



UNIVERSIDAD DE DEUSTO

**VENTANA DE BITS: PROCEDIMIENTO
PARA REDUCIR LA LATENCIA Y EL
CONSUMO ENERGÉTICO DE
PROTOCOLOS DE ANTI-COLISIÓN
RFID BASADOS EN QUERIES Y TAGS
PASIVOS**

Tesis doctoral presentada por Hugo Landaluce
dentro del Programa de Doctorado en Ing. Informática y Telecomunicación
Dirigida por Dr. Asier Perallos



UNIVERSIDAD DE DEUSTO

**VENTANA DE BITS: PROCEDIMIENTO
PARA REDUCIR LA LATENCIA Y EL
CONSUMO ENERGÉTICO DE
PROTOCOLOS DE ANTI-COLISIÓN
RFID BASADOS EN QUERIES Y TAGS
PASIVOS**

Tesis doctoral presentada por Hugo Landaluce
dentro del Programa de Doctorado en Ing. Informática y Telecomunicación

Dirigida por Dr. Asier Perallos

El doctorando

El director

Bilbao, Febrero de 2014

Ventana de bits: procedimiento para reducir la latencia y el consumo energético de protocolos de anti-colisión RFID basados en queries y tags pasivos

Autor: Hugo Landaluce

Director: Asier Perallos

Texto impreso en Bilbao

Primera edición, Febrero de 2014

*A mis padres, Marisa y Joserra,
por ayudarme a ser quien soy.*

Abstract

Radio Frequency Identification (RFID) is used to identify stored codes into tags using radio frequency. These tags are attached to different objects to produce benefits in applications such as: traceability and localization of those objects or access control. RFID, also, fits very well in the paradigm of *Ubiquitous computing* or the *Internet of things*.

The ability to establish the communication between several readers and tags presents RFID as a very flexible technology. The participants in the communication share the same channel (the air), which can cause garbled waveforms if more than one device transmit at the same time. These interferences are one of the main problems in RFID technology and it is called *the tag collision problem*. Some of the drawbacks this causes are an increase in the identification time and in the energy consumption of the system.

The problem mentioned above can be mitigated using restricted versions of multi-access network protocols that are applied to RFID, called anti-collision protocols. Different kinds of anti-collision strategies have been proposed and an improvement opportunity has been discovered in query based protocols related to the number of bits transmitted by the tags. A novel procedure has been proposed to be applied to these kind of protocols in order to decrease the time to identify all the tags in the antenna range and the energy consumption, considering passive tags. Furthermore, this procedure has been applied to two different query based protocols, and three anti-collision protocols have been produced: the QwT, the CwT and the CwT+.

With the main purpose of validating the hypothesis proposed, a test workbench has been developed so that all the proposed designs could

be tested and compared to other existing protocols in the literature. In this manner, the procedure designed shows the improvements produced on the different query based protocols in the time to identify the tags and the energy consumed in the operation.

Resumen

La tecnología de identificación por radio frecuencia (RFID) se emplea para detectar el código almacenado en los tags empleando ondas de radio frecuencia. Asociando un tag a cada objeto, se le aporta una nueva funcionalidad que provee grandes beneficios en aplicaciones como trazabilidad y localización de objetos, control de acceso y además, se adapta muy bien al paradigma de computación ubicua o *Internet de las cosas*.

RFID presenta una gran flexibilidad, permitiendo múltiples tags y lectores. Dado que el canal de transmisión de esta tecnología es el aire, cuando más de un tag transmiten al mismo tiempo, sus ondas de radio frecuencia interfieren, causando una modificación de los datos recibidos por el lector, y conformando el llamado *problema de las colisiones de tag*. Éste es el principal causante del incremento de tiempo en la identificación de los tags y en el consumo energético del sistema.

Para mitigar este problema, se han planteado los protocolos de anti-colisión, una versión restringida de los protocolos de multi-acceso de la teoría de redes aplicados a RFID. De todas las versiones existentes en la literatura, se ha buscado un protocolo de baja complejidad que provea tags de bajo costo y se ha encontrado una oportunidad de mejora relacionada con el número de bits transmitidos por los tags, en los protocolos basados en *queries*. Para ello, se ha propuesto el diseño de un procedimiento que, aplicado a este tipo de protocolos, consiga disminuir el tiempo de identificación del conjunto de tags y el consumo empleado en el proceso, en caso de que los tags sean pasivos. Así, aplicando el procedimiento a dos protocolos de anti-colisión existentes se han diseñado tres nuevos protocolos, el QwT, el CwT y el CwT+.

Con el objetivo de validar la hipótesis formulada, se ha desarrollado un banco de pruebas donde poder evaluar las características de los diseños aquí presentados y compararlos con algunos de los protocolos más recientes presentes en la literatura. De esta forma, se ha podido afirmar que el procedimiento diseñado, aplicable a protocolos basados en *queries*, produce mejoras en el tiempo de identificación y el consumo energético.

Agradecimientos

Antes de comenzar a profundizar en la presente tesis doctoral, me gustaría agradecer a todas aquellas personas que me han ayudado a conseguir esta importante meta en mi vida. Durante todo el tiempo que han durado mis estudios doctorales he crecido como investigador y como persona, y he conocido nuevos amigos y contactos; ha sido una gran experiencia.

Comenzaré agradeciendo a *Asier Perallos*, la persona que ha hecho posible todo esto. Quiero agradecerle su ayuda en todo momento, siempre ha estado ahí disponible para ayudarme a saltar esos obstáculos del camino a pesar de sus otras ocupaciones. Su experiencia y templanza ha sido muy importante para abordar aspectos confusos, que siempre ha tratado con opiniones objetivas y sumamente útiles aportando una nueva dimensión al problema. Gracias por iluminarme el camino y por hacerme ver que era posible, aún cuando todo era negro.

A *Lars Bengtsson* y *Johan Karlsson*, por darme la posibilidad de realizar la estancia en su departamento de la Universidad Chalmers y por sus consejos y útiles opiniones acerca del trabajo aquí presentado. Me gustaría agradecer igualmente a *Magnus Önnheim* y a *Emil Gustavsson* por su inestimable ayuda con las matemáticas y también a *Rubén Titos* y a *Kasyab Subramaniyan*, por hacer que mi estancia en Göteborg fuera tan agradable y dejarme recuerdos imborrables.

Me gustaría agradecer, también, a *Aitzol Zuloaga* por su útil orientación en el master y los inicios de la tesis, y a *Ignacio Angulo* por ofrecerme su inestimable ayuda y apoyo con el 'cacharreo'. Como no, a la gente que vive mi día a día, a todo el laboratorio de Mobility, porque es un placer estar entre vosotros. Al fantástico *Team Science*, por estar

rodeado de tanto conocimiento. Me gustaría destacar a *Eneko Osaba*, por sus impías correcciones, a *Idoia de la Iglesia*, por su coherencia y a *Asier Moreno* por sus tranquilos pero muy acertados apuntes. Respirad ahora, que dentro de poquito os tocará pasar por este trance. También a *Ignacio García Zuazola*, por darme su visión de la investigación y ser tan duro conmigo en nuestras publicaciones, lo que se ha convertido en éxitos, y a *Enrique Onieva* por su experta orientación y sus correcciones.

Por otro lado, agradecer a mis padres, que me ayudaron a tomar la difícil decisión de empezar este largo camino que es la tesis doctoral, por proporcionarme mis valores y ser un ejemplo admirable. A mi hermana por ser como es, un soplo de aire fresco. Y al resto de mi familia, porque sin todos vosotros no sería el mismo hoy en día. Quiero agradecer a mis amigos que también han influido en mí, y también, a los *Quaoar* por traer a mi vida algo más que ingeniería e investigación.

Y por supuesto a *Sandra*, por aguantarme día a día, por vivir la tesis conmigo y darme su apoyo y ánimos cuando los he necesitado. Por su paciencia, su cariño y por ser la persona que es, que hace que me acueste y me levante con una sonrisa.

¡Muchas gracias a tod@s!

Hugo Landaluce

Febrero de 2014

Índice general

Índice de figuras	xiii
Índice de tablas	xxi
1 Introducción y motivación	1
1.1 Tecnologías de identificación automática (auto-ID)	2
1.2 Tecnología RFID	6
1.2.1 Contexto histórico	6
1.2.2 Características de RFID	7
1.3 El problema de las colisiones de tag	11
1.4 Trabajo de tesis	14
1.4.1 Hipótesis	15
1.4.2 Objetivos	16
1.4.2.1 Objetivos específicos	16
1.5 Metodología de la Investigación	17
1.6 Estructura de la tesis	19
2 Estado del arte	21
2.1 Protocolos de multi-acceso: Capa física	23
2.2 Probabilísticos o basados en Aloha	27
2.2.1 Aloha (puro) (PA)	27
2.2.2 Slotted Aloha (SA)	28
2.2.3 Framed Slotted Aloha (FSA)	29
2.2.4 Dynamic framed slotted-Aloha (DFSA)	30

ÍNDICE GENERAL

2.2.4.1	El algoritmo Q (SC)	32
2.2.4.2	Enhanced Framed Slotted Aloha (EDFSA)	34
2.2.5	Comparación de los métodos probabilísticos	35
2.3	Deterministas o basados en Árbol Binario	37
2.3.1	Tree Splitting (TS)	38
2.3.2	Query Tree (QT)	40
2.3.3	Protocolos Bit-tracking	45
2.3.3.1	Binary Search (BS)	49
2.3.4	Bitwise arbitration (BTA)	50
2.3.5	Comparación de los protocolos basados en árbol	53
2.4	Híbridos	55
2.4.1	Protocolos Aloha en estructuras de árbol	55
2.4.1.1	Tree Slotted Aloha (TSA)	55
2.4.1.2	FSA en estructura de árbol QT	56
2.4.2	Protocolos de árbol en estructuras de Aloha	57
2.4.2.1	TS en estructuras FSA	58
2.4.2.2	QT en estructura FSA	58
2.4.3	Comparación de los algoritmos híbridos	59
2.5	Estándares de anti-colisión RFID	60
2.6	Conclusiones	62
3	Colisiones y oportunidades de mejora	65
3.1	Protocolos basados en queries	66
3.2	Métodos para la reducción de bits en los mensajes	67
3.2.1	Aviso de colisión	68
3.2.2	Corte de portadora	69
3.2.3	Short-long queries	70
3.2.4	RN16	71
3.2.5	Comparación de métodos para la reducción de bits en protocolos basados en QT	72
3.3	Oportunidades de mejora	73
3.3.1	Análisis del protocolo QT	73
3.4	Conclusiones	75

4	Metodología de la ventana	77
4.1	Aclaraciones previas	78
4.2	Ventana de bits	80
4.2.1	Ventajas e inconvenientes de la ventana	84
4.2.2	Aplicación de la ventana a protocolos basados en <i>queries</i>	85
4.3	QT con ventana constante	87
4.3.1	Análisis de la ventana aplicada al protocolo QT	93
4.3.1.1	Cálculo del número de slots del protocolo QT con ventana de bits	98
4.3.2	Simulaciones para diferentes valores de W	101
4.3.3	Conclusiones	106
4.4	Query window Tree (QwT)	107
4.4.1	Funciones de ventana	113
4.4.2	Comparativa entre ventanas	119
4.4.3	Conclusiones	121
4.5	CT con ventana Constante	122
4.5.1	Bit-tracking y ventana	123
4.5.2	Funcionamiento del protocolo	125
4.5.3	Análisis de la ventana aplicada al protocolo CT	128
4.5.3.1	Breve análisis del protocolo CT	128
4.5.3.2	Análisis del protocolo CT con la ventana de bits	129
4.5.4	Simulaciones para diferentes valores de W	131
4.5.5	Conclusiones	135
4.6	Collision window Tree (CwT)	136
4.6.1	Funciones de ventana	138
4.6.2	Comparativa entre ventanas	144
4.6.3	Collision window Tree plus (CwT+)	146
4.6.3.1	Funciones de ventana	152
4.6.3.2	Comparativa de las funciones	153
4.6.4	Conclusiones	156

ÍNDICE GENERAL

5	Experimentación y resultados	159
5.1	Estrategia de validación de la hipótesis	160
5.2	Banco de pruebas	162
5.2.1	Presentación del escenario	163
5.2.2	Medidas del protocolo	165
5.3	Simulaciones y resultados de QwT	170
5.3.1	Influencia del CRC en la ventana	178
5.3.2	Comparativa del QwT con ventana cuadrática	179
5.3.3	Conclusiones	185
5.4	Simulaciones y resultados de CwT	186
5.4.1	Influencia de la relación del <i>data rate</i> entre lector y tags	194
5.4.2	Comparativa del CwT con ventana lineal	197
5.4.3	Comparativas del protocolo CwT+	204
5.4.4	Conclusiones	207
6	Conclusiones	209
6.1	Visión general del trabajo	210
6.1.1	Contribuciones principales	213
6.2	Conclusiones de la experimentación	218
6.3	Limitaciones identificadas	221
6.4	Líneas futuras de trabajo	223
6.5	Consideraciones finales	225
A	Conclusions	227
A.1	General overview of the thesis	228
A.1.1	Main contributions	230
A.2	Conclusions of the experimentation	235
A.3	Identified limitations	238
A.4	Future work	239
A.5	Concluding remarks	240
	Bibliografía	245

Índice de figuras

1.1	Sistema RFID	8
1.2	Sistema RFID con acoplamiento inductivo.	10
1.3	Sistema RFID con acoplamiento electromagnético	11
1.4	Ejemplo de lectura de 3 tags, donde se produce una colisión y una identificación	15
1.5	Metodología de investigación empleada	19
2.1	Problema de las colisiones de tag	22
2.2	SDMA adaptativo con antena controlada direccionalmente	23
2.3	FDMA, canales de frecuencia adaptativos	24
2.4	Ejemplo de ejecución CDMA	24
2.5	Ejemplo de ejecución TDMA	25
2.6	Taxonomía de los protocolos de anti-colisión	26
2.7	Ejemplo de Aloha puro (PA)	28
2.8	Ejemplo de slotted Aloha (SA)	29
2.9	Ejemplo de framed slotted Aloha (FSA)	30
2.10	Ejemplo de dynamic framed slotted Aloha (DFSA)	31
2.11	Ajuste del parámetro Q	33
2.12	Ejemplo del algoritmo TS con $B = 2$	38
2.13	Ejemplo de identificación de 4 tags empleando el protocolo QT	42
2.14	Ejemplo de la codificación Manchester	46
2.15	Ejemplo de identificación de 4 tags empleando el protocolo CT	47
2.16	Ejemplo de detección de 4 tags mediante el protocolo BS	49
2.17	Ejemplo del protocolo ID-BTS	51

