i>Carrier Sense Multiple Access – Collision Avoided</i> .

- CWSN: Red de Sensores Inalámbrica Cognitiva, *Cognitive Wireless Sensor Network*.
- DSSS: Espectro Ensanchado de Secuencia Directa, *Direct Sequence Spread Spectrum*.
- EA: Algoritmo Evolutivo, *Evolutionary Algorithm*.
- EC: Computación Evolutiva, *Evolutionary Computing*.
- ED: Detección de Energía, *Energy Detection*.
- EEC: Códigos para Estimación de Error, *Error Estimating Codes*.
- FFD: Dispositivo de Función Completa, *Full Function Device*.
- FL: Lógica Difusa, *Fuzzy Logic*.
- GA: Algoritmo Genético, *Genetic Algorithm*.
- GP: Programación Genética, *Genetic Programming*.
- GTS: Intervalo de Tiempo Garantizado, *Guaranteed Time Slot*.
- ISM: Industrial, Científico y Médico, *Industrial, Scientific and Medical*.
- LQI: Indicación de Calidad de Enlace, *Link Quality Indication*.
- LR-WPAN: Red de Área Personal Inalámbrica de Baja Tasa de Transferencia de Información, *Low-Rate Wireless Personal Area Network*.
- MAC: Subcapa de Control de Acceso al Medio, *Medium Access Control*.
- O-QPSK: Modulación por Desplazamiento de Fase en Cuadratura Offset, *Offset Quadrature Phase Shift Keying*.

OSI:	Interconexión de Sistemas Abiertos, <i>Open System Interconnection</i> .
PHY:	Capa Física, <i>Physical Layer</i> .
PIB:	Base de Información de Red de Área Personal, <i>PAN Information Base</i> .
PN:	Pseudo-Ruido, <i>Pseudo-Noise</i> .
PSD:	Densidad Espectral de Potencia, <i>Power Spectral Density</i> .
PSSS:	Espectro Ensanchado de Secuencia Paralela, <i>Parallel Sequence Spread Spectrum</i> .
QoS:	Calidad de Servicio, <i>Quality of Service</i> .
RFD:	Dispositivo de Función Reducida, <i>Reduced Function Device</i> .
SDR:	Radio Definido por Software, <i>Software Defined Radio</i> .
SINR:	Relación Señal a Ruido más Interferencia, <i>Signal to Interference and Noise Ratio</i> .
UML:	Lenguaje de Modelado Unificado, <i>Unified Modeling Language</i> .
WSN:	Red de Sensores Inalámbrica, <i>Wireless Sensor Network</i> .

Capítulo 1

Introducción

Una red de sensores inalámbrica (WSN, *Wireless Sensor Network*) es una red de dispositivos, denominados nodos, que recogen información de variables físicas del ambiente donde se despliegan y la comunican a un nodo coordinador a través de enlaces inalámbricos. El nodo coordinador puede utilizar los datos recolectados localmente o transferirlos a una aplicación para que sean analizados y procesados por parte de los usuarios. Cada nodo está equipado con una unidad de procesamiento, una interfaz de comunicación inalámbrica, así como sensores y/o actuadores. Las aplicaciones de una WSN son muy diversas, pueden citarse por ejemplo, el monitoreo ambiental, la automatización, la salud, y la seguridad, entre otras.

Una WSN dispone de recursos limitados: potencia, memoria y capacidades de procesamiento, esto se debe a que la mayoría de los nodos operan a partir de baterías, y muchas veces en lugares donde no es fácil restituirlas, y por lo tanto se espera que duren por tiempos prolongados. El rango de alcance de los nodos es relativamente corto, alrededor de decenas de metros, así que para dar cubrimiento a espacios mayores, los nodos transmiten los datos recolectados utilizando múltiples saltos, dando lugar a diversas topologías de red, entre ellas en estrella y nodales.

La comunicación entre los nodos se realiza mediante enlaces inalámbricos, definidos a partir de canales de radiofrecuencia, por lo general de poco ancho de banda, y asignados en bandas de frecuencia compartidas con otras tecnologías y estándares de comunicación. Esto hace que los enlaces de radio varíen en tiempo y espacio,

donde el ruido e interferencia pueden comprometer el desempeño en la transmisión y recepción de los datos. De esta manera, es necesario que las WSNs dispongan de cierto grado de cognición que les permita ser conscientes de su ambiente de radio y se adapten a él con el fin de lograr una comunicación más confiable. Es decir, puedan usar de forma eficiente los recursos disponibles: espectro y potencia.

El concepto de radio cognitivo (CR, *Cognitive Radio*) fue definido como un mecanismo que proporciona cierta inteligencia a los dispositivos, de tal manera que puedan ser conscientes de su ambiente de radio, tomen decisiones de acuerdo a las medidas realizadas y ajusten sus parámetros de operación para adaptarse a las condiciones actuales y así presenten un mejor desempeño. CR es una tecnología prometedora para manejar el problema del uso del espectro radioeléctrico, el cual en la actualidad es escaso debido a la gran proliferación de tecnologías inalámbricas. Para ello, el mecanismo de cognición identifica huecos de espectro disponibles en bandas licenciadas y los utiliza de forma oportunista, desde luego sin crear dificultad alguna a los usuarios licenciados cuando éstos requieran el uso del canal; o en un contexto de uso compartido de espectro, identifica el mejor canal disponible para su uso y así lograr un mejor desempeño en la transmisión de los datos.

Una WSN que utilice mecanismos de cognición con el fin de mejorar su desempeño puede ser vista como una red de sensores inalámbrica cognitiva (CWSN, *Cognitive Wireless Sensor Network*), es decir una red que sea consciente de su ambiente y realice cambios adaptativos a su modo de operación, o utilice espacios blancos de frecuencia de forma oportunista.

Esta tesis propone un modelo de red de sensores inalámbrica cognitiva que incorpore los mecanismos de cognición: sensado, aprendizaje y razonamiento; y permita la selección dinámica de canal o de capa física y así lograr un mejor desempeño en la transmisión y recepción de los datos. Así mismo el diseño de un motor de aprendizaje que los implemente, centrando esfuerzos en el nivel físico del modelo de referencia de interconexión de sistemas abiertos (OSI, *Open System Interconnection*).

La metodología empleada es una metodología de diseño de sistema, donde se especificaron los componentes de una CWSN, se definió su funcionamiento, y se

analizaron a partir de dos escenarios de aplicación: 1) la selección de canales de radio con menor ruido e interferencia, para un esquema de uso compartido de la banda de operación, y 2) la selección de capas físicas para logra un mejor desempeño de error. Se ha especificado con suficiente detalle una CWSN y se ha diseñado un motor de aprendizaje utilizando redes neuronales con alimentación hacia delante para la caracterización de desempeño. La valoración del desempeño del motor de aprendizaje se realizó mediante simulación utilizando Matlab.

1.1 Justificación de la tesis

Generalmente las redes de sensores inalámbricas operan en bandas no licenciadas donde coexisten con otras tecnologías inalámbricas, y por ende el espectro de radio es más escaso y con mayores niveles de ruido e interferencia. Si los nodos de una WSN disponen de ciertas funcionalidades de cognición: sensado, aprendizaje y razonamiento, pueden determinar el canal con menor nivel de ruido e interferencia y utilizarlo para realizar la transmisión y recepción de datos; este proceso puede ser dinámico permitiéndole a la red de sensores mantener un desempeño deseado.

En la actualidad, radio cognitivo está en una fase naciente, y las comunidades científica y académica se encuentran realizando diferentes investigaciones con el fin de establecer su viabilidad, estructura y métodos que implementará. Por lo tanto, los resultados alcanzados en este proyecto constituyen bases de diseño que pueden ser de interés, y pueden permitir la formulación de nuevos proyectos de investigación y desarrollo, tanto al interior de la Facultad de Ingeniería Electrónica y Telecomunicaciones como hacia el exterior.

Es pertinente pensar que la incorporación de cognición en una WSN involucre la posibilidad de utilizar técnicas de inteligencia artificial que permitan lograr dicha cognición. En esta tesis se estudia la posibilidad de utilizar redes neuronales para el diseño del motor de aprendizaje, en especial como mecanismo para caracterizar el desempeño de error en la transmisión y recepción de datos, y así tomar decisiones más aproximadas a las condiciones reales del ambiente de radio y desempeño a nivel físico que mejore la operación de la WSN. Los resultados logrados son

importantes para establecer la factibilidad del uso de estos métodos y su alcance en el contexto de radio cognitivo.

1.2 Contribución de la tesis

El presente estudio tiene un significado importante en el avance de las redes de sensores inalámbricas cognitivas, plantea un modelo de red que cubre las funciones definidas en radio cognitivo y demuestra cómo puede implementarse utilizando redes neuronales como técnica de inteligencia artificial.

Los resultados de este trabajo contribuyen al estudio y desarrollo de temáticas de interés en el Departamento de Telecomunicaciones de la FIET, y contribuyen a generar visibilidad de su que hacer a nivel nacional, mediante la divulgación de los resultados alcanzados que se hagan de este proyecto.

Se elaboraron dos artículos científicos para objeto de publicación de los resultados alcanzados en esta Tesis, los cuales fueron arbitrados y recibieron su aprobación en: 1) I Seminario Taller Latinoamericano de Instrumentación, Control y Telecomunicaciones (SICOTEL2012), ISBN 978-958-46-0496-5, organizado por la Universidad del Quindío, artículo: "*Modelo de Redes de Sensores Inalámbricas Cognitivas*"; y 2) Revista Entre Ciencia e Ingeniería, de la Universidad Católica de Pereira, ISSN 1909-8367, artículo: "*Selección Dinámica de Canal en IEEE 802.15.4 utilizando Redes Neuronales Artificiales*".

1.3 Organización de la tesis

El documento se encuentra organizado de la siguiente manera:

Capítulo 1. Introducción

Capítulo 2. Marco teórico

Capítulo 3. Modelo de redes de sensores inalámbricas cognitivas

Capítulo 4. Mecanismo de aprendizaje en redes de sensores inalámbricas cognitivas

Capítulo 5. Escenarios de operación del motor de aprendizaje y métricas de desempeño

Conclusiones y recomendaciones

Anexo A. Análisis cuantitativo de las capas físicas de IEEE 802.15.4

Anexo B. Simulador de red NS-2 para el análisis de desempeño de IEEE 802.15.4

Anexo C. El método Levenberg-Marquardt para problemas de aproximación de curva utilizando mínimos cuadrados no lineales

Capítulo 2

Marco teórico

Este capítulo presenta una descripción de los fundamentos teóricos de las tecnologías estudiadas en esta tesis: redes de sensores inalámbricas, radio cognitivo y técnicas de inteligencia artificial.

2.1 Redes de sensores inalámbricas

Una red de sensores inalámbrica (WSN, *Wireless Sensor Network*) es una red de dispositivos, denominados nodos, que recogen información de variables físicas del ambiente donde se despliegan y la comunican a un nodo coordinador a través de enlaces inalámbricos. Una WSN puede ser utilizada para diferentes aplicaciones, tales como vigilancia y seguridad, monitoreo y protección del ambiente, monitoreo y cuidado de la salud, y automatización doméstica, entre otras (Briascó et al., 2006).

Los sensores utilizados pueden ser de diferentes tipos, tales como sísmicos, magnéticos, térmicos, visuales, infrarrojos, acústicos, de radar, etc., y permiten monitorear una amplia variedad de condiciones ambientales entre las que se incluyen temperatura, humedad, movimiento vehicular, condiciones de luminosidad, presión, niveles de ruido, presencia o ausencia de cierta clase de objetos, niveles de fortaleza mecánica de objetos, o dirección, velocidad y tamaño de un objeto, entre otras. Además, pueden realizar un sensado continuo, detectar e identificar eventos, determinar ubicaciones, y realizar control local a través de actuadores. El patrón de

interacción entre los nodos puede ser uno a muchos, muchos a muchos o muchos a uno, y además pueden ser móviles (Mottola & Picco, 2006).

Por ejemplo, algunas aplicaciones ambientales de las redes de sensores incluyen el seguimiento y registro de los movimientos de aves, animales pequeños e insectos; el monitoreo de las condiciones ambientales que afectan las plantas y los animales; la irrigación; macro-instrumentos para el monitoreo de la tierra en gran escala y exploración del planeta; detección química/biológica; agricultura de precisión; monitoreo biológico, de la tierra y ambiental en el campo marino, terreno y atmosférico; detección de incendios forestales; investigación meteorológica o geofísica; detección de avalanchas; y estudio de contaminación (Akyildiz, 2002).

La figura 2.1 muestra un ejemplo típico de la aplicación de una WSN, en este caso los nodos sensores toman datos de contaminación o incendios en una amplia zona de recursos forestales, y comunican estos datos a una estación central para que sean monitoreados por los usuarios.

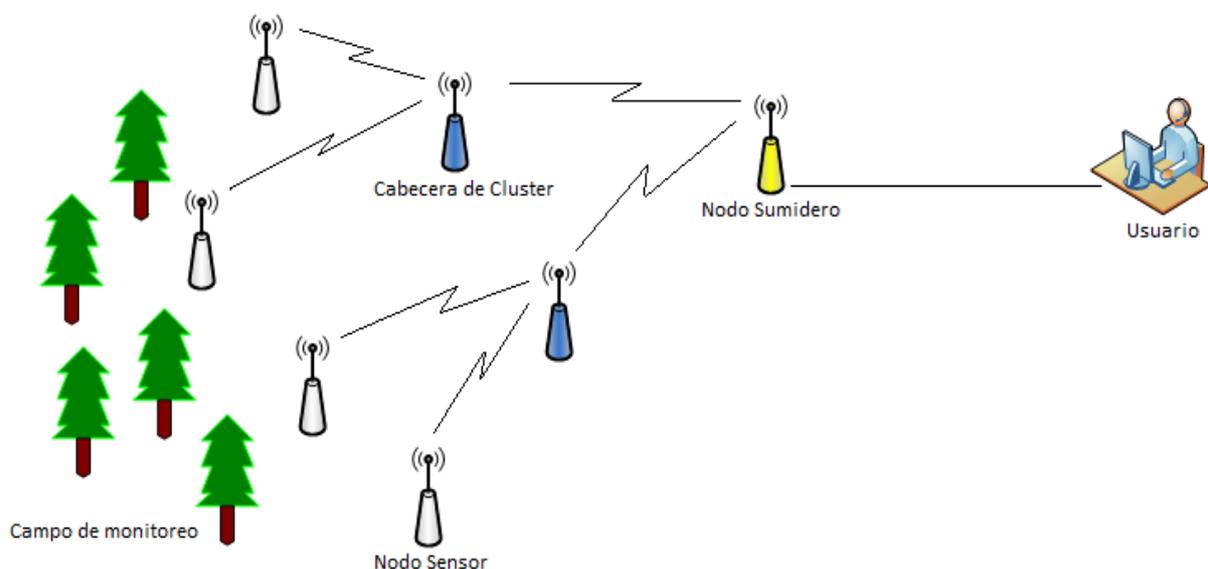


Figura 2.1. WSN para el monitoreo de incendios en zonas forestales

Los nodos sensores realizan procesamientos de propósito general y de red, además de las tareas de sensado específicas de la aplicación; pueden ser desplegados en

áreas densas ampliamente separadas mediante el agrupamiento de nodos sensores, *clusters*, coordinados por un nodo cabecera. Los nodos sensores transmiten sus datos a través de enlaces inalámbricos a la cabecera de *cluster* y ésta los comunica al nodo coordinador de la red o nodo sumidero; este nodo provee conectividad cableada o inalámbrica para que los usuarios monitoreen, analicen los datos recogidos y puedan tomar acciones de control en caso de ser necesarias.

El principal esfuerzo de estandarización en WSNs es la tecnología inalámbrica IEEE 802.15.4 (IEEE, 2006), la cual define una red de área personal inalámbrica de baja tasa de transferencia de información (LR-WPAN, *Low-Rate Wireless Personal Area Network*), y se caracteriza por su operación en las bandas de frecuencia para uso industrial, científico y médico (ISM, *Industrial, Scientific and Medical*), soporte para dispositivos de función completa (FFD, *Full Function Devices*) y de función reducida (RFD, *Reduced Function Devices*), bajo consumo de potencia, auto-organización, y conformación de topologías flexibles.

Una LR-WPAN IEEE 802.15.4 puede operar en dos topologías básicas: en estrella y par a par. En la topología en estrella la comunicación se establece entre dispositivos FFD o RFD y un único controlador central, llamado el coordinador de la red de área personal. Un dispositivo RFD o FFD típicamente tiene alguna aplicación asociada y es el punto de inicio o terminación para las comunicaciones de red, mientras el nodo coordinador puede también tener una aplicación específica, pero además es responsable del inicio, enrutamiento y finalización de la comunicación en la red, y por lo tanto puede contar con suministro continuo de potencia, mientras que los dispositivos RFD o FFD asociados probablemente usarán baterías.

Por su parte, la topología par a par también tiene un coordinador de red de área personal, sin embargo, se diferencia de la topología en estrella en que cualquier dispositivo puede comunicarse con los demás, siempre y cuando se encuentren en el mismo espacio de operación. Esta topología permite la conformación de redes más complejas, tales como las de árbol de *clusters*, y en malla. Una red par a par puede ser *ad hoc*, auto-organizada, auto-mejorada y puede permitir la comunicación a partir de múltiples saltos, es decir mediante el reenvío de mensajes de un nodo a otro. La figura 2.2 muestra ejemplos de estas topologías.

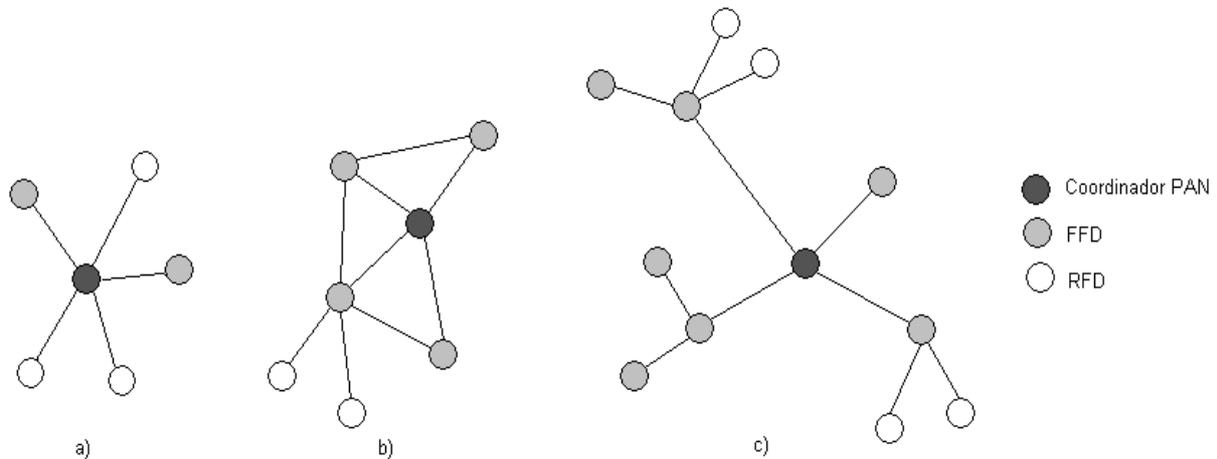


Figura 2.2. Nodos y topologías de red

Las capas físicas de IEEE 802.15.4 operan en tres bandas no licenciadas y emplean técnicas de espectro ensanchado. IEEE 802.15.4 especifica un total de 27 canales *half-duplex* distribuidos en las tres bandas de frecuencia y se organizan así: 1) Banda de 868 MHz: un solo canal con tasa de datos de 20 kbps, -92 dBm de sensibilidad en recepción y rango de transmisión ideal aproximadamente igual a 1 km; 2) Banda de 915 MHz: 10 canales de 40 kbps, sensibilidad y rango igual al anterior; y 3) Banda ISM 2.4 GHz: 16 canales de 250 kbps, sensibilidad requerida de -85 dBm y rango de transmisión ideal de 220 m. El rango de transmisión ideal es calculado considerando que, aunque cualquier potencia aceptable legalmente es permitida, los dispositivos que cumplan con IEEE 802.15.4 deben ser capaces de transmitir a -3 dBm. Además, los dispositivos que cumplan con el estándar, con el fin de usar la energía de forma eficiente, permanecen activos únicamente durante un tiempo corto.

IEEE 802.15.4 especifica la capa física (PHY, *Physical Layer*) y la subcapa de control de acceso al medio (MAC, *Medium Access Control*) de la pila de protocolos de OSI. Las funciones de la capa PHY son: activación y desactivación del transmisor y receptor de radio, detección de energía en el canal actual, indicación de calidad de enlace para tramas recibidas, percepción de canal libre, selección de la frecuencia del canal y, transmisión y recepción de los datos. Las funciones de la subcapa MAC son: generación de paquetes de sincronización (denominados *beacons*), soporte para asociación y des-asociación de red, acceso al canal y, provisión de enlace confiable entre dos entidades MAC.

IEEE 802.15.4 define cuatro capas PHY:

- Capa PHY de espectro ensanchado de secuencia directa (DSSS, *Direct Sequence Spread Spectrum*) en 868/915 MHz que emplea modulación por desplazamiento de fase binaria (BPSK, *Binary Phase Shift Keying*).
- Capa PHY de DSSS en 868/915 MHz que emplea modulación por desplazamiento de fase en cuadratura *offset* (O-QPSK, *Offset Quadrature Phase Shift Keying*).
- Capa PHY de espectro ensanchado de secuencia paralela (PSSS, *Parallel Sequence Spread Spectrum*) en 868/915 MHz que emplea BPSK y modulación por desplazamiento de amplitud (ASK, *Amplitude Shift Keying*).
- Capa PHY DSSS en 2450 MHz que emplea modulación O-QPSK.

La tabla 2.1, tomada de (IEEE, 2006) presenta las bandas de frecuencia y las tasas de datos de las capas PHY de IEEE 802.15.4. También, en el anexo A se presenta un análisis cuantitativo de las capas físicas del estándar de comunicación IEEE 802.15.4, este análisis será utilizado para caracterizar el desempeño de las capas físicas como se verá en los capítulos siguientes.

Tabla 2.1. Capas físicas IEEE 802.15.4

PHY (MHz)	Frec. (MHz)	Tasa de chip (kchip/s)	Modulación	Tasa de bit (kb/s)	Tasa de símb. (ksymbol/s)	Símbolos
868/915	868-868.6	300	BPSK	20	20	Binarios
	902-928	600	BPSK	40	40	Binarios
868/915 (opcional)	868-868.6	400	ASK	250	12.5	PSSS 20 bit
	902-928	1600	ASK	250	50	PSSS 5 bit
868/915 (opcional)	868-868.6	400	O-QPSK	100	25	16 est. ort.
	902-928	1000	O-QPSK	250	62.5	16 est. ort.
2450	2450-2483.5	2000	O-QPSK	250	62.5	16 est. ort.

Con respecto a la capa MAC, IEEE 802.15.4 usa un protocolo basado en el algoritmo de acceso múltiple con sensado de portadora y evasión de colisiones (CSMA-CA, *Carrier Sense Multiple Access – Collision Avoided*), el cual requiere sensar el canal antes de transmitir para reducir la probabilidad de colisiones con otras transmisiones en curso. El estándar define dos modos de operación diferentes: uno donde se utilizan *beacons* para sincronización y otro donde no.

En el modo de operación sin transmisión de *beacons*, los nodos usan el protocolo CSMA-CA sin intervalo de tiempo para acceder al canal y transmitir sus paquetes. El algoritmo se implementa usando unidades de tiempo llamadas periodos de *backoff*. Primero, cada nodo retardará cualquier actividad para un número aleatorio de periodos de *backoff*, después de este retardo, se sensa el canal durante una unidad de tiempo y si el canal está libre, el nodo inmediatamente inicia la transmisión. De lo contrario, el nodo entra nuevamente al estado de *backoff*. Existe un número máximo de veces que el nodo puede tratar de acceder al canal, cuando se alcanza este número máximo, el algoritmo finaliza y la transmisión no ocurre.

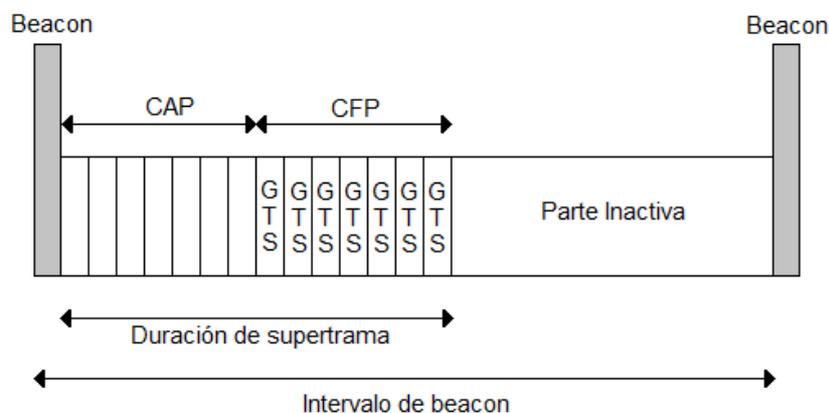


Figura 2.3. Estructura de supertrama en IEEE 802.15.4

Por su parte, en el modo de operación con transmisión de *beacons*, el acceso al canal es gestionado a través de una supertrama, cuya estructura se muestra en la figura 2.3. Una supertrama inicia con un paquete de sincronización transmitido por el coordinador de la red de área personal inalámbrica. Esta puede estar dividida en dos partes, una activa y otra inactiva. La parte inactiva permite que los nodos vayan al

modo de bajo consumo de energía. A su vez, la parte activa está dividida en dos subpartes, el periodo de acceso por contención (CAP, *Contention Access Period*) y el periodo libre de contención (CFP, *Contention-Free Period*), compuesto de intervalos de tiempo garantizados (GTSs, *Guaranteed Time Slots*) que pueden ser asignados por el nodo controlador a nodos específicos.

2.2 Radio cognitivo

De acuerdo a Haykin (2005), un dispositivo de radio cognitivo (CR, *Cognitive Radio*), soportado por un radio definido por software (SDR, *Software Defined Radio*), es un sistema de comunicación inalámbrico inteligente que es consciente de su ambiente, puede aprender de la experiencia y puede hacer cambios a ciertos parámetros de operación (potencia de transmisión, frecuencia portadora y esquema de modulación-codificación, entre otros) para adaptarse en tiempo real a los estímulos de radiofrecuencia presentes en el ambiente. El principal objetivo de un CR es tener comunicaciones confiables haciendo un uso eficiente del espectro de radio para satisfacer las necesidades del usuario (Baldo, 2008).

El concepto de radio cognitivo fue definido originalmente por el Dr. Mitola (1999a), y es descrito a partir del ciclo de cognición (Mitola, 1999b) que se muestra en la figura 2.4. En él, un radio recibe información acerca de su ambiente de operación (Mundo exterior) a través de la observación directa o a través de señalización (Observa), esta información se evalúa (Orienta) para determinar su importancia. De acuerdo a esta evaluación, el radio determina sus alternativas (Planea) y selecciona una alternativa (Decide) en una manera que presumiblemente podría mejorar los resultados de la evaluación. Asumiendo que fue necesario un cambio en la forma de onda, el radio implementa la alternativa (Actúa) ajustando sus recursos y realizando la señalización apropiada. Estos cambios se reflejan en el perfil de interferencia presentado por el radio cognitivo en el mundo exterior. Como parte de este proceso, el radio usa estas observaciones y decisiones para mejorar la operación del radio (Aprende), quizá creando nuevos estados de modelado, generando nuevas alternativas, o creando nuevas evaluaciones.

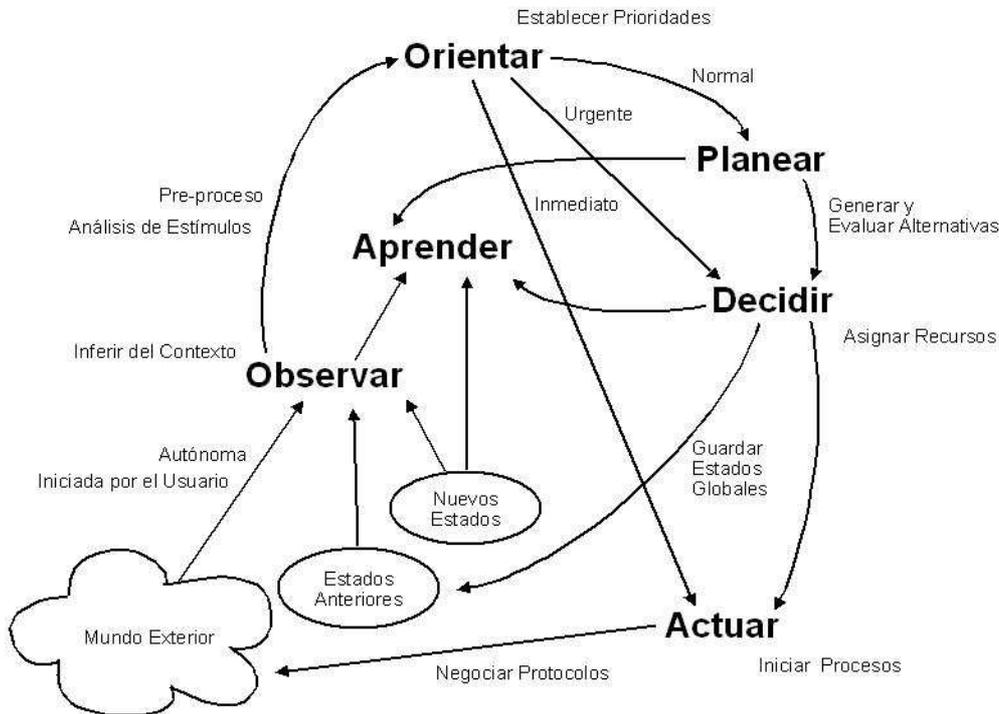


Figura 2.4. Ciclo de cognición

Chapin & Doyle (2007) describen tres tipos de radios cognitivos: CR básico, CR de razonamiento y CR de aprendizaje. Un CR básico es un sistema radio que sensa y se adapta al ambiente pero no necesariamente incluye alguna técnica de razonamiento o aprendizaje. Este medirá todos los canales disponibles y seleccionará un canal con el ruido y nivel de interferencia más bajo. Un CR de razonamiento es una mejora del CR básico el cual puede determinar automáticamente los límites legales de operación en un esquema de uso compartido de espectro. El CR de aprendizaje dispone de información de realimentación actualizada para la toma de decisiones.

El objetivo final de un CR es obtener el mejor espectro disponible a través de funciones cognitivas y la capacidad de reconfiguración de los nodos de radio. Dado que la mayoría del espectro ya ha sido asignado, el reto más importante es compartir el espectro licenciado sin interferir con la transmisión de otros usuarios licenciados como se ilustra en la figura 2.5 (Akyildiz, 2009). El CR permite el uso de espectro no utilizado temporalmente, el cual es referido como hueco de espectro o espacio blanco (Haykin, 2005). Si esta banda es posteriormente utilizada por un usuario

licenciado, el radio cognitivo se mueve a otro hueco de espectro o se mantiene en la misma banda, alterando su estrategia de transmisión para evitar interferencia.