ÍNDICE DE FIGURAS

2.18	Ejemplo de ejecución del algoritmo TSA	56
2.19	Ejemplo de identificación del MS	57
2.20	Ejemplo de identificación del FQT	59
3.1	Proceso de carga y descarga del condensador de un tag	70
3.2	Probabilidades del protocolo QT. (a) $n = 400$; (b) $n = 1000$. . .	74
4.1	Comparativa entre el protocolo QT y el SIBT: número medio de bits transmitidos por tag con $k = 128$	81
4.2	Comparativa entre el protocolo QT y el SIBT: (a) número de slots empleados en detectar el conjunto de tags, (b) número total de bits (lector + tags) empleados en el proceso	81
4.3	Ejemplo de funcionamiento de un CRC-5	83
4.4	Comando del lector y respuestas de los tags en un protocolo basado en <i>queries</i>	87
4.5	Estructura del ID del tag	88
4.6	Esquema de funcionamiento del protocolo: (a) procedimiento del lector y (b) procedimiento de cada tag	89
4.7	Ejemplo de ejecución del protocolo QT con ventana, $W = 2$. . .	90
4.8	Identificación del mismo ejemplo de la Figura 4.7 con el protocolo QT	92
4.9	Árbol formado por el protocolo QT con ventana $W = 2$	94
4.10	Probabilidades del protocolo QwT para un conjunto de tags $n = 1000$: (a) probabilidades para diferentes valores de W ; (b) probabilidad de slot go-On variando L y W	98
4.11	Posibilidades de visita del nodo v	99
4.12	Resultados ante diferentes valores de W aplicados sobre el protocolo QT, empleando la eq. 4.19	100
4.13	Influencia del tamaño de la ventana en los bits transmitidos por cada tag del protocolo QT	101
4.14	Influencia del tamaño de la ventana sobre el protocolo QT: (a) en el número de slots (t), (b) en la eficiencia de slots	102
4.15	Slots go-On generados por la utilización de la ventana	103

ÍNDICE DE FIGURAS

4.16	Influencia del tamaño de la ventana sobre el protocolo QT: (a) en el número de colisiones, (b) en el de slots libres	104
4.17	Influencia del tamaño de la ventana sobre el protocolo QT: (a) en el número total de bits transmitidos por el lector y los tags, (b) en el número de bits transmitidos por el lector	105
4.18	Esquema de aplicación de la función en el protocolo QwT	109
4.19	Diagrama de flujo del protocolo QwT: (a) lector y (b) tag	112
4.20	Representación de $f(L)$ lineal	114
4.21	Variación del comportamiento del protocolo QwT con función Lineal, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits	115
4.22	Variación del comportamiento del protocolo QwT con ventana lineal para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n	115
4.23	Representación de $f(L)$ cuadrática	116
4.24	Variación del protocolo QwT con función cuadrática, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits	116
4.25	Variación del protocolo QwT con ventana cuadrática para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n	117
4.26	Representación de $f(L)$ exponencial	118
4.27	Variación del protocolo QwT con función exponencial, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits	118
4.28	Variación del protocolo QwT con ventana cuadrática para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n	119
4.29	Consumos del QwT con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de go-On slots, (b) número medio de bits transmitidos por cada tag.	120

ÍNDICE DE FIGURAS

4.30	Consumos del QwT con diferentes tipos de ventana identificando diferentes grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.	121
4.31	Interrogación de 3 tags empleando el protocolo CT con ventana, $W = 5$	124
4.32	Esquema de funcionamiento del protocolo CT con ventana constante: (a) procedimiento del lector y (b) de cada tag	125
4.33	Ejemplo de ejecución del protocolo CT con ventana, $W = 2$	126
4.34	Resolución del ejemplo mostrado en la figura 4.33 mediante el protocolo CT.	127
4.35	Resultados ante diferentes valores de W aplicados sobre el protocolo CT, empleando la eq. 4.19	130
4.36	Influencia del tamaño de la ventana en los bits transmitidos por cada tag del protocolo CT, con $k = 96$ bits	131
4.37	Influencia del tamaño de la ventana sobre el protocolo CT: (a) en el número de slots τ , (b) en la eficiencia de slots η	132
4.38	Slots go-On generados por la utilización de la ventana	133
4.39	Influencia del tamaño de la ventana sobre el protocolo CT: (a) en el número de colisiones, (b) en el de slots libres	134
4.40	Influencia del tamaño de la ventana sobre el protocolo CT: (a) en el número total de bits transmitidos por el lector y los tags, (b) en el número de bits transmitidos por el lector	135
4.41	Diagrama de flujo del protocolo CwT: (a) lector y (b) tag	137
4.42	Variación del comportamiento del protocolo CwT con función Lineal, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits	139
4.43	Variación del comportamiento del protocolo CwT con ventana lineal para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n	140
4.44	Variación del protocolo CwT con función cuadrática, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits	141

4.45	Variación del protocolo CwT con ventana cuadrática para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n	142
4.46	Variación del protocolo CwT con función exponencial, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits	143
4.47	Variación del protocolo CwT con ventana exponencial para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n	143
4.48	Consumos del protocolo CwT con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de go-On slots, (b) número medio de bits transmitidos por cada tag, siendo $k = 128$ bits	144
4.49	Consumos del protocolo CwT con diferentes tipos de ventana identificando diferentes grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.	145
4.50	Ejemplo de ejecución del protocolo CwT+	147
4.51	Ejemplo de ejecución del protocolo CwT+ en caso de colisión	148
4.52	Diagrama de flujo del protocolo CwT+: (a) lector y (b) tag	150
4.53	Mínimos valores de slots de tipo go-On variando β en: (a) función Lineal, (b) función Cuadrática y (c) función Exponencial	153
4.54	Valores de los bits transmitidos por los tags alrededor de los valores de β seleccionados en: (a) función Lineal, (b) función Cuadrática y (c) función Exponencial sabiendo que $k = 128$ bits	154
4.55	Consumos del protocolo CwT+ con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de go-On slots, (b) número medio de bits transmitidos por cada tag.	155
4.56	Consumos del protocolo CwT+ con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.	156
4.57	Consumos del protocolo CwT+ con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.	156

ÍNDICE DE FIGURAS

5.1	Esquema del banco de pruebas empleado	162
5.2	Escenario de las simulaciones	164
5.3	Promedio de bits transmitidos por cada tag de los protocolos QT y QwT empleando los tres tipos de ventanas propuestas y dos tamaños de CRC.	172
5.4	Comparativa de QwT con QT en: (a) bits transmitidos por el lector, (b) bits totales transmitidos entre lector y tags	173
5.5	Latencias de los protocolos QwT y QT	174
5.6	Energía consumida por los protocolos QwT y QT considerando tags pasivos	175
5.7	Energía consumida por los protocolos QwT y QT considerando tags activos	176
5.8	Influencia de la variación de la longitud del CRC sobre el protocolo QwT comparado con el QT en: (a) el número de bits transmitidos y (b) latencia	178
5.9	Ejemplo de error empleando un CRC de 1 bit, o bit de paridad	179
5.10	Número promedio de bits transmitidos por cada tag en un ciclo completo de interrogación del protocolo QwT y diversos protocolos de la literatura existente	180
5.11	Comparativa del protocolo QwT con diversos protocolos de la literatura existente en: (a) eficiencia de bits de los tags y (b) slots totales necesarios para identificar la totalidad del conjunto de tags	181
5.12	Comparativa del protocolo QwT y diversos protocolos de la literatura existente en: (a) los bits transmitidos por su correspondiente lector, y (b) los bits totales transmitidos entre lector y tags	182
5.13	Comparativa de las latencias medias del protocolo QwT y diversos protocolos de la literatura existente	184
5.14	Número de slots libres del protocolo QwT comparado con diversos protocolos de la literatura existente	185
5.15	Energía consumida por el protocolo QwT comparado con diversos protocolos de la literatura existente empleando tags pasivos	186

ÍNDICE DE FIGURAS

5.16 Promedio de bits transmitidos por cada tag de los protocolos CT y CwT empleando los tres tipos de ventanas propuestas. En la figura (b) se muestra el rango de bits [190-215] ampliado	187
5.17 Comparativa de CwT con CT en: (a) bits transmitidos por el lector, (b) bits totales transmitidos entre lector y tags	189
5.18 Latencias de los protocolos CwT y CT	190
5.19 Energía consumida por los protocolos CwT y CT considerando tags pasivos	191
5.20 Energía consumida por los protocolos CwT y CT considerando tags activos	192
5.21 Latencias medias de los protocolos variando el conjunto de tags existente en la zona de interrogación y el ratio entre la velocidad del lector y los tags: (a) QwT comparado con su predecesor QT, (b) CwT comparado con CT	195
5.22 Latencias medias de los protocolos: (a) QwT comparado con el FTTP, (b) CwT comparado con el OQTT; variando el conjunto de tags existente en la zona de interrogación y el ratio entre la velocidad del lector y los tags	196
5.23 Número promedio de bits transmitidos por cada tag en un ciclo completo de interrogación del protocolo CwT y diversos protocolos de la literatura existente. Se muestra en (b) el rango de bits [180-230] ampliado	198
5.24 Comparativa del protocolo CwT con diversos protocolos de la literatura existente en: (a) eficiencia de bits de los tags y (b) slots totales necesarios para identificar la totalidad del conjunto de tags	200
5.25 Comparativa del protocolo CwT y diversos protocolos de la literatura existente en: (a) los bits transmitidos por su correspondiente lector, y (b) los bits totales transmitidos entre lector y tags	201
5.26 Comparativa de las latencias medias del protocolo CwT y diversos protocolos de la literatura existente	202
5.27 Número de slots libres del protocolo CwT comparado con diversos protocolos de la literatura existente	204

ÍNDICE DE FIGURAS

5.28	Energía consumida por el protocolo CwT comparado con diversos protocolos de la literatura existente empleando tags pasivos	205
5.29	Comparativa del protocolo CwT+ con el CwT y diversos protocolos de la literatura existente en número promedio de bits transmitidos por cada tag en un ciclo completo de interrogación. Se muestra en (b) el rango de bits [180-220] ampliado	206
5.30	Comparativa del protocolo CwT+ con el CwT y diversos protocolos de la literatura existente en eficiencia de bits transmitidos por los tags	206
5.31	Comparativa del protocolo CwT+ con el CwT y diversos protocolos de la literatura existente en latencia	207

Índice de tablas

1.1	Comparativa entre las diferentes tecnologías de auto-ID	5
1.2	Actividades más relevantes y objetivo específico al que contribuyen	18
2.1	Comparativa de los métodos probabilísticos	36
2.2	Ejemplo de ejecución del protocolo TS	39
2.3	Identificación mediante el protocolo QT	43
2.4	Identificación mediante el protocolo CT	48
2.5	Ejemplo del protocolo ID-BTS	52
2.6	Comparativa de los métodos deterministas	54
2.7	Estándares RFID y sus protocolos de anti-colisión	61
3.1	Métodos de ahorro de bits en el protocolo QT	72
4.1	Términos empleados y su notación	79
4.2	Ejecución detallada del ejemplo mostrado en la Figura 4.7 del QT con $W = 2$	91
4.3	Bits transmitidos por los tags de las Figuras 4.7 y 4.8	93
4.4	Ejecución detallada del ejemplo mostrado en la Figura 4.33 del CT con $W = 2$	126
4.5	Bits transmitidos por los tags en los ejemplos de las Figuras 4.33 y 4.34	128
5.1	Potencias empleadas por un sistema RFID comercial	169
5.2	Valores de las distintas variables empleadas en la simulación	171
5.3	Tag bits/slot de los protocolos QwT y QT	171

ÍNDICE DE TABLAS

5.4	Bits/slot de los protocolos QwT y QT	173
5.5	Latencia/slot de los protocolos QwT y QT	174
5.6	Tag bits/slot de los protocolos comparados en la Figura 5.10	180
5.7	Bits/slot de los protocolos comparados en la Figura 5.12	183
5.8	Latencia de los protocolos comparados en la Figura 5.13	184
5.9	Tag bits/slot de los protocolos CwT y CT	188
5.10	Bits/slot de los protocolos CwT y CT	189
5.11	Latencia/slot de los protocolos CwT y CT	190
5.12	Tag bits/slot de los protocolos comparados en la Figura 5.23	199
5.13	Bits/slot de los protocolos comparados en la Figura 5.25	200
5.14	Latencia de los protocolos comparados en la Figura 5.26	203

*Toda tecnología lo suficientemen-
te avanzada es indistinguible de la
magia*

Arthur C. Clarke

1

Introducción y motivación

El control del flujo de productos en toda la cadena de suministro, desde la producción en la industria hasta la entrega al cliente, pasando por su almacenaje, es un aspecto que debe ser controlado y monitorizado. La logística y el coste de inventarios tienen la necesidad de conocer la cantidad y el tipo de productos que se transportan o tienen almacenados, ya que la gestión errónea de todos los bienes inmersos en la cadena de suministro suponen costes muy elevados. La realización de las entregas en el menor tiempo y de la manera más eficiente posible, así como el seguimiento de los artículos conociendo dónde están o han estado, son algunas de las principales necesidades de la cadena de suministro. Estos productos se registran en bases de datos que muestran los elementos producidos, almacenados o entregados. Además, deben ser actualizadas ya que cuando alguno de esos productos pasa de un estado a otro, dichas bases de datos quedan obsoletas.

Para controlar esos movimientos de forma constante, se crearon las tecnologías de identificación automática. Mediante estas tecnologías se pueden realizar operaciones como obtener información de trazabilidad de los productos, impedir su falsificación o controlar la seguridad, entre otras. En este primer capítulo se introducen las principales tecnologías de auto identificación, para seguidamente, orientarse hacia la tecnología *Radio Frequency IDentification* o identificación por radio

1. Introducción y motivación

frecuencia (RFID), que es en la que se centra esta tesis. Una vez introducida la tecnología se procederá a presentar uno de los problemas existentes que se pretende resolver mediante la solución propuesta en la presente tesis. Al final del capítulo se habrá expuesto de forma clara la problemática existente que se pretende afrontar, con el fin de darle una solución.

1.1 Tecnologías de identificación automática (auto-ID)

La identificación automática (auto-ID) [Finkenzeller 99] nació en 1949 de la mano de Norman Woodland y Bernard Silver [Woodland 52]. Ellos diseñaron el *bull's eye*, un código de barras circulares concéntricas que derivó en el *Universal Product Code* (UPC), dando paso al conocido código de barras. A partir de este punto, se han ido desarrollando diferentes métodos de auto-ID. Las principales tecnologías empleadas son:

- ◇ El código de barras. Es el primero y más famoso método de auto-ID. Éste refleja la luz enviada por un scanner que la convierte en energía eléctrica, y es interpretada como datos. Actualmente se emplea en la gran mayoría de productos comerciales, principalmente para controlar el stock. Existen también códigos de barras de dos dimensiones, *dataMatrix*, en los que la cantidad de información contenida es mucho mayor. Esta tecnología posee limitaciones como la necesidad de que los objetos sean manipulados para ser escaneados o el requerimiento de una línea de visión directa entre el scanner y el código.
- ◇ El reconocimiento óptico de caracteres (OCR), usado por primera vez en la década de los 60. Actualmente se usa en producción, campos de servicios y administrativos, y en algunos bancos para el registro de cheques. Los inconvenientes de estos sistemas residen en su alto precio, y en la complejidad de los lectores en comparación con otros sistemas de identificación.
- ◇ Los procedimientos biométricos. Son sistemas que identifican personas por comparación de características individuales que no admiten equivocación. Podemos hablar de sistemas identificadores por huella dactilar, identificadores por voz y en menor número, identificadores por retina.

1.1 Tecnologías de identificación automática (auto-ID)

- ◇ La *Smart-Card*. Es un sistema de almacenamiento electrónico de datos, con capacidad adicional para procesarlos. Por conveniencia, suelen instalarse dentro de una tarjeta de plástico del tamaño de una tarjeta de crédito. Las primeras Smart-Cards se lanzaron en 1984 como tarjetas telefónicas. El contacto con el lector proporciona la alimentación, el reloj y una interfaz serie bidireccional (puerto E/S). Una de las principales ventajas de estas tarjetas inteligentes es la facilidad de almacenaje de información, así como la protección que posee ante posibles accesos indeseados. Además, son seguras y baratas. Su desventaja es la vulnerabilidad a contactos con ropa, corrosión y suciedad.