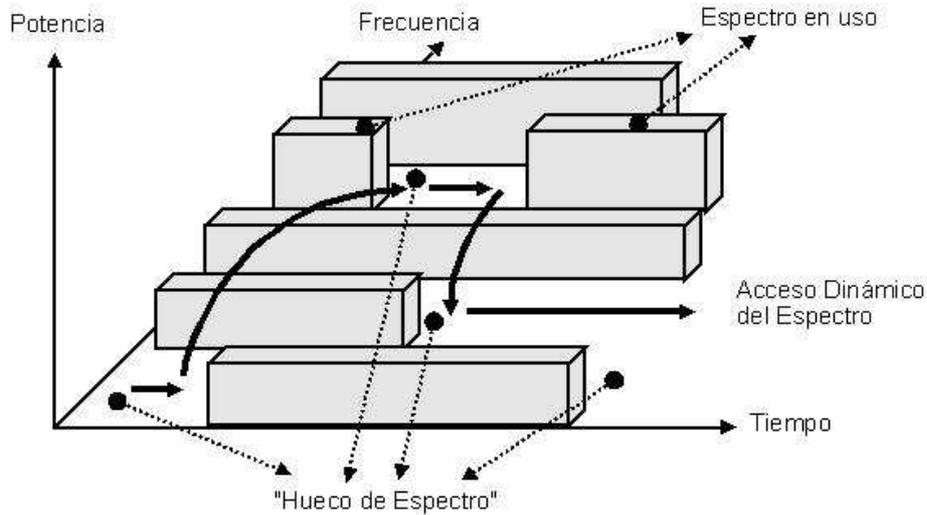


Figura 2.5. Concepto de hueco de espectro

La noción de radio cognitivo puede ser extendida a redes de radio cognitivo. Para Haykin (2007a, b), una red de radio cognitivo es un sistema de comunicación inalámbrico multiusuario inteligente con las siguientes habilidades: 1) Sensar el ambiente de radio para ser consciente de él; 2) Aprender del ambiente y realizar adaptaciones en respuesta a sus variaciones; 3) Facilitar la comunicación entre múltiples usuarios de una manera auto-organizada; 4) Controlar los recursos de comunicación entre los múltiples usuarios; y 5) Crear la experiencia de intención y auto-percepción.

Dos aproximaciones alternativas para construir una red de radio cognitiva de acuerdo a Chapin & Doyle (2007) incluyen lo siguiente: 1) Usar un canal licenciado fuera de banda como canal común dedicado disponible para todos los usuarios, esta aproximación tienen como desventaja que el canal común desempeña un rol similar a las estaciones base, incrementando el costo en el uso de recursos y siendo un elemento vulnerable en el sistema; y 2) Explotar el principio de redes auto-organizadas para construir una red *Ad Hoc*.

Sin embargo, las redes de radio cognitivo imponen retos únicos debido a la escasez y variación del espectro disponible, así como a los diversos requisitos de calidad de servicio (QoS, *Quality of Service*) que aplicaciones actuales pueden demandar (Akyildiz, 2006). Especialmente, en redes de radio cognitivo ad hoc (CRAHNS, *Cognitive Radio Ad Hoc Networks*), la arquitectura multi-salto distribuida, la topología de red dinámica, y la variación de disponibilidad de espectro en tiempo y lugar son algunos de los factores clave que la distinguen. Estos retos necesitan técnicas de diseño novedosas que simultáneamente consideren un amplio rango de problemas de comunicación abarcando varias capas de la pila de protocolos (Akyildiz, 2009).

Radio cognitivo extiende las funcionalidades de un radio definido por software. La tecnología SDR permite que algunas o todas las funciones de la capa física sean definidas por software (SDR Forum, 2008), brindándole mayor flexibilidad. SDR puede considerarse como una arquitectura abierta, la cual crea una plataforma de comunicación mediante la interconexión de bloques de construcción estandarizados (Haghighat, 2002). En CR, el radio software obtiene los parámetros de radio y las estadísticas, que permiten lograr la percepción del estado interno y el ambiente de radio, necesarias para el aprendizaje, generación de alternativas y toma de decisiones del motor de radio cognitivo; y además realiza la reconfiguración necesaria de acuerdo a las decisiones de adaptabilidad del CR.

2.3 Técnicas de inteligencia artificial

En esta sección se presenta una introducción a diferentes técnicas de inteligencia artificial: redes neuronales artificiales, computación evolutiva y lógica difusa, las cuales han sido diseñadas para, en cierto sentido, emular el funcionamiento de sistemas biológicos.

2.3.1 Redes neuronales artificiales

De acuerdo a Haykin (1999), las redes neuronales artificiales (ANNs, *Artificial Neural Networks*) son estructuras de computación alternativas, creadas con el propósito de reproducir las funciones del cerebro humano. Las neuronas naturales reciben señales electroquímicas de otras neuronas a través de las uniones sinápticas que

conectan el axón de las neuronas emisoras y las dendritas de las receptoras. De acuerdo a los impulsos recibidos, la neurona computa y envía su propia señal. El potencial interno asociado es el que controla el proceso de emisión, si se supera un cierto umbral, se envía un impulso eléctrico al axón, en caso contrario no se envía.

Las ANNs están constituidas por un conjunto de neuronas, unidades de procesamiento de información, dispuestas en varias capas e interconectadas entre sí mediante conexiones con pesos, las cuales realizan cálculos simples basados en la información que reciben de neuronas vecinas. Las ANNs usan un proceso de aprendizaje por analogía donde los pesos de las conexiones se ajustan automáticamente para reproducir un conjunto de patrones representativo del problema a aprender. La organización y forma de conexión de las neuronas permiten definir diferentes arquitecturas de redes, dependiendo de la presencia de una sola capa o múltiples capas, y la existencia o no de lazos de realimentación.

Una breve reseña histórica en este campo, destaca a McCulloch y Pitts (1943) quienes definieron algunos conceptos sobre el funcionamiento de las neuronas biológicas. Hebb (1949) definió la base del aprendizaje Hebbiano, a partir de la formulación de diferentes especulaciones fisiológicas. Rosenblatt (1957) construyó modelos neuronales, los cuales posteriormente generaron el concepto de Perceptrón. Widrow (1959) diseñó la red neuronal de elementos lineales adaptativos, o Adaline. Widrow y Hoff (1960), probaron matemáticamente que en determinadas circunstancias el error entre la salida deseada para la red y la obtenida por ella ante una entrada determinada podía ser minimizado hasta el límite que se desee. Minsky y Pappert (1969) publicaron el libro *Perceptrones*, el cual mostró las limitaciones del perceptrón desarrollado por Rosenblatt, en cuanto a su condición de separabilidad lineal. Después de varios años de recesión en el desarrollo de las ANNs, John Hopfield (1982) mostró que tienen el potencial para una operación exitosa, y propuso cómo podría ser desarrollado, esto marcó el reinicio de las investigaciones en este campo. Posteriormente, Rumelhart, Hinton y Williams (1986), presentaron el algoritmo de retropropagación de error, y desde entonces han ganado una posición importante en las técnicas de inteligencia artificial y han sido utilizadas en muchas aplicaciones prácticas.

La figura 2.6 muestra el modelo de una neurona (Haykin, 1999), donde se identifican tres elementos básicos: 1) un conjunto de sinapsis o enlaces de conexión, cada uno de los cuales es caracterizado por un peso o ponderación; 2) las señales de entrada, las cuales se ponderan por las sinapsis respectivas de la neurona y se combinan linealmente; y 3) una función de activación que limita la amplitud de la señal de salida de la neurona, típicamente en el intervalo $[0, 1]$ o alternativamente en $[-1, 1]$. Este modelo también incluye un umbral aplicado externamente, b_j , que tiene el efecto de incrementar o disminuir la señal de entrada a la función de activación, dependiendo si su valor es positivo o negativo, respectivamente.

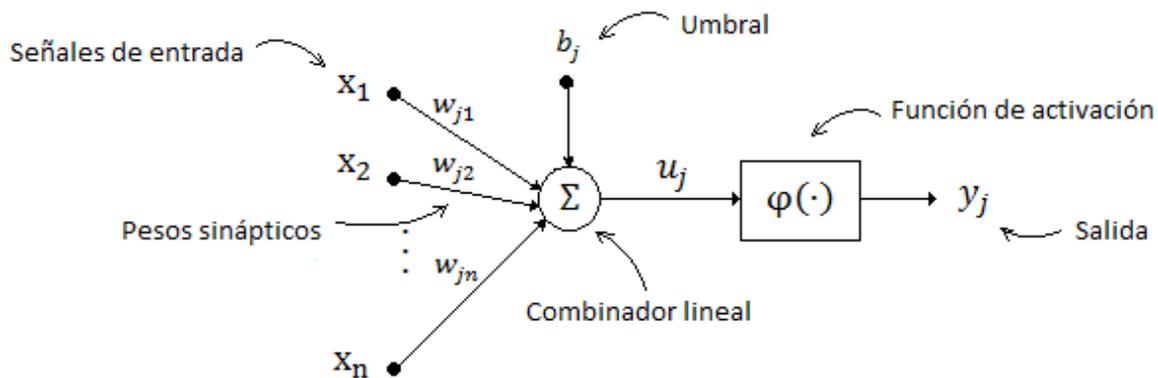


Figura 2.6. Modelo de una neurona

En términos matemáticos, se puede describir la neurona j mediante las ecuaciones 2.1 y 2.2.

$$u_j = \sum_{k=1}^n w_{jk} x_k \quad (2.1)$$

$$y_j = \varphi(u_j + b_j) \quad (2.2)$$

Donde x_1, x_2, \dots, x_n son las señales de entrada; $w_{j1}, w_{j2}, \dots, w_{jn}$ son los pesos sinápticos de la neurona j ; u_j es la combinación lineal de las señales de entrada; b_j es el umbral; $\varphi(\cdot)$ es la función de activación; y y_j es la señal de salida de la neurona.

Diferentes funciones pueden ser utilizadas como función de activación, preferiblemente aquellas que son derivables en todo punto y son acotadas por abajo y arriba. La figura 2.7 presenta las funciones de activación: a) Threshold, b) lineal acotada, y c) sigmoideal.

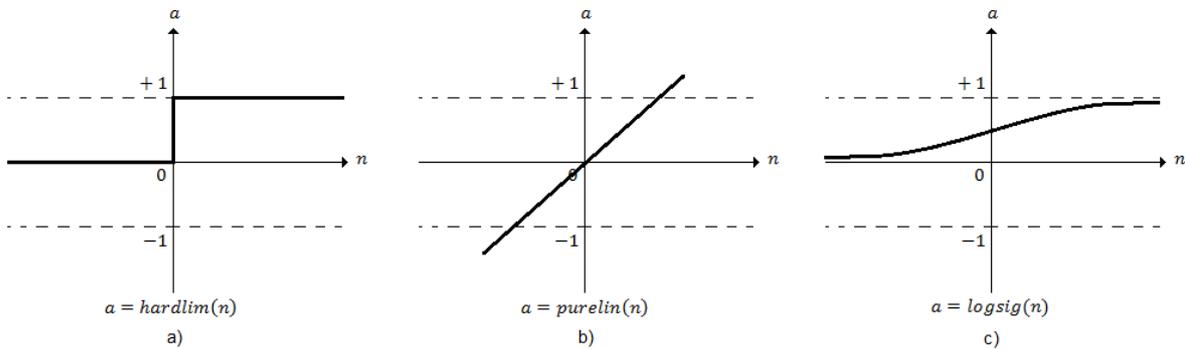


Figura 2.7. Ejemplos de funciones de activación

En una ANN el aprendizaje permite obtener los valores precisos de los pesos de las conexiones neuronales de tal manera que la red pueda resolver un problema específico de forma eficiente. El proceso general de aprendizaje consiste en ir facilitando a la red de ejemplos de un conjunto de aprendizaje representativo del problema a resolver y modificando los pesos de sus conexiones en base a un determinado esquema de aprendizaje. El aprendizaje se basa en el uso de ejemplos, por lo que la capacidad de una red para resolver un problema está ligada de forma fundamental al tipo y representatividad de los ejemplos de que dispone en el proceso de aprendizaje.

Generalmente, el proceso de aprendizaje es un proceso cíclico. La finalización de este proceso se realiza en función de un criterio de convergencia, que depende del tipo de red utilizada o del tipo de problema a resolver. Así, el periodo de aprendizaje finaliza cuando se alcanza un número fijo de ciclos de aprendizaje, el error desciende de una cantidad preestablecida o cuando la modificación de los pesos sea irrelevante.

Se distinguen tres tipos principales de esquemas de aprendizaje: supervisado, no supervisado y por refuerzo, y existe un gran número de métodos de aprendizaje. El algoritmo de retropropagación del error o *backpropagation* propuesto por Rumelhart (1986) fue el primer algoritmo importante para redes multicapa con alimentación hacia adelante. La idea principal de este algoritmo es modificar gradualmente los pesos de la red en la dirección que le indica el gradiente descendente de los pesos respecto a la función de error. A pesar de que este algoritmo es una aproximación

muy útil para el aprendizaje en este tipo de redes, plantea dos inconvenientes de importante consideración que se describen a continuación (Pérez, 2010):

- Velocidad de convergencia lenta. De manera general, los algoritmos de aprendizaje basados en gradiente descendente presentan una convergencia lenta hacia la solución. Este hecho es debido a que el algoritmo de aprendizaje avanza muy lentamente en aquellas regiones de la superficie de error en las que sus derivadas son prácticamente nulas. Esta situación es crítica en las mesetas de la función de error.
- La convergencia hacia el mínimo global no está garantizada. La minimización de la función de error es la que dirige el aprendizaje de la red. De manera general, los algoritmos de aprendizaje utilizan solamente información del estado actual para llevar a cabo tal minimización. Esto ocurre en el método de retropropagación del error. El problema que se plantea es consecuencia de los mínimos locales que puede presentar la función a optimizar. Si el estado inicial de la red, determinado por los valores iniciales asignados a los pesos, están en el área de atracción de un mínimo local de la función entonces el algoritmo se verá atraído por éste.

Con el objeto de resolver los problemas que se acaban de describir, a lo largo de los años se han planteado y desarrollado diferentes alternativas (Pérez, 2010). Entre ellas: modificaciones del algoritmo estándar, mejoras en la función de error, métodos basados en mínimos cuadrados y métodos de segundo orden, los cuales emplean las segundas derivadas de la función de error para incrementar la velocidad de convergencia. Los métodos de segundo orden se encuentran entre los algoritmos de aprendizaje más rápidos. Como ejemplos relevantes de este tipo de métodos se pueden mencionar: Levenberg-Marquardt y quasi-Newton. El método de Levenberg-Marquardt combina el gradiente y la aproximación de Gauss-Newton de la hessiana de la función del error en la regla de actualización de los pesos. La influencia de cada término viene determinada por un parámetro que se actualiza automáticamente. Los métodos quasi-Newton, al igual que el método de Newton, utilizan una aproximación local cuadrática de la función de error pero aplican una aproximación de la inversa de la matriz hessiana (Madsen et al., 2004).

2.3.2 Computación evolutiva

La computación evolutiva (EC, *Evolutionary Computing*) (Engelbrecht, 2007) está compuesta por diferentes algoritmos que se basan en la evolución de una población hacia una solución de cierto problema, y típicamente son utilizados para la optimización de funciones multi-dimensionales. La población de posibles soluciones evoluciona de una generación a la siguiente, hasta que lleguen a una solución aceptable. Entre estos algoritmos se tienen: algoritmos genéticos (GAs, *Genetic Algorithms*), programación genética (GP, *Genetic Programming*) y algoritmos evolutivos (EAs, *Evolutionary Algorithms*).

Un GA se implementa como un procedimiento de búsqueda computarizada y optimización que usa los principios genéticos y de selección natural. Por lo general, un GA incluye tres operaciones genéticas fundamentales: selección, cruzamiento y mutación, utilizadas para modificar las soluciones y seleccionar los individuos más apropiados para que existan en generaciones siguientes. Los GAs son un procedimiento iterativo que consiste de una población de individuos limitada en tamaño, cada uno representado por una cadena de símbolos finita (genoma) que codifica una posible solución en un espacio de problema dado.

La programación genética es una implementación especial de GAs, que usa material genético jerárquico no limitado en tamaño, los miembros de una población o cromosomas son programas en estructura de árbol y los operadores genéticos trabajan sobre las ramas de éstos árboles. Por su parte los algoritmos evolutivos no requieren separación entre espacios de recombinación y evaluación, además los operadores genéticos trabajan directamente sobre la estructura actual.

2.3.3 Lógica difusa

La lógica difusa (FL, *Fuzzy Logic*) (Zadeh, 1975) fue desarrollada para representar conocimiento no cierto e impreciso, provee un medio aproximado pero efectivo de describir el comportamiento de los sistemas que son muy complejos, poco definidos o cuyo análisis matemático no es simple. FL es una lógica de razonamiento aproximado, con las siguientes características: 1) los valores de verdad difusos se expresan en términos lingüísticos, por ejemplo: verdad, absolutamente verdad, más o menos verdadero, no verdadero, falso y completamente falso; 2) tablas de verdad

imprecisas; y 3) reglas de inferencia cuya validez es más bien aproximada que exacta.

Diferente a los conjuntos clásicos, los elementos de un conjunto difuso tienen grados de pertenencia a ese conjunto. El grado de pertenencia a un conjunto difuso indica la certidumbre (o incertidumbre) que ese elemento tiene en ese conjunto. Formalmente, sea X el dominio, o universo de discurso, y $x \in X$ un elemento específico del dominio X . Entonces el conjunto difuso A es caracterizado por la función de relación de pertenencia 2.3:

$$\mu_A: X \rightarrow [0, 1] \quad (2.3)$$

Por lo tanto, para todo $x \in X$, $\mu_A(x)$ indica la certidumbre para la cual el elemento x pertenece al conjunto difuso A . Para conjuntos de dos valores, $\mu_A(x)$ es 0 ó 1.

Los conjuntos difusos pueden ser definidos para dominios discretos (finitos) o continuos (infinitos). La notación usada para denotar conjuntos difusos es diferente de acuerdo al tipo de dominio sobre el cual ese conjunto es definido. En el caso de un dominio discreto X , el conjunto difuso puede ser expresado en la forma del vector n_x -dimensional 2.4,

$$A = \{(\mu_A(x_i)/x_i) | x_i \in X, i = 1, \dots, n_x\} \quad (2.4)$$

O como en 2.5 usando notación de sumatoria,

$$A = \mu_A(x_1)/x_1 + \mu_A(x_2)/x_2 + \dots + \mu_A(x_{n_x})/x_{n_x} = \sum_{i=1}^{n_x} \mu_A(x_i)/x_i \quad (2.5)$$

Donde la suma no debe ser confundida con la suma algebraica. El uso de la notación de suma sirve como una indicación que A es un conjunto de pares ordenados. Un conjunto difuso continuo, A , es denotado en 2.6:

$$A = \int_X \mu(x)/x \quad (2.6)$$

Nuevamente, la notación de integral no debe ser interpretada algebraicamente.

Las funciones de pertenencia son la esencia de los conjuntos difusos. Una función de pertenencia, también denominada función característica del conjunto difuso, define el conjunto difuso. La función es usada para asociar un grado de pertenencia a cada uno de los elementos del dominio del correspondiente conjunto difuso. Las funciones de pertenencia para conjuntos difusos pueden ser de cualquier forma o tipo según lo determinado por expertos en el dominio sobre el cual los conjuntos son definidos. Las funciones de pertenencia deben satisfacer las siguientes restricciones:

- Una función de pertenencia debe ser limitada abajo por 0 y arriba por 1.
- El rango de una función de pertenencia debe ser por lo tanto $[0, 1]$
- Para cada $x \in X$, $\mu_A(x)$ debe ser única. Esto es, el mismo elemento no puede mapear diferentes grados de pertenencia para el mismo conjunto difuso.

Entre conjuntos difusos se definen diferentes operadores difusos, tales como igualdad de conjuntos difusos, conjuntos difusos contenidos, complemento de un conjunto difuso, intersección de conjuntos difusos y operador producto, entre otros.

Un controlador difuso puede ser visto como una función estática no lineal que mapea entradas del controlador a salidas del mismo. Es usado para controlar algún sistema, o planta. El sistema tiene una respuesta deseada que debe ser mantenida ante las entradas que sean recibidas. Las entradas al sistema pueden cambiar el estado del sistema lo cual causa un cambio en la respuesta. La tarea del controlador es tomar la acción correctiva dando un conjunto de entradas que aseguran la respuesta deseada. El diseño de un controlador difuso involucra los siguientes aspectos: un universo de discurso debe ser definido, y los conjuntos difusos y funciones de pertenencia para los espacios de entrada y salida tienen que ser diseñados. Con la ayuda de un experto humano, se definen las reglas lingüísticas que describen el comportamiento dinámico, y se decide cómo se implementa el codificador difuso, el motor de inferencia y el decodificador difuso. Un tipo de controlador ampliamente utilizado es el controlador difuso Mamdani (Berenji, 1992).

2.4 Conclusiones

El concepto de radio cognitivo involucra diversas tecnologías y áreas de conocimiento: procesamiento de señales, radio definido por software, inteligencia artificial, optimización, protocolos de red, calidad de servicio, gestión de espectro y lenguajes de especificación, entre otras. Más aún, puede concebirse para diferentes tipos de redes, no solo inalámbricas. Esto conlleva al planteamiento de diversas posibilidades de trabajo en este campo. En especial, para desarrollar el objetivo de diseño de un motor de aprendizaje basado en técnicas de inteligencia artificial para redes de sensores inalámbricas cognitivas, con énfasis en el nivel físico del modelo de referencia OSI, es apropiado proponer un modelo de redes de sensores inalámbricas que defina las funciones cognitivas que deben ser soportadas y especificar así estas novedosas funciones.

Las redes neuronales artificiales son sistemas de procesamiento de información adaptativos no programados que pueden en forma autónoma desarrollar capacidades operacionales en respuesta a un ambiente de información; ellas representan un modelo óptimo para lograr capacidades de memoria y aprendizaje en una red de radio cognitivo. El uso de redes neuronales multicapa con alimentación hacia delante para la implementación de funciones de aprendizaje constituye una aproximación razonable, debido a la simplicidad y eficiencia que este tipo de redes ofrece.

Los algoritmos que componen el área de la computación evolutiva están enfocados en la búsqueda de soluciones a partir de una población de individuos inicial y su evolución entre generaciones. De acuerdo al análisis realizado se concluye que no deben ser abordados en esta tesis, principalmente debido a que se busca aplicar el concepto de radio cognitivo en un tipo de redes caracterizadas por sus limitados recursos computacionales, siendo necesario mantener una complejidad razonable.

La anterior conclusión también es válida para la técnica lógica difusa. Se analizó su uso para el diseño de la función de control del motor de aprendizaje propuesto, sin embargo, se encontró que la base de reglas resultantes estaba compuesta por dos valores: cambiar de canal o no, por ejemplo. Por lo tanto, se llegó a la conclusión que la complejidad del motor de aprendizaje se incrementaría sin obtenerse

contraprestaciones al respecto. No obstante, Baldo (2007) propone su uso para el diseño de un controlador cognitivo desde un punto de vista *cross-layer*, donde las diferentes capas de la pila de protocolos puedan exportar métricas lingüísticas, por ejemplo canal bueno, canal malo, calidad de servicio pobre, aceptable, buena, etc., y el controlador sea independiente de los detalles de implementación de las redes, esquemas de codificación-modulación, tasa de datos, etc. Desde luego, esta apreciación es adecuada en un contexto donde los dispositivos de red tienen mayores capacidades computacionales, lo que no ocurre en las redes de sensores inalámbricas.

Capítulo 3

Modelo de redes de sensores inalámbricas cognitivas

Una red de sensores inalámbrica cognitiva (CWSN, *Cognitive Wireless Sensor Network*) es una red de comunicaciones donde los nodos incorporan funciones de observación, orientación, planeación, decisión, actuación y aprendizaje, que les permiten ser conscientes de su ambiente de radio, decidir alternativas y adaptarse para mejorar su desempeño en la transmisión y recepción de datos.

Este capítulo presenta un modelo de redes de sensores inalámbricas cognitivas, donde se especifica el modelo conceptual de una CWSN, la arquitectura de red y se describen funcionalmente sus componentes. Para lo cual se utiliza el lenguaje de modelado unificado (UML, *Unified Modeling Language*) de la OMG (2005).

3.1 Modelo conceptual

La figura 3.1 presenta el modelo conceptual de una CWSN, en él se han definido seis funciones: PHY, sensado, aprendizaje, control, persistencia de datos y señalización.

A continuación se presenta una descripción general de estas funciones.

- PHY: define la capa física de los nodos de la red, es decir implementa los esquemas de codificación y modulación que procesan las señales para el envío y recepción de datos. Consiste en un *framework* SDR que dota a los

nodos de la WSN con la flexibilidad necesaria mediante capacidades de reconfiguración.

Esta función también realiza las mediciones del ambiente de radio necesarias, entre ellas: densidad espectral de potencia (PSD, *Power Spectral Density*) y relación señal a ruido más interferencia (SINR, *Signal to Interference and Noise Ratio*).

- Sensado: coordina el procedimiento por medio del cual se recoge información del ambiente de radio donde operan los nodos, y depende directamente de la función PHY para la obtención de las mediciones necesarias.
- Aprendizaje: modela el desempeño en la transmisión y recepción de datos, medido a partir de la tasa de errores de bit (BER, *Bit Error Rate*), esta consciencia se logra a partir de la información proporcionada por la función de sensado.
- Control: realiza un proceso de razonamiento para inferir una decisión sobre las acciones que se deben tomar para mejorar el desempeño, en caso de que esto sea factible, por ejemplo la selección de un mejor canal para realizar la transferencia de datos o la selección de un esquema de capa PHY diferente.
- Persistencia de datos: la información de sensado, desempeño y decisiones de control deben ser persistentes en el tiempo, con el objetivo que puedan ser utilizadas e intercambiadas por las funciones del nodo cognitivo y entre nodos de la red, de esto se encarga esta función. Puede ser implementada mediante una base de información de red de área personal (PIB, *PAN Information Base*) como en el caso de una red IEEE 802.15.4, o como una base de datos relacional asumiendo que las capas superiores de la red disponen de capacidades para tal fin.
- Señalización: implementa los protocolos necesarios para el intercambio de información entre los nodos de una CWSN, la cual permitirá coordinar la operación distribuida cognitiva de la red, así como realizar la negociación de recursos necesaria para llevar a cabo la adaptabilidad de la red.

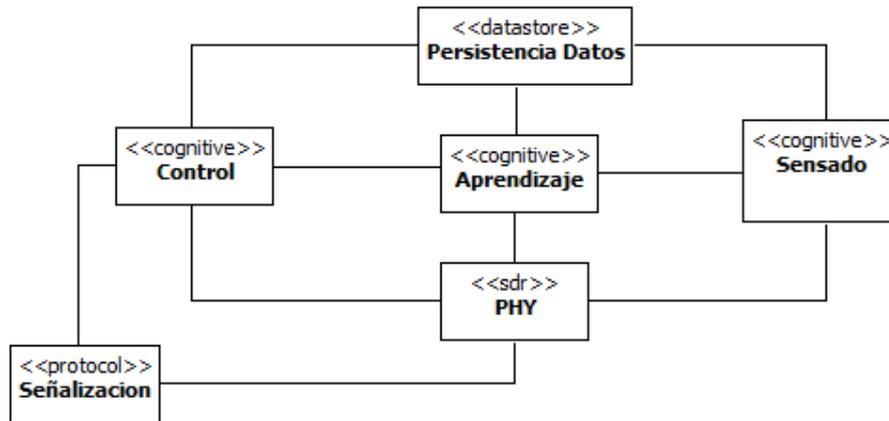


Figura 3.1. Modelo conceptual de una CWSN

3.2 Arquitectura de red

La arquitectura de red de una CWSN hereda las características de una WSN tradicional, siendo posibles dos estructuras principales: en estrella y árbol de *clusters*, representadas en las figuras 3.2 y 3.3 respectivamente, las cuales se describen a continuación.

- El subsistema denominado Aplicaciones representa los servicios que la CWSN ofrece a los usuarios, tales como: almacenamiento de los datos recolectados por los nodos sensores, monitoreo, control y análisis para una aplicación determinada.
- El nodo Coordinador implementa el modelo conceptual CWSN principalmente, debido a que por lo general dispone de mayores recursos computacionales y posiblemente de suministro continuo de energía.
- Un nodo Sensor realiza las funciones de sensado propias de la aplicación, y tiene capacidades de reconfiguración que le permitan adaptarse según las decisiones tomadas por el nodo Coordinador. Para algunas aplicaciones de control y automatización estos nodos también tendrán funciones de actuación.

- En una red de sensores *peer-to-peer* es posible formar grupos de nodos, denominados *clusters*, coordinados por un nodo Cabecera, este nodo realiza el sensado cognitivo, es decir, toma medidas del ambiente de radio y de desempeño necesarias para lograr la capacidad de percepción en la red cognitiva, y las envía al nodo Coordinador; así mismo, infiere sobre el desempeño y toma acciones necesarias para mejorarlo. Esta última topología, permite que la CWSN sea desplegada en un área más amplia y sea consciente del ambiente de radio para diferentes condiciones de espacio – tiempo.

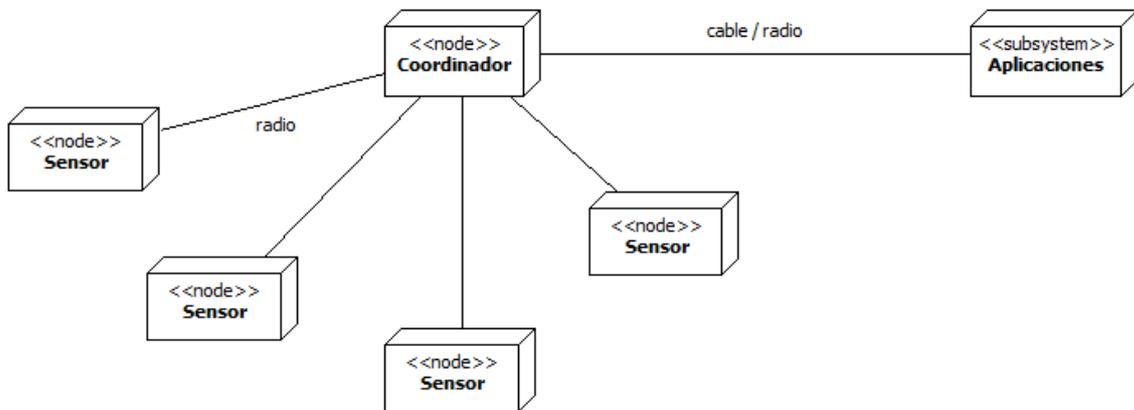
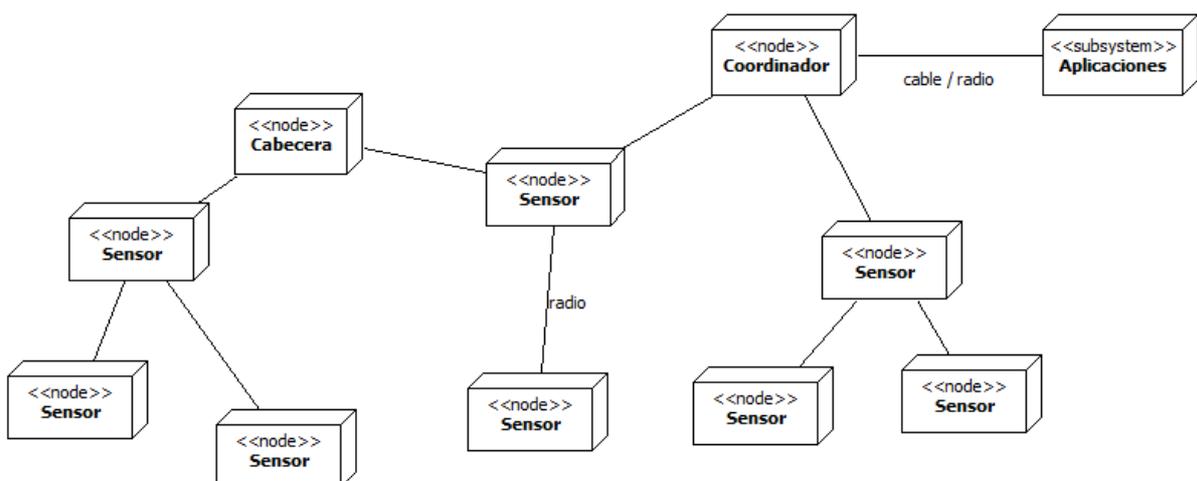


Figura 3.2. CWSN en estrella

Figura 3.3. CWSN en árbol de *clusters*

3.3 Arquitectura de los nodos

Las funciones cognitivas definidas en el modelo conceptual deben ser distribuidas en los nodos de una CWSN. Los nodos sensores son dispositivos con fuertes restricciones computacionales y de energía, y deben mantenerse con la menor complejidad posible. La figura 3.4 muestra la arquitectura de un nodo sensor, donde se requieren funciones de capa física reconfigurables y un mecanismo de señalización para el intercambio de información y sincronización entre nodos de la red; además de las funciones propias de la aplicación, las cuales no se muestran en el gráfico.