- ◇ La tecnología *RFID*. Emplea ondas de radio frecuencia para leer los nodos que contienen la información. Esta tecnología presenta una serie de ventajas con respecto a otros métodos de auto-ID:
 - No se requiere la intervención de una persona para leer los datos. La identificación de los elementos es independiente de la posición o la distancia hasta un radio de 10 metros.
 - No es necesaria una línea de visión directa entre el lector y el soporte de datos. Empleando ondas de radio frecuencia, el lector RFID puede obtener la información almacenada en los soportes de datos a través de diferentes materiales.
 - Dispone de mayor capacidad de manejo de información, lo que le permite identificar cada producto de forma individual y no cada tipo de producto, como sucede con los códigos de barras. Además, el identificador de cada elemento y la información almacenada asociada a dicho identificador pueden modificarse dinámicamente.
 - Permite ocultar el identificador a las personas, de manera que se puede evitar conocer el número de identificación simplemente mirando el dispositivo. Puede almacenarse en el interior del objeto a identificar, evitando además posibles daños o roturas que puedan impedir su identificación.

1. Introducción y motivación

De todas estas tecnologías de auto-ID presentadas, actualmente la más empleada es el código de barras, aunque la tecnología RFID la está desplazando progresivamente y se está teniendo en cuenta en mayor medida, gracias a sus características no intrusivas y a las ventajas que presenta con respecto a otros métodos de auto-ID. Aun considerando todas estas ventajas, los códigos de barras no van a ser sustituidos a corto plazo por RFID, dado que se encuentran ampliamente extendidos. No obstante, se prevé que ambas tecnologías convivan aprovechándose de sus respectivas ventajas. A continuación, en la Tabla 1.1, se muestra una comparativa de los principales métodos existentes de auto-ID [Finkenzeller 99].

La tecnología RFID se va asentando y ganando terreno a muchos de los mencionados sistemas de auto-ID en la sociedad actual. Aplicaciones muy diversas están sirviendo para demostrar el potencial que muestra esta tecnología. Etiquetando todos los objetos con RFID, todo puede ser identificado, creando numerosos beneficios en la cadena de suministro, en el ámbito industrial, en aplicaciones de seguimiento y en la localización de activos o trazabilidad de mercancías. Existen multitud de investigaciones orientadas al uso de esta tecnología en un futuro no muy lejano dentro de estándares logísticos. Cabe destacar que una de las características más beneficiosas de esta tecnología es la facilidad con la que se adapta al paradigma de la *Inteligencia ambiental* o la *Computación ubicua*. Objetos como libros, contenedores de barcos, llaves de coche, maletas, ropa u otros objetos pueden convertirse en nodos de una red que envíen sus identificadores y datos. A esta red de comunicaciones entre objetos se le denomina también el *Internet de las cosas*. Etiquetando todos estos objetos e incluso incorporando sensores a estas etiquetas, en adelante tags, se les dota de capacidad computacional, logrando acercarse a la predicción que realizó Mark Weiser en 1991 [Weiser 91] sobre la computación ubicua y su relación directa con este paradigma. Dichos objetos permanecen conectados al mundo virtual y pueden ser controlados remotamente, actuando como puntos de acceso a los servicios de Internet.

RFID es una herramienta de computación invisible, es decir, que es capaz de situarse en nuestra periferia de forma que resulte invisible para nuestra percepción y posteriormente pasar a ser el centro de atención cuando la necesitemos [Héctor Ramos 06]. Además, las características no intrusivas de la tecnología per-

1.1 Tecnologías de identificación automática (auto-ID)

Características	Código de barras	OCR	Biometría	Smart-Card	RFID
Cantidad de datos (bytes)	1-100	1-100	-	16-64k	16-64k
Interpretable por personas	Limitado	Sí	Difícil	No	No
Línea de visión	Sí	Sí	Sí	-	No
Influencia de la suciedad	Muy alta	Muy alta	-	-	No influye
Degradación / Desgaste	Limitado	Limitado	-	Contactos	No influye
Complejidad electrónica	Muy Baja	Media	Muy Alta	Baja	Media
Velocidad lectura	Baja $\sim 4 s$	Baja $\sim 3 s$	Baja $> 9 s$	Baja $\sim 4 s$	Alta $\sim 5 m.s$
Distancia lector-datos	0 – 50 <i>cm</i>	< 1 <i>cm</i>	Contacto	Contacto	0 – 10 <i>m</i>

Tabla 1.1: Comparativa entre las diferentes tecnologías de auto-ID

1. Introducción y motivación

miten que las máquinas u objetos desarrollen sus propósitos sin necesidad de que el hombre se preocupe de que los realizan [Want 04].

1.2 Tecnología RFID

La identificación por radio frecuencia (RFID) es una tecnología de auto identificación inalámbrica empleada para detectar el código almacenado en una etiqueta con un microchip, empleando ondas de radio frecuencia (RF). La principal idea de esta tecnología es la de marcar con un tag cada uno de los objetos que se desean identificar o rastrear. De esta forma cada objeto puede ser identificado gracias a los mensajes que envían sus correspondientes tags, bien de solo-lectura o de lectura-escritura, a un lector especializado (lector RFID o interrogador), sin necesidad de que exista una línea de visión directa.

Los numerosos beneficios que proporciona la tecnología RFID permiten que cada vez más empresas implanten RFID en sus almacenes o aplicaciones, y que se realicen multitud de investigaciones orientadas al uso de esta tecnología en un futuro próximo. Por otro lado, la amplia expansión del código de barras provocará que la tecnología RFID deba previsiblemente convivir con él, antes de sustituir una tecnología por otra.

1.2.1 Contexto histórico

La tecnología RFID no tiene una historia ni un descubridor claro, ya que surgió por la aportación de numerosos investigadores y gracias a la aplicación de avances en otros campos tecnológicos. En pocas decenas de años, los sistemas de RFID se han ido transformando, de simples apariciones en artículos de revistas científicas, a toda una realidad.

Los primeros desarrollos de esta tecnología [Landt 05] se produjeron durante la 2ª Guerra Mundial, gracias a los importantes avances conseguidos en el estudio del radar y la reflexión de las ondas de radio. Los problemas para diferenciar los aviones enemigos de los aliados llevaron a los ingenieros británicos a desarrollar un transmisor en cada avión, que al recibir una señal de radar contestara con otra señal conocida y fuera identificado como aliado o enemigo, el denominado *identification*,

friend or foe (IFF). Dado que todas las investigaciones fueron de índole militar, la mayoría se realizaron en secreto.

Los primeros experimentos conocidos datan de 1948, y en 1960 se realizaron los primeros con tags de 1 bit. Aún así, no fue hasta 1970 cuando se comenzó a trabajar de forma muy activa en RFID y se produjeron grandes avances instaurando el tag completamente pasivo con un rango de decenas de metros.

Con la implantación de la tecnología *Complementary Metal-Oxide Semiconductor* (CMOS) para la fabricación de circuitos integrados, el tag RFID mejoró en funcionalidad y se redujo su tamaño. En la década de los 80, la aparición del ordenador personal (PC), permitió la captura de datos de los dispositivos RFID de forma más conveniente y económica, y los tags pasaron a fabricarse con tecnología CMOS y memorias EEPROM. En los 90, se instalaron los primeros lectores multi-protocolo y se extendió la utilización del RFID en diversas aplicaciones comerciales. Paralelamente, la posibilidad de fabricar el diodo Schotky con tecnología CMOS [Tran 07] permitió la aparición de tags de alta frecuencia.

Hoy en día, la tecnología RFID se está extendiendo a la cadena de suministro ya que la naturaleza del RFID se adecúa perfectamente a sus necesidades. Además, el coste de fabricación de los tags actuales, compuestos por un pequeño circuito integrado CMOS y una antena impresos sobre una pegatina, ha descendido considerablemente para competir de forma directa con el código de barras.

1.2.2 Características de RFID

El principal cometido de un sistema RFID es leer los códigos almacenados en etiquetas o tags, empleando para ello las ondas de radio frecuencia. Un sistema RFID, por tanto, está compuesto principalmente por tres elementos [Finkenzeller 99]: uno o varios tags, un lector y un subsistema de control o almacenamiento (ver Figura 1.1).

- ◇ Uno o varios *tags*. Contenedores de los datos o identificadores (ID), están formados por un microchip y una pequeña antena, y se adhieren al objeto que se desea identificar. Teniendo en cuenta cómo son alimentados los tags se dividen en:

1. Introducción y motivación

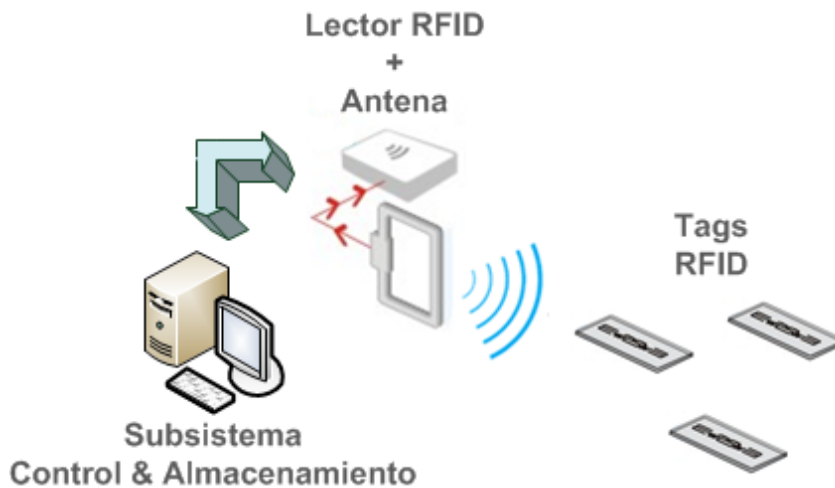


Figura 1.1: Sistema RFID

- *Pasivos*: no tienen ningún tipo de fuente de alimentación o batería. La energía la obtienen de la señal que el lector envía en el momento de la detección. El tag se energiza con la señal recibida, la modula y la retransmite de vuelta. Esto provoca que el alcance de estos tags esté limitado a rangos típicos de 3-5 m.
- *Activos*: están alimentados por una fuente de alimentación interna, generalmente una batería. Al no depender de la señal del lector para energizarse, son capaces de interpretar señales más débiles, y retransmitirlas con más potencia, incrementando el rango de acción hasta un total de 15 metros aproximadamente.
- ◇ Un *lector*. Compuesto por un módulo de RF (transmisor - receptor), una unidad de control y una o más antenas, es el encargado de establecer la comunicación con los tags y transmitir la información obtenida. En la literatura existente se le denomina comúnmente lector, aunque en numerosas aplicaciones son capaces de escribir datos en los propios tags.
- ◇ Un *subsistema de control* y almacenamiento de datos. Conectado al lector, puede controlar su funcionamiento o almacenar los datos que éste le proporcione.

La comunicación de los sistemas RFID entre el lector y el tag se produce empleando el mismo canal de transmisión, el aire. La mayoría de los sistemas RFID operan en la banda Industrial Científico Médica (ISM) y son designadas por la Unión Internacional de Telecomunicaciones (ITU) [Regulations 08]. En función de las características de alcance y velocidad que se desean obtener en el sistema se emplean diferentes frecuencias, siendo las más empleadas 13,56 MHz en alta frecuencia (HF) y un rango de 860 - 960 MHz en ultra alta frecuencia (UHF), del que solo se pueden usar 865,6 – 867,6 MHz en España [Barthel 06]. HF permite coberturas desde varios centímetros hasta un metro, y transmite la información modulando un campo magnético generado por el lector. Por otro lado, las frecuencias UHF son capaces de alcanzar hasta 10 metros de distancia, y emplean una onda electromagnética que varía su amplitud (modulación por *backscatter* [Dobkin 12]) para la transmisión de los datos.

El proceso de comunicación entre un lector y un tag suele comenzar con la transmisión de un comando por parte del lector. Éste es recibido por el tag, que energiza su circuitería con la señal recibida (en el caso de tags pasivos), y emite una respuesta mientras el lector mantiene la señal alimentadora, llamada onda continua o *Continuous Wave* (CW). Para establecer la comunicación entre los dispositivos se debe producir un acoplamiento entre ellos, que puede ser magnético o electromagnético (*backscatter*). La principal diferencia entre ambos es el campo de operación: cercano (*near field*) o lejano (*far field*). El límite teórico entre ambos se calcula como $\frac{\lambda}{2\pi}$ donde λ es la longitud de onda [Finkenzeller 99, Want 06]. A continuación se expone una breve descripción de cada uno de los acoplamientos.

- ◇ Acoplamiento inductivo: en este tipo de comunicación se emplea el campo magnético generado por la señal de radio del lector para inducir una corriente en una bobina alojada en el tag. Se induce una tensión alterna (AC) en dicha bobina que carga un condensador conectado a continuación. Juntos forman un circuito LC, que rectifica y proporciona la tensión continua (DC) necesaria para energizar el microchip del tag cuando se sintoniza a la frecuencia de diseño. La comunicación entre el tag y el lector se produce modulando la carga o impedancia del tag acorde con su identificador, y se suele emplear en el rango de baja frecuencia (LF) y alta frecuencia (HF). Para una frecuencia

1. Introducción y motivación

de 13,56 MHz, típica de RFID en HF, la longitud de onda es de 22 metros y su límite de campo cercano es de 3,52 metros. Por ello, este tipo de sistema tiene un alcance máximo de 3 metros aproximadamente. En la Figura 1.2 se muestra un esquema del acoplamiento inductivo.

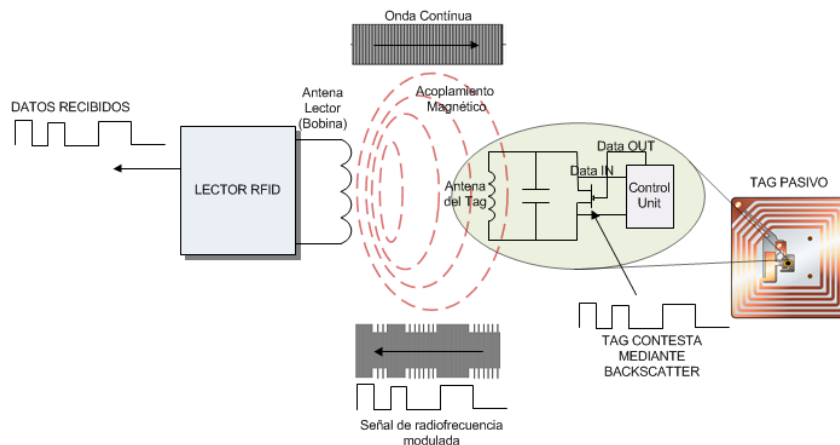


Figura 1.2: Sistema RFID con acoplamiento inductivo

- ◇ Acoplamiento electromagnético (*backscatter*): la principal característica de este sistema es que el rango de lectura es mayor, alcanzando más distancia. En el caso de captar la energía en campo lejano (*far field*), la transmisión y recepción debe ser diferente a la del campo cercano ya que más allá del límite calculado anteriormente la comunicación no es eficiente [Scharfeld 01]. Por esta razón se emplean antenas resonantes, generalmente dipolos de tamaños cercanos a la mitad de la longitud de onda de la frecuencia de transmisión. El lector RFID envía una onda electromagnética (EM) que crea una diferencia de potencial en los extremos del dipolo del tag. Éste emplea la energía de la señal de interrogación para cargar un condensador a través de un circuito *front-end* y provee de alimentación a la etiqueta y su microchip. Entonces el tag varía la amplitud de dichas ondas EM que se reflejan en la antena en concordancia con los datos digitales a transmitir (ver Figura 1.3). Esta forma de comunicación se denomina modulación por *backscatter* y permite al tag modular la señal del campo electromagnético tanto en amplitud como en fase. La señal reflejada y modulada por la etiqueta es interpretada posteriormente

1.3 El problema de las colisiones de tag

por el lector, demodulándola en un receptor. Este método se emplea en UHF y en la banda de microondas, y puede alcanzar distancias de hasta 15 m.