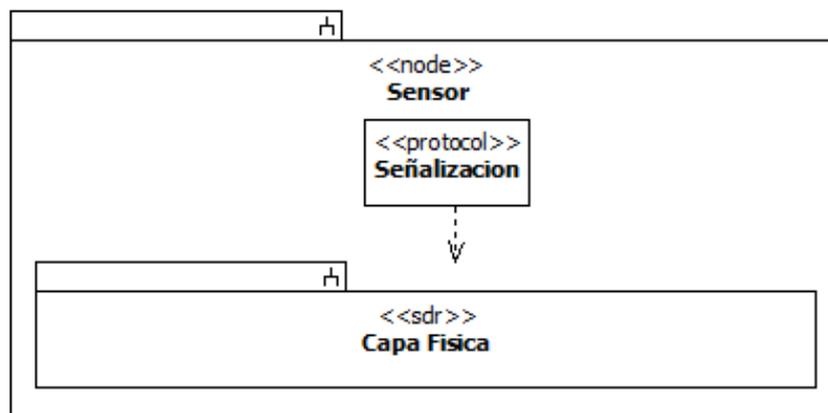


Figura 3.4. Arquitectura nodo sensor

A diferencia de un nodo sensor, los nodos cabecera realizan funciones de sensado cognitivo y de control. La primera permite obtener la información de desempeño y ambiente de radio propia para un grupo de nodos sensores. Las funciones de un nodo cabecera son coordinar dicho grupo de sensores, reenviar los datos de aplicación de los nodos sensores, o datos de aplicación sensados localmente, y enviar la información de sensado cognitivo al nodo coordinador. La segunda permite generar las alternativas y tomar las decisiones que mejoren el desempeño.

Además, al igual que en un nodo sensor, se requiere del mecanismo de señalización para el intercambio de información y sincronización entre nodos de la red. Las

capacidades de reconfiguración de la capa física son necesarias para implementar las instrucciones de adaptabilidad de la CWSN. La figura 3.5 muestra la arquitectura de un nodo cabecera.

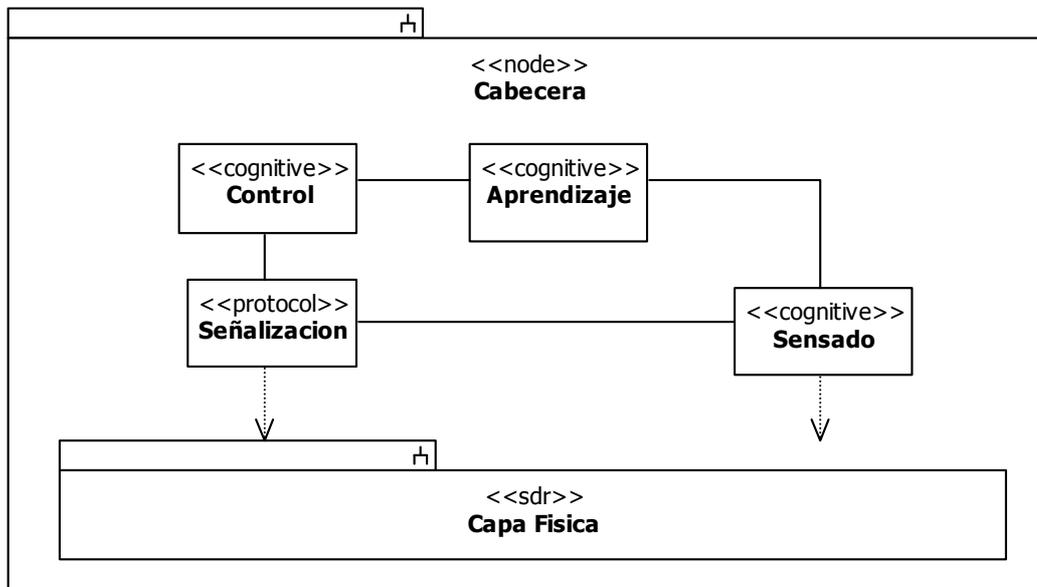


Figura 3.5. Arquitectura nodo cabecera

El nodo coordinador de una CWSN implementa las funciones cognitivas: sensado, aprendizaje y control, además de persistencia de datos, señalización y reconfiguración como se ilustra en la figura 3.6. Esto implica que el nodo coordinador debe disponer de mayores recursos computacionales y seguramente de alimentación continua de energía. En él se realiza el aprendizaje del desempeño a nivel físico de la red, posiblemente a partir de la implementación de funciones de ajuste de los parámetros de una función que lo caracterice, de acuerdo a los datos de las muestras y de tal manera que sea posible relacionar las medidas del ambiente de radio con las estimaciones de desempeño de error de la capa física.

Además, este nodo debe implementar un controlador para la toma de decisiones, la persistencia de datos tanto de sensado como de las decisiones, configuraciones, y los mecanismos de señalización necesarios para el intercambio y sincronización de información entre los nodos de la red.

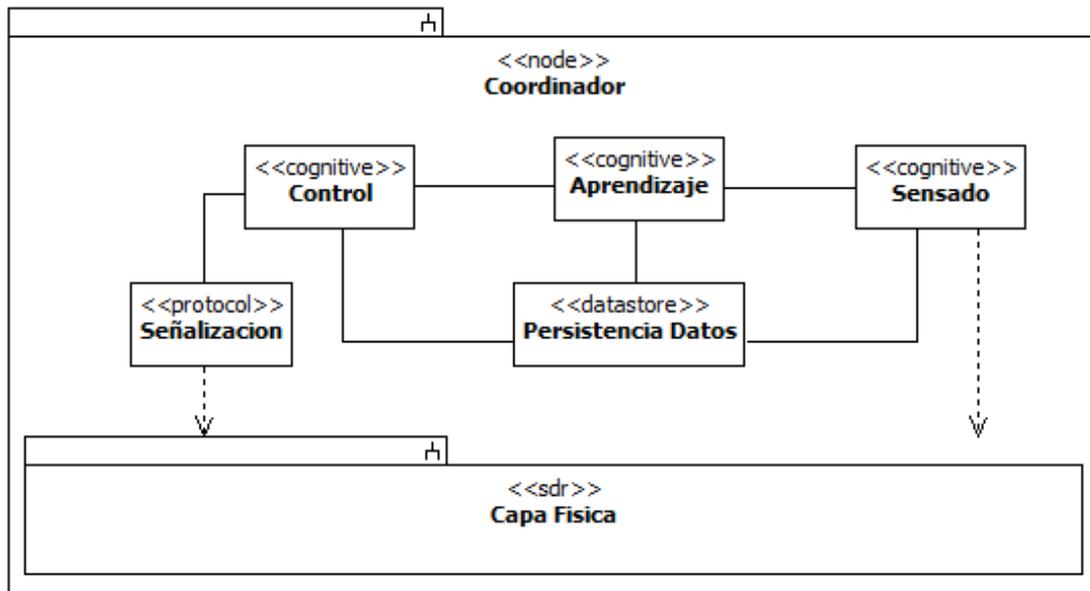


Figura 3.6. Arquitectura nodo coordinador

Capítulo 4

Mecanismo de aprendizaje en redes de sensores inalámbricas cognitivas

En este capítulo se presenta el diseño de las funciones de cognición: sensado, aprendizaje y control especificadas en el modelo de CWSN, y que ha sido definido como el motor de aprendizaje de una CWSN.

4.1 Sensado

La función de sensado se realiza a partir del envío de instrucciones y sincronización entre el nodo coordinador y los nodos cabecera, en el caso de una red en árbol de *clusters*, o la intención de inicio de sensado por parte del nodo coordinador para una red en estrella. El sensado consiste en una serie de medidas/estimaciones iterativas del ambiente de radio, por ejemplo la SINR y la BER para los canales o capas físicas disponibles. La SINR puede medirse a partir de operaciones de detección de energía, junto a medidas de calidad de enlace, como se realiza en IEEE 802.15.4.

El desempeño de la capa física, definido generalmente por el esquema de codificación y modulación empleado, se determina a partir de la tasa de error de bit (BER). La BER no es una medida del ambiente de radio, más bien es una estimación de la calidad del enlace radio y se calcula a partir de la siguiente relación

$$BER = \# \text{ bits erróneos} / \# \text{ bits total transmitidos} \quad (4.1)$$

La medida de la BER puede realizarse mediante el envío de una secuencia de pseudo-ruido (PN, *Pseudo-Noise*) por parte de nodos sensores y/o de cabecera que pueda ser conocida por el nodo coordinador y pueda ser comparada para calcular los errores producidos.

Diferentes métodos han sido propuestos para estimar la BER en el canal de comunicaciones. Chen et al. (2010) investigaron el concepto de códigos para estimación de error (EEC, *Error Estimating Codes*), los cuales sin realizar una corrección de los errores que se presenten, estiman la fracción de bits errados en una trama de datos, y cuya información sirve para orientar la realización de procesos dinámicos, tales como adaptación de tasa de datos, retransmisiones de paquetes conscientes de BER, enrutamiento consciente de BER y selección de portadora. Smit et al. (2003) propusieron un método que estima la BER de un canal inalámbrico empleando los datos recibidos por un receptor RAKE (Price, 1958), este método no requiere conocimiento acerca de las características del canal o condiciones externas precisas; esta información puede ser usada para adaptar la operación del receptor y así lograr una calidad de servicio deseada para una aplicación o protocolo determinado.

Así mismo, el uso de códigos correctores de errores puede permitir la estimación de la BER y una mejora en el desempeño de error. Li et al. (2010) plantean la incorporación de un módulo corrector de errores (FEC, *Forward Error Correction*) e intercalación en las funciones de capa física de IEEE 802.15.4 para mejorar el desempeño de error, y que permita una reducción de la energía necesaria para realizar la transmisión. Shen & Abedi (2007) propusieron un esquema de codificación para mejorar el desempeño de error del estándar IEEE 802.15.4, el trabajo consistió en aplicar diferentes esquemas de codificación a la capa PHY de IEEE 802.15.4, se compararon los desempeños de error sobre canal AWGN y se seleccionó un código convolucional con intercalador aleatorio para un mejor desempeño manteniendo baja la complejidad de implementación.

Una vez se realicen las medidas por parte de la función de sensado, se deben almacenar en la base de información para que pueda ser utilizada posteriormente.

La figura 4.1 muestra el diagrama de flujo de la función de sensado. El sensado se realiza para todos los canales disponibles en la CWSN, e inicia mediante la señalización necesaria para coordinar el envío y recepción de tramas de datos de secuencias PN. Cuando una secuencia llega al nodo coordinador en el canal i , se realiza la medición de la SINR, diferentes técnicas pueden utilizarse como el periodograma Welch (Matinmiko, 2008), o bien, esta medida es realizada por implementaciones actuales de redes de sensores por ejemplo, en redes 802.15.4 (IEEE, 2006).

La medida de la BER se realiza para las tramas recibidas, posiblemente requiriendo la incorporación de un bloque FEC en los nodos, tanto para la codificación, decodificación, la detección y conteo de errores. Li (2010) propone la implementación de codificación FEC eficiente para una red WSN. Las medidas realizadas se almacenan en una base de información, denominada parámetros ambiente de RF.

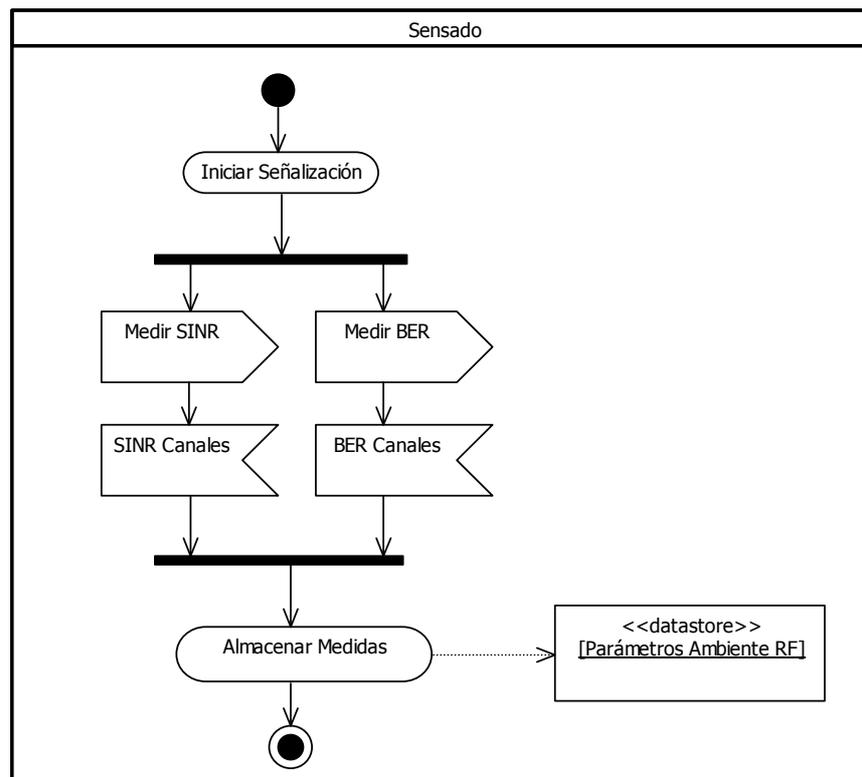


Figura 4.1. Diagrama de flujo función de sensado

4.2 Aprendizaje

Para esta función se propone el uso de una red neuronal multicapa con alimentación hacia delante. El aprendizaje permite establecer la relación existente entre la SINR y BER, mediante la determinación de una función de regresión, la cual pueda ser utilizada por la función de control para estimar la BER a partir de la SINR de una trama recibida por el nodo coordinador y de esta manera pueda tomar una decisión que mejore el desempeño de error.

La figura 4.2 muestra el diagrama de flujo de esta función, en primer lugar se crea la red neuronal y se entrena a partir de las medidas obtenidas por la función de sensado, SINR vector de entrada y BER vector de salida. Los pesos de la red neuronal definen una función de estimación que generaliza la calidad del canal a partir de sus condiciones actuales. Es claro que el ambiente de radio puede variar, por ejemplo debido a que nuevos sistemas inalámbricos sean instalados en la proximidad de la WSN, y por lo tanto se requiere que la red sea re-entrenada periódicamente para actualizar sus pesos para que represente las condiciones de radio y de desempeño reales en el lugar donde ha sido desplegada la WSN.

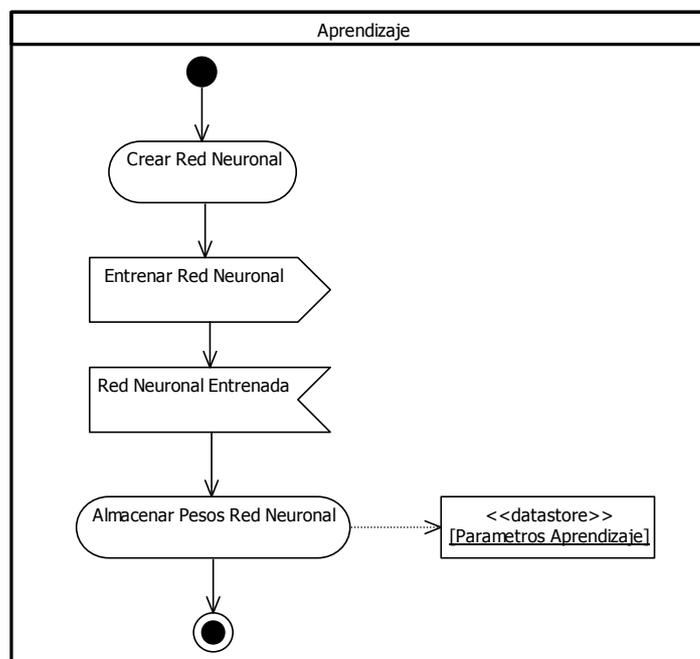


Figura 4.2. Diagrama de flujo función de aprendizaje

Se describe a continuación la implementación de la función de aprendizaje soportada en redes neuronales multicapa con alimentación hacia delante. Para ello se describe el método de aprendizaje utilizado y el algoritmo de cálculo realizado por la red neuronal.

4.2.1 Método de aprendizaje red neuronal

Considerando la red neuronal de la figura 4.6. Los datos de entrada a la red, $x_{1s} = SINR$, se propagan hacia delante multiplicándose por los pesos de las conexiones de las cuatro neuronas de la capa oculta w_{1i} , donde a cada entrada ponderada se le suma el umbral b_i , y luego se aplica la función de activación f_i . El subíndice $i = 1, \dots, 4$, indica el peso, el umbral y la función de activación para cada neurona de la capa oculta. La salida de cada neurona de la capa oculta se multiplica por el peso de las conexiones de la neurona de salida f_5 , donde se le suma el umbral b_5 y se combinan linealmente mediante la función de activación f_5 , para producir la salida de la red $y_{1s} = BER$.

El subíndice s se utiliza para indicar el número de la muestra presente en la entrada, y la salida producida por la red en un instante de tiempo determinado. La función de activación utilizada en las neuronas de la capa oculta es la función tagsig, definida por la ecuación 4.2, mientras la neurona de salida utiliza la función lineal como función de activación, ecuación 4.3. Estas funciones se muestran en las figuras 4.7a y 4.7b respectivamente

$$f_i(v) = \frac{2}{1+e^{-2v}} - 1, \quad i = 1, \dots, 4 \quad (4.2)$$

$$f_5(v) = k * v \quad (4.3)$$

El método de aprendizaje seleccionado es el método de Levenberg-Marquardt, desde el punto de vista de ajuste de curva de datos como se presenta en el anexo B, pero adaptado para que permita el entrenamiento de la red. Esto se describe a continuación.

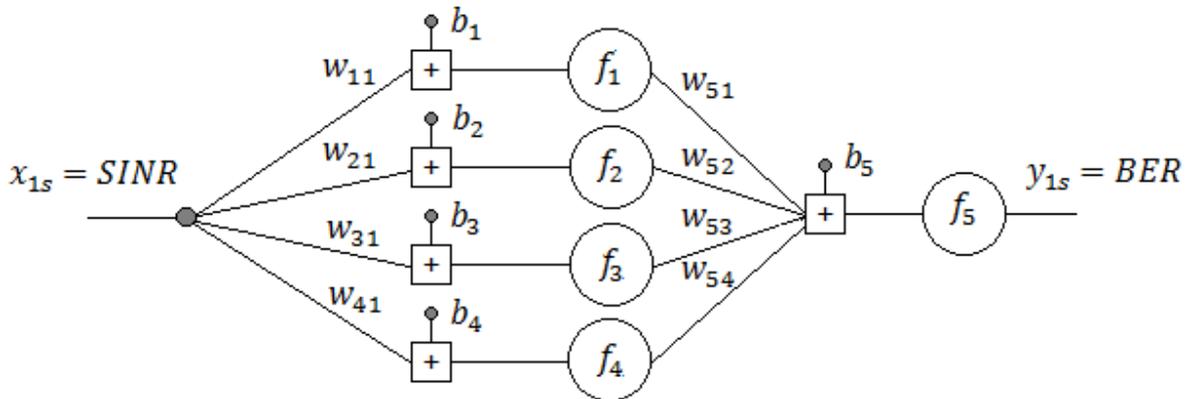


Figura 4.3. Red neuronal para caracterización de desempeño

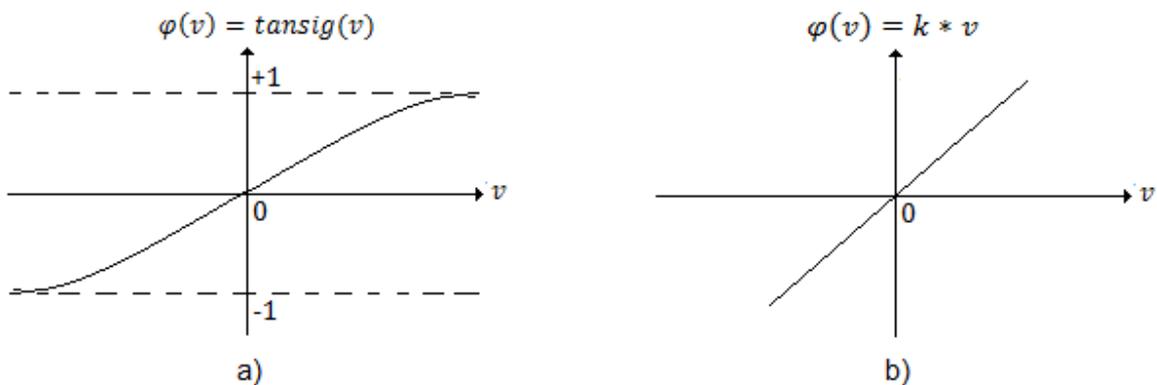


Figura 4.4. Función de activación: a) tangsig, b) lineal

La función de ajuste o función de aproximación es la función de transferencia que relaciona la salida de la red con su entrada. Esta función está definida por los pesos de las conexiones, los umbrales de cada neurona y las funciones de activación. La función se ajusta de acuerdo a los datos de ejemplo que se utilizarán en el entrenamiento, los cuales representan las medidas de SIRD y BER obtenidas por la función de sensado. Para la red neuronal de la figura 4.6, la salida de cada neurona de la capa oculta está dada por

$$z_{is} = f_i(w_{i1}x_{1s} + b_i), \quad i = 1, \dots, 4 \quad (4.4)$$

La salida y de la red está dada por la ecuación 4.5

$$y_{1s} = f_5(\sum_{i=1}^4 w_{5i} z_{is} + b_5) \quad (4.5)$$

Asumiendo que $v = 1$ en la función de activación de la neurona de salida, se tiene

$$y_{1s} = \sum_{i=1}^4 w_{5i} z_{is} + b_5 \quad (4.6)$$

Y finalmente, reemplazando la ecuación 4.4 en 4.6, la salida de la red es

$$y_{1s} = \sum_{i=1}^4 w_{5i} f_i(w_{i1} x_{1s} + b_i) + b_5 \quad (4.7)$$

Y cuya forma vectorial es

$$y_{1s} = f_5(\mathbf{l}w * f_1(\mathbf{i}w * x_{is} + \mathbf{b}_i) + \mathbf{b}_l) \quad (4.8)$$

La actualización de los pesos y umbrales de la red para ajustarse a los valores de entrada-salida que se le presenten como ejemplos en la fase de aprendizaje se realiza utilizando el algoritmo de aprendizaje Levenberg-Marquardt. Como se ha mencionado, este método de aprendizaje se caracteriza por ofrecer una rápida convergencia y es eficiente para caracterizar el desempeño de error (BER vs SINR).

Sea p el vector de parámetros a actualizar en las iteraciones del método de aprendizaje, la cantidad de elementos de este vector depende del número de conexiones y neuronas de la red, siendo igual a diez para la red utilizada. El método requiere un valor inicial de estos parámetros, para lo cual se generan diez números aleatorios con distribución normal de media cero y varianza igual a uno, ecuación 4.9, además de valores mínimos y máximos, para los cuales se utilizó $-2 * p_{init}$ y $+2 * p_{init}$, respectivamente. La elección de este rango obedece a valores promedio encontrados en experimentos realizados con la red.

$$\mathbf{p}_{init} = N(0, 1) \quad (4.9)$$

Con los patrones de entrada-salida de ejemplo suministrados a la red, junto con los vectores de parámetros \mathbf{p}_{min} , \mathbf{p}_{max} y \mathbf{p}_{init} , la red neuronal puede entrenarse, no sin antes indicar los criterios de desempeño de su aprendizaje deseados, tales como máximo número de iteraciones, tolerancia de convergencia para gradiente, tolerancia de convergencia para parámetros y tolerancia de convergencia para Chi-cuadrado (Gavin, 2011).

Además, para un mejor desempeño en el aprendizaje de la red, es importante que los vectores de datos de ejemplo estén normalizados, y por lo tanto, una vez determinados los pesos y umbrales de la red, será necesario normalizar el valor de *SINR* de entrada a calcular por la red y des-normalizar su valor de *BER* de salida.

El resumen del algoritmo se presenta a continuación. Los criterios de finalización del algoritmo son: ϵ_1 , convergencia dada por el gradiente; ϵ_2 , convergencia en los parámetros, y ϵ_3 , convergencia en el criterio Chi-cuadrado.

begin

```

k := 0;  p := p0;  λ := λ0
found := false
while (not found) and (k < kmax)
    k := k + 1;
    Solve [JTWJ + λ diag(JTWJ)] hlm = JTW(y - ŷ)
    Solve χ2(p) =  $\frac{1}{2}\mathbf{y}^T\mathbf{W}\mathbf{y} - \mathbf{y}^T\mathbf{W}\hat{\mathbf{y}} + \frac{1}{2}\hat{\mathbf{y}}^T\mathbf{W}\hat{\mathbf{y}}$ 

    if max(|hi/pi|) < ε2
        found := true
    if χ2/m < ε3
        found := true
    if max(|JTW(y - ŷ)|) < ε1
        found := true

```

```

else
    if  $\chi^2(\mathbf{p}) - \chi^2(\mathbf{p} + \mathbf{h}) > \epsilon_3 \mathbf{h}^T (\lambda \mathbf{h} + \mathbf{J}^T \mathbf{W}(\mathbf{y} - \hat{\mathbf{y}}))$ 
         $\mathbf{p} := \mathbf{p} + \mathbf{h}_{lm}$ 
         $\lambda := \lambda/10$ 
    else
         $\lambda := 10 * \lambda$ 
end

```

4.2.2 Algoritmo de cálculo red neuronal

Una vez entrenada la red, es decir, una vez que los pesos y umbrales han sido ajustados. El algoritmo de cálculo de la red neuronal obedece a la ecuación 4.7 ó 4.8, a partir del cual la red calcula o estima un valor de salida para el valor de entrada. La versión vectorial de esta función de cálculo, ecuación 4.8, es preferible para su implementación en Matlab.

begin

```

 $x_{1s} := \text{sinr}$ 
 $\mathbf{iw} := [w_{11} \ w_{21} \ w_{31} \ w_{41}]^T$ 
 $\mathbf{bi} := [b_1 \ b_2 \ b_3 \ b_4]^T$ 
 $\mathbf{lw} := [w_{51} \ w_{52} \ w_{53} \ w_{54}]^T$ 
 $\mathbf{bl} := [b_5]$ 

```

$$f_1(v) := \frac{2}{1+e^{-2v}} - 1$$

$$f_5(v) := k * v$$

$$x_{nmin} := \min(\mathbf{SINR}); \ x_{nmax} := \max(\mathbf{SINR}); \ y_{nmin} := -1; \ y_{nmax} := 1$$

$$x_{dmin} := -1; \ x_{dmax} := 1; \ y_{dmin} := \min(\mathbf{BER}); \ y_{dmax} := \max(\mathbf{BER})$$

$$x_n := (y_{nmax} - y_{nmin}) * (x_{1s} - x_{nmin}) / (x_{nmax} - x_{nmin}) + y_{nmin}$$

$$y_{1s} = f_5(\mathbf{lw} * f_1(\mathbf{iw} * x_{is} + \mathbf{b}_i) + \mathbf{b}_l)$$

$$y_d := ber := (y_{1s} - x_{dmin}) * (y_{dmax} - y_{dmin}) / (x_{dmax} - x_{dmin}) + y_{dmin}$$

end

4.3 Control

La función de control utiliza el aprendizaje de desempeño de la red CWSN para realizar un razonamiento acerca del significado de ese aprendizaje, y de acuerdo a él, decidir los ajustes necesarios para lograr un desempeño establecido, definido a partir de una BER umbral. De forma general, si el desempeño de error supera la BER umbral, el canal se clasifica como canal malo y el control decide cambiar de canal, cuando hayan más canales disponibles, o el cambio de capa física, cuando haya más de una capa física definida para la red; por otro lado, si el desempeño de error no supera la BER umbral, el canal se clasifica como canal bueno, e indica que no se requiere tomar ninguna acción inmediata.

La función de control realiza el razonamiento acerca de la BER estimada para el canal actual, así como determina cuál es el mejor canal disponible y de acuerdo a ello decide si realizar el cambio de canal o no. La figura 4.3 muestra el diagrama de flujo para la tarea de razonamiento realizado por el control para el caso donde es posible seleccionar el canal de radio con la mayor SINR disponible.

La función de control, es implementada a partir de un controlador lógico, que junto a las otras funciones de cognición definidas, conforman un motor de aprendizaje. El controlador es el encargado de dirigir las actividades de las funciones cognitivas con el fin de mejorar el desempeño de la CWSN. La figura 4.4 presenta la interacción y los responsables de su implementación para el procedimiento de selección de canal.

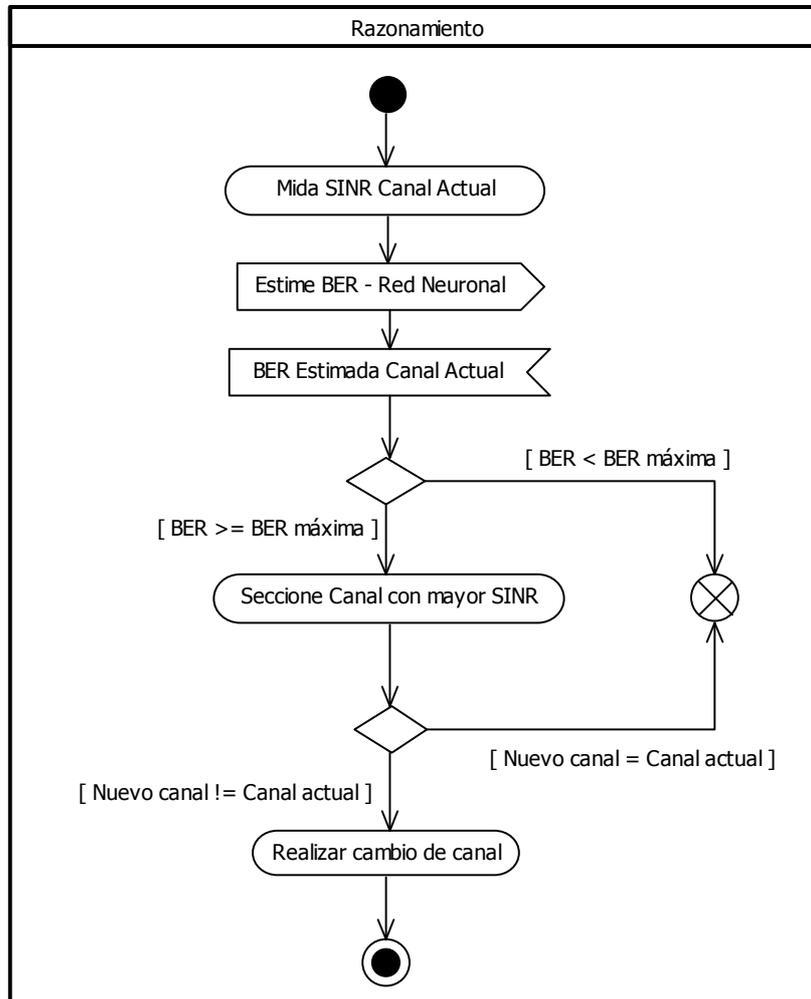


Figura 4.5. Diagrama de flujo función de control para selección de canal

Por su parte, la figura 4.5 presenta la vista de estados de la operación del motor de aprendizaje. Esta inicia en un canal de comunicación preestablecido, llevando el motor de aprendizaje al estado aprendiendo, donde se realizan las funciones de sensado y aprendizaje. El estado modelado representa la red neuronal entrenada con la capacidad de caracterizar el desempeño de error. Una vez se ha modelado este desempeño, se pasa al estado razonando, a partir del evento razonar, en este estado se realiza la transmisión y recepción de los datos propios de la aplicación de la WSN, donde se mide la SINR para cada trama de datos recibida, y se estima la BER a partir de la red neuronal que fue entrenada en la función de aprendizaje.

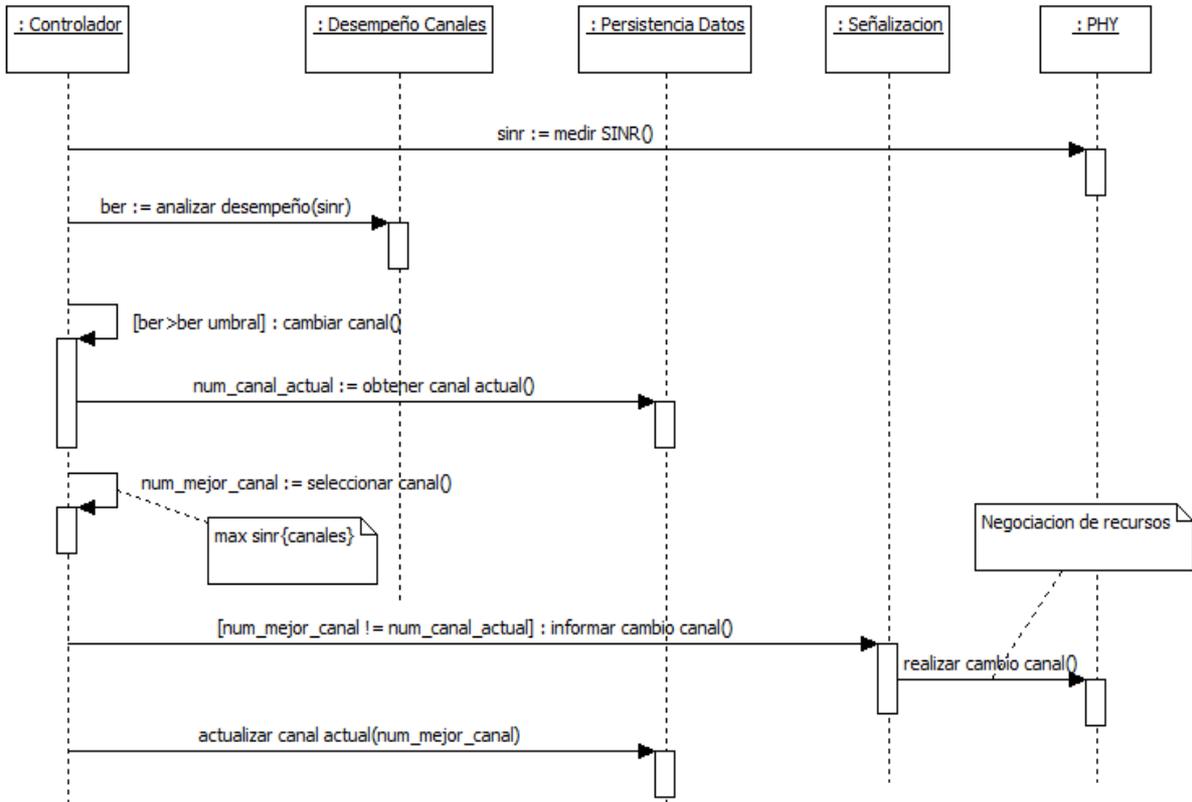


Figura 4.6. Diagrama de interacción función de control

El valor de BER estimado se utiliza para tomar una decisión si el canal es bueno o no, y dependiendo de este resultado se pasa al estado actuando, en el cual, cuando la BER estimada supera la BER umbral, se ejecuta el procedimiento necesario para negociar, realizar y mantener un nuevo canal, de lo contrario no se realiza ningún cambio. Del estado razonando puede pasarse al estado aprendiendo o evaluando, al estado aprendiendo se pasa de forma periódica, con el objetivo de que la red neuronal que modela el desempeño sea re-entrenada y de esta manera caracterice el desempeño de una forma acertada, modelando los cambios en el ambiente de radiofrecuencia

Por su parte, al estado evaluando, se pasa una vez se haya realizado un cambio en el canal de comunicación utilizado o de capa física, para medir la efectividad del motor de aprendizaje, en este estado son enviadas tramas de PN para realizar la evaluación de una forma similar a la operación del estado razonando. Si el resultado

de la evaluación es exitoso se realiza un conteo de éxito, con el fin calcular una tasa de mejora del motor de aprendizaje, de lo contrario, es necesario realizar un re-aprendizaje, para actualizar el mecanismo de caracterización de desempeño.

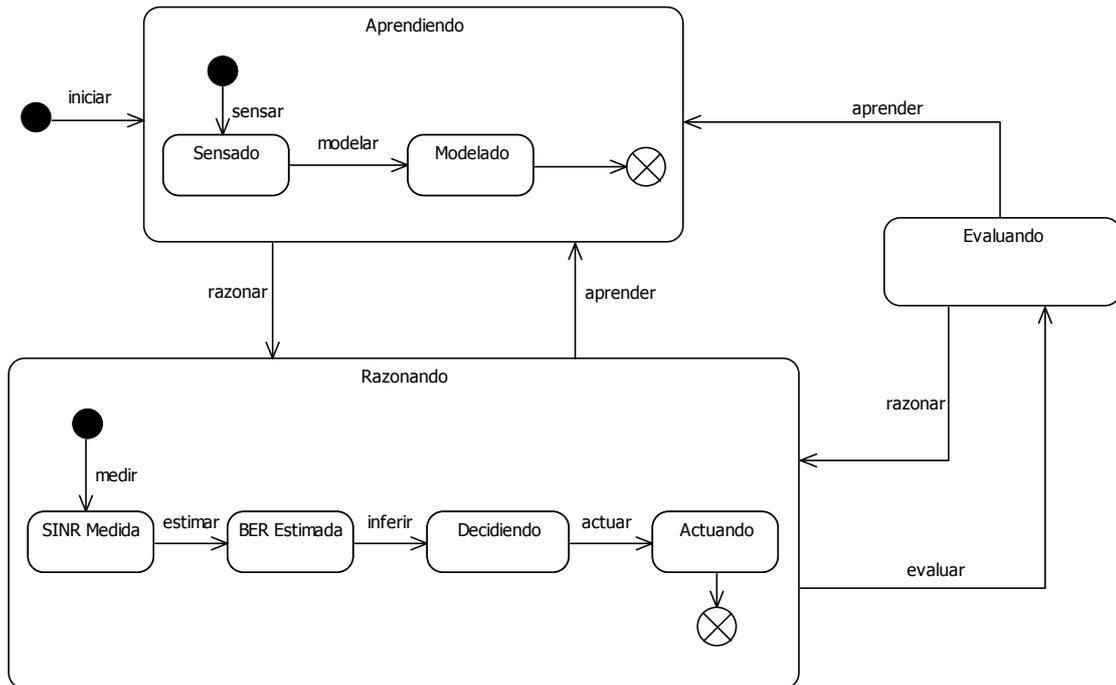


Figura 4.7. Motor de aprendizaje en CWSN

4.4 Persistencia de datos

La figura 4.8 presenta el modelo entidad-relación de la función de persistencia de datos. Este modelo está conformado por cuatro tablas: sensado, medidas ambientales, desempeño y cabeceras. La tabla sensado tiene cinco atributos: id, numero_canal, sinr, ber y cabecera_id, y es utilizada para almacenar y recuperar los datos de sensado. La tabla medidas ambientales permite registrar los datos de sensado representativos, tales como la SINR y BER promedios para cada canal de comunicación. La tabla desempeño representa los parámetros de la red neuronal que caracteriza el desempeño de error de la red, sus valores permiten la ejecución del algoritmo de cálculo de la red neuronal y así estimar la BER a partir de la SINR medida para un paquete de datos recibidos por los nodos cabecera y/o coordinador.

La tabla cabeceras define el número de árboles de *clusters* que conforman la CWSN y es utilizada para relacionar los datos de sensado, las medidas ambientales y los parámetros de desempeño para cada *cluster* de la red; en el caso particular de una red en estrella, donde se dispone de nodos sensores conectados de forma directa con el nodo coordinador, se asume que el nodo coordinador es también el nodo cabecera de *cluster* y por ende la tabla cabeceras tendrá un solo registro donde el campo número tendrá el valor uno. Por su parte, las tablas *capa_fisica_sensado* y *capa_fisica_ma* son utilizadas para relacionar las medidas de sensado y ambientales con las capas físicas disponibles, las cuales serán identificadas mediante un código de tipo entero.

El esquema relacional de la figura 4.8 puede ser implementado en un sistema de gestión de bases de datos, y sus tablas pobladas con datos. Para recuperar información de la base de datos resultante se requiere un lenguaje de consulta formal conocido como álgebra relacional. Al aplicar operaciones del álgebra relacional se forman nuevas tablas, las cuales pueden ser nombradas y almacenadas. Se dispone de cuatro operaciones básicas: unión, intersección, diferencia y producto cartesiano; y cuatro operaciones especiales: selección, proyección, unión y división. El álgebra relacional también incluye seis operadores de comparación: igual, diferente, menor que, mayor que, menor igual que y mayor igual que (Halpin & Morgan, 2008).

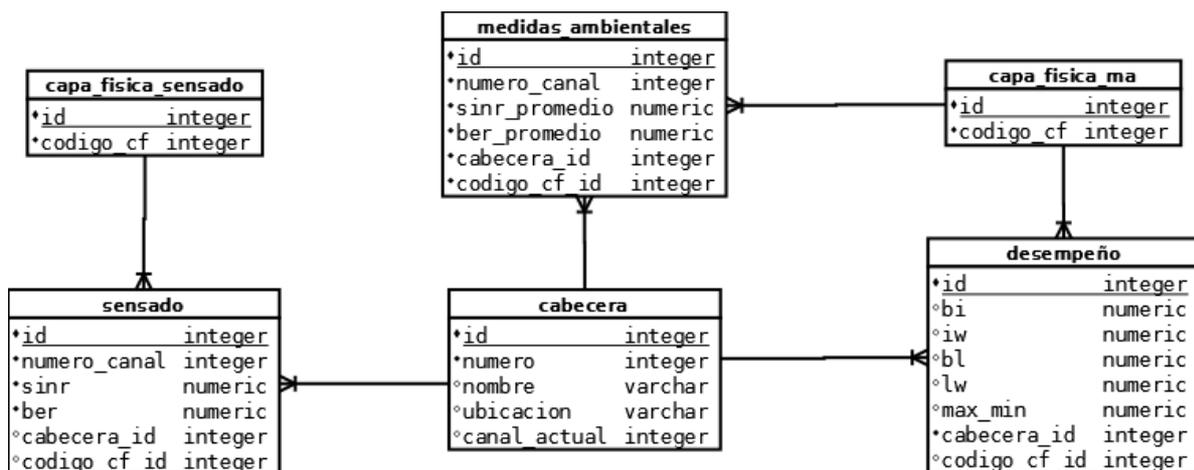


Figura 4.8. Modelo entidad-relación función de persistencia de datos

En este capítulo se ha presentado el diseño de las funciones cognitivas: sensado, aprendizaje, control y persistencia de datos. En especial, para la función de aprendizaje donde se emplea una red neuronal multicapa con alimentación hacia adelante, y se ha ilustrado a partir de la selección dinámica de canal. Como se verá en el capítulo 5, el mecanismo de aprendizaje, realizado a partir del modelado del desempeño de error frente a la relación señal a ruido más interferencia será utilizado para el segundo caso de operación del motor de aprendizaje: la selección dinámica de capa física.

Capítulo 5

Escenarios de operación del motor de aprendizaje y métricas de desempeño

En este capítulo se presenta una descripción de los escenarios de operación del motor de aprendizaje y las métricas de desempeño empleadas para su valoración. Los escenarios de operación esperados para el motor de aprendizaje son:

- Selección dinámica de canal, y
- Selección dinámica de capa física

Tanto la selección dinámica de canal como la de capa física son iniciadas por el motor de aprendizaje, el cual es visto como un agente autónomo desplegado en el nodo Coordinador que dirige las funciones cognitivas de una WSN. La figura 5.1 muestra estos escenarios como casos de uso de UML.

La selección dinámica de canal y de capa física se fundamenta en la función de razonamiento, que a su vez depende de las funciones de aprendizaje y sensado; la función de aprendizaje se divide en dos sub-funciones: el entrenamiento y la estimación, el entrenamiento caracteriza el desempeño de error, mientras que la estimación permite predecir el desempeño de error para una relación de señal a ruido más interferencia presente en el canal de radio.

Las secciones 5.1 y 5.2 presentan los escenarios de operación, tanto para una CWSN en topología en estrella como para una red en árbol de *clusters*, y consideran un canal AWGN como canal de comunicación.

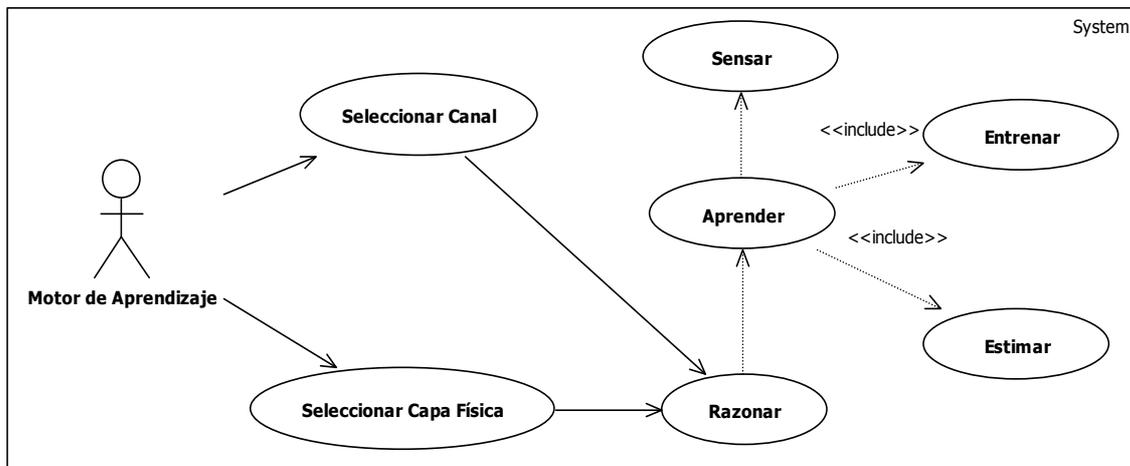


Figura 5.1. Escenarios de operación motor de aprendizaje

Para analizar la operación del motor de aprendizaje y valorar su desempeño en los escenarios planteados, se recurrió a la simulación de una CWSN mediante Matlab.

Es importante indicar que al iniciar este trabajo de investigación, se había planteado el uso del simulador de red NS para valorar el desempeño del motor de aprendizaje, y para ello se realizaron simulaciones de WSNs en estrella, árbol de *clusters* y nodales (Véase el anexo B), lo cual permitió una mayor comprensión de la operación de estas redes. Sin embargo, dado que NS es de mayor utilidad para la simulación de protocolos de las capas de enlace y red de OSI, no fue utilizado para el análisis y valoración del motor de aprendizaje en la capa física objeto de este proyecto.

La metodología de simulación empleada combina dos aproximaciones, la primera es la simulación de sistema de telecomunicaciones, desarrollada principalmente a partir de las pautas presentadas por Astaiza (2007) y complementadas por Gavini (2007) cuyo resultado es el análisis cuantitativo de las capas físicas presentado en el anexo A; y la segunda, la simulación de redes neuronales, tanto para la clasificación de patrones, como para aproximación de funciones a datos de entrada-salida.

5.1 Selección dinámica de canal

Este escenario considera una WSN donde se dispone de una sola capa física pero diferentes canales de radio para realizar la transmisión y recepción de datos, tal como sucede en una WSN basada en la tecnología de comunicación IEEE 802.15.4 en la banda de 2400 MHz donde se tienen 16 canales de radio, numerados desde el 11 al 24.

En este escenario el mecanismo de selección dinámica de canal es una cualidad importante para garantizar un buen desempeño de error de la WSN, debido a que la banda de operación es compartida por otras tecnologías inalámbricas, quienes pueden ser fuente de interferencia para la WSN y viceversa.

5.1.1 CWSN en estrella

El motor de aprendizaje incorporado en la WSN requiere la realización de las funciones cognitivas según lo definido en el modelo CWSN. Considérese una CWSN en estrella como la de la figura 3.2, un nodo coordinador que soporta las funciones de sensado, aprendizaje y control, y cinco nodos sensores. Considérese también una sola capa física, por ejemplo la definida en IEEE 802.15.4 mediante el esquema DSSS-OQPSK; y disponibilidad de 16 canales de radio para realizar la transmisión y recepción de los datos.

Una vez desplegada la CWSN, el nodo coordinador iniciará el envío de *beacons* de red, utilizados para sincronización, y se utilizará el canal 11 como canal de comunicación inicial. Una vez los nodos sensores se hayan asociado a la red, el motor de aprendizaje ejecuta sus funciones para:

- Enviar mensajes de sincronización, señalización y control hacia los nodos para coordinar el envío de tramas de datos PN que serán utilizadas para realizar las tareas de la función de sensado, para cada uno de los 16 canales disponibles.

- Entrenar la red neuronal que caracteriza el desempeño de error, el resultado del entrenamiento arrojará los pesos de las conexiones y umbrales de las neuronas de la red.
- La función de control seleccionará el mejor canal disponible para la comunicación y comunicará la decisión a los nodos de la red para que se realice la adaptación.
- Una vez realizada la selección de canal, a cada trama de datos propia de la aplicación de la WSN recibida por el nodo coordinador se le mide la SINR y se estima la BER utilizando la red neuronal entrenada.
- La función de control infiere sobre la calidad del canal de comunicaciones utilizado actualmente y toma la decisión de realizar cambio de canal o no.

Las figuras 5.2, 5.3 y 5.4 presentan las interacciones generadas entre los nodos de la red para este escenario a partir de un diagrama de secuencias.

La figura 5.2 representa las interacciones cuando se ejecuta la función de sensado. El motor de aprendizaje inicia esta función en el nodo coordinador de la WSN, los mensajes dos y tres permiten fijar el número de tramas esperadas y el número de canales a escanear. El número de tramas permite que las medidas de la SINR y cálculos de la BER sean representativos, por ejemplo a partir de la determinación de SINRs y BERs promedio. El envío de las tramas de datos PN se realiza para cada canal y para cada nodo. Para esto, en el nodo sensor se generan las tramas de datos, se realiza el mapeo de bits a símbolos, de símbolos a chips, se modulan utilizando la técnica O-QPSK, se retarda el canal Q un tiempo $T_b/2$ y se conforman los pulsos de media onda. Esto se representa por los mensajes cinco a diez. Luego se realiza la transmisión de cada trama sobre el canal de radio AWGN, el cual modela el ruido y la interferencia presente en el ambiente de radio.

En el nodo coordinador, las tramas de datos recibidas se filtran en media onda, se realiza la medición de la SINR y se almacena su valor. En el diagrama de la figura 5.2 el mecanismo de gestión de información se ha denominado Persistencia de Datos y se espera que sea desplegado en el nodo coordinador. Seguidamente, se

realiza la sincronización de bits, la demodulación y el des-ensanchamiento para obtener los datos en recepción, donde se realiza el conteo de los bits erróneos y el cálculo de la BER. Finalmente, los mensajes 21 a 26 representan la determinación de medidas representativas, a partir del cálculo de SINRs y BERs promedio para cada canal de radio disponible.

La figura 5.3 presenta las interacciones que se dan en la función de aprendizaje, en particular cuando se realiza el entrenamiento de una red neuronal que caracteriza el desempeño de error. En una topología en estrella, la caracterización de desempeño se realiza en el nodo coordinador y no requiere el intercambio de mensajes hacia o desde los nodos sensores.

El entrenamiento de la red neuronal se realiza recuperando los datos de SINRs y BERs promedio, normalizándolos y posteriormente aplicándolos como vectores de entrada-salida de ejemplo para el ajuste de pesos y umbrales. Al finalizar el procedimiento de entrenamiento, el algoritmo entrega los pesos, umbrales y, valores máximos y mínimos para SINRs y BERs, definidos como los parámetros de la red neuronal para caracterización de desempeño. Así como el valor de error logrado en el ajuste de pesos y umbrales, bien sea cuando se alcance un valor de error esperado o cuando el algoritmo termine el número máximo de iteraciones sin alcanzar el valor de convergencia. Este error, basado en el criterio de convergencia Chi-cuadrado permite evaluar el desempeño de la función de aprendizaje y si es el caso, repetir el procedimiento de aprendizaje hasta que el error sea menor a un error máximo esperado, o se cumpla un número preestablecido de iteraciones.

Las interacciones entre los nodos de la red para la función de razonamiento, representadas en el diagrama 5.4, inician cuando el nodo coordinador recibe una trama de datos enviada por un nodo sensor, a la cual se le mide la SINR y se estima la BER utilizando la red neuronal entrenada previamente en la caracterización de desempeño. Si la BER del canal actual es mayor a una BER umbral, se selecciona el canal con BER mínima y se realiza el procedimiento de señalización entre los nodos para realizar los ajustes necesarios para efectuar el cambio de canal.

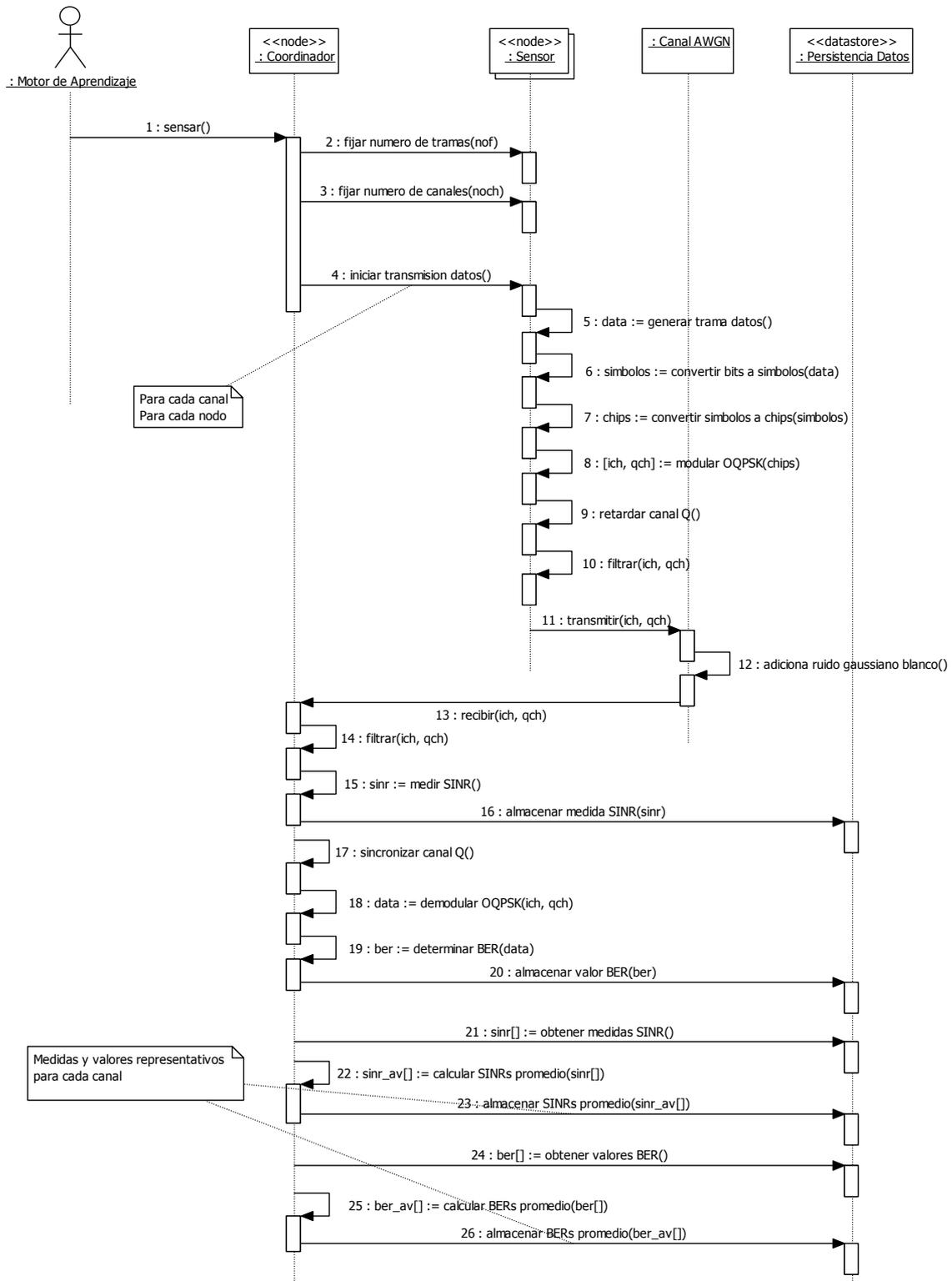


Figura 5.2. Interacciones función de sensado

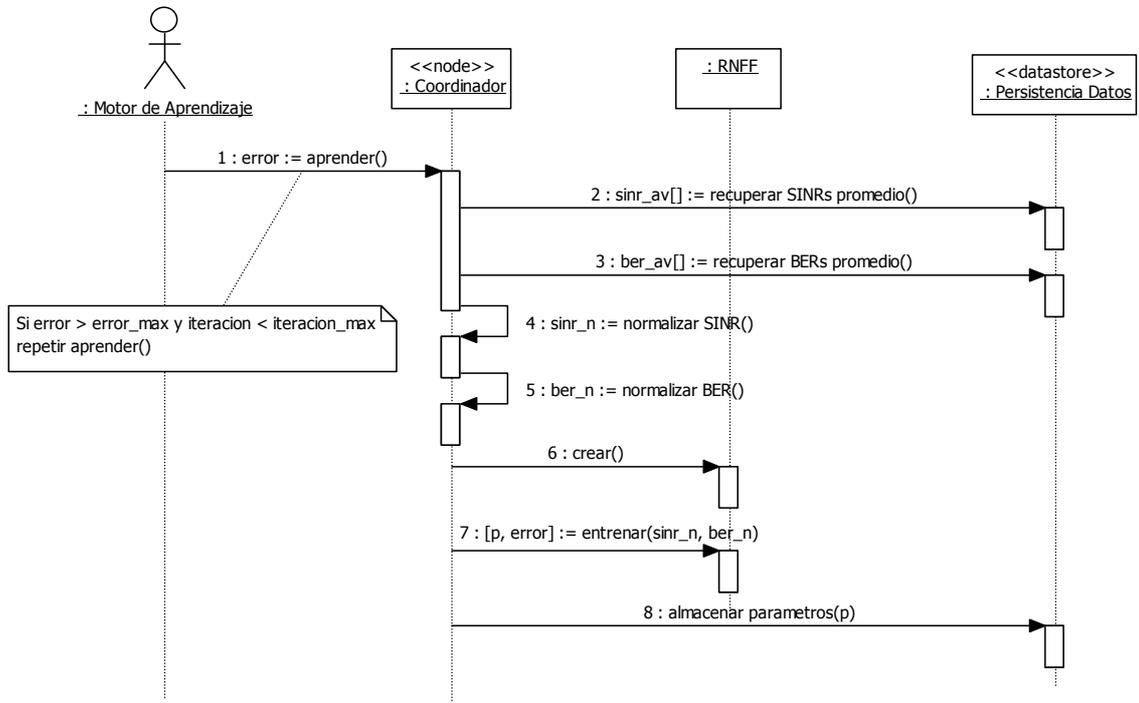


Figura 5.3. Interacciones función de aprendizaje

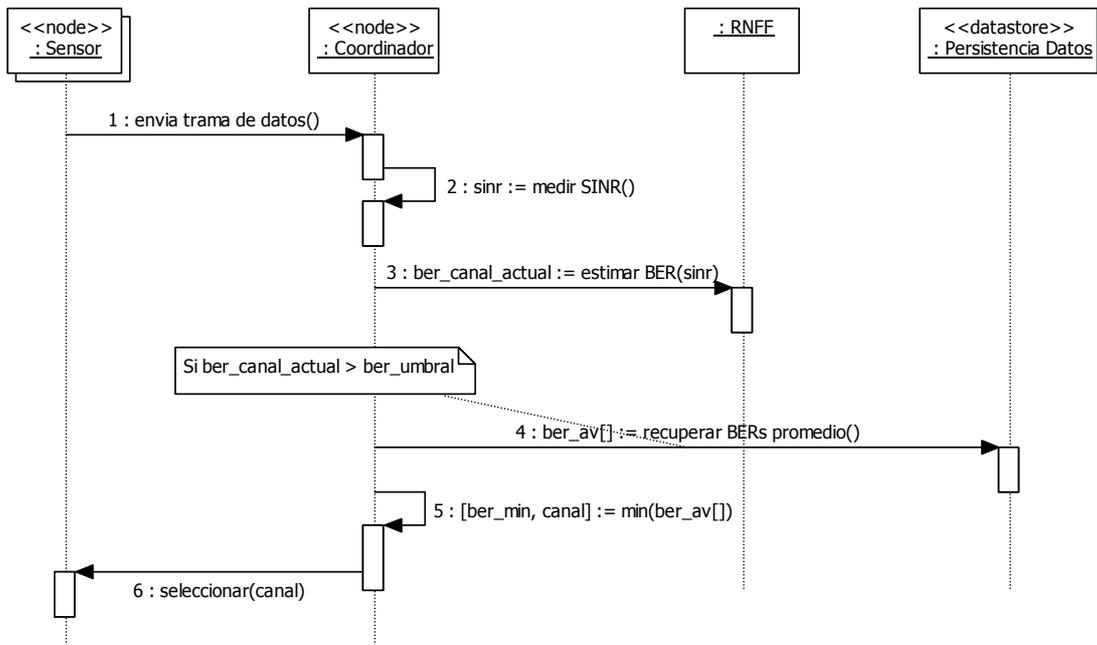


Figura 5.4. Interacciones función de razonamiento

Este escenario Selección dinámica de canal para una CWSN en estrella se desarrolló en Matlab, con el fin que pueda ser simulado y analizado. El programa de simulación implementa las funciones de sensado, aprendizaje y razonamiento a partir de las especificaciones del modelo CWSN, y los diagramas de secuencia de las figuras 5.2 a 5.4.

El escenario de simulación está conformado por seis nodos, un nodo coordinador y cinco nodos sensores en una topología en estrella. Se disponen de 16 canales de radio y cada nodo realiza el envío de 10 tramas de datos, cada trama de 27 bytes, 5 de ellos son bytes de la cabecera y 22 corresponden a la carga útil de la trama. Estos 22 bytes representan secuencias de PN utilizadas para estimar la BER para cada canal de radio. Inicialmente se generan 16 números pseudo-aleatorios con distribución uniforme entre 0 y 1, que representan valores de SINR de las tramas recibidas bajo condiciones de no interferencia. El rango seleccionado obedece a que se espera que 802.15.4 opere en ambientes ruidosos y en presencia de interferencia. Posteriormente, se ejecutan de forma secuencial las funciones de sensado, aprendizaje y razonamiento, cuyos resultados se grafican en la figura 5.5.