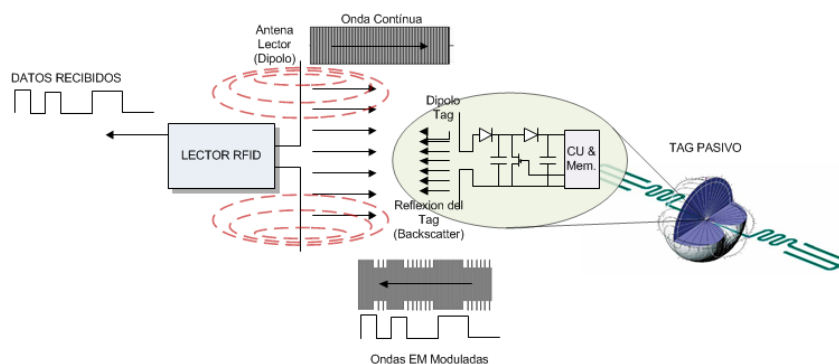


Figura 1.3: Sistema RFID con acoplamiento electromagnético

Los datos básicos contenidos en cada tag constan básicamente de un número identificativo o identificador (ID), que transmite al lector cuando éste desea reconocer el entorno. Estos identificadores varían desde unos pocos bits hasta 512 bits, siendo 96 bits el tamaño más empleado actualmente. En esta cantidad de información se recogen datos como el tipo de producto, la empresa fabricante o el número de palé en el que ha sido transportado. Además, en determinados tags, este ID puede ser modificado dinámicamente un número cuasi ilimitado de veces, dotando al sistema de una gran versatilidad.

1.3 El problema de las colisiones de tag

En un sistema RFID se puede encontrar al menos un lector y uno o varios tags. Cuando el lector necesita saber qué tags están a su alrededor, envía un mensaje de interrogación al canal de transmisión. Dado que el canal de transmisión es el aire, y es un canal único, debe ser compartido por todos los participantes en la comunicación. Con el auge de la utilización de los rangos de frecuencias HF, UHF y de microondas, el alcance de detección de los lectores se ha ampliado en gran medida. Por tanto, es habitual que en el mismo rango de lectura existan varios tags que contesten simultáneamente a la interrogación del lector, mezclándose sus señales y pudiendo llegar a cancelarse unas con otras. A las interferencias que

1. Introducción y motivación

se producen entre los diferentes mensajes enviados por los lectores o los tags, y que causan la ilegibilidad o pérdida de éstos, se las conoce como colisiones de mensajes. En el campo de RFID este problema es conocido como **el problema de las colisiones de tag** [Finkenzeller 99, Abraham 02, Yang 10] y representa una de las áreas de investigación vigentes en el marco de esta tecnología. La colisión de mensajes obliga al lector a solicitar los identificadores de nuevo, es decir, que los tags deban reenviar su ID, malgastando el ancho de banda del canal, incrementando el tiempo total en completar el ciclo de identificación y aumentando el consumo total del sistema.

El problema de las colisiones de mensajes no es nuevo en el campo de las redes inalámbricas. Cualquier red que emplea varios nodos en un mismo canal de transmisión necesita un protocolo de acceso múltiple o multi-acceso. Esto es, un protocolo que, empleando diferentes técnicas y métodos, establece una serie de reglas de acceso al medio con el objetivo de evitar o minimizar las colisiones [Abramson 70, Massey 80]. Aplicando esta situación a RFID, el lector y los tags equivalen a los nodos que conforman una serie de redes punto a punto (lector-tag) compartiendo el mismo canal de comunicación. Para evitar las transmisiones simultáneas se debe realizar una multiplexación de estas redes punto a punto para que no se solapen y se produzcan las colisiones. Sin embargo, el problema de las colisiones de tag representa un caso único y los protocolos de multi-acceso o resolución de colisiones clásicos existentes no pueden ser aplicados a las transmisiones de un sistema RFID directamente por las siguientes restricciones ([Abraham 02]):

- ◇ La falta de una fuente de alimentación interna en el caso de los tags pasivos hace que éstos deban ser alimentados por medio de una señal proporcionada por el lector.
- ◇ El número total de tags a leer es desconocido.
- ◇ Los tags no pueden comunicarse entre sí. Por tanto, el grueso del algoritmo de anti-colisión debe residir en el lector.
- ◇ Las limitaciones de memoria de los tags y su escasa capacidad computacional hacen que el protocolo deba ser simple desde el punto de vista principalmente del tag.

1.3 El problema de las colisiones de tag

Mientras que los protocolos de multi-acceso clásicos persiguen mejorar el rendimiento, la estabilidad y la disminución del tiempo de retardo de los paquetes, los protocolos de resolución de colisiones de RFID apuestan por reducir el tiempo para identificar todos los tags existentes y la energía consumida en el proceso. Por las características que les diferencian y porque las metas que persiguen son diferentes a las de los protocolos de multi-acceso, los protocolos de resolución de colisiones en RFID se denominan **protocolos de anti-colisión**.

El problema de las colisiones implica por tanto, el desarrollo de un protocolo de anti-colisión eficiente y rápido [Shih 06]. Abraham dice en [Abraham 02] que las características deseables para un protocolo de comunicaciones entre tag y lector son:

- ◇ *Mínima latencia*. El tiempo empleado en la lectura de los tags situados en el entorno del lector debe ser el menor posible para que resulte imperceptible.
- ◇ *Reducido consumo* energético. Tanto en el caso de tags activos como pasivos el consumo debe minimizarse. Para ello se debe minimizar la complejidad del hardware y reducir la cantidad de información intercambiada entre los elementos del sistema.
- ◇ *Exactitud y confiabilidad*. El protocolo debe identificar todos los tags del entorno del lector.
- ◇ *Independencia de una línea de visión*. El tag a identificar puede estar en cualquier zona del rango de lectura.
- ◇ *Máxima robustez*. El protocolo debe trabajar de forma independiente ante diferentes condiciones medio-ambientales.
- ◇ *Escalabilidad*. El protocolo debe acomodarse al número de tags presentes en el rango de lectura.

Las características esperadas de un lector, por tanto, son la identificación de múltiples tags con el mínimo retardo o tiempo necesario y el empleo para ello del

1. Introducción y motivación

menor consumo posible. Ante esta problemática, se presenta un nuevo reto orientado al arbitraje de los mensajes de los diferentes tags en respuesta a las interrogaciones del lector.

Existen diferentes tipos de protocolos de anti-colisión, como se verá más adelante en el estado del arte. Cada uno de éstos posee una serie de ventajas e inconvenientes, pero no existe un tipo claramente destacado. Esta tesis se centra en los protocolos basados en *queries*, o protocolos sin memoria, que pertenecen al tipo de protocolos basados en árbol. Este tipo de protocolos son capaces de identificar todos los tags situados en la zona de interrogación del lector, y su costo electrónico es muy reducido. Generalmente son protocolos relativamente simples, acorde a los requerimientos anteriormente expuestos por Abraham. Sin embargo, el hecho de que los tags no posean memoria interna en el proceso de identificación obliga al lector y a los tags a incrementar la cantidad de bits transmitidos. El lector debe comunicar en cada interrogación la información necesaria para que los tags sepan si deben contestar o no. Además, la propia naturaleza de este tipo de protocolos genera una gran cantidad de colisiones al inicio del ciclo de lectura. El problema que se genera entonces, es que dada la simplicidad de estos dispositivos, se ven obligados a transmitir su ID al completo sin el conocimiento de que vaya a llegar a su destino. Es decir, que en una colisión, los tags que hayan transmitido habrán malgastado el tiempo y la energía necesarias para transmitir su ID (ver Figura 1.4). Además, la longitud de los IDs de los tags ha aumentado en los últimos años, comprendiendo una mayor cantidad de información y provocando que los bits desperdiciados en la colisión generen un gasto innecesario mayor. Esto quiere decir, que las colisiones van a provocar un aumento del tiempo necesario para identificar los tags y del consumo energético del sistema. Los protocolos de este tipo, por tanto, presentan una gran capacidad de mejora en este aspecto.

1.4 Trabajo de tesis

En esta tesis se propone una solución para resolver el problema de las colisiones de tag, atendiendo a los requisitos y necesidades impuestas por los protocolos de anti-colisión. Después de analizar de forma crítica y en profundidad el problema encontrado en el estado del arte actual, se plantea la siguiente hipótesis a validar

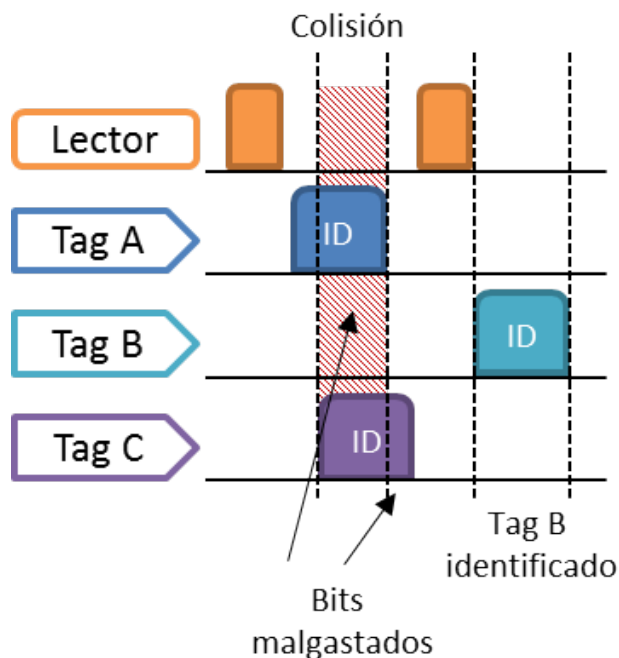


Figura 1.4: Ejemplo de lectura de 3 tags, donde se produce una colisión y una identificación

durante el desarrollo de la tesis doctoral. Además, se plantean una serie de objetivos que delimitan el ámbito de trabajo a realizar, dividiéndose éstos en generales y específicos.

1.4.1 Hipótesis

Una vez identificado el problema y con la intención de resolverlo, se plantea la siguiente hipótesis:

«Es posible diseñar un procedimiento que, aplicado sobre un protocolo de anti-colisión RFID basado en *queries* ya existente, consiga reducir el número de bits transmitidos por los tags, mejorando el comportamiento del protocolo en latencia y consumo energético en entornos de tags pasivos.»

Es decir, se va a desarrollar un procedimiento que pueda ser aplicado a protocolos de anti-colisión cuyos tags no requieran memoria en el proceso de identifica-

1. Introducción y motivación

ción, simplificando el hardware en gran medida y reduciendo el costo electrónico de los tags. Además, este procedimiento pretenderá reducir el número de bits que transmiten los tags, que se verá reflejado en una disminución del intercambio total de información entre lector y tags, del tiempo necesario para identificar el conjunto total de tags y del consumo energético del sistema RFID. Finalmente, se pretende que el protocolo resultante no solo consiga mejores resultados que el protocolo original al que se le ha aplicado el procedimiento, sino que mejore las latencias y consumos de diversos protocolos de la literatura reciente.

1.4.2 Objetivos

El objetivo general que se pretende alcanzar con esta tesis es *desarrollar un procedimiento, denominado ventana de bits, que modifique los protocolos basados en queries existentes, conformando así nuevos protocolos*. Se le denomina ventana a un marco variable que define el número de bits que transmite un tag. Con este método se pretende mejorar las características de latencia y consumo energético del protocolo al que se le aplica dicho procedimiento (protocolo origen). Además, dado que existen multitud de ellos en la literatura actual, se pretende que los protocolos conformados también ofrezcan mejores prestaciones que los desarrollados recientemente ante las mismas condiciones de trabajo. Todo ello pretende llevarse a cabo sin incrementar en gran medida la complejidad del sistema RFID y de los tags.

1.4.2.1 Objetivos específicos

Los principales objetivos específicos que se derivan de la hipótesis y que a su vez son necesarios para conseguir el objetivo general son:

- ◇ Definir los requerimientos y el comportamiento deseado para la ventana de bits [OE1]. O lo que es lo mismo, analizar exhaustivamente los protocolos de anti-colisión existentes para identificar aquellos que pueden ser mejorados con este procedimiento y determinar el proceso que éste debe seguir.
- ◇ Desarrollar el procedimiento de la ventana de bits [OE2]. Éste debe mantener la simplicidad de los tags y reducir su número de bits transmitidos.

Primeramente, se estudiará el protocolo escogido con la ventana constante para, una vez obtenidas las conclusiones, plantear el funcionamiento dinámico al procedimiento, con la intención de disminuir la latencia y el consumo energético.

- ◇ Configurar el entorno de pruebas [OE3]. Se trata de plantear un banco de pruebas donde se comparen los principales protocolos de anti-colisión existentes, directamente relacionados con los protocolos propuestos. Para ello se empleará un simulador donde se emulen los comportamientos de los diversos protocolos planteados y puedan ser comparados entre sí.
- ◇ Evaluar los resultados obtenidos de las simulaciones en diferentes aspectos [OE4]. Inicialmente, se observará el número de bits transmitidos por los tags, ya que es donde actúa principalmente la ventana de bits. Y posteriormente, se estudiará el total de bits transmitidos, su influencia en la latencia del sistema y su consumo energético.

En cuanto a las actividades que posibilitarán la consecución de los objetivos generales y por consiguiente la validación de la hipótesis, la Tabla 1.2 recoge algunas de las más representativas y su vínculo con el objetivo específico al que contribuyen.

1.5 Metodología de la Investigación

Dado que la tecnología RFID avanza a un ritmo trepidante, ha sido necesaria la utilización de una metodología de investigación que se adecúe a esta velocidad. Por ello, es necesario un ciclo continuo de revisión del estado del arte, que contemple la aparición de posibles mejoras de forma continuada.

Con estas necesidades, se plantea un proceso definido de forma iterativa y cíclica, en el que se vayan planteando soluciones cada vez más refinadas mediante la completitud de ciclos, en cada uno de los cuales se ponen en marcha nuevas ideas, que son puestas en práctica y comprobadas en el ciclo siguiente.

Esta metodología cíclica que se muestra en la Figura 1.5, puede resumirse en los siguientes puntos:

1. Introducción y motivación

ACTIVIDADES	OBJ. ESP.
Análisis de los protocolos de anti-colisión existentes	OE1
Análisis de la eficiencia de transmisión de bits y su relación con la latencia y el consumo energético	OE1
Desarrollo del procedimiento/técnica	OE2
Aplicación a un protocolo en formato constante	OE2
Aplicación a un protocolo en formato dinámico	OE2
Desarrollo de un entorno de simulación de protocolos de anti-colisión	OE3
Diseño y configuración del entorno de experimentación	OE3
Ejecución de las pruebas en el simulador	OE3
Evaluación de los resultados de la experimentación	OE4

Tabla 1.2: Actividades más relevantes y objetivo específico al que contribuyen

- ◇ Actualización del conocimiento del estado del arte, empleando bibliografía relacionada y publicaciones de la comunidad científica en este ámbito. Esto permitirá conocer las limitaciones de los protocolos existentes y por tanto las oportunidades de mejora de este área de conocimiento.
- ◇ Diseño de la solución que solventa los problemas en los que se centra la tesis. Para ello, se va a definir una hipótesis y unos objetivos, que constituirán el enlace de unión entre los problemas y la solución.
- ◇ Experimentación y evaluación, para medir de forma cuantitativa el resultado obtenido.
- ◇ Validación con la comunidad científica, de tal forma que se determine la relevancia de la investigación.
- ◇ Adaptación del procedimiento diseñado, teniendo en cuenta los resultados obtenidos y las correcciones recibidas por parte de la comunidad científica.
- ◇ Análisis de los resultados, permitiendo la validación de la hipótesis planteada.