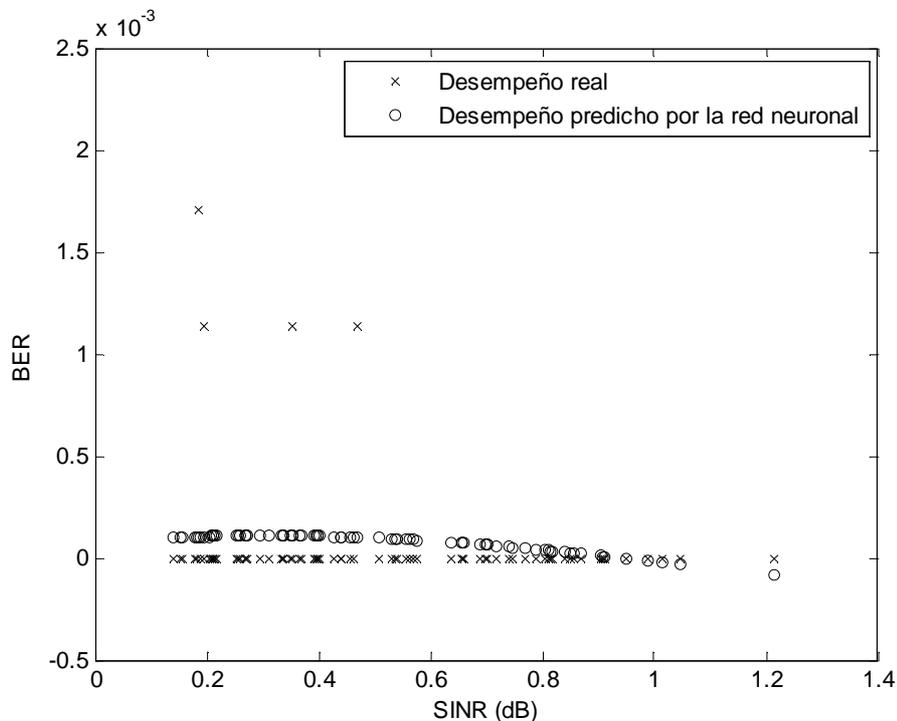


Figura 5.5. Caracterización del desempeño canales sin interferencia

Los pesos de las conexiones y umbrales obtenidos en la simulación se consignan en la tabla 5.1. Los nombres dados en la tabla corresponden al nombre de los vectores presentados en el algoritmo de cálculo de red neuronal de la sub-sección 4.2.2.

Tabla 5.1. Pesos y umbrales obtenidos para la red neuronal cuando no se presenta interferencia

Nombre	Valor
<i>iw</i>	$[-0.2373 \quad 0.4037 \quad 2.6450 \quad 1.3545]^T$
<i>bi</i>	$[0.5147 \quad -0.1274 \quad -2.0959 \quad -0.0361]^T$
<i>lw</i>	$[1.5589 \quad -5.0000 \quad 0.3105 \quad 1.0923]$
<i>bl</i>	$[-1.1903]$

Si se asume que el canal actual es el canal 22, cuando una trama de datos llegue al nodo coordinador, éste estimará la BER para la SINR medida, la regla de inferencia establece que si la BER estimada es mayor a la BER umbral, el canal se clasifica como malo y es necesario realizar la selección de un nuevo canal, de lo contrario no se realiza ninguna acción. Para el caso que se ha considerado, la BER estimada para un trama cuya SINR sea de 0.96 dB es de $7.48 * 10^{-5}$, que es menor a la BER umbral $= 2 * 10^{-4}$, tomando este valor del análisis de desempeño de 802.15.4 en la banda de 2400 MHz del anexo A. Por lo tanto, el canal es bueno, al igual que el desempeño, y no se requiere realizar ninguna acción.

Si ahora, el ambiente de radiofrecuencia de la CWSN cambia, debido a la presencia de fuentes de interferencia que afecten los canales. Por ejemplo un dispositivo IEEE 802.11b interferente, que transmita en el canal 11 de 802.11b cuya frecuencia central es 2462 MHz y su ancho de banda de 22 MHz, la BER de los canales 21, 22, 23 y 24 de 802.15.4, cuyas frecuencias centrales son 2455, 2460, 2465 y 2470 MHz, respectivamente, cada uno con un ancho de banda de 2 MHz, aumentará notablemente, pudiendo ser de alrededor 10^{-1} (IEEE, 2006). En estas condiciones el canal 22 ya no es el mejor canal para la transmisión.

La SINR de los canales expuestos a la fuente interferente estará entre -8 y -7 dB. Las funciones de sensado y aprendizaje se realizan de forma periódica con el objetivo de

modelar los cambios del ambiente de radio. De esta manera el desempeño real y caracterizado puede ser como el que se presenta en la figura 5.6.

En estas condiciones, la función de razonamiento infiere que debe realizarse el cambio de canal, debido a que la BER estimada supera a la BER umbral, por ejemplo, para SINR igual a -7.5 dB la BER estimada es 0.1105. La selección del nuevo canal se realiza buscando el canal que tenga la menor BER promedio, para el caso en consideración uno de los canales diferente a los canales 21, 22, 23 y 24, ya que se ha asumido que estos canales no han sufrido interferencia y siguen siendo buenos. Una vez realizada la selección de canal, el nodo coordinador gestiona el cambio de canal con los nodos sensores utilizando la función de señalización.

Los pesos de las conexiones y umbrales obtenidos en la simulación se consignan en la tabla 5.2.

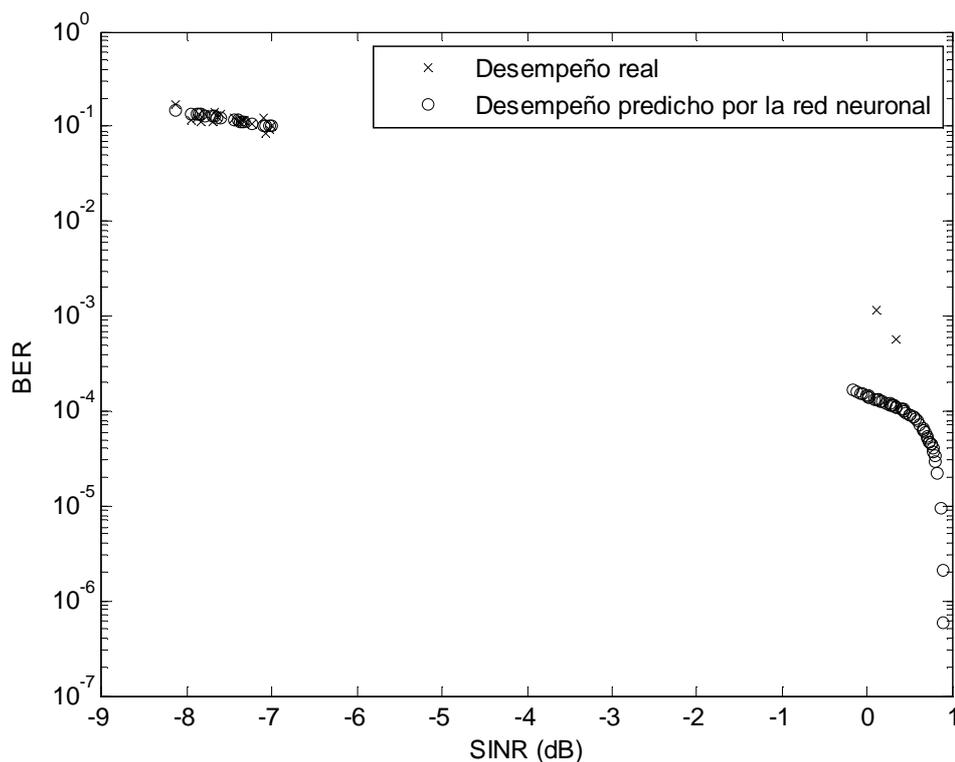


Figura 5.6. Caracterización del desempeño en presencia de una fuente 802.11b interferente

Tabla 5.2. Pesos y umbrales obtenidos para la red neuronal cuando se presenta interferencia

Nombre	Valor
<i>iw</i>	$[4.4645 \ 0.4301 \ -3.3123 \ 0.2904]^T$
<i>bi</i>	$[4.2389 \ 0.0935 \ -1.6836 \ 0.3742]^T$
<i>lw</i>	$[-0.4957 \ -0.5337 \ 0.7303 \ 0.8904]$
<i>bl</i>	$[-0.0354]$

Teniendo en cuenta la estructura de supertrama de la figura 2.3, el intervalo de *beacon* está definido por la duración de la supertrama y la duración de la parte inactiva. A su vez, en el modo de acceso por contención se dispone de 16 intervalos de tiempo de 60 símbolos de duración para el envío de datos entre los nodos sensores y el nodo coordinador. Con el fin de estimar el tiempo que toma la función de sensado, considere que las tramas de PN para esta función son enviadas por cada nodo y para cada canal en uno de los intervalos de tiempo del CAP. Además, si el orden de *beacon* es 8 y el orden de supertrama es 7, el intervalo de *beacon* será de 3.932 segundos (IEEE, 2006). Así, si cada nodo envía 10 tramas de sensado en los 16 canales disponibles, el tiempo necesario será $10 \times 16 \times 3.932 = 629.12$ segundos, equivalente a 10.5 minutos aproximadamente.

Esta estimación no tiene en cuenta el tiempo requerido para realizar cambios de canal y sincronización entre los nodos de la red, sin embargo, permite notar que la función de sensado puede ser relativamente intensiva en cantidad de tramas y tiempo, y da pie a pensar en alternativas para su implementación. Por ejemplo, puede realizarse el sensado en cada canal a partir de los datos propios de la aplicación, mediante la codificación para estimación de error (Chen et al, 2010). En este caso, el nodo coordinador gestiona el envío de datos en cada canal hasta que haya recogido los datos de sensado suficientes, y ejecuta las funciones de aprendizaje y razonamiento para seleccionar el mejor canal disponible.

5.1.2 CWSN en árbol de *clusters*

Al igual que en una CWSN en estrella, en esta topología la función de sensado es iniciada por el motor de aprendizaje desplegado en el nodo coordinador, el cual

inicia, a su vez, el sensado en los nodos cabecera, y son estos los encargados de dirigir el envío y recepción de las tramas para sensado con los nodos sensores asociados. Esto se representa en el diagrama de interacción de la figura 5.7. Cada nodo cabecera se encarga de enviar peticiones de almacenamiento de los datos de sensado al módulo de Persistencia de datos. Puede notarse que el procedimiento es similar al realizado por una CWSN en estrella, sin embargo es importante tener en cuenta que la gestión de la persistencia de los datos ahora es mayor ya que es necesario relacionar los datos de sensado con cada nodo cabecera de la red. Finalmente, es el nodo coordinador el encargado de obtener las medidas de SINR y BER representativas para cada nodo.

La función de aprendizaje es realizada por el nodo coordinador mediante el entrenamiento de redes neuronales que caracterizan el desempeño de error. Cada nodo cabecera requiere una función de caracterización que modela el desempeño en el ambiente de radio de influencia, y de esta manera, el nodo coordinador debe fijar los parámetros de la función de caracterización en cada nodo cabecera, esto se ilustra en el diagrama de interacciones para la función de aprendizaje de la figura 5.8.

Por su parte, la función de razonamiento debe ser desplegada en cada nodo cabecera, y actúa cuando recibe tramas de datos enviadas por los nodos sensores como se muestra en el diagrama de interacciones de la figura 5.9. La red neuronal que modela el desempeño estima la BER para una SINR de una trama de datos recibida y la compara con una BER máxima.

Es de importancia resaltar que cada nodo cabecera seleccionará el mejor canal disponible para los nodos sensores asociados. Así mismo, el nodo coordinador realiza las funciones de sensado y aprendizaje para las tramas de datos recibidas, bien sea por los nodos cabecera o por nodos sensores, modela el desempeño que el ambiente de radio le presenta y, ejecuta la función de razonamiento para inferir la calidad del canal de radio utilizado, y de esta manera tomar la mejor decisión posible para garantizar un buen desempeño para toda la red. Nótese entonces, que en la función de razonamiento se presenta una interacción entre todos los nodos de la red, e incluso al momento de obtener los valores de BERs promedio de la función Persistencia de datos ya que ésta se encuentra desplegada en el nodo coordinador.

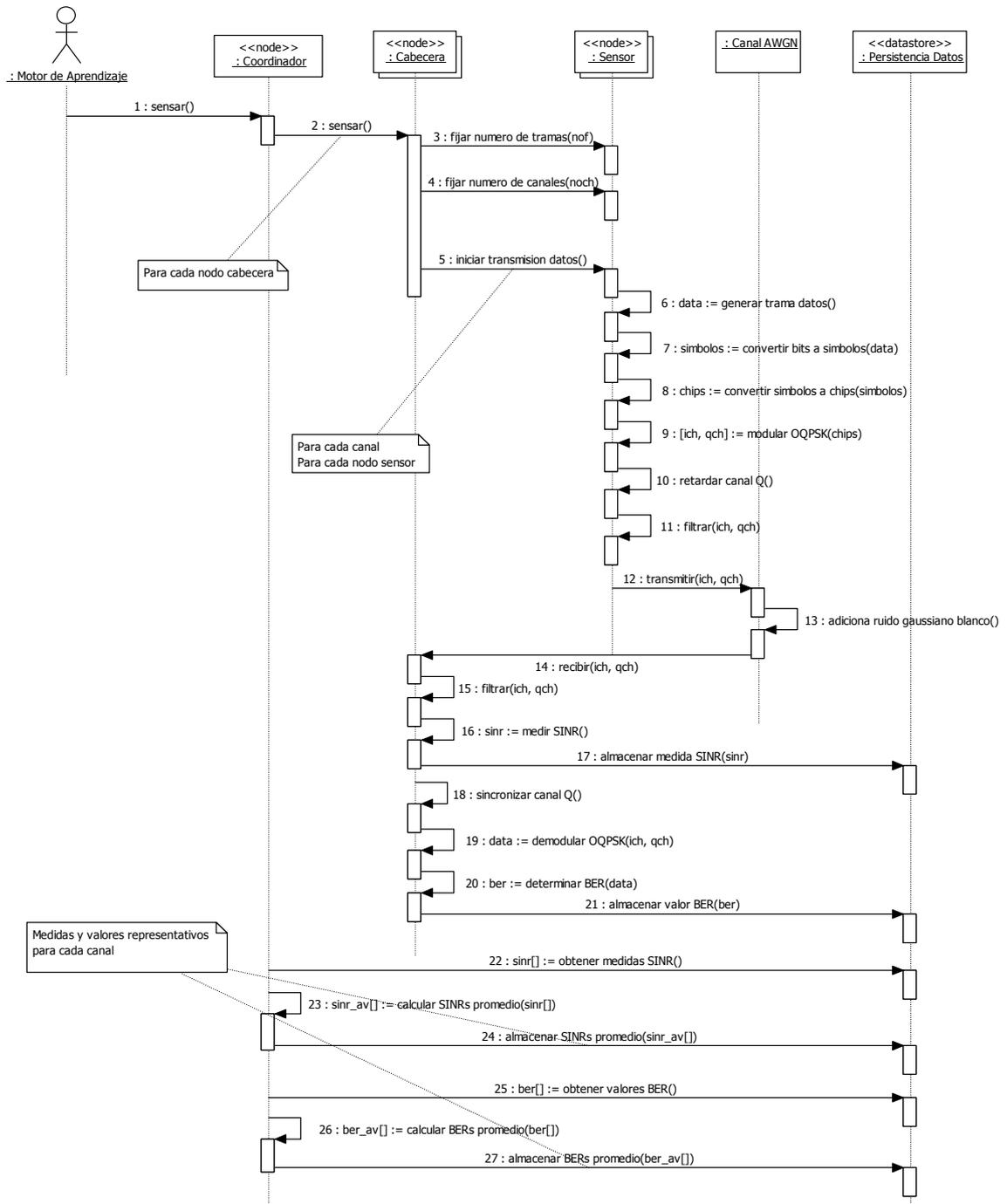


Figura 5.7. Interacciones función de sensado

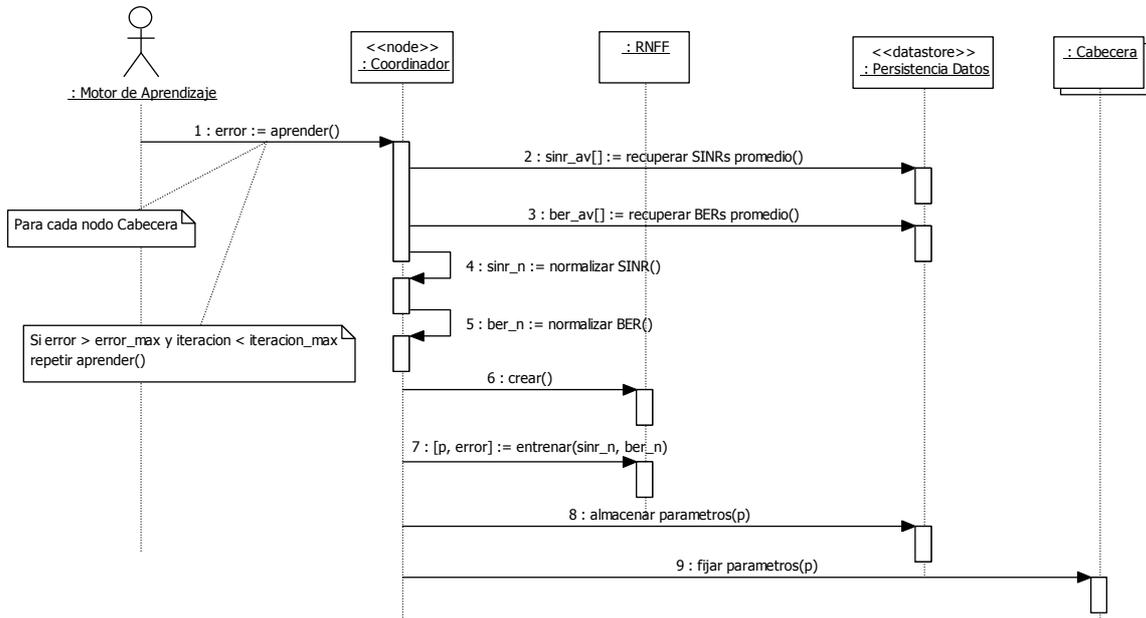


Figura 5.8. Interacciones función de aprendizaje

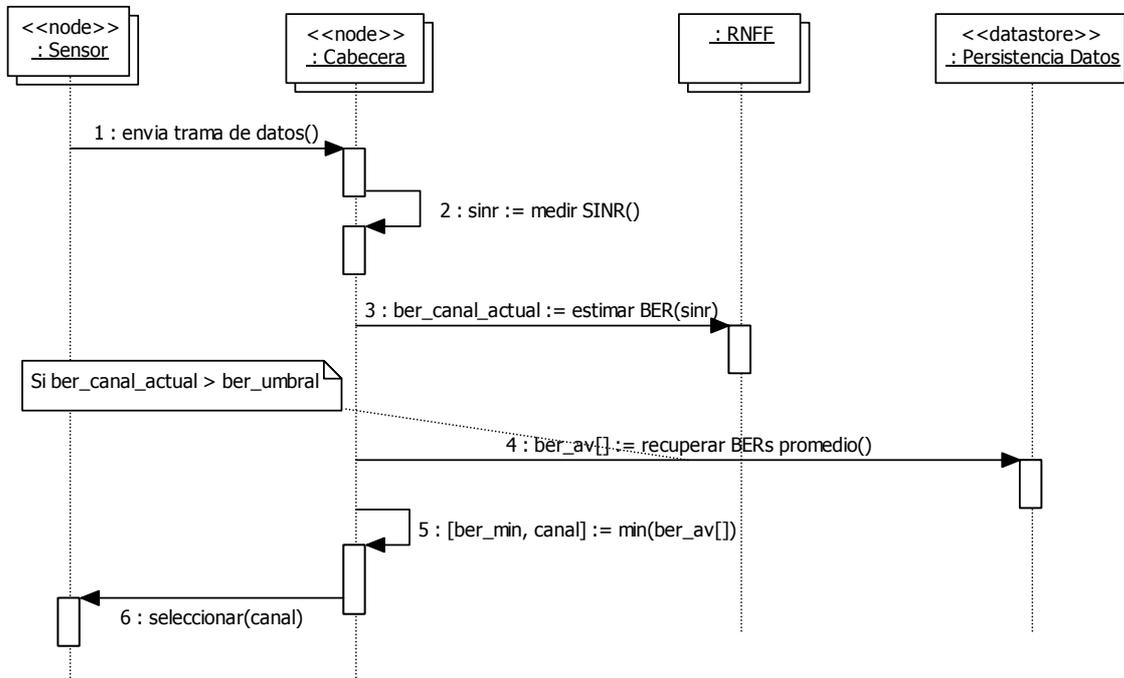


Figura 5.9. Interacciones función de razonamiento

5.2 Selección de capa física

Para este escenario, considérese que una WSN tiene dos capas físicas las cuales pueden seleccionarse de forma dinámica. Como se observa en los resultados de análisis de desempeño de error para 802.15.4 en la banda de 915 MHz presentados en el anexo A, la capa PHY BPSK tiene un mejor desempeño de error, aunque a expensas de una menor tasa de datos.

Por otra parte, la capa PHY ASK permite una mayor tasa de datos pero su desempeño de error disminuye. De acuerdo a estas consideraciones, este escenario es importante cuando se requiere lograr una mayor confiabilidad en la transferencia de datos, no importando en mayor medida su velocidad de transferencia.

5.2.1 CWSN en estrella

Para ilustrar la operación de selección de capa física en una CWSN en estrella conformada por un nodo coordinador y cinco nodos sensores, la función de sensado del motor de aprendizaje debe recoger información de SINR y BER para cada capa disponible, para ello la red inicia en una capa por defecto, por ejemplo la capa PHY ASK, donde la tasa de datos es mayor. Una vez los nodos sensores se hayan asociado a la red, el motor de aprendizaje ejecuta sus funciones para:

- Enviar mensajes de sincronización, señalización y control hacia los nodos para coordinar el envío de tramas de datos PN que serán utilizadas para realizar las tareas de la función de sensado, para cada capa PHY disponible.
- Entrenar redes neuronales que caractericen el desempeño de error en cada capa PHY, el resultado del entrenamiento arrojará los pesos de las conexiones y umbrales de las redes neuronales.
- La función de control seleccionará la mejor capa PHY para la comunicación y comunicará la decisión a los nodos de la red para que se realice la adaptación.

- Una vez realizada la selección de capa PHY, a cada trama de datos propia de la aplicación de la WSN recibida por el nodo coordinador se le mide la SINR y se estima la BER utilizando la red neuronal entrenada.
- La función de control influye sobre la calidad de la transmisión en la capa PHY utilizada actualmente y toma la decisión de realizar cambio a otra capa PHY o no.

Las figuras 5.10, 5.11 y 5.12 presentan las interacciones generadas entre los nodos de la red para este escenario a partir de diagramas de secuencias. La figura 5.10 representa las interacciones cuando se ejecuta la función de sensado.

El motor de aprendizaje inicia esta función en el nodo coordinador de la WSN, los mensajes dos y tres permiten fijar el número de tramas esperadas y el número de muestras a realizar. El número de tramas y de muestras permite que las medidas de la SINR y cálculos de la BER sean representativos, por ejemplo a partir de la determinación de SINRs y BERs promedio.

El envío de las tramas de datos PN se realiza para cada capa PHY disponible y para cada nodo. Para esto, en el nodo sensor se generan las tramas de datos, se realiza el mapeo de bits a símbolos, de símbolos a chips, se modulan utilizando técnicas digitales ASK o BPSK, y se conforman los pulsos de media onda. Esto se representa por los mensajes cinco a nueve. Luego se realiza la transmisión de cada trama sobre el canal de radio AWGN, el cual modela el ruido y la interferencia presente en el ambiente de radio.

En el nodo coordinador, las tramas de datos recibidas se filtran en media onda, se realiza la medición de la SINR y se almacena su valor. En el diagrama de la figura 5.2 el mecanismo de gestión de información se ha denominado Persistencia de Datos y se espera que sea desplegado en el nodo coordinador. Seguidamente, se realiza la sincronización de bits, la demodulación y el des-ensanchamiento para obtener los datos en recepción, donde se realiza el conteo de los bits erróneos y el cálculo de la BER. Finalmente, los mensajes 21 a 26 representan la determinación de medidas representativas, a partir del cálculo de SINRs y BERs promedio para cada capa PHY disponible.

La figura 5.11 presenta las interacciones que se dan en la función de aprendizaje, en particular cuando se realiza el entrenamiento de redes neuronales que caracterizan el desempeño de error para dos capas PHY definidas por las técnicas de modulación ASK y BPSK. En una topología en estrella, la caracterización de desempeño se realiza en el nodo coordinador y no requiere el intercambio de mensajes hacia o desde los nodos sensores. El entrenamiento de la red neuronal se realiza recuperando los datos de SINRs y BERs promedio, normalizándolos y posteriormente aplicándolos como vectores de entrada-salida de ejemplo para el ajuste de pesos y umbrales.

Al finalizar el procedimiento de entrenamiento, el algoritmo entrega los pesos, umbrales y, valores máximos y mínimos para SINRs y BERs, definidos como los parámetros de las redes neuronales para caracterización de desempeño. Así como los valores de error obtenidos en el ajuste de pesos y umbrales, bien sea cuando se alcance un valor de error esperado o cuando el algoritmo termine el número máximo de iteraciones sin alcanzar el valor de convergencia. Este error, basado en el criterio de convergencia Chi-cuadrado permite evaluar el desempeño de la función de aprendizaje y si es el caso, repetir el procedimiento de aprendizaje hasta que el error sea menor a un error máximo esperado, o se cumpla un número preestablecido de iteraciones.

Es de notar que la función de aprendizaje requiere dos redes neuronales para caracterizar el desempeño de cada capa física, la razón está en que los patrones de datos entrada – salida que sirven de entrenamiento para la red ahora son diferentes para cada capa física.

Las interacciones entre los nodos de la red para la función de razonamiento, representadas en el diagrama 5.12, inician cuando el nodo coordinador recibe una trama de datos enviada por un nodo sensor, a la cual se le mide la SINR y se estima la BER utilizando la red neuronal entrenada previamente en la caracterización de desempeño. Si la BER para el canal actual es mayor a una BER umbral, se determina si en otra capa PHY disponible puede lograrse un mejor desempeño, en tal caso se selecciona dicha capa PHY y se realiza el procedimiento de señalización entre los nodos para realizar los ajustes necesarios para efectuar el cambio de capa PHY.

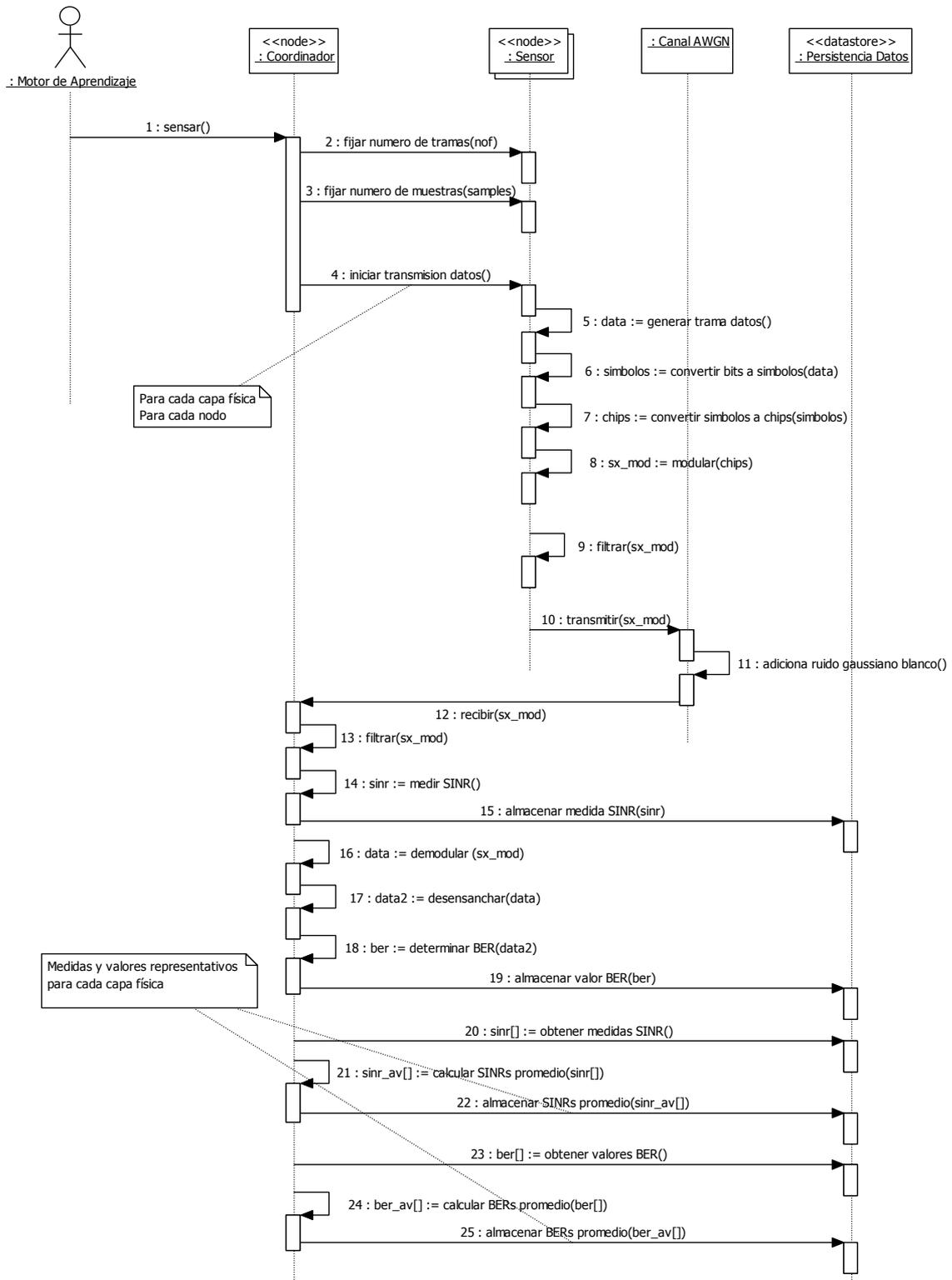


Figura 5.10. Interacciones función de sensado

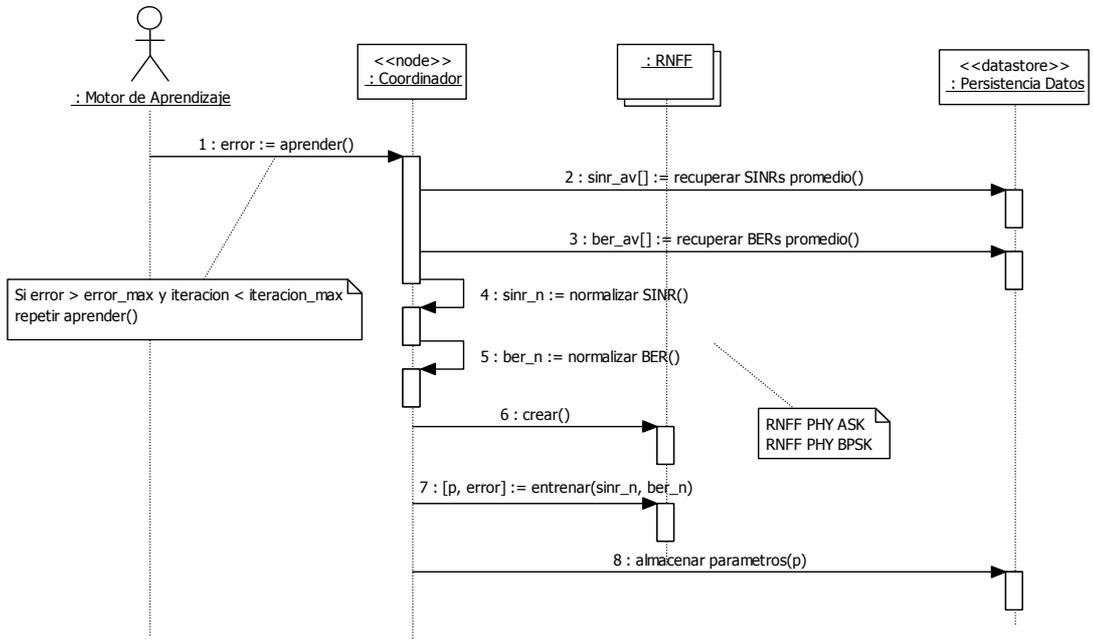


Figura 5.11. Interacciones función de aprendizaje

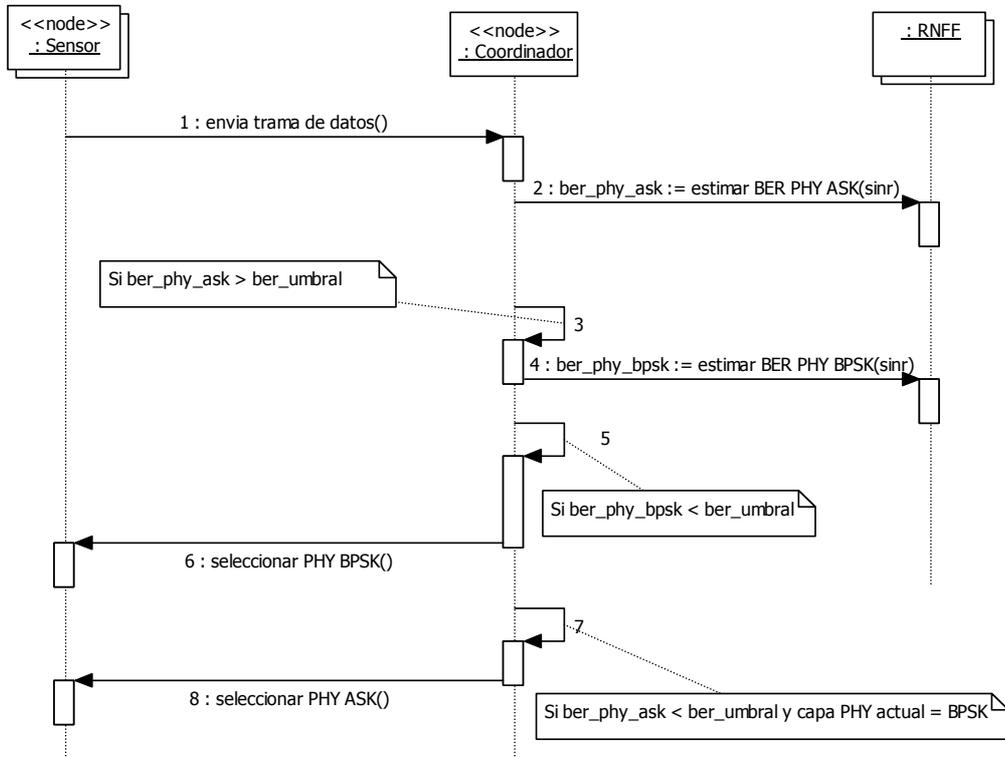


Figura 5.12. Interacciones función de razonamiento

Este escenario Selección dinámica de capa física para una CWSN en estrella se desarrolló en Matlab, con el fin que pueda ser simulado y analizado. El programa de simulación implementa las funciones de sensado, aprendizaje y razonamiento a partir de las especificaciones del modelo CWSN, y los diagramas de secuencia de las figuras 5.10 a 5.12.

El escenario de simulación está conformado por seis nodos, un nodo coordinador y cinco nodos sensores en una topología en estrella. Se disponen de dos capas PHY definidas por los esquemas de codificación-modulación PSSS-ASK y DSSS-BPSK, donde cada nodo realiza el envío de 10 tramas de datos PN, cada trama de 22 bytes. El envío de estas tramas se repite diez veces, lo cual es definido por el número de muestras. Inicialmente se generan cinco números pseudo-aleatorios con distribución uniforme entre 1 y 2, que representan valores de SINR de las tramas recibidas bajo condiciones de no interferencia. Posteriormente, se ejecutan de forma secuencial las funciones de sensado, aprendizaje y razonamiento, cuyos resultados se grafican en la figuras 5.13 y 5.14. Los valores de los pesos y umbrales para las redes neuronales que modelan el desempeño de las capas PHY ASK y PHY BPSK se consignan en las tablas 5.3 y 5.4 respectivamente.

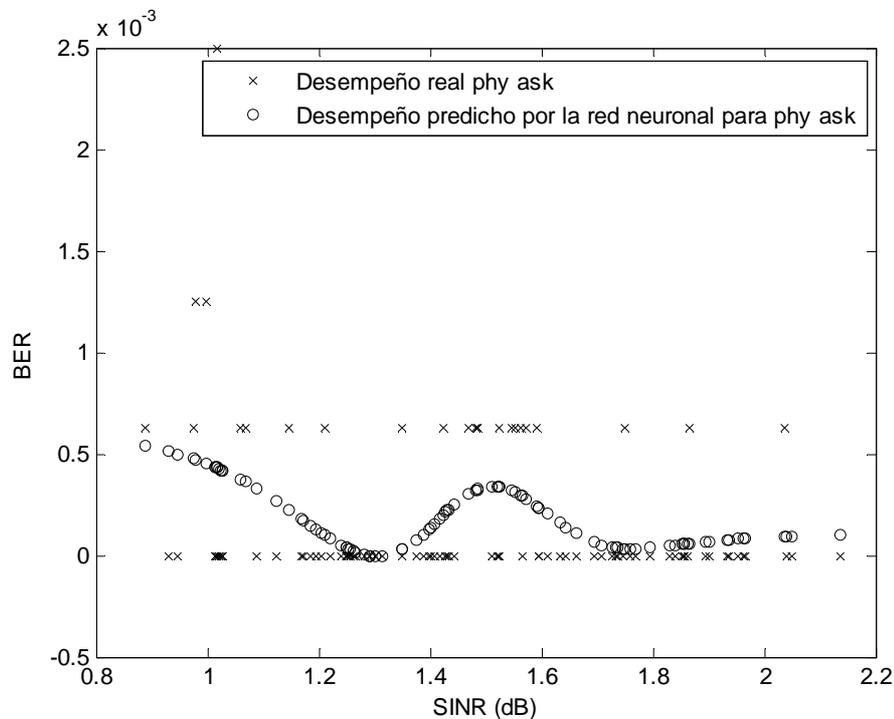


Figura 5.13. Caracterización del desempeño sin interferencia PHY ASK

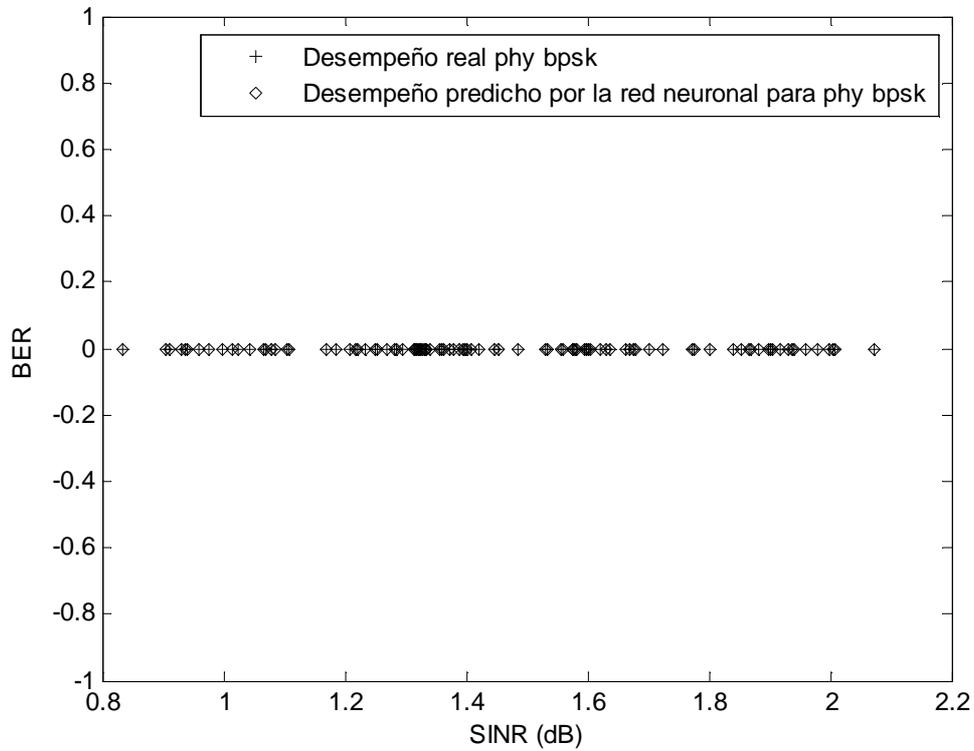


Figura 5.14. Caracterización del desempeño sin interferencia PHY BPSK

Tabla 5.3. Pesos y umbrales obtenidos para la red neuronal PHY BPSK

Nombre	Valor
<i>iw</i>	$[-0.1173 \quad -0.5137 \quad 0.5557 \quad 0.2117]^T$
<i>bi</i>	$[0.4683 \quad 0.9284 \quad 0.8960 \quad -0.3361]^T$
<i>lw</i>	$[2.0255 \quad -0.4239 \quad -1.2013 \quad -3.0492]$
<i>bl</i>	$[-0.8191]$

Tabla 5.4. Pesos y umbrales obtenidos para la red neuronal PHY ASK

Nombre	Valor
<i>iw</i>	$[0.9849 \quad 0.6227 \quad -2.1589 \quad -0.3884]^T$
<i>bi</i>	$[0.8061 \quad -0.4748 \quad -2.6131 \quad 1.5928]^T$
<i>lw</i>	$[-1.0224 \quad -0.5841 \quad 0.3774 \quad 0.6478]$
<i>bl</i>	$[-0.1111]$

Si la SINR es 1.7 dB , la función de control estima la BER en $5.8411 * 10^{-5}$, y la capa PHY ASK presenta una buena calidad. Si las condiciones del ambiente de radio cambian, por ejemplo una SINR entre $-1 \text{ y } 0 \text{ dB}$, las funciones de sensado y aprendizaje modelan el desempeño actual como se muestra en la figuras 5.15 y 5.16. Si la SINR llega a ser de -0.6 dB para una trama de datos recibida, el control estima una BER de 0.0022, llegando a ser un desempeño bajo para la capa física actual, PHY ASK.

De acuerdo a este resultado de desempeño, la alternativa es cambiar a la capa PHY BPSK, así que el control debe determinar si el desempeño de error es bueno para esta nueva capa PHY. Si se estima la BER para la capa PHY BPSK el resultado de error es $4.1482 * 10^{-5}$, indicando que en esta capa física, la calidad es buena y se procede a seleccionarla, señalar y realizar la adaptación necesaria por los nodos de la red.

Como se ha mencionado, ahora la transmisión es más confiable aunque la velocidad de transferencia es menor. Si posteriormente el ruido e interferencia disminuyen, lo que es equivalente a que la SINR aumente, la función de aprendizaje deberá modelar estos cambios y permitirá que el control pueda seleccionar nuevamente la capa PHY ASK.

Para la estructura de supertrama de la figura 2.3, el intervalo de *beacon* está definido por la duración de la supertrama y la duración de la parte inactiva. A su vez, en el modo de acceso por contención se dispone de 16 intervalos de tiempo de 60 símbolos de duración para el envío de datos entre los nodos sensores y el nodo coordinador.

Con el fin de estimar el tiempo que toma la función de sensado, considere que las tramas de PN para esta función son enviadas por cada nodo y para cada capa física en uno de los intervalos de tiempo del CAP. Además, si el orden de *beacon* es 8 y el orden de supertrama es 7, el intervalo de *beacon* será de 3.932 segundos (IEEE, 2006). Así, si cada nodo envía 10 tramas de sensado en cada capa física y para 20 muestras cada uno, el tiempo necesario será $10 \times 20 \times 3.932 = 786.4$ segundo, equivalente a 13.1 minutos aproximadamente.

Esta estimación no tiene en cuenta el tiempo requerido para realizar cambios de capa física y sincronización entre los nodos de la red, sin embargo, permite notar que la función de sensado puede ser relativamente intensiva en cantidad de tramas y tiempo, y da pie a pensar en alternativas para su implementación. Por ejemplo, puede realizarse el sensado en cada capa física a partir de los datos propios de la aplicación, mediante la codificación para estimación de error (Chen et al, 2010). En este caso, el nodo coordinador gestiona el envío de datos en cada capa física hasta que haya recogido los datos de sensado suficientes, y ejecuta las funciones de aprendizaje y razonamiento para seleccionar la mejor capa física disponible.

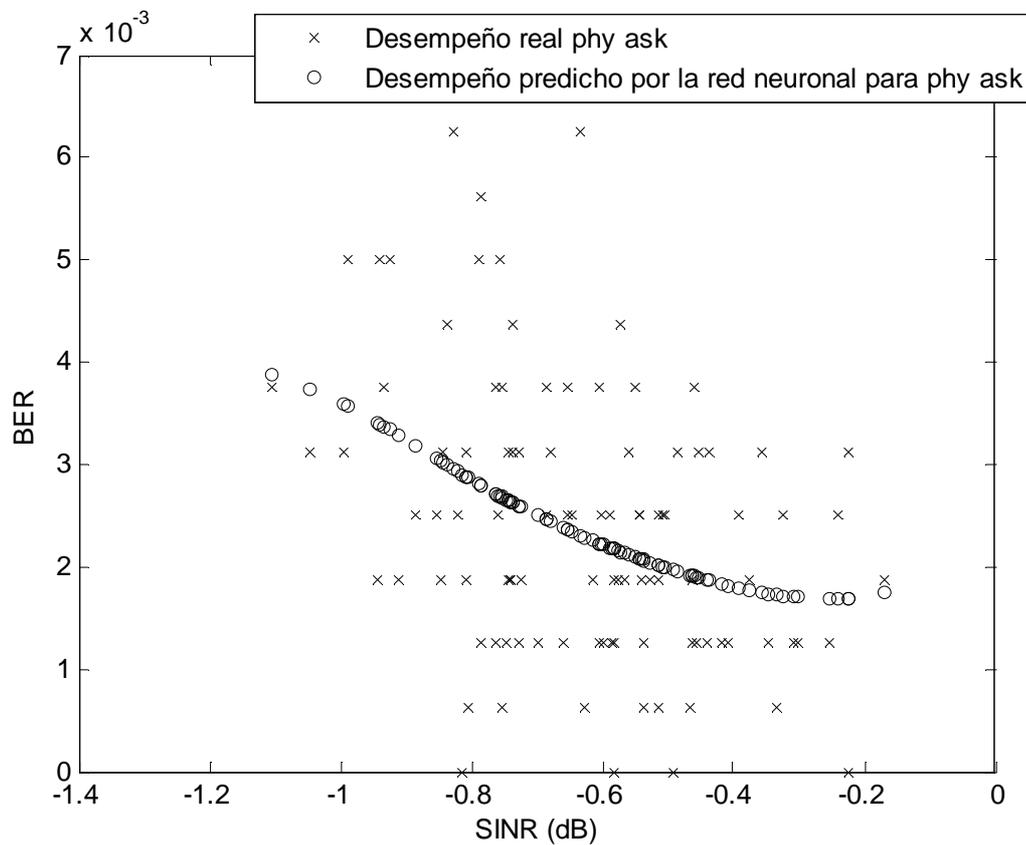


Figura 5.15. Caracterización del desempeño PHY ASK en presencia de interferencia

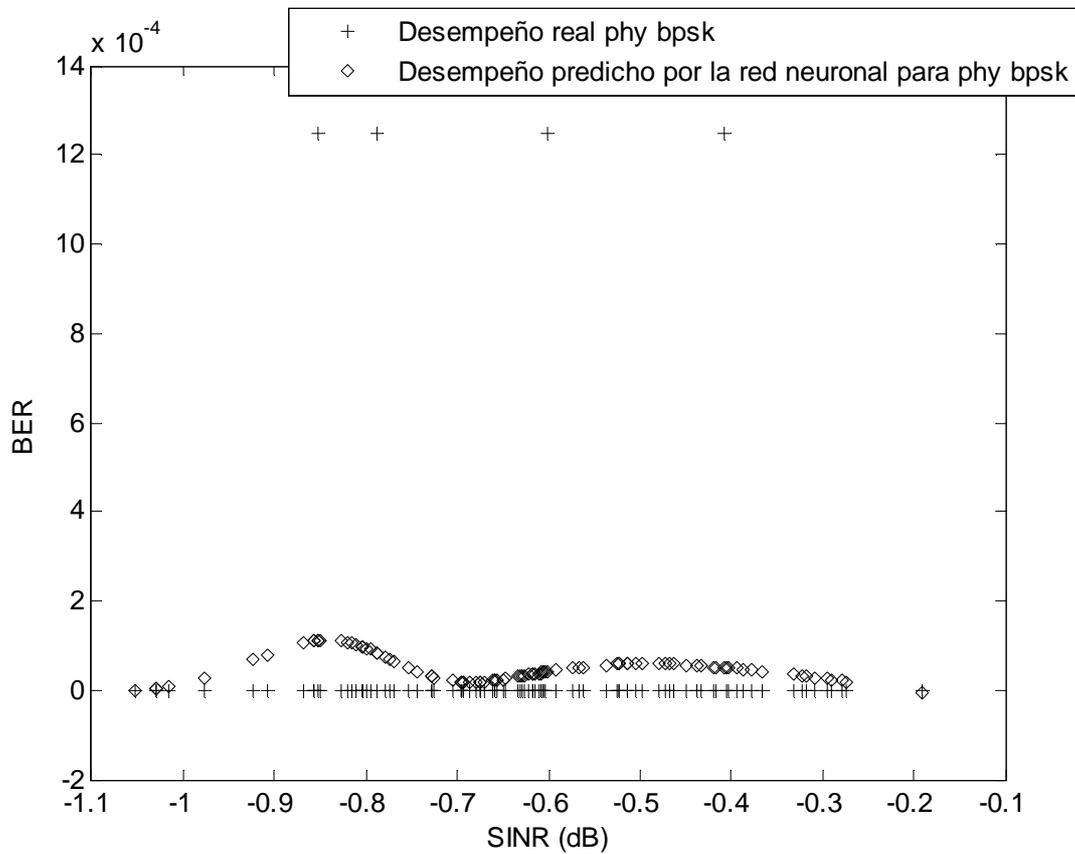


Figura 5.16. Caracterización del desempeño PHY BPSK en presencia de interferencia

5.2.2 CWSN en árbol de clusters

Al igual que en una CWSN en estrella, en esta topología la función de sensado es iniciada por el motor de aprendizaje desplegado en el nodo coordinador, el cual inicia, a su vez, el sensado en los nodos cabecera, y son estos los encargados de dirigir el envío y recepción de las tramas para sensado con los nodos sensores asociados.

Esto se representa en el diagrama de interacción de la figura 5.17. Cada nodo cabecera se encarga de enviar peticiones de almacenamiento de los datos de sensado al módulo de Persistencia de datos. Puede notarse que el procedimiento es similar al realizado por una CWSN en estrella, sin embargo es importante tener en

cuenta que la gestión de la persistencia de los datos ahora es mayor ya que es necesario relacionar los datos de sensado con cada nodo cabecera de la red. Finalmente, es el nodo coordinador el encargado de obtener las medidas de SINR y BER representativas para cada nodo cabecera.

La función de aprendizaje es realizada por el nodo coordinador mediante el entrenamiento de redes neuronales que caracterizan el desempeño de error. Cada nodo cabecera requiere dos funciones de caracterización que modelan el desempeño en el ambiente de radio de influencia para las capas PHY ASK y BPSK, y de esta manera, el nodo coordinador debe fijar los parámetros de las funciones de caracterización en cada nodo cabecera, esto se ilustra en el diagrama de interacciones para la función de aprendizaje de la figura 5.18.

Por su parte, la función de razonamiento debe ser desplegada en cada nodo cabecera, y actúa cuando recibe tramas de datos enviadas por los nodos sensores como se muestra en el diagrama de interacciones de la figura 5.19. La red neuronal que modela el desempeño estima la BER para una SINR de una trama de datos recibida y la compara con una BER máxima.

Es de importancia resaltar que cada nodo cabecera seleccionará la mejor capa PHY disponible para los nodos sensores asociados. Así mismo, el nodo coordinador realiza las funciones de sensado y aprendizaje para las tramas de datos recibidas, bien sea por los nodos cabecera o por nodos sensores, modela el desempeño que el ambiente de radio le presenta y, ejecuta la función de razonamiento para inferir la calidad de la capa PHY utilizada, y de esta manera tomar la mejor decisión posible para garantizar un buen desempeño para toda la red. Nótese entonces, que en la función de razonamiento se presenta una interacción entre todos los nodos de la red.

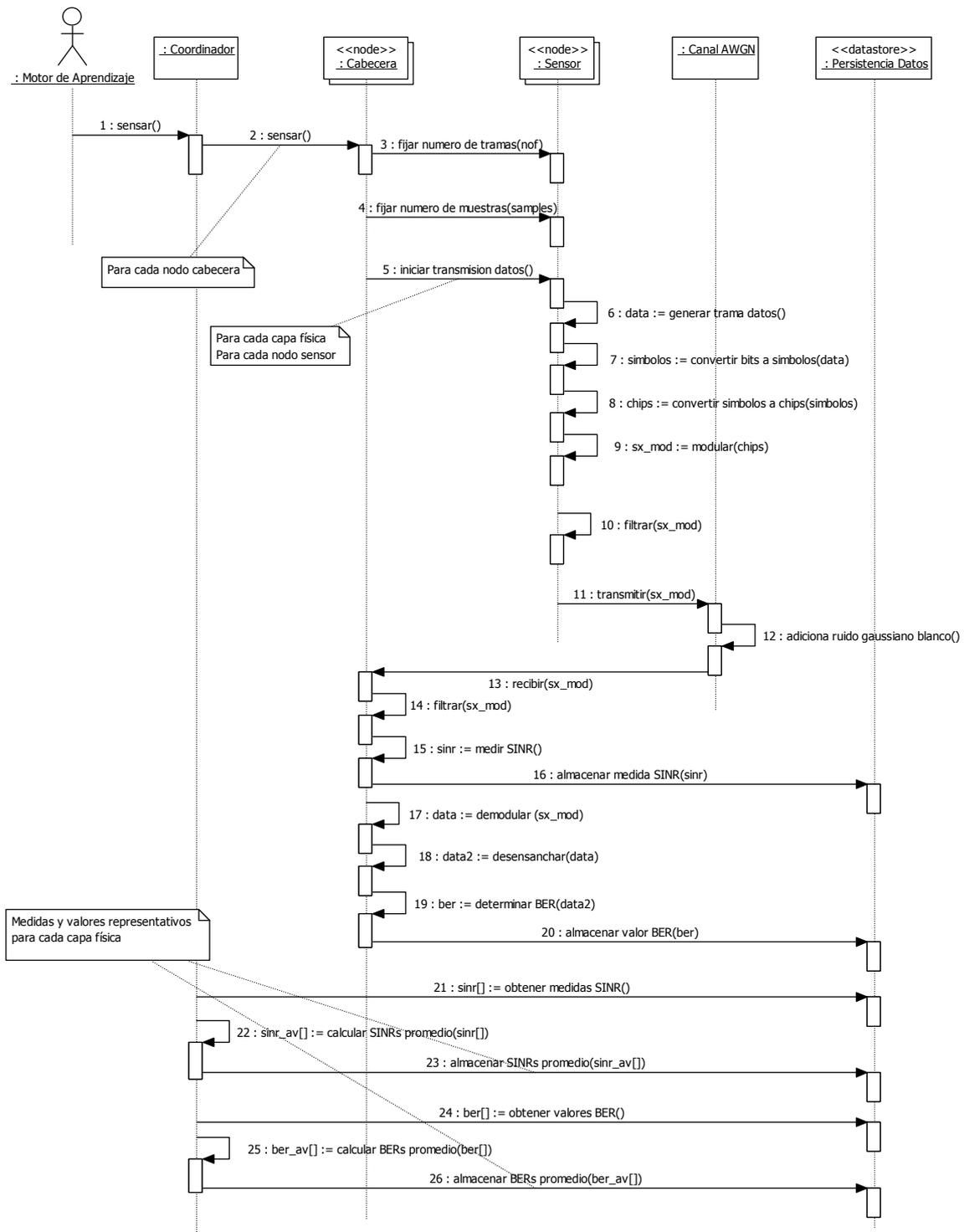


Figura 5.17. Interacciones función de sensado

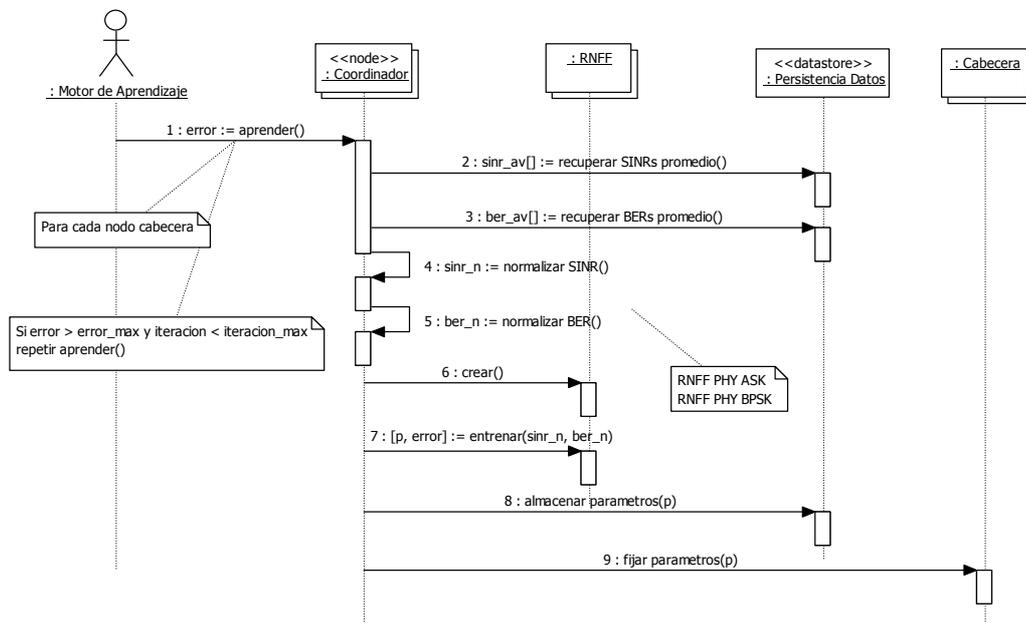


Figura 5.18. Interacciones función de aprendizaje

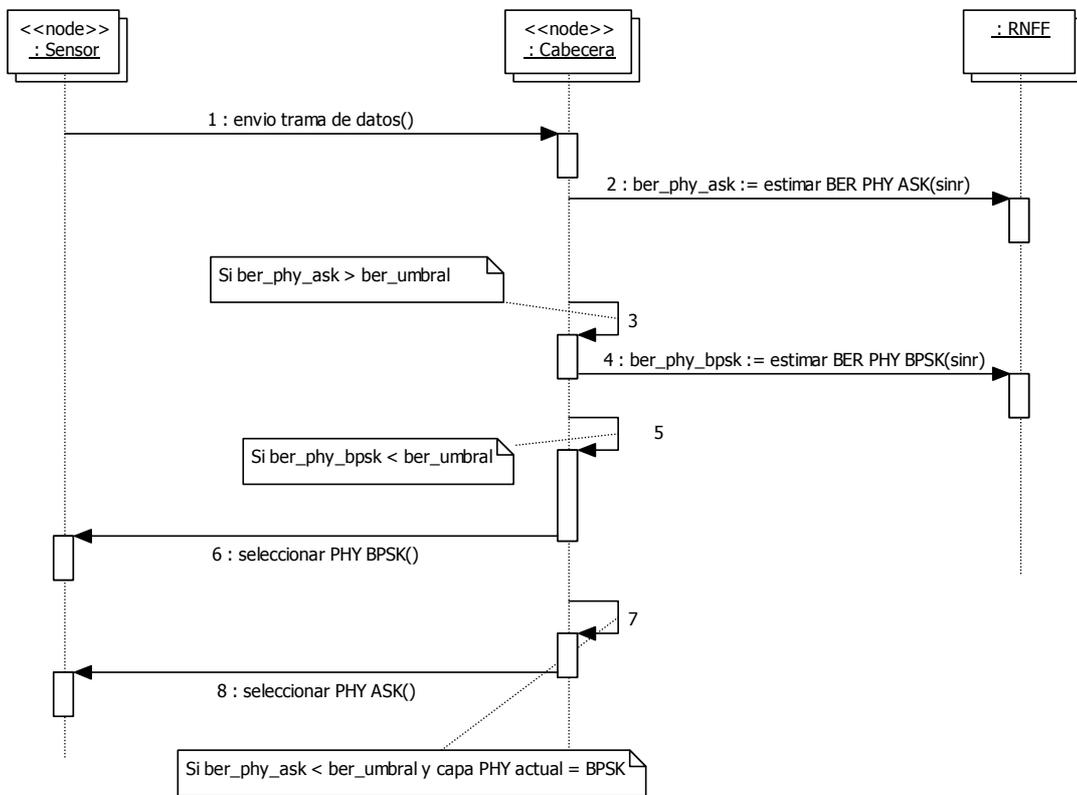


Figura 5.19. Interacciones función de razonamiento

5.3 Métricas de desempeño

Para la evaluación del desempeño del motor de aprendizaje se consideran las métricas: BER y criterio de error Chi-cuadrado.

- La BER define la calidad de la transmisión y recepción de datos sobre el canal de radio. Esta métrica es el insumo principal para la operación del motor de aprendizaje, ya que le permite al nodo controlador y/o nodos cabecera ser auto-conscientes de su estado de operación (desempeño).

Más aún, la función de control utiliza esta métrica para razonar acerca de las tramas de la SINR actual y determinar si el canal utilizado es bueno o malo, o bien si una capa física presenta un buen desempeño o no. Bajo los escenarios planteados, se espera que el desempeño de error se mantenga por debajo de la BER umbral aún en presencia de variaciones en el ambiente de RF, gracias a los cambios adaptativos inferidos, señalizados y realizados por los nodos de la red.

- El criterio de error Chi-cuadrado es una medida de la cualidad de ajuste de la función a los datos de entrenamiento en la función de aprendizaje, es decir, mide en qué grado el desempeño caracterizado corresponde con el desempeño real.

La figura 5.20 presenta la curva de desempeño obtenida al aplicar el método de aprendizaje Levenberg-Marquard en el entrenamiento de la red neuronal que caracteriza el desempeño de error en el escenario selección de canal. Puede notarse que la convergencia se alcanza en la iteración número ocho del método y el criterio de error Chi-cuadrado alcanza el valor de $2 * 10^{-6}$ el cual fue establecido como parámetro de desempeño.

La figura 5.21 presenta la curva de desempeño obtenida al aplicar el método de aprendizaje Levenberg-Marquard en el entrenamiento de la red neuronal que caracteriza el desempeño de error para la capa física PHY ASK. Lo propio hace la figura 5.22 para la capa física PHY BPSK.