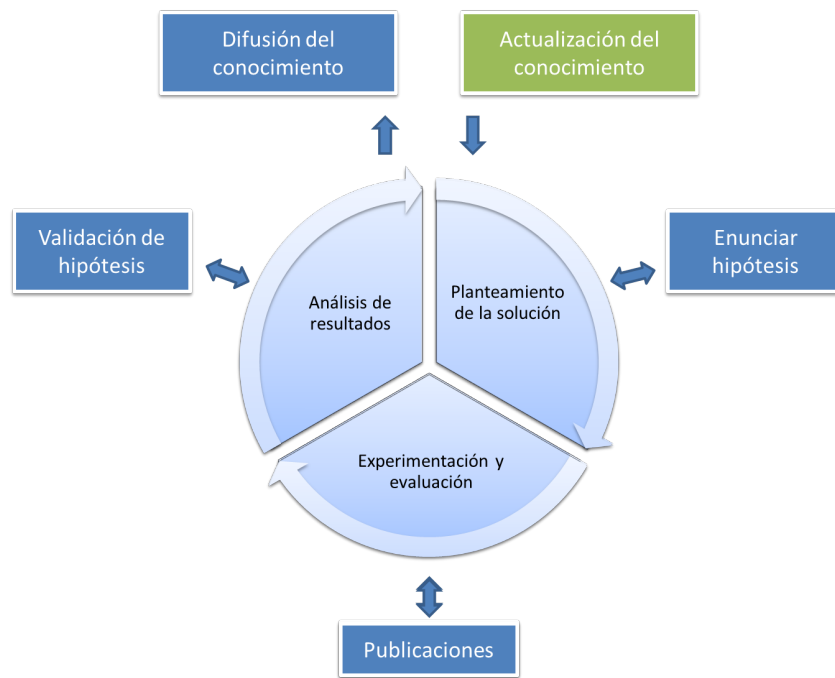


Figura 1.5: Metodología de investigación empleada

- ◇ Divulgación de los conocimientos obtenidos durante la investigación, contribuyendo a la producción científica y a la generación de conocimiento.

1.6 Estructura de la tesis

En esta sección se presenta la estructura de la presente tesis doctoral. Ésta consta de seis capítulos:

- ◇ El primer capítulo, el presente, asienta los conceptos básicos y el ámbito de trabajo de esta tesis. Primeramente se ha introducido la tecnología RFID como base para presentar e identificar el problema de las colisiones que esta tesis pretende resolver. Adicionalmente se han presentado las características deseables de la solución, la hipótesis de trabajo de la tesis y los objetivos que guían su desarrollo.
- ◇ El segundo capítulo proporciona al lector una visión general del estado del arte de los protocolos de anti-colisión de RFID, de manera que permita en-

1. Introducción y motivación

tender las diferentes estrategias existentes y por qué se han escogido los protocolos basados en *queries* como base de la tesis. El capítulo repasa los protocolos basados en Aloha o probabilísticos, los protocolos basados en árbol, donde se encuentran los basados en *queries*, y los híbridos, conformados a partir de las estrategias anteriores. En cada uno de los tipos se presentan los protocolos más representativos y algunas de las últimas mejoras aportadas.

- ◇ El tercer capítulo describe en mayor profundidad el problema existente con el exceso de transmisión de bits de tags en los protocolos basados en *queries*, y por qué no ha podido ser solucionado hasta el momento. Se aportan diferentes soluciones planteadas de forma teórica y se presentan las oportunidades de mejora.
- ◇ El cuarto capítulo describe el procedimiento principal del trabajo de tesis: la ventana de bits. Una vez expuestas las oportunidades de mejora, se presenta la solución principal de esta tesis. Inicialmente, se definen las características del procedimiento para, a continuación ser aplicado a dos protocolos ya existentes en la literatura, conformando dos nuevos protocolos de anti-colisión. Además se añade una mejora al segundo de los protocolos desarrollados, conformando finalmente una contribución de tres protocolos de anti-colisión. Cada uno de estos protocolos contiene sus correspondientes pruebas de ajuste de la ventana para obtener su mejor rendimiento, formadas por una sección con ventana constante y otra aportando dinamismo al procedimiento.
- ◇ El quinto capítulo aborda la experimentación como medio para la validación de la hipótesis enunciada en la tesis. Se analiza la hipótesis y se justifica mediante la presentación de un banco de pruebas. El desarrollo de un simulador permite comparar los protocolos presentados con sus originales sin la ventana de bits y posteriormente, con algunos de los protocolos de la literatura reciente. Se muestran, de esta manera, los resultados de las simulaciones realizadas, se analizan los datos obtenidos y se comprueba la veracidad de la hipótesis.
- ◇ El sexto capítulo recoge finalmente las conclusiones de la tesis, así como las líneas futuras de trabajo y mejora.

*Daría todo lo que sé por la mitad de
lo que ignoro*

René Descartes

2

Estado del arte

La identificación por radio frecuencia es un método de auto-ID inalámbrico empleado para detectar el código almacenado en una etiqueta (en adelante tag) con un microchip empleando ondas de radio frecuencia. La idea principal de esta tecnología es la de marcar con un tag cada uno de los objetos que se desean identificar o rastrear. De esta forma pueden identificarse gracias a los mensajes que intercambian con el lector, sin necesidad de que exista una línea de visión directa entre ellos.

RFID tiene la ventaja de ser una tecnología que admite gran flexibilidad, pero que por otro lado crea ciertos problemas. En esta tesis se va a tratar el problema de las colisiones de tag (ver Figura 2.1). Dado que un lector puede interrogar a varios tags que se encuentren en su zona de alcance, existe la posibilidad de que los mensajes de dichos tags interfieran entre sí y puedan cancelarse, este fenómeno se conoce como colisión de mensajes. Cuando el lector RFID detecta que se ha producido una colisión, solicita la retransmisión de los paquetes, lo que provoca un aumento del tiempo necesario para identificar los tags presentes y un incremento del consumo energético del sistema. Por ello, este problema debe minimizarse.

Como ya se ha comentado anteriormente, el problema de las colisiones no es novedoso. En la teoría de redes ya se ha estudiado anteriormente y se han desa-

2. Estado del arte

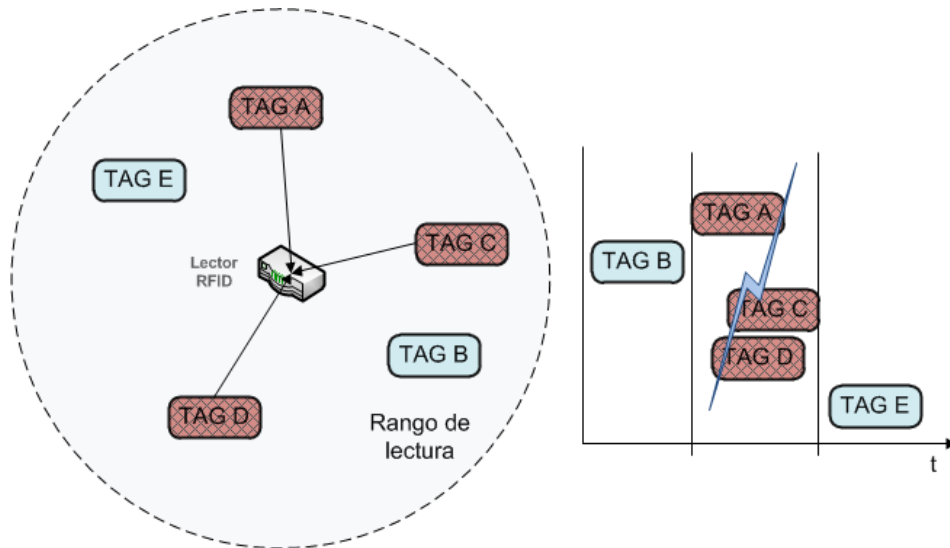


Figura 2.1: Problema de las colisiones de tag

rollado métodos para organizar las respuestas de los tags y reducir así las colisiones [Abramson 70, Roberts 75, Capetanakis 79, Massey 80, Kaplan 85]. Este tipo de procedimientos son denominados métodos de multi-acceso. Sin embargo, estos métodos no pueden aplicarse a RFID de forma directa, ya que la tecnología en cuestión impone una serie de restricciones a la hora de diseñarlos, como que el número de tags a identificar es desconocido, y que las comunicaciones tag-tag no son posibles debido a la simplicidad de su hardware. Por ello cuando los métodos de multi-acceso se aplican a la tecnología RFID reciben el nombre de protocolos de anti-colisión.

Los sistemas RFID se están empleando en aplicaciones de detección masiva de tags en diferentes entornos. Dado que los lectores emplean un medio común para comunicarse con los tags, el aire, aparece la necesidad de un protocolo de anti-colisión que establezca una serie de reglas de acceso al medio para realizar transmisiones satisfactorias. Estos protocolos tratan de evitar o minimizar las colisiones con diferentes técnicas y métodos. Para mediar en las comunicaciones de los diferentes tags, existen numerosas alternativas que tratan de dar solución a este problema. Una primera división en 4 métodos se realiza a nivel de capa física.

2.1 Protocolos de multi-acceso: Capa física

Los protocolos de multi-acceso presentados a continuación separan los mensajes de los tags desde la capa física. Los cuatro principales métodos existentes son [Finkenzeller 99, Shih 06, Klair 10, Liu 10]: multi-acceso por división del espacio (SDMA), multi-acceso por división en frecuencia (FDMA), multi-acceso por división de código (CDMA) y multi-acceso por división en tiempo (TDMA).

- ◇ *SDMA*. Multi-acceso por división del espacio: Las técnicas SDMA interrogan cada área o porción del espacio de forma consecutiva. Su implementación se lleva cabo empleando arrays de antenas direccionales que, controlados electrónicamente, pueden variar la directividad global de la antena realizando barridos o apuntando a zonas específicas del espacio donde está situado cada tag. En la Figura 2.2 se muestra un gráfico explicativo del método. Estos métodos requieren de un alto coste de implementación para el sistema de control de la antena adaptativa y su uso está restringido a unas pocas aplicaciones específicas [Yu 08, Sayeed 09].

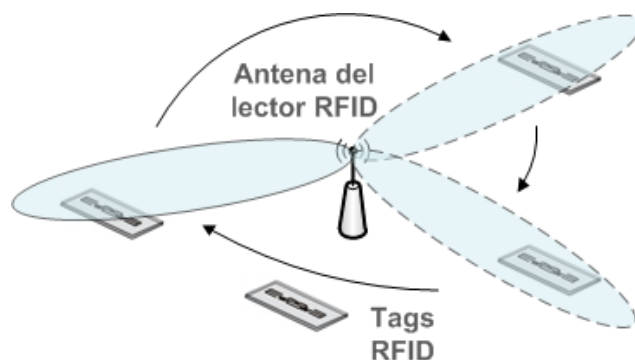


Figura 2.2: SDMA adaptativo con antena controlada direccionalmente

- ◇ *FDMA*. Multi-acceso por división en frecuencia: En los sistemas FDMA, el canal de comunicación se divide en subcanales que emplean diferentes señales portadoras de frecuencia, de forma que los participantes pueden transmitir en cada una de ellas sin interferirse. En la Figura 2.3 se muestra un ejemplo del procedimiento. El FDMA requiere de un alto nivel de prestaciones en los lectores ya que debe disponer de un receptor específico para cada

2. Estado del arte

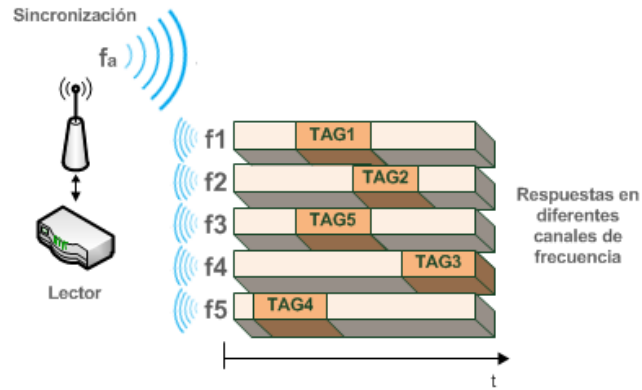


Figura 2.3: FDMA, canales de frecuencia adaptativos

frecuencia, así como tags capaces de transmitir en diferentes frecuencias. Liu et al. realiza unas simulaciones de un protocolo que emplea FDMA y TDMA en [Liu 09] y Geng et al. en [Geng 08].

- ◇ *CDMA*. Multi-acceso por división de código: Los métodos CDMA emplean técnicas de modulación para cubrir la mayor cantidad posible del espectro de frecuencias basadas en códigos pseudo-aleatorios. Requieren que los tags multipliquen su ID por una secuencia de números pseudo-aleatorios antes de transmitirlo. Un ejemplo del método se muestra en la Figura 2.4. Añade una gran complejidad a los participantes en la comunicación y puede ser computacionalmente exigente para los tags. Además son sistemas muy caros y de alto consumo [Loeffler 10, Loeffler 11]. También en [Mutti 08, Wang 07a], Mutti y Floerkemeier, y Wang et al. utilizan los métodos CDMA y TDMA,

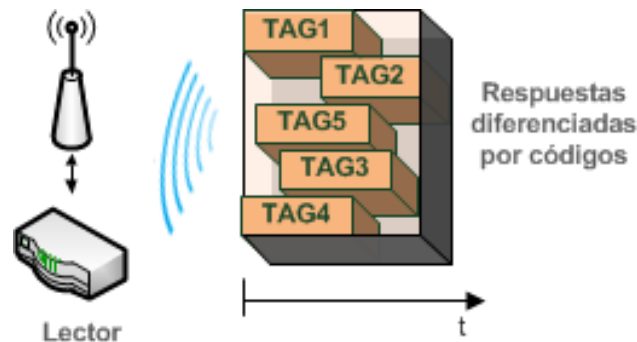


Figura 2.4: Ejemplo de ejecución CDMA

2.1 Protocolos de multi-acceso: Capa física

que se explica a continuación.

- ◇ *TDMA*. Multi-acceso por divisiones en el tiempo: Los procedimientos TDMA dividen la capacidad del canal entre los diferentes participantes de forma cronológica. Es decir, cada participante se organiza para utilizar el canal durante cierto tiempo y una vez terminado lo deja libre para otro usuario. Estos métodos son los más extendidos en el campo de las comunicaciones digitales y móviles. Al igual que en RFID, que también son los más empleados para resolver el problema de las colisiones de tag, ver Figura 2.5.