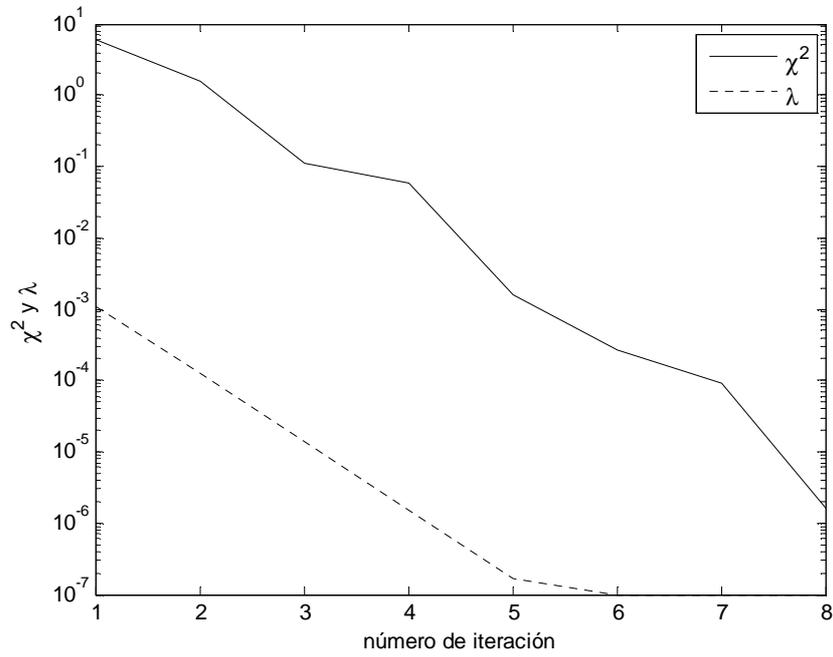


Figura 5.20. Desempeño algoritmo de aprendizaje en el escenario selección de canal

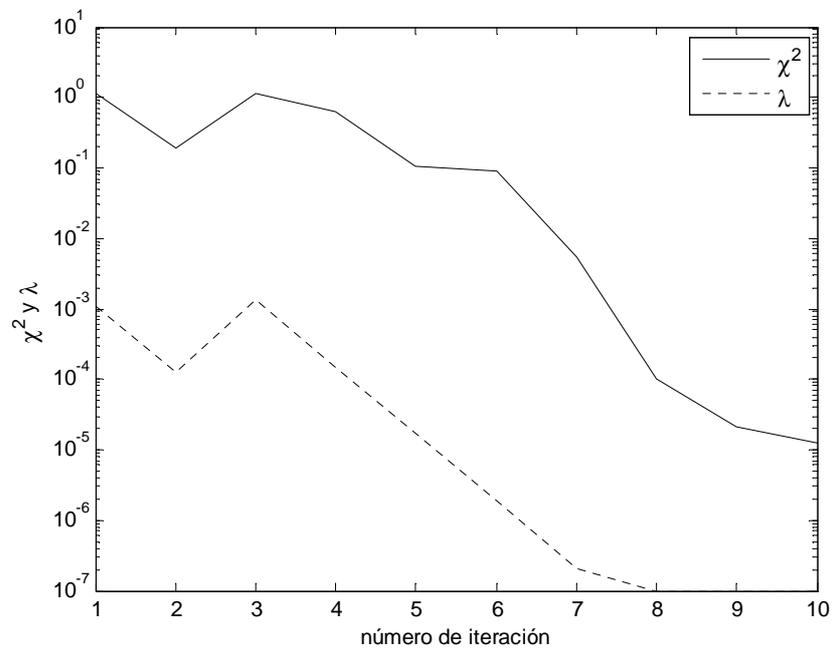


Figura 5.21. Desempeño algoritmo de aprendizaje en para la capa PHY ASK

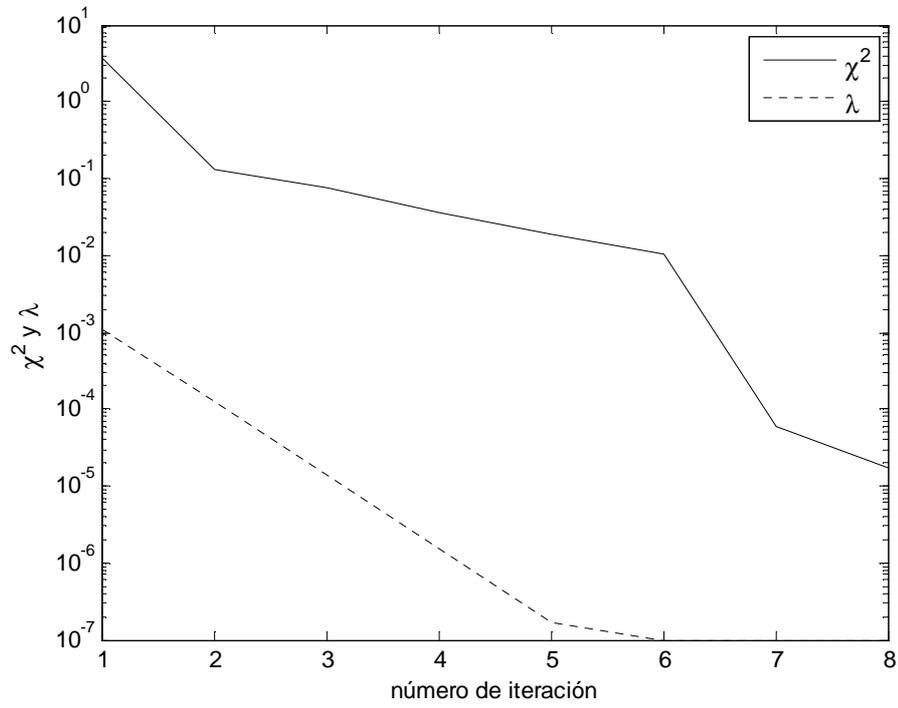


Figura 5.22. Desempeño algoritmo de aprendizaje en para la capa PHY BPSK

Conclusiones y recomendaciones

A continuación se presentan las principales conclusiones y recomendaciones generadas en el desarrollo de este trabajo, así como algunos trabajos futuros propuestos.

Conclusiones y recomendaciones:

En el modelo de redes de sensores inalámbricas cognitivas las funciones cognitivas deben ser soportadas en su mayoría por el nodo coordinador, con la restricción que éste sea un dispositivo con mayores recursos computacionales para soportar las nuevas capacidades de procesamiento, y posiblemente con suministro continuo de energía. Por otra parte, los nodos cabecera y los nodos sensores tratan de mantenerse tan simples como sea posible, ya que es importante el ahorro de energía, dado que los nodos por lo general funcionan a partir de baterías, las cuales no pueden ser reemplazadas con frecuencia en muchas de las situaciones.

Los escenarios de operación del motor de aprendizaje: selección de canal y selección de capa física, han sido planteados para una red en estrella. Estos escenarios pueden ser extendidos para una red nodal, con la salvedad que el sensado se hace de forma distribuida, en los nodos cabecera, mientras que las funciones de aprendizaje y control son realizadas por el nodo coordinador.

La habilidad para detectar la ocupación de canal y su selección dinámica es un mecanismo importante para la coexistencia de diferentes tecnologías inalámbricas

que operen en la banda ISM. Se ha considerado un mecanismo para la selección dinámica de canal, empleando una red neuronal artificial que caracteriza de forma eficiente la BER frente a la SINR de los 16 canales disponibles en IEEE 802.15.4 en la banda de 2400 MHz. Esta caracterización permite seleccionar el canal con menor BER disponible de acuerdo al ambiente de radio y de esta manera lograr un mejor desempeño en la transmisión de información.

Se ha considerado la función de aprendizaje desde el punto de vista de la caracterización del desempeño de la capa física, tanto para los canales como para las capas físicas disponibles, y se ha planteado el uso de redes neuronales multicapa con alimentación hacia delante para su implementación práctica. Esta técnica de inteligencia artificial puede ser simple, sin embargo en una red de sensores inalámbrica, donde existe una amplia restricción de recursos computacionales, constituye de por sí un importante esfuerzo de investigación.

Una red neuronal multicapa con alimentación hacia delante, un tipo de red ampliamente conocida y que presenta buenos resultados cuando se utiliza como aproximador de funciones como en este caso. Sin embargo, en la literatura se presenta una gran variedad de clases de redes neuronales y podría ser importante analizar su desempeño. Es más, investigaciones recientes en este campo, revelan nuevos métodos de aprendizaje, que pregonan ser más eficientes y con menor costo computacional. De esta manera, queda abierta una puerta importante para investigación, tanto en métodos de aprendizaje y optimización, como en la misma aplicación de redes neuronales para la solución de diversos problemas en el procesamiento de información y señales en sistemas de telecomunicaciones.

Una vez la función de aprendizaje realiza el entrenamiento de una red neuronal para caracterizar el desempeño, ésta tiene la propiedad de estimar de forma rápida la BER a partir de la SINR presente en el canal, y puede ser utilizada con frecuencia para predecir la calidad de transmisión en el canal. No obstante, la fase de entrenamiento requiere de un mayor número de recursos computacionales y por ende puede realizarse en periodos de tiempo más largos que la fase de predicción de tal manera que no se afecte el desempeño de una WSN.

Las nuevas funciones de reconfiguración, sensado y señalización adicionales para estos nodos implican un mayor procesamiento y por lo tanto mayor consumo de energía; este tema debe ser analizado en mayor detalle con el fin de garantizar implementaciones factibles del modelo propuesto.

Trabajos futuros:

El algoritmo de cálculo de una red neuronal con alimentación hacia delante requiere la implementación de operaciones lineales: suma de entradas ponderadas y umbrales, y operaciones no lineales: función de activación tansig. Así mismo, el método de entrenamiento Levenberg-Marquardt ejecuta de forma iterativa operaciones vectoriales y numéricas, cuyo costo computacional depende del número de ejemplos de entrenamiento y de neuronas empleadas. Como trabajo futuro es necesario realizar un análisis detallado del costo computacional de la implementación de estos algoritmos en una CWSN y de esta manera aproximarse a realizaciones prácticas del motor de aprendizaje propuesto.

El intercambio de mensajes, instrucciones y sincronismo entre los nodos de la red que permita coordinar las funciones de sensado, aprendizaje y control, requieren un esfuerzo considerable en el proceso de diseño e implementación de redes de sensores inalámbricas cognitivas. Como trabajo futuro se plantea el uso de un simulador de red, por ejemplo NS, para diseñar y validar un protocolo de red que posteriormente pueda ser considerado para implementaciones en equipos reales.

La medición del desempeño de error en implementaciones prácticas del motor de aprendizaje requiere métodos de estimación eficientes, donde la incorporación de bits adicionales permitan estimar la BER manteniendo las restricciones de las redes de sensores inalámbricas: bajo consumo de potencia y bajo costo. Esto constituye un trabajo futuro en el marco de una CWSN. Es más, es de interés analizar una aproximación mixta de estimación y corrección de errores para el motor de aprendizaje, donde se realice corrección de errores cuando la BER estimada presente un nivel relativamente bajo y sea más eficiente realizar la corrección de errores, o se decida realizar cambio de canal o de capa física cuando el nivel de error sea mayor y la corrección de error no implique mayor ventaja de procesamiento.

Como es planteado en el anexo A, la coexistencia de redes de sensores inalámbricas, tales como redes IEEE 802.15.4, con otras redes inalámbricas, surge del fenómeno de interferencia presente en el área de operación por el uso de canales o bandas de frecuencia compartidas. Para cuantificar el desempeño de coexistencia se utilizó el modelo propuesto en el estándar IEEE 802.15.4, un modelo simple que considera la potencia de transmisión, recepción y la distancia de separación entre fuente interferente y nodo sensor, que permite estimar la BER para determinados valores de estos parámetros de operación. Un campo importante de investigación futura es el análisis de desempeño de coexistencia que considere otros modelos de interferencia, que permitan ampliar la operación del motor de aprendizaje en acceso oportunista de espectro en una CWSN.

Es conveniente implementar un prototipo de una CWSN donde puedan implementarse los mecanismos de selección dinámica de canal y capa física propuestos. Como *framework* SDR puede emplearse hardware reconfigurable o implementaciones de fuente abierta como GNURadio. Los dos casos constituyen una alternativa viable para continuar con la investigación y desarrollo en redes de sensores inalámbricas cognitivas.

Anexo A

Análisis cuantitativo de las capas físicas de IEEE 802.15.4

El análisis cuantitativo de la capas físicas del estándar de comunicación para redes de área personal inalámbricas 802.15.4 se realiza con el fin de evaluar su desempeño, específicamente la tasa de error de bit (BER, *Bit Error Rate*) frente a la variación de la relación señal a ruido más interferencia (SINR, *Signal to Interference and Noise Ratio*) sobre un canal con ruido gaussiano blanco aditivo (AWGN, *Additive White Gaussian Noise*), dado que el efecto de la señal interferente sobre la señal deseada se asume similar a ruido AWGN en el mismo ancho de banda (IEEE, 2006).

Este análisis se realiza mediante simulación, utilizando la herramienta de cálculo Matlab. El modelo de simulación de las capas físicas considera las funciones de dispersión de espectro y modulación. En IEEE 802.15.4, como se mencionó en el capítulo 2, se dispone de tres bandas de frecuencia: 868, 915 y 2400 MHz. Para las dos primeras bandas se han especificado tres capas físicas definidas por los esquemas de dispersión y modulación: PSSS-ASK, DSSS-BPSK y DSSS-OQPSK, mientras que para la tercera banda, solo se dispone de la capa física DSSS-OQPSK. La figura A.1 presenta el diagrama de bloques de las funciones de la capa física DSSS-OQPSK implementado en Matlab que permitió obtener las curvas de desempeño de error. Este modelo considera la simulación de la capa PHY en banda base y ha sido adaptado del trabajo realizado por Gavini (2007), y se apoya en los fundamentos de simulación para sistemas de telecomunicaciones propuestos por Astaiza et al. (2007).

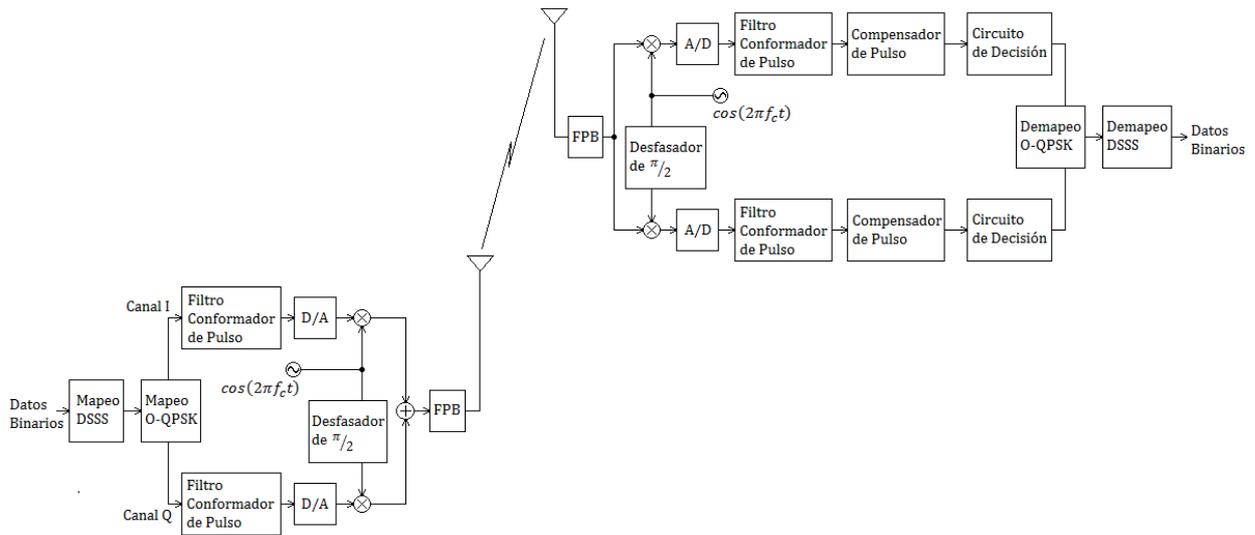


Figura A.1. Modelo de simulación IEEE 802.15.4 capa física DSSS-OQPSK

A.1 Capa física IEEE 802.15.4 en 2400 MHz

Cada byte de la unidad de datos de protocolo de la capa PHY (PPDU, *PHY Protocol Data Unit*) se procesa a través de las funciones de ensanchamiento y modulación, para esto los cuatro bits menos significativos (LSBs, *Less Significant Bits*) de cada byte se mapean en un símbolo de datos, y los cuatro bits más significativos (MSBs, *Less Significant Bits*) en el símbolo siguiente. Luego, cada símbolo se mapea en una secuencia de ruido pseudo-aleatorio (PN, *Pseudo-random Noise*) de 32 chips. Finalmente, las secuencias de chips se modulan sobre la portadora usando O-QPSK con pulsos conformados en media onda. Los chips de índice par se modulan sobre la portadora en fase (I, *In-phase*), y los chips de índice impar se modulan sobre la portadora de fase en cuadratura (Q, *Quadrature-phase*). Debido a que cada símbolo de datos se representa por una secuencia de 32 chips, la tasa de chips es de 2.0 Mchip/s. Para formar el offset entre la modulación de chip en fase I y fase Q, los chips de fase Q deben ser retardados en un tiempo T_c con respecto a los chips en fase I, donde T_c es el inverso de la tasa de chips.

Esta capa física permite mantener un buen desempeño de error en canales AWGN, principal factor que determina la calidad de transmisión, junto a una eficiencia espectral razonable. La codificación DSSS dispersa la señal en un ancho de banda más amplio y permite lograr ganancias de codificación y procesamiento, y la modulación OQPSK con pulsos conformados en media onda logra un alto grado de ocupación espectral mientras presentan muy poca fuga espectral fuera de la banda de la señal. El objetivo principal es que el sistema tenga una sensibilidad excelente, presente poca interferencia a otros usuarios en el espectro, e implementaciones de bajo consumo de potencia (Lanzisera & Pister, 2007).

El límite de sensibilidad fundamental de IEEE 802.15.4 en canales AWGN puede ser caracterizado considerando cálculos de margen de enlace que incluyan las características de señalización, ganancia de codificación y ganancia de procesamiento del estándar (Lanzisera & Pister, 2007). La sensibilidad de un receptor, definida como la potencia de señal mínima que garantiza un desempeño de error determinado, está dada por la ecuación (A.1)

$$P_{min} = kT \cdot B \cdot NF \cdot SNR_{min} \quad (A.1)$$

Donde k es la constante de Boltzmann, T es la temperatura absoluta, B es el ancho de banda de comunicación, NF es la figura de ruido del receptor, y SNR_{min} es la relación entre la potencia de la señal en banda base a la potencia de ruido en el demodulador mínima. SNR_{min} está dada por

$$SNR_{min} = \frac{(E_b/N_0)_{min}^{OQPSK}}{C \cdot P} \quad (A.2)$$

Donde $(E_b/N_0)_{min}^{OQPSK}$ es la relación de energía por bit a ruido mínima requerida para el esquema de modulación OQPSK, C es la ganancia de codificación, y P es la ganancia de procesamiento.

El valor SNR_{min} depende de la tasa de error tolerada, dada por una tasa de error de paquete máxima permisible (PER, *Packet Error Rate*) de 1% para un paquete de referencia de 22 bytes. Estos 22 bytes son codificados en 44 símbolos de $k = 4$ bits cada uno (IEEE, 2006). Un solo error de símbolo provoca un error de paquete por lo

cual la tasa de error de símbolo (SER, *Symbol Error Rate*) puede ser calculada a partir de

$$SER = \frac{PER}{\text{Símbolos por Paquete}} = \frac{1\%}{44} = 2.3 \times 10^{-4} \quad (\text{A.3})$$

La SER puede emplearse para estimar la BER promedio del sistema. Un error de símbolo no corresponde a un único error de bit en el caso típico debido a que un error de símbolo puede ser generado hasta por k errores de bit. En promedio, aproximadamente $k/2$ errores de bit ocurren por cada error de símbolo resultando en una BER requerida de 1.15×10^{-4} . La figura A.2 presenta los resultados de simulación del desempeño de error de la capa física en la banda de 2400 MHz. Puede notarse que para SINR pequeñas la BER toma valores grandes, por ejemplo para SINR igual a -10 dB, la BER alcanza el valor de 2×10^{-1} , mientras que para SINR mayores la BER disminuye, por ejemplo en el caso de SINR igual a 1 dB, la BER es de 10^{-5} . La PER máxima permitida se halla alrededor de 0 dB de SINR, donde la BER es aproximadamente 2×10^{-4} .

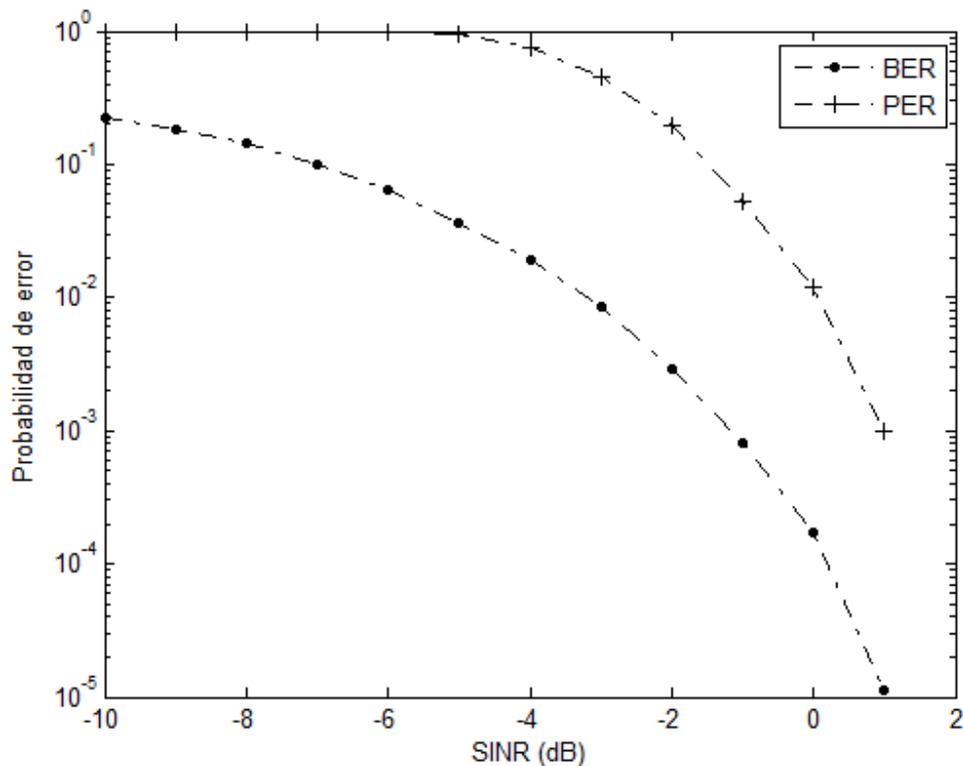


Figura A.2. Desempeño capa física O-QPSK IEEE 802.15.4 en la banda de 2400 MHz

A.2 Capa PHY BPSK en 868/915 MHz

La tasa de datos de esta capa PHY debe ser de 20 kbps cuando opera en la banda 868 MHz y 40 kbps cuando opera en la banda de 915 MHz, la modulación de chip se realiza en DSSS con BPSK y para la codificación de símbolos de datos se utiliza codificación diferencial, como lo representa el diagrama de bloques funcional de referencia de la figura A.3. La codificación diferencial es la suma módulo-2 (or exclusiva) de un bit de datos sin codificar con el bit previo codificado, cada bit de entrada se mapea en una de dos secuencias PN de 15 chips, las secuencias de chip se modulan en la portadora usando BPSK con conformación de pulso de coseno realzado (con factor de *roll-off* igual a 1). La curva de desempeño para la capa PHY BPSK en 915 MHz se presenta en la figura A.5.



Figura A.3. Modelo funcional de referencia capa PHY BPSK

A.3 Capa PHY ASK en 868/915 MHz

Las tasas de datos de la capa PHY ASK deben ser de 250 kb/s en las dos bandas. Si se usa esta capa el dispositivo deberá ser capaz de operar usando la capa PHY BPSK 868/915 también. La capa PHY ASK emplea una técnica de modulación multicódigo llamada espectro ensanchado de secuencia paralela (PSSS), también conocida como multiplexación por división de código ortogonal (OCDM, *Orthogonal Code Division Multiplexing*).

Durante cada periodo de símbolo de datos, 20 bits de información para 868 MHz y 5 bits de información para 915 MHz son modulados separadamente en 20 y 5

secuencias PN casi ortogonales, respectivamente. Estas secuencias se suman linealmente para crear un símbolo de 32 chips multinivel para 868 MHz y un símbolo de 32 chips multinivel para 915 MHz. Luego cada símbolo se precodifica, y finalmente se modulan en una portadora usando ASK. El diagrama de bloques funcional de referencia de la figura A.4 especifica las funciones de modulación de dispersión. La curva de desempeño para la capa PHY ASK en 915 MHz se presenta en la figura A.5.

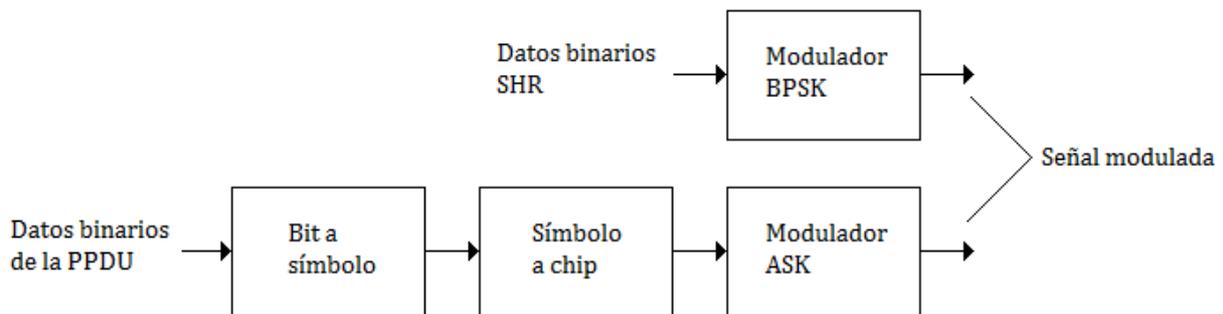


Figura A.4. Modelo funcional de referencia capa PHY ASK

A.4 Capa PHY O-QPSK en 868/915 MHz

La tasa de datos de la capa PHY O-QPSK debe ser de 100 kbps cuando opera en la banda 868 MHz y 250 kbps cuando opera en la banda 915 MHz, esta capa no es obligatoria, sin embargo cuando se use, el mismo dispositivo debe ser capaz de señalizar usando la capa PHY BPSK 868/915 MHz también.

La capa PHY O-QPSK emplea una técnica de modulación casi-ortogonal de 16 estados, su funcionamiento es similar al de la capa PHY O-QPSK de 2450 MHz, con la diferencia que las secuencias de PN utilizadas en la función de dispersión tienen una longitud de 16 chips. La curva de desempeño para la capa PHY O-QPSK en 915 MHz se presenta en la figura A.5.

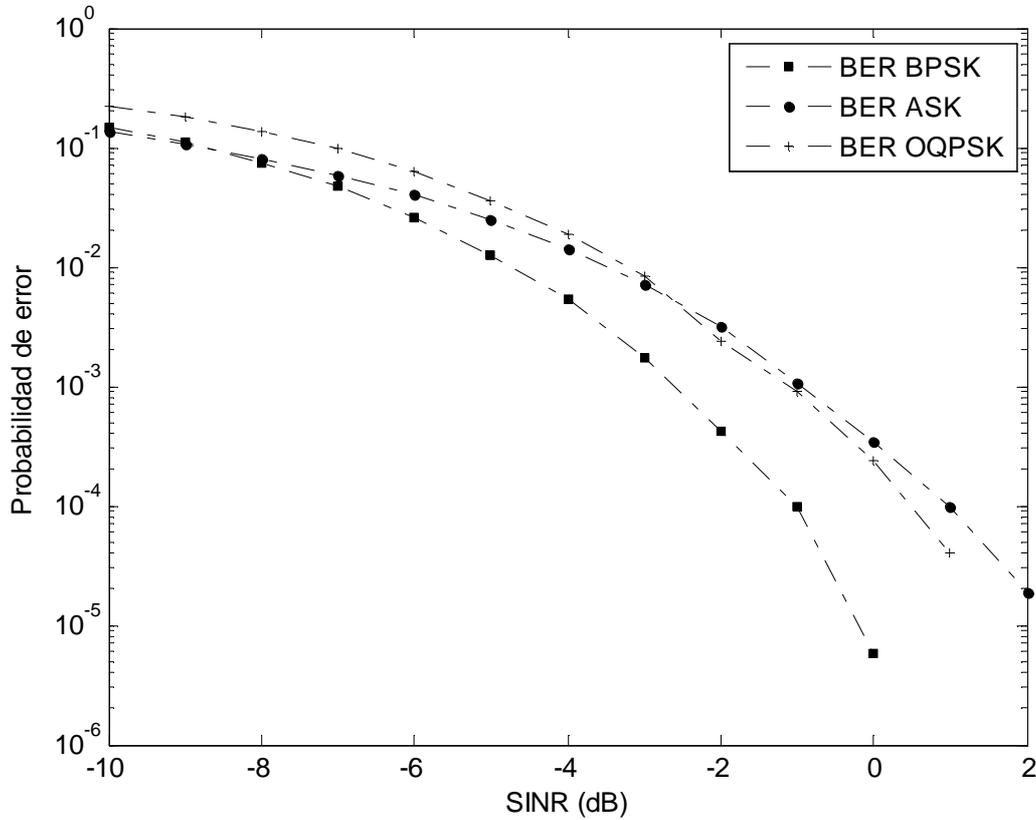


Figura A.5. Desempeño de error capas PHY IEEE 802.15.4 en la banda de 915 MHz

De la figura A.5 puede notarse que la capa PHY BPSK tiene el mejor desempeño de error de las tres capas PHY, donde incluso para valores negativos de SINR (en dB) se presenta tasas de error menores al valor umbral ($2 * 10^{-4}$); por el contrario, las capas PHY ASK y O-QPSK requieren que la SINR sea mayor que cero para lograr este desempeño deseado. Sin embargo, la tolerancia a ruido e interferencia de la capa PHY es contrarrestada por la menor tasa de transferencia de datos que presenta.

A.5 Análisis de coexistencia

Se espera que los dispositivos IEEE 802.15.4 “coexistan”, es decir, operen en proximidad a otros dispositivos inalámbricos. Esta condición es importante debido a que en la banda de 2400 MHz operan otras tecnologías, por ejemplo: IEEE 802.11b, 802.15.1, 802.15.3 y 802.16, lo cual puede degradar el desempeño de una red IEEE 802.15.4, requiriéndose la selección dinámica de canal para minimizar su efecto. Por su parte, en la banda de 915 MHz se dispone de tres capas físicas que pueden interactuar entre si y puede ser necesaria la selección dinámica de capa física para mantener los objetivos de desempeño de la red.

En esta sección se realiza el análisis de coexistencia para dispositivos IEEE 802.15.4 operando en la banda de 2400 MHz bajo la presencia de un dispositivo 802.11b como fuente de interferencia, y se estima la BER para diferentes valores de distancia de separación entre dicha fuente y un nodo receptor 802.15.4. Este análisis se basa en los resultados de coexistencia presentados en (IEEE, 2006).

A.5.1 Desempeño de coexistencia en la banda de 2400 MHz

El modelo de canal está definido por la ecuación A.1, se basa en el estándar IEEE 802.11 y ha sido adaptado por el estándar IEEE 802.15.2 (2003).

$$d = \begin{cases} 10^{\frac{(P_t - P_r - 40.2)}{20}}, & d < 8 \text{ m} \\ 8 * 10^{\frac{(P_t - P_r - 58.5)}{33}}, & d > 8 \text{ m} \end{cases} \quad (\text{A.4})$$

Donde P_t es la potencia de transmisión en dBm , P_r es la potencia de recepción en dBm y d es la distancia en mts . La sensibilidad para un nodo 802.15.4 en la banda de 2400 MHz es de -85 dBm , la potencia de transmisión para la fuente interferente 802.11b es de 14 dBm , el ancho de banda para 802.15.4 es de 2 MHz, la máscara espectral de potencia en 802.11b se define en la tabla A.1. Estos datos permiten determinar el nivel de SIR resultante y se utiliza para estimar la BER a partir de la ecuación A.5. Empleando Matlab se realizó la simulación de este escenario de coexistencia obteniéndose los resultados que se grafican en la figura A.6.

$$BER = \frac{8}{15} \times \frac{1}{16} \times \sum_{k=2}^{16} -1^k \binom{16}{k} e^{(20 \times SINR (\frac{1}{k} - 1))} \tag{A.5}$$

Tabla A.1. Máscara espectral de potencia en 802.11b

Frecuencia	Límite relativo
$f_c - 22 \text{ MHz} < f < f_c - 11 \text{ MHz}$ y $f_c + 11 \text{ MHz} < f < f_c + 22 \text{ MHz}$ y	-30 dBr
$f < f_c - 22 \text{ MHz}$ y $f > f_c + 22 \text{ MHz}$	-50 dBr

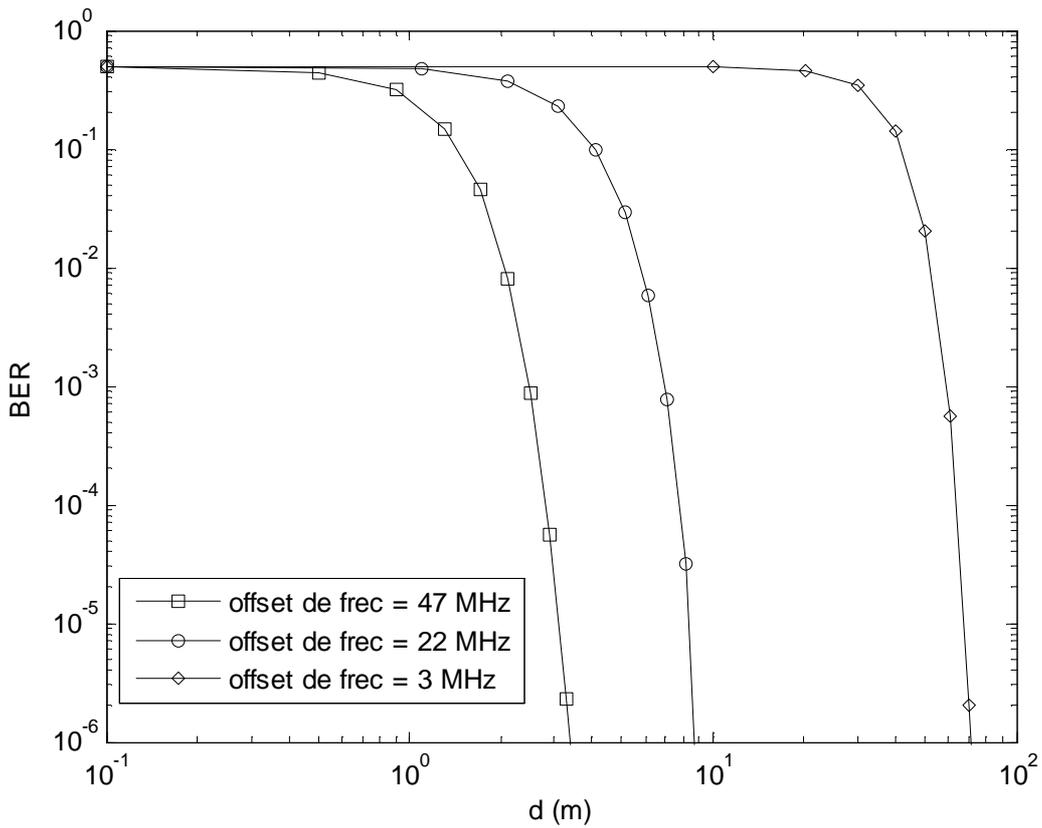


Figura A.6. Resultados de simulación desempeño de coexistencia

Los resultados de simulación de desempeño de coexistencia para una fuente interferente 802.11b, muestran que si la frecuencia de operación de la fuente interferente está alejada 3 MHz de la frecuencia central del canal actual de 802.15.4, la distancia de separación debe ser mayor a 60 m para que el nivel de interferencia generado no degrade el desempeño de la red; si la separación de frecuencias es de 22 MHz la distancia mínima es de 8 m; y por su parte, si es de 47 MHz se puede tolerar una distancia de separación cercana a 3 m.

Anexo B

Simulador de red NS-2 para el análisis de desempeño de IEEE 802.15.4

Este anexo describe el uso del simulador de red NS-2 para el análisis de desempeño de IEEE 802.15.4, un estándar diseñado para redes de área personal inalámbrica de baja transferencia de datos, bajo consumo de operación y bajo costo. Zheng y Lee (2006) desarrollaron un simulador NS-2 para IEEE 802.15.4 y muestran varios experimentos para estudiar sus características, incluyendo: 1) modo de baliza habilitado y modo sin baliza; 2) asociación, formación de árbol y autoconfiguración de red; aislamiento y relocalización del coordinador; 4) acceso múltiple por detección de portadora con evasión colisión (CSMA-CA); y 5) transmisiones de datos en intervalos de tiempo garantizados directos e indirectos.

NS (Versión-2) es un simulador de eventos discretos, orientado a objetos, desarrollado bajo el proyecto VINT como un esfuerzo conjunto entre la UC Berkeley, USC/ISI, LBL y Xerox PARK. Está escrito en C++ con Otcl como *front-end*. El simulador soporta una jerarquía de clases en C++ (jerarquía compilada), y una jerarquía de clases similar en el interprete Otcl (jerarquía interpretada). Las dos jerarquías están relacionadas una de la otra; desde la perspectiva del usuario, hay una correspondencia uno a uno entre una clase en la jerarquía interpretada y una en la jerarquía compilada.

Este simulador NS-2 para 802.15.4 ha sido utilizado en la Universidad de Tecnología Dresden (Marandin, 2008) para la simulación de 802.15.4/ZigBee. Marandin (2008)

ha desarrollado una estructura de proceso de simulación con base en el simulador 802.15.4 de Zheng (2006), la cual se representa en el diagrama de la figura B.1.

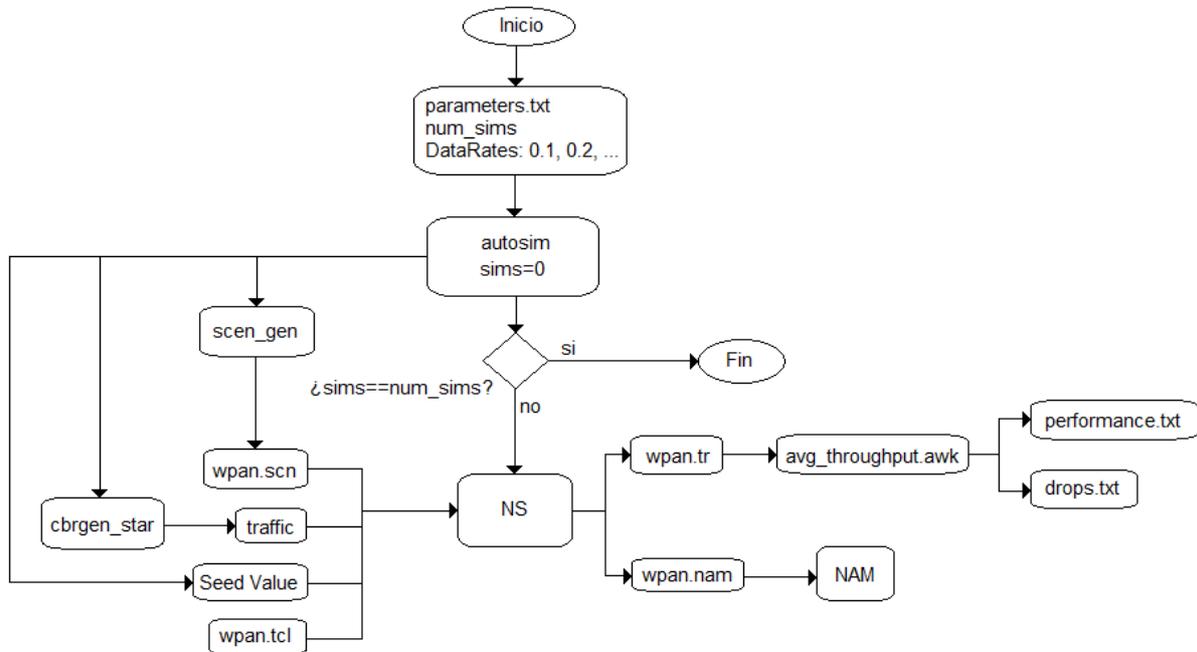


Figura B.1. Proceso de simulación IEEE 802.15.4/ZigBee

El programa autosim es una aplicación que automatiza el proceso de simulación. Si un reporte necesita producir resultados concluyentes, ellos deben ser generados en forma aleatoria tomando la media de varios resultados; wpan.tcl es un archivo que contiene parámetros y variables los cuales ayudan a cambiar el escenario de la red y configurar y controlar el proceso de simulación; scen_gen es una utilidad que genera automáticamente las coordenadas de los nodos para ubicarlos alrededor del coordinador, a lo largo de una circunferencia imaginaria; wpan.scn es el archivo de posición de los nodos, mantiene la posición de los nodos alrededor del coordinador; cbrgen_star es un archivo script tcl, capaz de generar un escenario de tráfico aleatorio entre los nodos; y avg_throughput.awk genera las métricas de desempeño.

Las métricas de desempeño que pueden evaluarse en el simulador NS-2 para IEEE 802.15.4/ZigBee son:

- *Throughput* de red: medida de la cantidad de datos transmitidos de la fuente al destino en un periodo de tiempo unitario (segundo).
- *Throughput* de un nodo = Bits de datos total recibido/Tiempo de ejecución de la simulación.
- *Throughput* de la red = Sumatoria del *throughput* de los nodos involucrados en la transmisión de datos/Número de nodos.
- Retardo extremo a extremo: es el tiempo que le toma a un paquete de datos alcanzar el nodo de destino.
- Retardo de paquete = Tiempo de recepción en el destino – Tiempo de transmisión en la fuente.
- Retardo promedio = Sumatoria de todos los retardos de paquetes/Número total de paquetes recibidos.
- Tasa de entrega: Porcentaje de los paquetes de datos transmitidos que son exitosamente recibidos.
- Tasa de entrega = Número de paquetes recibidos/Número de paquetes recibidos * 100.
- Consumo de energía: Porcentaje de energía consumida por un nodo con respecto a su energía inicial.
- Porcentaje de energía consumida = (Energía inicial – Energía final)/Energía inicial*100.
- Energía promedio consumida = Sumatoria del porcentaje de energía consumida por todos los nodos/Número de nodos

A manera de ejemplo, la tabla B.1 presenta la evaluación de la métrica de desempeño Tasa de Entrega obtenida a partir de la implementación del proceso de

simulación de la figura B.1 en NS-2, para una red IEEE 802.15.4 en estrella conformada por diez nodos: un nodo coordinador y nueve nodos sensores.

Tabla B.1. Desempeño de la tasa de entrega de paquetes en una red IEEE 802.15.4

Carga de Tráfico (Paquetes/segundo)	Paquetes Transmitidos (Paquetes)	Paquetes Recibidos (Paquetes)	Tasa de Entrega (%)
0.1	739	730	98.782138
1.0	7390	7390	100.000000
2.0	14747	14733	99.905065
3.0	22126	22083	99.805659
4.0	29513	29286	99.230847
5.0	36896	36638	99.300737
6.0	44248	43614	98.567167
7.0	51668	48652	94.162731
8.0	59142	47210	79.824828
9.0	66401	36520	54.999172
10.0	73735	32812	44.499898

Anexo C

El método Levenberg-Marquardt para problemas de aproximación de curva utilizando mínimos cuadrados no lineales

El método Levenberg-Marquardt es una técnica estándar usada para solucionar problemas de mínimos cuadrados no lineales. Los problemas de mínimos cuadrados surgen cuando se ajusta una función parametrizada a un conjunto de puntos de datos medidos mediante la minimización de la suma de los cuadrados de los errores entre los puntos de datos y la función. Los problemas de mínimos cuadrados no lineales surgen cuando la función es no lineal en los parámetros. Los métodos de mínimos cuadrados no lineales requieren ir mejorando los valores de los parámetros a medida que se vaya reduciendo la suma de los cuadrados de los errores entre la función y los puntos de datos medidos.

El método de ajuste de curva Levenber-Marquardt es realmente una combinación de dos métodos: el método de gradiente descendente y el método Gauss-Newton. En el método de gradiente descendente, la suma de los cuadrados de los errores es reducida mediante la actualización de los parámetros en la dirección de la reducción más grande del objetivo de mínimos cuadrados. En el método Gauss-Newton, la suma de los errores cuadrados es reducida asumiendo que la función de mínimos cuadrados es cuadrática localmente, y hallando el mínimo de la función cuadrática. El método Levenberg-Marquardt actúa como un método de gradiente descendente cuando los parámetros están lejos de su valor óptimo, y como el método Gauss-Newton cuando los parámetros están cerca de su valor óptimo. En este anexo se describen estos métodos, de acuerdo al planteamiento presentado por Gavin (2011).

Cuando se ajusta una función $\hat{y}(t; \mathbf{p})$ de una variable independiente t y un vector de n parámetros \mathbf{p} a un conjunto de puntos de datos (t_i, y_i) , es usual y conveniente minimizar la suma de los cuadrados ponderados de los errores (residuos ponderados) entre los datos medidos $y(t_i)$ y la función de ajuste de curva $\hat{y}(t_i; \mathbf{p})$. Esta medida de la calidad de ajuste de valor escalar es llamada el criterio de error Chi-cuadrado y está dado por las ecuaciones C.1 a C.3

$$\chi^2(\mathbf{p}) = \frac{1}{2} \sum_{i=1}^m \left[\frac{y(t_i) - \hat{y}(t_i; \mathbf{p})}{w_i} \right]^2 \quad (\text{C.1})$$

$$= \frac{1}{2} (\mathbf{y} - \hat{\mathbf{y}}(\mathbf{p}))^T \mathbf{W} (\mathbf{y} - \hat{\mathbf{y}}(\mathbf{p})) \quad (\text{C.2})$$

$$= \frac{1}{2} \mathbf{y}^T \mathbf{W} \mathbf{y} - \mathbf{y}^T \mathbf{W} \hat{\mathbf{y}} + \frac{1}{2} \hat{\mathbf{y}}^T \mathbf{W} \hat{\mathbf{y}} \quad (\text{C.3})$$

El valor w_i es una medida del error en la medida $y(t_i)$. La matriz de pesos \mathbf{W} es diagonal con $W_{ii} = 1/w_i^2$. Si la función \hat{y} es no lineal en los parámetros del modelo \mathbf{p} , la minimización de χ^2 con respecto a los parámetros debe ser realizada de forma iterativa. La meta de cada iteración es hallar una perturbación \mathbf{h} a los parámetros \mathbf{p} que reduzcan χ^2 .

C.1 El método de gradiente descendente

El método de descenso gradual es un método de minimización general el cual actualiza los valores de los parámetros en la dirección opuesta al gradiente de la función objetivo. Es un algoritmo altamente convergente para hallar el mínimo de funciones objetivo simples. El gradiente de la función objetivo chi-cuadrado con respecto a los parámetros es

$$\frac{\partial}{\partial \mathbf{p}} \chi^2 = (\mathbf{y} - \hat{\mathbf{y}}(\mathbf{p}))^T \mathbf{W} \frac{\partial}{\partial \mathbf{p}} (\mathbf{y} - \hat{\mathbf{y}}(\mathbf{p})) \quad (\text{C.4})$$

$$= -(\mathbf{y} - \hat{\mathbf{y}}(\mathbf{p}))^T \mathbf{W} \left[\frac{\partial}{\partial \mathbf{p}} \hat{\mathbf{y}}(\mathbf{p}) \right] \quad (\text{C.5})$$

$$= -(\mathbf{y} - \hat{\mathbf{y}})^T \mathbf{W} \mathbf{J} \quad (\text{C.6})$$

Donde la matriz Jacobiana $m \times n$ $\left[\frac{\partial}{\partial \mathbf{p}} \hat{\mathbf{y}} \right]$ representa la sensibilidad local de la función $\hat{\mathbf{y}}$ frente a la variación en los parámetros \mathbf{p} . La perturbación \mathbf{h} que mueve los parámetros en la dirección del descenso gradual está dada por

$$\mathbf{h}_{gd} = \alpha \mathbf{J}^T \mathbf{W} (\mathbf{y} - \hat{\mathbf{y}}), \quad (\text{C.7})$$

Donde el escalar positivo α determina la longitud del paso en la dirección de descenso gradual.

C.2 El método Gauss-Newton

El método Gauss-Newton es un método para minimizar una función objetivo de suma de cuadrados. Presume que la función objetivo es aproximadamente cuadrática en los parámetros cerca a la solución óptima. Para la mayoría de problemas de tamaño moderado, el método Gauss-Newton típicamente converge más rápido que los métodos de gradiente descendente.

La función evaluada con parámetros de modelo perturbados puede ser localmente aproximada a través de una expansión en series de Taylor de primer orden

$$\hat{\mathbf{y}}(\mathbf{p} + \mathbf{h}) \approx \hat{\mathbf{y}}(\mathbf{p}) + \left[\frac{\partial \hat{\mathbf{y}}}{\partial \mathbf{p}} \right] \mathbf{h} = \hat{\mathbf{y}} + \mathbf{J} \mathbf{h} \quad (\text{C.8})$$

Sustituyendo la aproximación para la función perturbada, $\hat{\mathbf{y}} + \mathbf{J} \mathbf{h}$, por $\hat{\mathbf{y}}$ en la ecuación (C.3),

$$\chi^2(\mathbf{p} + \mathbf{h}) \approx \frac{1}{2} \mathbf{y}^T \mathbf{W} \mathbf{y} + \frac{1}{2} \hat{\mathbf{y}}^T \mathbf{W} \hat{\mathbf{y}} - \frac{1}{2} \mathbf{y}^T \mathbf{W} \hat{\mathbf{y}} - (\mathbf{y} - \hat{\mathbf{y}})^T \mathbf{W} \mathbf{J} \mathbf{h} + \frac{1}{2} \mathbf{h}^T \mathbf{J}^T \mathbf{W} \mathbf{J} \mathbf{h} \quad (\text{C.9})$$

Esto muestra que χ^2 es aproximadamente cuadrática en la perturbación \mathbf{h} , y que la matriz Hessiana del criterio de ajuste Chi-cuadrado es aproximadamente $\mathbf{J}^T \mathbf{W} \mathbf{J}$.

La perturbación \mathbf{h} que minimiza χ^2 se halla al hacer $\frac{\partial \chi^2}{\partial \mathbf{h}} = 0$.

$$\frac{\partial}{\partial \mathbf{h}} \chi^2(\mathbf{p} + \mathbf{h}) \approx -(\mathbf{y} - \hat{\mathbf{y}})^T \mathbf{W} \mathbf{J} + \mathbf{h}^T \mathbf{J}^T \mathbf{W} \mathbf{J}, \quad (\text{C.10})$$

Y las ecuaciones normales resultantes para la perturbación Gauss-Newton son

$$[\mathbf{J}^T \mathbf{W} \mathbf{J}] \mathbf{h}_{gn} = \mathbf{J}^T \mathbf{W} (\mathbf{y} - \hat{\mathbf{y}}) \quad (\text{C.11})$$

C.3 El método Levenberg-Marquardt

El algoritmo Levenberg-Marquardt adaptativamente varía las actualizaciones de los parámetros entre la actualización de gradiente descendente y de Gauss-Newton,

$$[\mathbf{J}^T \mathbf{W} \mathbf{J} + \lambda \mathbf{I}] \mathbf{h}_{lm} = \mathbf{J}^T \mathbf{W} (\mathbf{y} - \hat{\mathbf{y}}), \quad (\text{C.12})$$

Donde valores pequeños de λ resultan en una actualización Gauss-Newton y valores grandes de λ resultan en una actualización de gradiente descendente. En una distancia grande del mínimo de la función, el método de gradiente descendente provee estabilidad y progreso convergente hacia la solución. A medida que la solución se aproxime al mínimo, λ se decrementa adaptativamente, el método Levenberg-Marquardt se aproxima al método Gauss-Newton, y la solución típicamente converge rápidamente al mínimo local.

La relación de actualización C.13 sugerida por Marquardt, se utiliza en el algoritmo implementado en Matlab del método Levenberg-Marquardt por Gavin (2011).

$$[\mathbf{J}^T \mathbf{W} \mathbf{J} + \lambda \text{diag}(\mathbf{J}^T \mathbf{W} \mathbf{J})] \mathbf{h}_{lm} = \mathbf{J}^T \mathbf{W} (\mathbf{y} - \hat{\mathbf{y}}) \quad (\text{C.13})$$

C.4 Implementación numérica

El Jacobiano ($\mathbf{J} \in \mathbb{R}^{m \times n}$) se aproxima numéricamente usando la ecuación de diferencias

$$J_{ij} = \frac{\partial \hat{y}_i}{\partial p_j} = \frac{\hat{y}(t_i; \mathbf{p} + \delta \mathbf{p}_j) - \hat{y}(t_i; \mathbf{p})}{\|\delta \mathbf{p}_j\|} \quad (\text{C.14})$$

Donde el elemento j -ésimo de $\delta \mathbf{p}_j$ es el único elemento diferente de cero y se fija a $\epsilon_2(1 + |p_j|)$. Si en una iteración $\chi^2(\mathbf{p}) - \chi^2(\mathbf{p} + \mathbf{h}) > \epsilon_3 \mathbf{h}^T (\lambda \mathbf{h} + \mathbf{J}^T \mathbf{W}(\mathbf{y} - \hat{\mathbf{y}}))$ entonces $\mathbf{p} + \mathbf{h}$ es suficiente mejor que \mathbf{p} , \mathbf{p} es reemplazado por $\mathbf{p} + \mathbf{h}$, y λ es reducido por un factor de diez. De otra manera λ se incrementa por un factor de diez, y el algoritmo procede a la siguiente iteración. La convergencia se logra si $\max(|h_i/p_i|) < \epsilon_2$, $\chi^2/m < \epsilon_3$, o $\max(|\mathbf{J}^T \mathbf{W}(\mathbf{y} - \hat{\mathbf{y}})|) < \epsilon_1$. De otra manera, las iteraciones terminan cuando el número de iteraciones excede un límite preestablecido.

C.5 Análisis de error

Una vez los parámetros de ajuste de curva óptimos \mathbf{p}_{fit} son determinados, se calculan parámetros estadísticos para la solución convergida usando valores de pesos, w_i^2 , igual al error de medida cuadrado promedio, σ_y^2 ,

$$w_i^2 = \sigma_y^2 = \frac{1}{n-m+1} (\mathbf{y} - \hat{\mathbf{y}}(\mathbf{p}_{fit}))^T (\mathbf{y} - \hat{\mathbf{y}}(\mathbf{p}_{fit})) \quad \forall i. \quad (\text{C.15})$$

La matriz de covarianza de los parámetros se calcula a partir de

$$\mathbf{V}_p = [\mathbf{J}^T \mathbf{W} \mathbf{J}]^{-1}, \quad (\text{C.16})$$

Los errores de los parámetros estándar asintóticos están dados por

$$\sigma_p = \sqrt{\text{diag}(\mathbf{J}^T \mathbf{W} \mathbf{J})^{-1}}, \quad (\text{C.17})$$

El error estándar asintótico es una medida de cómo la variabilidad inexplicable en los datos se propagan a la variabilidad en la solución, y es esencialmente una medida de error para los parámetros. El error estándar del ajuste está dado por

$$\sigma_{\hat{y}} = \sqrt{\text{diag}(\mathbf{J}[\mathbf{J}^T \mathbf{W} \mathbf{J}]^{-1} \mathbf{J}^T)} \quad (\text{C.18})$$

El error estándar del ajuste indica cómo la variabilidad en los parámetros afecta la variabilidad en la curva de ajuste. El error de predicción estándar asintótico refleja el error estándar del ajuste así como el error de medida cuadrado promedio.

$$\sigma_{\hat{y}_p} = \sqrt{\sigma_y^2 + \text{diag}(\mathbf{J}[\mathbf{J}^T \mathbf{W} \mathbf{J}]^{-1} \mathbf{J}^T)} \quad (\text{C.19})$$

Bibliografía

Akyildiz, I. F. et al. (2002). Wireless sensor networks: a survey, *Computer Networks* 38. Elsevier, pp. 393-422.

Akyildiz, I. F. et al. (2006). NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey. *Computer Networks Journal (Elsevier)* 50, 2127–2159.

Akyildiz, I. F. et al. (2009). CRAHNs: Cognitive radio ad hoc networks, *Ad Hoc Netw.*

Astaiza, E., Bermúdez, H., Muñoz, P. (2007). *Simulación de Sistemas de Telecomunicaciones*. ISBN: 978-958-44-2583-6. Armenia, Colombia.

Baldo, N. & Zorzi, M. (2007). Fuzzy Logic for Cross-layer Optimization in Cognitive Radio Networks. *IEEE Consumer Communications and Networking Conference (CCNC'07)*.

Baldo, N. & Zorzi, M. (2008). Learning and Adaptation in Cognitive Radios Using Neural Networks. *5th IEEE Consumer Communications and Networking Conference (CCNC 2008)*.

Berenji, H. (1992). *Fuzzy Logic Controllers. An Introduction to Fuzzy Logic Applications in Intelligent Systems*. Kluwer Academic Publishers. pág. 69-96.

- Briasco, M, et al. (2006), Cognitive antennas architecture for distributed sensor networks. Department of Biophysical and Electronic Engineering – University of Genova, Italy.
- Chapin, J. & Doyle, L. (2007). A Path Forwards for Cognitive Radio Research. 2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communication (CrownCom 2007).
- Chen, B, et al. (2010). Efficient Error Estimating Coding: Feasibility and Applications. SIGCOMM'10.
- Engelbrecht, A. (2007). Computational Intelligence, An Introduction. Second Edition. Wiley.
- Gavin, H. (2011). The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems. Duke University.
- Gavini, C. (2007). Quantifying Tradeoffs in the IEEE 802.15.4 protocol through simulation. EE464 Senior Design Project.
- Haghighat, A. (2002). A Review on Essentials and Technical Challenges of Software Defined Radio. IEEE. British Crown Copyright.
- Halpin, T. & Morgan, T. (2008). Information Modeling and Relational Databases (2 ed.). Morgan Kaufmann.
- Haykin, S. (1999). Neural Networks. A Comprehensive Foundation (2 ed.). Pearson Education.
- Haykin, S. (2005), Cognitive radio: brain-empowered wireless communications. Selected Areas in Communications, IEEE Journal on 23(2): 201-220.
- Haykin, S. (2007a). Cognitive radio and radio networks. INFWEST seminar in Helsinki.

Haykin, S. (2007b). Fundamental issues in cognitive radio. In: B. Bhargava and E. Hussain (ed.) Cognitive radio networks, Springer-Verlag.

Hebb D. (1949). The Organization of Behavior: A neuropsychological theory. New York: Wiley.

Hopfield, J. (1982). "Neural networks and physical systems with emergent collective computational abilities". En Proceedings of the National Academy of Sciences of the United States of America, tomo 79, pág. 2554-2558.

IEEE (2003). IEEE 802.15.3, Part 15.2: Coexistence of Wireless Personal Area Networks with Other Wireless Devices Operating in Unlicensed Frequency Bands. IEEE Computer Society.

IEEE (2006). IEEE 802.15.4, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal area Networks (LR-WPANs). IEEE Computer Society.

Lanzisera, S. & Pister, K. (2007). Theoretical and Practical Limits to Sensitivity in IEEE 802.15.4 Receivers. International Conference on Electronics, Circuits and Systems.

Levenberg, K. (1944). A Method for the Solution of Certain Problems in Least Squares. Quart. Appl. Math. 2, pág. 164-168.

Li, L. et al. (2010). An Energy-Efficient Error Correction Scheme for IEEE 802.15.4 Wireless Sensor Networks. IEEE Transactions on circuits and systems-II: Express Briefs, vol. 57, No. 3.

Madsen, K., Nielsen, H. & Tingleff, O. (2004). Methods for non-linear least squares problems. 2nd Edition. IMM.

Marandin, D. (2008). Simulation of IEEE 802.15.4/ZigBee with Network Simulator-2 (ns-2). Publicación Web. [En Línea]. Disponible: <http://www.ifn.et.tu-dresden.de/marandin/ZigBee/ZigBeeSimulation.html>. [Consultado: Enero, 2011]

Marquardt, D. (1963). "An Algorithm for Least Squares Estimation on Nonlinear Parameters". *SIAM J. APPL. MATH.* 11, pág. 431-441.

Matinmiko M., et al. (2008), Cognitive radio: An intelligent wireless communication system. Research Report VTT-R-02219-08. Centro de Investigación Técnica VTT de Finlandia.

McCulloch, W. & Pitts, W. (1943). "A logical calculus of the ideas immanent in nervous activity". *Bulletin of Mathematical Biophysics* 5, pág. 115-133.

Minsky, M. & Pappert, S. (1969). "Perceptrons: An introduction to computational geometry". The MIT Press.

Mitola III, J. & Maguire, G. (1999a). Cognitive Radio: Making Software Radios More Personal. *IEEE PCS Magazine's Special Issue on Software Radio.*

Mitola III, J. (1999b). Cognitive Radio for Flexible Mobile Multimedia Communications. Best Paper of the 6th International Workshop on Mobile Multimedia Communications.

Mottola, L. & Picco, G (2006). Programming Wireless Sensor Networks: Fundamental Concepts and State of the Art.

OMG (2005). Unified Modeling Language: Superstructure version 2.0. Object Management Group.

Perez, B. (2010). Algoritmos eficientes, incrementales y escalables para el aprendizaje en redes de neuronas artificiales. Tesis doctoral. Universidade da Coruña.

Price, R. & Green, P. (1958). A communication technique for multipath channels. *Proceedings of the IRE.* Volume 46. pp. 555–570.

Rosenblatt, F. (1957). Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Washington DC: Spartan.

Rumelhart, D., Hinton, G. & Williams, R. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. The MIT Press, Cambridge, MA.

SDR Forum (2008). *Cognitive Radio Definitions and Nomenclature*, Approved Document SDRF-06-P-0009-V1.0.0.

Shen, B. & Abedi, A. (2007). A Simple Error Correction Scheme for Performance Improvement of IEEE 802.15.4. ICWN.

Smit, L., Smit, G. & Hurink, J. (2003). BER Estimation for wireless links using BPSK/QPSK modulation. Progress.

Widrow, B. (1959). Adaptive sampled-data systems – a statistical theory of adaptation. En IRE WESCON Convention Record, tomo 4, pág. 74-85.

Widrow, B. & Hoff, M. (1960). Adaptive Switching Circuits. En IRE WESCON Convention Record, tomo 4, pág. 96-104.

Zadeh, L. (1975). Fuzzy Logic and Approximate Reasoning. *Synthese* **30**. Pág. 407-428. D. Reidel Publishing.

Zheng, J. and Lee, M. (2006). A comprehensive performance study of IEEE 802.15.4. *Sensor Network Operations*, IEEE Press, Wiley Interscience, Chapter 4, pp. 218-237.