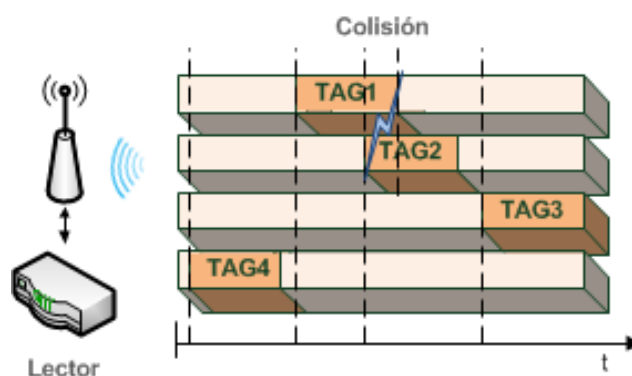


Figura 2.5: Ejemplo de ejecución TDMA

La presente tesis se centra en los métodos de multi-acceso TDMA. Estos ordenan cronológicamente las diferentes peticiones de comunicación en un único canal, separando en el tiempo las transmisiones de cada participante en la comunicación (tags). Dado que este tipo de procedimientos o protocolos son los más empleados en RFID, en posteriores secciones se realiza un estudio de los algoritmos de anti-colisión existentes que explotan la característica TDMA [Finkenzeller 99, Abraham 02, Shih 06, Tang 07, Wang 09, Klair 10, Geng 10, Zhu 11]. En la Figura 2.6 se muestran los principales algoritmos de anti-colisión existentes dentro de los métodos TDMA.

Numerosos protocolos han sido diseñados para la comunicación de sistemas inalámbricos, pero debido a las características específicas y los recursos limitados de los sistemas RFID, éstos han de ser modificados para adaptarse a dicha tecnología. Por tanto, los principales protocolos diseñados se han inspirado en otros, ya

2. Estado del arte

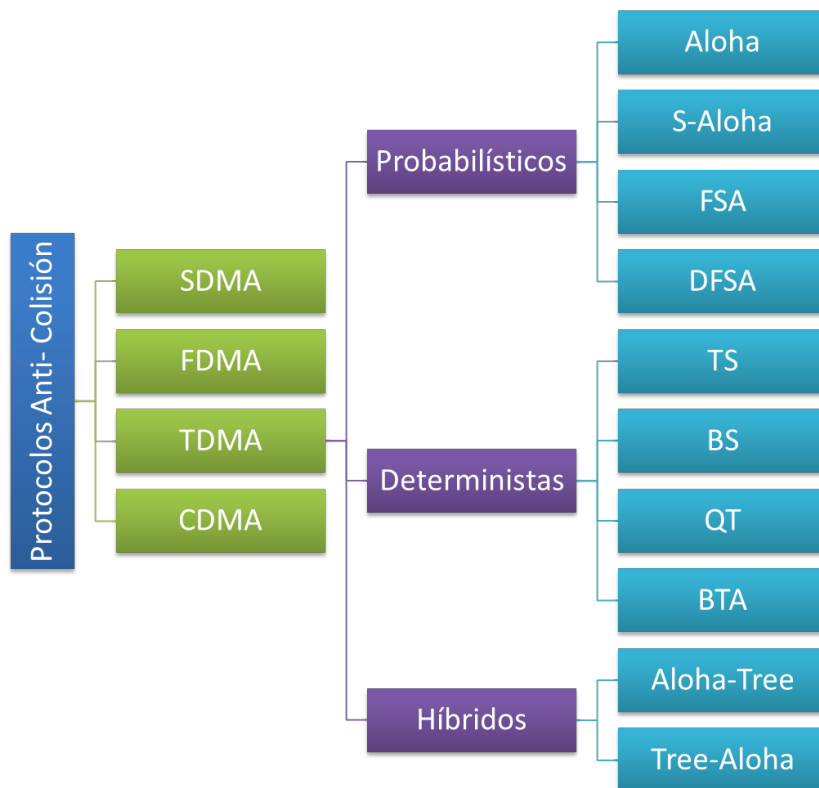


Figura 2.6: Taxonomía de los protocolos de anti-colisión

existentes en la teoría de redes y han sido adaptados para poder trabajar con RFID. Teniendo en cuenta los dos componentes básicos, el lector y uno o varios tags, el primero trata de identificar todos los tags que se encuentran en su zona de interrogación sin conocer a priori el número de tags existentes. Así pues, los métodos de identificación pueden ser de dos tipos [Finkenzeller 99, Shih 06, Klair 10]: métodos controlados por tag (*tag-driven*) y los controlados por el lector (*reader-driven*).

- ◇ Controlados por tag (*tag-driven*): que un método esté controlado por el tag implica que se trata de un procedimiento asíncrono ya que no existe un reloj centralizado que gobierna a los participantes, y por tanto el lector no controla la transferencia de datos. Los procesos controlados por tag suelen ser muy lentos e inflexibles. En otro tipo de clasificaciones estos métodos también pueden ser conocidos como *Tag-Talk-First* (TTF).

- ◇ Controlados por el lector (*reader-driven*): la gran mayoría de los procedimientos están controlados por el lector. Éstos se consideran síncronos ya que en este caso es el lector quien controla la comunicación y nadie habla a no ser que sea preguntado por éste. Por ello, también son conocidos como *Reader-Talk-First* (RTF). Los principales protocolos de tipo TDMA se pueden englobar dentro de esta categoría.

A continuación se realiza una clasificación detallada de los 3 principales tipos de protocolos de anti-colisión: métodos estocásticos o basados en ALOHA, métodos deterministas o basados en árbol y métodos híbridos que contienen ambas naturalezas.

2.2 Probabilísticos o basados en Aloha

En los métodos probabilísticos o estocásticos, los tags responden en tiempos generados aleatoriamente. Funcionan de forma efectiva cuando el número de tags no es muy elevado, pero debido a su naturaleza probabilística, es necesaria la estimación del conjunto de tags [Burdet 04, Klair 07a]. Los métodos que conforman esta categoría son: Aloha, slotted Aloha, framed slotted Aloha, dynamic framed slotted Aloha y enhanced framed slotted Aloha.

2.2.1 Aloha (puro) (PA)

Es un protocolo desarrollado en la universidad de Hawaii en 1970 por N. Abramson [Abramson 70]. Aplicado a la tecnología RFID, puede englobarse en la categoría TTF, ya que son los tags quienes toman la iniciativa en la comunicación. Este protocolo permite a los tags transmitir en el momento en que lo deseen, siempre que dispongan de la alimentación proporcionada por el lector. Dada la elevada importancia que adquieren los tags en la comunicación se les suele denominar SuperTags [Microelectronics 11]. En la Figura 2.7 se muestra un ejemplo de detección de 2 tags.

Para que el ID del tag sea transmitido correctamente hasta el lector, los otros tags no deberán transmitir durante al menos dos veces el periodo de transmisión de cada tag. En la Figura 2.7 se observa cómo se produce una colisión parcial si hay

2. Estado del arte

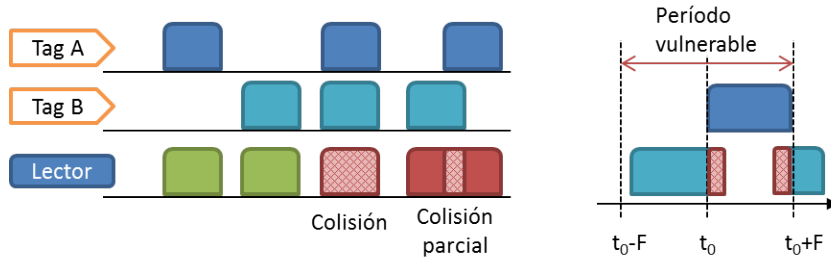


Figura 2.7: Ejemplo de Aloha puro (PA)

otra transmisión entre $t_0 - F$ y $t_0 + F$, siendo F la duración de una transmisión. Cuando se produce la colisión, los tags involucrados esperan un tiempo aleatorio y retransmiten de nuevo.

Dentro del protocolo PA existen una serie de variantes [Burdet 04, Klair 07a]:

- ◇ PA con *Muting*. En esta variante, los tags una vez identificados permanecen en silencio, reduciendo el tráfico a medida que se van identificando.
- ◇ PA con *Slow Down*. En vez de silenciar a los tags identificados, con esta opción los tags aumentan el período con el que envían su ID al lector disminuyendo la probabilidad de colisión.
- ◇ PA con *Fast Mode*. El lector envía un comando de silencio al detectar que un tag está transmitiéndole su ID. Este comando afecta al resto de tags que permanecen a la espera hasta que el lector envía un comando ACK una vez ha recibido correctamente el ID del tag que estaba transmitiendo.
- ◇ Otras variantes. Las variantes arriba mencionadas también han sido combinadas creando otros algoritmos como PA con *Fast Mode* y *Muting*, y PA con *Fast Mode* y *Slow Down*.

2.2.2 Slotted Aloha (SA)

El protocolo Slotted Aloha (SA) presenta una evolución del protocolo PA que mejora el rendimiento de éste. Al igual que el PA, es una adaptación de un protocolo presentado para teoría de redes [Roberts 75]. En el SA los tags transmiten su ID en ranuras limitadas de tiempo, de ahora en adelante *slots*. De esta manera se evitan las

colisiones parciales que se producen en el PA. En la Figura 2.8 se muestra un ejemplo de detección de dos tags. También se observa que el período de vulnerabilidad es ahora de un ciclo o slot, a diferencia del protocolo PA.

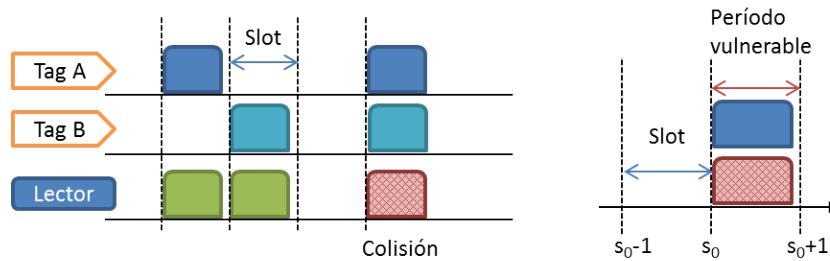


Figura 2.8: Ejemplo de slotted Aloha (SA)

Al protocolo SA también se le pueden aplicar técnicas similares a las presentadas en el PA [Burdet 04, Klair 07a]:

- ◇ SA con *Muting/Slow Down*. Funcionamiento idéntico al explicado en el protocolo PA.
- ◇ SA con *Early End*. Si no se detecta transmisión en un slot, el lector lo cierra e inicia el siguiente para no desperdiciar ese tiempo. Para ello se emplean dos comandos de inicio y fin de transmisión (*SOF* y *EOF*), usándose el segundo para dar por finalizado el slot cuando el lector lo considera necesario.
- ◇ Otras variantes. Se han diseñado otras variantes combinando la característica *Early End* con *Muting* o con *Slow Down*.

2.2.3 Framed Slotted Aloha (FSA)

En los protocolos basados en PA y SA, un tag con período de respuesta alto colisionará más frecuentemente con otros tags que podrían haber transmitido de forma correcta. Los protocolos Framed Slotted Aloha (FSA) obligan a cada tag a transmitir únicamente una vez en cada agrupación de slots, de ahora en adelante *frame*. En [Semiconductors 02], se describe el protocolo comercial *I-Code* que emplea FSA para la identificación de los tags.

2. Estado del arte

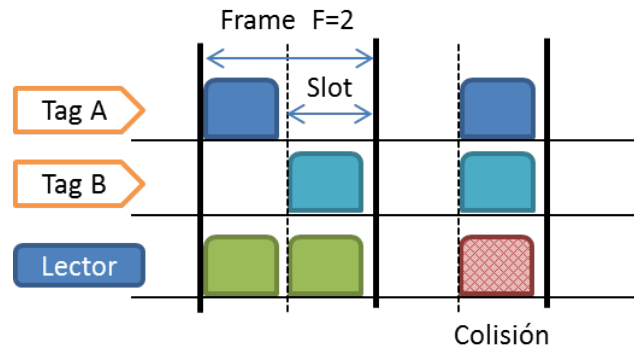


Figura 2.9: Ejemplo de framed slotted Aloha (FSA)

A este tipo de protocolo también se le suele conocer como Basic Framed Slotted Aloha (BFSA) ya que el tamaño del frame permanece fijo a lo largo del proceso de lectura y posee las mismas variantes que sus antecesores aplicando *Muting* y/o *Early End*.

El BFSA sufre de un incremento exponencial del tiempo de identificación cuando el número de tags es superior al tamaño del frame escogido [Cha 05, Cheng 07]. Hwang et al. [Hwang 06] resuelven este problema con una variación del BFSA que limita la respuesta de los tags. El lector envía cadenas de bits y contestan aquellos tags que al compararlo con parte de su ID obtienen un valor menor.

Otra variante presentada por Wang et al. [Wang 07b] es el método Detection and Jump (DJ). El lector define una trama, y los tags contestan un valor de longitud establecida, inferior al de su ID para consumir menos bits, y obtener los slots en los que ha contestado solo un tag. El lector almacena las posiciones donde ha contestado un solo tag y posteriormente, mediante un parámetro S , va saltando de un slot a otro, leyendo los tags que habían contestado correctamente en la primera interrogación. El mismo autor presenta el Extended Detection Jump [Wang 07a], en el que propone emplear el DJ y un método CDMA.

2.2.4 Dynamic framed slotted-Aloha (DFSA)

Como solución al problema del incremento del tiempo de identificación mencionado en [Cha 05, Cheng 07], se aplica a RFID un algoritmo de control de transmisión de paquetes de la teoría de redes desarrollado por Schoute denominado Dynamic

2.2 Probabilísticos o basados en Aloha

Frame Length Aloha [Schoute 83, Wieselthier 89]. La filosofía es similar a la del FSA, aunque en este caso, al comienzo de cada frame el lector comunica a los tags un nuevo tamaño de frame que se ha encargado de estimar previamente, teniendo en cuenta el número de slots colisionados (c), de slots vacíos (i) y de respuestas correctas (s). En la Figura 2.10 se muestra un ejemplo del protocolo DFSA, donde F es el tamaño del frame.

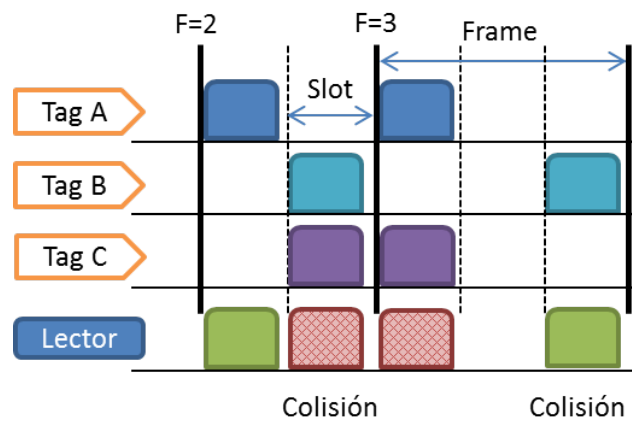


Figura 2.10: Ejemplo de dynamic framed slotted Aloha (DFSA)

El lector emplea la información del número de colisiones, identificaciones y slots libres para determinar una estimación de los tags existentes en la zona de interrogación, y con ello, el tamaño óptimo del frame en cada ciclo de lectura. El tamaño óptimo es aquel que ofrece la máxima eficiencia y el mínimo tiempo de identificación y se da cuando el tamaño del frame coincide con el número de tags existentes [Cha 05]. En general, se emplean dos metodologías para la estimación de los tags.

- ◇ Estimación estática: se basa en el empleo de un multiplicador fijo y obtiene resultados erróneos cuando el número de tags supera el tamaño del frame seleccionado.
- ◇ Estimación dinámica: obtiene el valor estimado por medios estadísticos sin depender de valores fijos, permitiendo una mayor precisión en los resultados.

2. Estado del arte

En la literatura existente se han presentado un gran número de herramientas de estimación [Wang 07c, Klair 10]. Schoute, impulsor de este tipo de métodos, propone en [Schoute 83], una estimación estática basada en el número de colisiones detectadas en un frame ($n_{est} = 2, 39 * c$). Dado que este método se diseñó para la teoría de redes, Cha et al. [Cha 05, Cha 06] lo aplican a RFID empleando *Muting* y Zhen et al., en [Bin 05], proponen incluir los tags identificados en la expresión ($n_{est} = 1, 4*(s+2, 39*c)$). Previamente, Vogt [Vogt 02a, Vogt 02b] había propuesto otro estimador más sofisticado empleando la desigualdad de Chebyshev; el error entre la tupla (i, s, c) y sus correspondientes valores esperados (a_i, a_s, a_c) , eq. 2.1.

$$n_{est}(F, i, s, c) = \min \left| \begin{pmatrix} a_i^{F,n} \\ a_s^{F,n} \\ a_c^{F,n} \end{pmatrix} - \begin{pmatrix} i \\ s \\ c \end{pmatrix} \right| \quad (2.1)$$

Según las pruebas realizadas por Klair et al. [Klair 07b], el método que mayor precisión demuestra en la práctica es el método de Vogt [Vogt 02a, Vogt 02b]. Mientras que cuando el número de tags crece por encima del tamaño del frame, el método propuesto por Cha et al. [Cha 05, Cha 06] es el que mejor se comporta en ese aspecto. Además de éstos, se han desarrollado un gran número de estimadores para el protocolo DFSA [Kodialam 06, Khandelwal 07, Wu 10, Pupunwiwat 10]. No obstante, dado que éste no es el propósito de esta tesis, no se va extender esta información.

Sin embargo, sí que vale la pena mencionar que, frente a la complejidad de estos estimadores [Floerkemeier 06b, Zhu 11], EPCGlobal ha presentado un estándar incluido en la norma ISO 18000C. Se le conoce como el algoritmo Q [EPCglobal 08].

2.2.4.1 El algoritmo Q (SC)

El algoritmo propuesto por EPCglobal, conocido también como Slot Counter (SC), evita la estimación en su primera versión y destaca por su sencillez y reducido coste de computación. Además, dicho coste es independiente del número de tags existentes en la zona de interrogación, al contrario del resto métodos.

Mediante el parámetro Q , un entero entre 0 y 8 que se envía a los tags, se especifica el nuevo tamaño del frame (2^Q) que se puede actualizar en cada slot de lectura. Una vez recibido el parámetro, cada tag escoge un slot para transmitir su

identificador. Cuando el lector analiza la respuesta obtenida, actualiza el parámetro Q , aumentándolo o disminuyéndolo un valor C si se ha producido una colisión o no ha contestado ningún tag respectivamente. En caso de que la respuesta haya sido satisfactoria, el valor de Q no varía. En la Figura 2.11, se muestra el esquema de actualización de este parámetro.

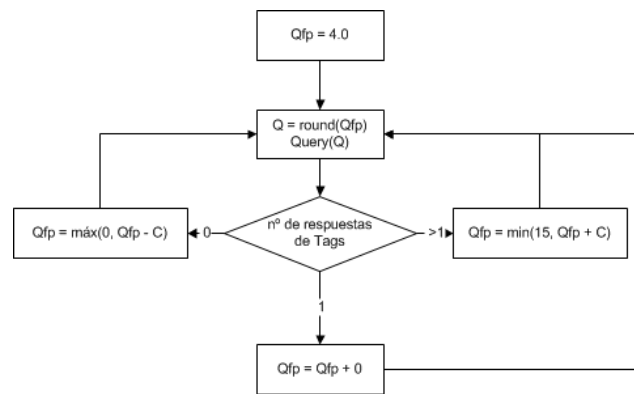


Figura 2.11: Ajuste del parámetro Q

Dando valores del orden de $[0, 1 - 0, 5]$ al parámetro C , el algoritmo ofrece mejores resultados en slots consumidos y eficiencia que los métodos DFSA con estimadores [Namboodiri 12]. También se ha propuesto la utilización de la expresión $0, 8/Q$ [Wang 06, Helmut 09], ya que si C toma un valor muy elevado el algoritmo converge rápido pero se producen oscilaciones alrededor del punto óptimo. Por otro lado, si C toma un valor pequeño, converge lentamente hacia dicho óptimo. Además de esto, numerosos autores en la literatura proponen diferentes estrategias para calcular ese parámetro C [Bueno-Delgado 09]:

- ◇ Lee et al. optimizan el algoritmo Q sustituyendo C por C_i para compensar Q con los slots libres y C_c con las colisiones [Lee 07].
- ◇ Kim y Kim proponen, además de actualizar Q de la forma tradicional, añadir un comando *QueryAdjustPlus* que permite ampliar el frame de una forma más afinada. Es decir, si se desea modificar el frame mediante la Q , la única manera es multiplicarlo por 2 o dividirlo entre 2 ya que el tag interpreta el valor como 2^Q . El autor propone enviar otro parámetro P , que se suma

2. Estado del arte

al calculado con Q . De esta forma la flexibilidad del frame es mayor y se mejora la eficiencia del protocolo origen [Kim 11].

- ◇ Fan et al. proponen en [Fan 08b, Fan 08a] estimar el tamaño del frame mediante la desigualdad de Chebyshev empleada por Vogt en la eq. 2.1. Con el valor estimado, los autores calculan el tamaño del frame y ajustan Q para enviárselo a los tags solo cuando se ha completado la transmisión de un frame completo. Si se da el caso de que Q necesita ser actualizado y no se ha transmitido el frame al completo, la Q se actualiza empleando el método tradicional comentado anteriormente.
- ◇ Floerkemeier propone aplicar la estimación Bayesiana al cálculo del valor de Q en [Floerkemeier 06a, Floerkemeier 07]. Este método, empleado después de cada slot, ofrece buenos resultados en rendimiento. Sin embargo, el costo computacional es muy alto.

Una de las limitaciones de las diferentes variantes de DFSA es el acotamiento del tamaño del frame a 256 slots ([Vogt 02a, EPCglobal 08]) o 512 ([Center 03]). Si el número de tags excede este límite, se pueden producir colisiones persistentes que dificultan mucho la identificación completa de los tags. Además, los estimadores sufren de un alto costo computacional, y algunos no pueden ofrecer resultados muy exactos. Para solucionar este problema se propone el siguiente protocolo.

2.2.4.2 Enhanced Framed Slotted Aloha (EDFSA)

El protocolo Enhanced Framed Slotted Aloha (EDFSA), propuesto por Lee et al. [Lee 05], divide los tags existentes en M subgrupos si el total de éstos supera el tamaño máximo del frame. Lee propone una tabla de tamaños de frame con la que se consigue una eficiencia máxima. Una vez se han dividido los tags en los subgrupos correspondientes, el lector interroga a cada grupo por separado empleando el tamaño del frame más adecuado.

También se han propuesto una serie de modificaciones para mejorar el protocolo:

- ◇ Peng et al. proponen el Variant Enhanced Framed Slotted Aloha (VEDFSA) [Peng 07]. Los autores indican que en el EDFSA, una vez establecidos los grupos de tags, cada uno de ellos se identifica empleando DFSA. También proponen mejorar la eficiencia ajustando esos grupos de forma dinámica.
- ◇ Wasikon y Suradi mejoran el VEDFSA con su Hasten Dynamic Frame Slotted Aloha (HEDFSA) [Wasikon 08] asignando una posición de slot a los tags que no han sido leídos.

2.2.5 Comparación de los métodos probabilísticos

Los protocolos de anti-colisión explicados presentan numerosas similitudes con los métodos de multi-acceso en redes. Sin embargo, los recursos limitados de los tags RFID añaden una serie de inconvenientes al problema de las colisiones: deben ser capaces de trabajar consumiendo lo menor posible; el número de tags a detectar es desconocido; la detección de colisiones es complicada, ya que las señales de los diferentes tags llegan con potencias muy diferentes en función de la distancia; y finalmente, los tags no pueden comunicarse entre ellos.

En los métodos probabilísticos los tags responden en momentos aleatorios, que dependen del control que tenga el lector sobre los tags y la naturaleza de su algoritmo. Sarma en [Sarma 03] ha establecido una serie de parámetros que pueden definir el rendimiento de un algoritmo. Éstos son: la velocidad de lectura de los tags, el ancho de banda de la señal del lector, el ancho de banda de la señal de retorno, la cantidad de información de estado almacenable en cada tag, la tolerancia del algoritmo al ruido, el costo de realización del tag, el costo de realización del lector, la tolerancia a tags que entran y salen del proceso de detección, y el rango en el que los tags pueden ser leídos. Comparando los métodos probabilísticos y teniendo en cuenta algunas de estas características se obtiene la Tabla 2.1.

El protocolo PA proporciona el diseño más simple de lector, donde se requiere de un generador de números aleatorios (RN), que defina los momentos en los que transmiten los tags mediante el circuito de transmisión/recepción (TX/RX). Aunque obtiene, también, el rendimiento más bajo. Con la aparición de los slots en el SA, el rendimiento aumenta, al igual que las exigencias en el costo de los tags. Este último requiere de un circuito de sincronización para ajustar las respuestas de los

2. Estado del arte

	PA	SA	FSA	DFSA
Características	Transmite en tiempos aleatorios	Transmite en slots sincronizados	Transmite 1 vez cada F slots	Igual que BFSFA pero con F variable
En colisión	Espera aleatoria	Espera x slots aleatorios	Espera F slots	
RTF/TTF	TTF	RTF	RTF	RTF
Rendimiento	18.4 % [Finkenzeller 99]	36.8 %	42.6 % [Schoute 83]	
Desventajas	si tags $\uparrow \Rightarrow$ cols $\uparrow \uparrow$	si tags $\uparrow \Rightarrow$ cols $\uparrow \uparrow$ Sincronización	Tags necesitan F Lector-Tag	Lector sofisticado Cálcs. avanzados
Costo del tag	Generador RN Temporizador y TX/RX Circ. Sincro		Gen. RN y circ. sincro. Memoria comandos TX/RX	
Costo del sistema	Muy Bajo	Bajo	Alto	Muy Alto
Complejidad del protocolo	Muy Bajo	Bajo	Alto	Muy Alto

Tabla 2.1: Comparativa de los métodos probabilísticos

tags a los tiempos delimitados por los slots. El DFSA ofrece el mejor rendimiento, especialmente en su variante EDFSA para entornos con alta densidad de tags, aunque demanda un mayor costo tanto de lector como de tags, por la inclusión de memoria.

El tiempo de identificación del conjunto de tags necesario de los protocolos PA y SA es muy alto, sin embargo, éste mejora considerablemente empleando las técnicas de *fast mode*, *early end* y *muting*, afectando muy positivamente al protocolo PA y reduciendo su tiempo de identificación hasta ser el más rápido de este conjunto. En el caso de los protocolos FSA, el BFSFA es el que muestra el mayor tiempo de identificación, y el DFSA, al igual que el EDFSA, muestran to-

2.3 Deterministas o basados en Árbol Binario

do su potencial cuando se les aplican las modificaciones de *muting* y *early end* [Klair 10, Klair 07c, Klair 07a].

Dado que los métodos basados en Aloha son probabilísticos, todos ellos adolecen del *tag starvation problem*. Este problema provoca que haya ciertos tags que no sean identificados nunca por no encontrar slots libres o colisionar en todos sus intentos [Yang 10, Yan 10, Zhu 11].

2.3 Deterministas o basados en Árbol Binario

Los métodos basados en árbol binario nacen de los algoritmos diseñados por Capetanakis [Capetanakis 79], donde realiza un estudio exhaustivo de los algoritmos de árbol de búsqueda. Éstos han sido analizados en profundidad en [Massey 80, Kaplan 85, Yan 11a]. Los nodos transmiten paquetes en instantes de tiempo concretos cuando son solicitados por el nodo receptor. Si se produce una transmisión simultánea de dos o más nodos en un mismo instante, se produce una colisión y el algoritmo de árbol crea una división en dos subconjuntos de los nodos causantes. El primer subconjunto transmite a continuación, mientras que el segundo espera a que se resuelva el primero. Si cualquiera de los dos subconjuntos se vuelve a encontrar con otra colisión, se repite el mismo proceso recursivamente hasta que todos los nodos transmiten la información solicitada sin impedimentos. Este tipo de algoritmos han sido aplicados a la tecnología RFID de forma muy efectiva. Los nodos que transmiten se corresponden con los tags, y el receptor que solicita, con el lector. Los tags envían sus identificadores (ID) en respuesta a las peticiones o *queries* del lector. La resolución de todas las colisiones implica que todos los tags han sido identificados. Los mensajes de los tags se envían en slots o ranuras de tiempo. En el caso de los protocolos basados en árbol, un slot determina un período de tiempo variable, en el que el tag es interrogado, responde con su ID y espera la confirmación del lector. La gran ventaja de la utilización de estos métodos es que, a nivel de protocolo, todos los tags interrogados por el lector RFID van a ser identificados, independientemente del número. Por ello, estos algoritmos son denominados deterministas.

La división de los subconjuntos que se produce en los nodos donde existe una colisión se realiza empleando diferentes métodos en función del tipo de protocolo

2. Estado del arte

empleado. Los principales protocolos basados en árbol binario son: Tree Splitting (TS), Query Tree (QT), Bit-Tracking y el Bitwise arbitration (BTA) [Finkenzeller 99, Shih 06, Klair 10].

2.3.1 Tree Splitting (TS)

Aplicado a RFID por Hush y Wood [Hush 98], el protocolo Tree Splitting (TS), de tipo *reader-driven* o controlado por lector, emplea un árbol virtual para organizar e identificar cada uno de los tags contenidos en la zona de interrogación del lector. Cada vez que se produce una colisión entre un grupo de tags se crea una división en B subconjuntos (siendo $B > 1$). Estos subconjuntos se hacen cada vez más pequeños a medida que se van produciendo colisiones, hasta que llega un punto en el que queda un único tag en el subgrupo y es posible identificarlo. El lector interroga consecutivamente a cada uno de estos subgrupos, permaneciendo a la espera los tags que están fuera del subgrupo elegido. Cada vez que el algoritmo alcanza una hoja del árbol, es decir, un subgrupo con un único tag, lo identifica y pasa a resolver los subgrupos que se encuentran a la espera en orden *last-in first-out* (LIFO).

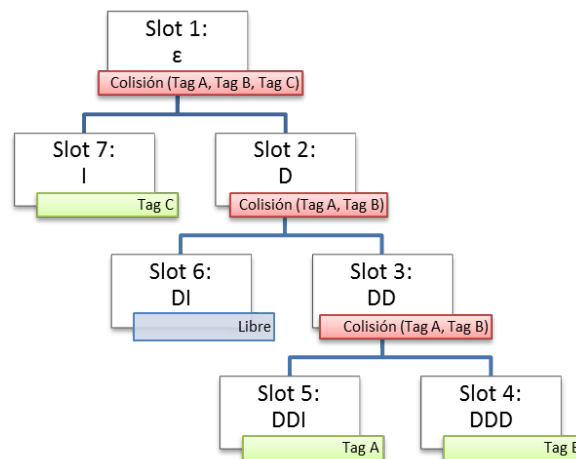


Figura 2.12: Ejemplo del algoritmo TS con $B = 2$

En [Bertsekas 92] Bertsekas et al. implementan un contador interno en los tags y éstos lo van modificando en función de las indicaciones del lector, enviando su ID

2.3 Deterministas o basados en Árbol Binario

	Operación	Nodo	Tag A	Tag B	Tag C
Slot 1	Reset Contadores Colisión / Tag C + 1	ε	0	0	0
Slot 2	Colisión / Tag C + 1	I	0	0	1
Slot 3	Colisión / Tag B + 1 Tag C + 1	DD	0	0	2
Slot 4	Tag A ident / Tag B - 1 Tag C - 1	DDD	0, ID	1	3
Slot 5	Tag B ident / Tag C - 1	DDI	-	0, ID	2
Slot 6	Slot libre / Tag C - 1	D	-	-	1
Slot 7	Tag C ident	L	-	-	0, ID

Tabla 2.2: Ejemplo de ejecución del protocolo TS

siempre que su contador valga 0. El protocolo se inicia reseteando los contadores de los tags (comando ε) y cada vez que el lector recibe los datos de los tags, éste les contesta el estado de la transacción. Los tags involucrados en una colisión suman un número aleatorio entre 0 y 1 a su contador, mientras que el resto suman 1. Si se ha producido una respuesta única y satisfactoria o si no ha contestado nadie, los tags disminuyen en 1 su contador con la confirmación del lector. En la Figura 2.12 se muestra un ejemplo que viene detallado en la Tabla 2.2.

Como variaciones de este método se presentan:

- ◇ Adaptive Binary Splitting (ABS) [Myung 06b, Myung 06a, Myung 07]: esta variación se centra en mejorar las lecturas posteriores al primer ciclo de interrogación. Los tags del ABS poseen un contador adicional al de los tags del TS y el lector un contador de terminación. El primer ciclo de lectura es idéntico al del TS. El segundo ciclo aprovecha la información aprendida en el primero para, no solo reducir las colisiones, sino también los slots libres. Al final del último ciclo realizado, cada tag posee un valor de contador diferente, lo que facilita mucho las posteriores lecturas, eliminando al menos las colisiones entre las etiquetas existentes.
- ◇ Parallel Splitting (PS) [Yeh 07, Yeh 11]: los autores proponen una variación del protocolo TS. En el ciclo de lectura hasta que se identifica el primer tag,

2. Estado del arte

cada vez que se produce una colisión, los tags desplazan su bit de contador a la izquierda (multiplican por 2) y añaden un 0 o 1. De esta forma los contadores se expanden en mayor medida y se reduce el número de colisiones. Una vez identificado el primer tag, se procede a la identificación del resto empleando el protocolo TS.

- ◇ Fast Tree Traversal Protocol (FTTP) [Choi 10]: Choi et al. aplican la estimación MLE, *Maximum Likelihood Estimation*, y con ella ajustan un parámetro k . El protocolo se ejecuta como el TS hasta que identifica el primer tag. Este proceso cubre los nodos izquierdos del árbol. A continuación, el protocolo retrocede por los nodos derechos de forma descendente. Mediante la estimación MLE, el protocolo estima que en ese nodo tiene que haber el mismo número de tags que en su nodo hermano, es decir, el número de tags identificados hasta ese momento. Con ese dato calcula k y define el número de bifurcaciones del árbol en el siguiente nivel. Posteriormente, si se produce una colisión en alguna de esas bifurcaciones, las resuelve empleando el protocolo TS. Este protocolo reduce el tiempo de identificación y el número de colisiones.

Dado que existe un componente probabilístico en este tipo de protocolos, puede llegar a producirse un problema similar al del *tag starvation* de los protocolos basados en Aloha [Hush 98, Abraham 02]. Puede ocurrir que dos tags escojan el mismo número aleatorio varias veces aunque la probabilidad de que se dé esta situación es muy baja. De cualquier manera, para evitar este tipo de problema, el protocolo debe usar identificadores únicos en la resolución de las colisiones, eliminando la aproximación probabilística del número aleatorio. Un método que se basa en esta estrategia es el que se emplea en la siguiente sección.

2.3.2 Query Tree (QT)

El protocolo Query Tree (QT), también conocido como protocolo *memoryless* o sin memoria, se refiere a que los tags no necesitan almacenar datos en variables o contadores para la ejecución del protocolo, como es el caso de los protocolos TS. Esa es la gran ventaja de este algoritmo diseñado por Law [Law 00], que identifica

2.3 Deterministas o basados en Árbol Binario

los tags jerárquicamente y adquiere la forma de un árbol. El tag no tiene en cuenta tramas ni estados anteriores o futuros, únicamente la petición actual del lector. Al igual que el TS, se trata de un método *reader-driven* y por tanto, es el lector quien envía prefijos denominados *queries* a los tags de su rango de lectura. Inicialmente, el lector envía un *query* ε al que contestan todos los tags y posteriormente, se forman cadenas de bits $q_1, q_2 \dots q_{L-1}, q_L$ donde $q_i \in \{0, 1\}$ y L representa su longitud. Las probabilidades de colisión en los primeros *queries* suelen ser altas ya que están formados por pocos bits. Los tags cuyos identificadores coincidan con el *query* enviado contestan su ID al lector. Si existen varios tags, contestan al mismo tiempo y se produce una colisión. El lector detecta dicha colisión y añade otro bit (q_{L+1}) más al *query* creando una bifurcación, $q_1, q_2 \dots q_{L-1}, q_L, 0$ y $q_1, q_2 \dots q_{L-1}, q_L, 1$. De esta forma, el lector ahora debe enviar dos tramas de la misma manera que lo hizo con la anterior. En caso de obtener una respuesta de uno de los tags sin interferencias, da por finalizada esa rama del árbol y continúa con las restantes. Por otro lado, si envía un *query* que no obtiene respuesta, el lector interpreta que en esa rama no existen más tags y también la abandona. De esta manera, solo se forman bifurcaciones en caso de producirse una colisión, repitiendo este proceso hasta que todos los tags quedan unívocamente identificados. Cada ciclo satisfactorio, el lector comunica a los tags el ID del tag identificado para que éste no conteste más hasta el siguiente ciclo. A continuación se muestra en la Figura 2.13 un ejemplo de detección de cuatro tags mediante este protocolo y en la Tabla 2.3, el procedimiento seguido.

El propio Law [Law 00] propone una serie de modificaciones para reducir el tiempo de identificación del conjunto de tags del protocolo:

- ◇ Atajando (*Short cutting*): esta mejora consiste en evitar nodos en los que se prevea que va a haber una colisión, “atajando” y evitando ciertos *queries* que producirían más retardo. Considerando un nodo en el que se ha producido una colisión, el protocolo QT añadiría un 0 y un 1 al *query* enviado. Y seguidamente procedería al envío del primero generado q_0 , siendo q el *query* enviado. Si se da el caso de que ningún tag contesta a ese prefijo, se puede deducir que en q_1 van a existir más de dos tags y por tanto se va a producir

2. Estado del arte

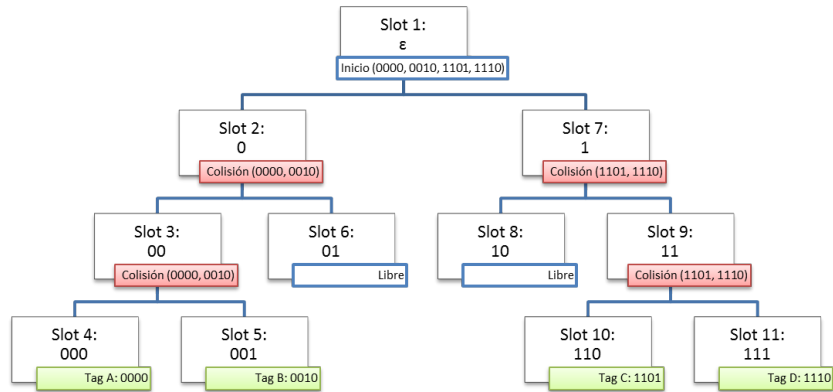


Figura 2.13: Ejemplo de identificación de 4 tags empleando el protocolo QT

otra colisión. En consecuencia, el método se salta ese prefijo y añade otro 0 y 1 a q_1 , generando dos nuevas *queries* (q_{10} , q_{11}) donde antes habría uno.

- ◇ Avance agresivo: en esta técnica cada *query* se extiende en más de 1 bit cada ciclo. Por ejemplo, en una colisión generaría cuatro *queries* en vez de dos (q_{00} , q_{01} , q_{10} , q_{11}). Generalmente suele emplear más slots que la versión QT simple. El árbol pasa de ser binario o 2-ario, a 3-ario, 4-ario ó k -ario.
- ◇ Categorización: si el lector tiene información extra acerca de los tags, pueden subdividirse en grupos, por ejemplo de tags cuyos prefijos de ID sean similares, de forma que el lector identifique cada grupo de forma independiente.

Otros autores también han propuesto numerosas variantes del algoritmo QT. Son las mostradas a continuación:

- ◇ Adaptive Query Tree (AQT): es una variación del algoritmo QT que presenta ventajas de rapidez en la detección de todos los tags del rango de la antena del lector. Desarrollado por Myung et al. [Myung 06b, Myung 06c, Myung 07], se basa en almacenar en memoria el conjunto de *queries* detectados en el ciclo de lectura anterior. El sistema puede evitar que se produzcan colisiones entre los llamados *staying tags*, es decir, los tags que no han variado de un

2.3 Deterministas o basados en Árbol Binario

	Query	Pila Lector	Respuesta Tags
Slot 1	Query vacío	[-]	Colisión
Slot 2	0	[1]	Colisión
Slot 3	00	[01,1]	Colisión
Slot 4	000	[001,01,1]	Tag A Ident.
Slot 5	001	[01,1]	Tag B Ident.
Slot 6	01	[1]	Vacía
Slot 7	1	[-]	Colisión
Slot 8	10	[11]	Vacía
Slot 9	11	[-]	Colisión
Slot 10	110	[111]	Tag C Ident.
Slot 11	111	[-]	Tag D Ident.

Tabla 2.3: Identificación mediante el protocolo QT

ciclo de lectura a otro. Solo se producirían dichas colisiones en caso de que aparezca en el escenario un nuevo tag con un identificador que comparta *query* con alguno de los existentes. En ese caso se producen una serie de situaciones que el protocolo resuelve de forma que los *queries* innecesarios para la detección de cada tag son eliminados y el resto almacenados para el siguiente ciclo de lectura. AQT reduce las colisiones pero requiere de slots libres que solo consumen energía sin detectar ningún tag nuevo. De cualquier manera, estos ciclos libres son útiles y necesarios para detectar todos los tags existentes en su rango de lectura, ya que si no el algoritmo solo podría detectar los nuevos tags que colisionarán con los existentes.

- ◇ Enhanced Binary Query Protocol (EBQP) [Yan 09, Yan 11b]: durante el proceso de identificación, el protocolo obtiene datos de los ID de los tags. Esto contribuye a poder dividirlos en grupos y utilizar el QT para identificar cada uno de ellos, reduciendo las colisiones y evitando posibles *queries* que habrían sido innecesarios.
- ◇ Smart Trend Traversal (STT) [Pan 09, Pan 11]: este protocolo va generando los sucesivos *queries* en base a una serie de reglas que definen los autores.

2. Estado del arte

Estas reglas generan indicaciones en base a las respuestas recibidas a lo largo del proceso. En función de cómo transcurre el proceso de identificación, el lector indica si el *query* debe ampliarse con 0, 1 o desecharse. De esta forma va estableciendo un camino de identificación directa, reduciendo las colisiones.

También se han propuesto diferentes protocolos de tipo árbol k -ario, en el que los *queries* se generan de forma más agresiva.

- ◇ *k*-ary Tree-based Anti-collision Scheme (k-TAS) [Chen 11]: el autor utiliza información adicional obtenida de las colisiones, como cuál ha sido la causa de dicha colisión, para reducirlas y evitar los slots libres. Para ello, los tags añaden a la respuesta de su identificador un código generado a partir de su ID. El lector emplea los códigos recibidos para calcular el valor de k y generar un árbol k -ario en cada colisión, el cual favorece la reducción de las colisiones.
- ◇ Unified Q -ary tree [Pupunwiwat 09]: Pupunwiwat et al. proponen un protocolo Q -ario diseñado de forma empírica, que obtiene mejores comportamientos que el QT. Los autores proponen utilizar dos tipos de protocolos Q -arios y realizar combinaciones de, por ejemplo, un árbol 4-ario con un 8-ario, que se modifique a lo largo del ciclo de interrogación. De esta forma, no solo reduce las colisiones, sino también los slots libres, causantes del aumento del tiempo de identificación en este tipo de protocolos.

Dado que los protocolos basados en el QT utilizan el ID de los tags para realizar la identificación, también son muy sensibles a las distribuciones que éstos presentan. Es decir, que dependiendo de cómo sean los ID del conjunto de los tags presentes en la zona de interrogación, el protocolo QT se va a comportar de diferente manera.

Gracias a la estandarización de la tecnología RFID (ver Sección 2.5), se establecen diferentes campos dentro del ID de cada tag [EPC-Global 11] para indicar datos como: la empresa fabricante, tipo de producto, serie de fabricación, etc. Existen protocolos que sacan provecho de esta característica de los tags:

2.3 Deterministas o basados en Árbol Binario

- ◇ Sidewalk [Koh 08]: este protocolo está diseñado para sacar partido de las partes comunes de los IDs de los tags de la zona de interrogación. El lector avanza en la generación de *queries* hasta la lectura del primer tag, de manera que ha profundizado por la rama más externa del árbol. Después de esto, comienza a generar *queries* de esa longitud, avanzando de forma *horizontal*, es decir, por el mismo nivel de dicho árbol. En caso de producirse algún slot libre, el lector disminuirá el nivel, generando *queries* de menor longitud.
- ◇ Knowledge Query Tree (KQT) [Bonuccelli 07b]: el protocolo presentado por Bonuccelli et al. emplea el conocimiento adquirido del proceso de identificación, su número aproximado y el tipo de distribución de datos estadísticos de la comunicación. De esta forma saca partido del conocimiento de los ID de los tags acelerando el proceso en las sucesivas identificaciones.
- ◇ Intelligent Query Tree (IQT) [Bhandari 06]: este protocolo acelera el proceso de identificación de tags con prefijos comunes. El protocolo almacena los *queries* que más se han empleado y explota los patrones de los identificadores de los tags, como el vendedor y el tipo de producto. Esto implica que el lector identifica primeramente un prefijo común que ignora en sucesivas lecturas. Por tanto, este método necesita de varios ciclos de identificación para resultar efectivo.
- ◇ El protocolo propuesto en [Yeo 10], actúa suponiendo que los IDs de los tags tienen una estructura concreta. Yeo et al. comienzan la identificación de los tags por el bit de menor peso, *Least Significant Bit* (LSB), en lugar de empezar de por el de mayor peso, *Most Significant Bit* (MSB), como realizan el resto de protocolos. Solicita a los tags que envíen un tamaño de ID concreto que posteriormente varía en función del tipo de slot recibido. El protocolo está basado en el QT, y solo ofrece buenos resultados cuando los bits de mayor peso de los ID son constantes.

2.3.3 Protocolos Bit-tracking

La tecnología bit-tracking, también llamada *collision tracking*, permite al lector conocer la posición exacta en la que se ha producido una colisión, es decir, que puede

2. Estado del arte

trazar la colisión al bit exacto en el que ha sucedido. Para ello emplea la codificación Manchester [Finkenzeller 99]. Esta codificación define el valor del bit como una transición de voltaje. Así, un 0 se representa como una transición positiva y un 1 como una negativa. Una colisión ocurre cuando dos o más tags transmiten diferentes bits y se cancelan entre sí. Aunque un estado de *no-transición* no está permitido en la normativa de la codificación Manchester, puede emplearse para trazar la posición de la colisión a un bit de forma individual.

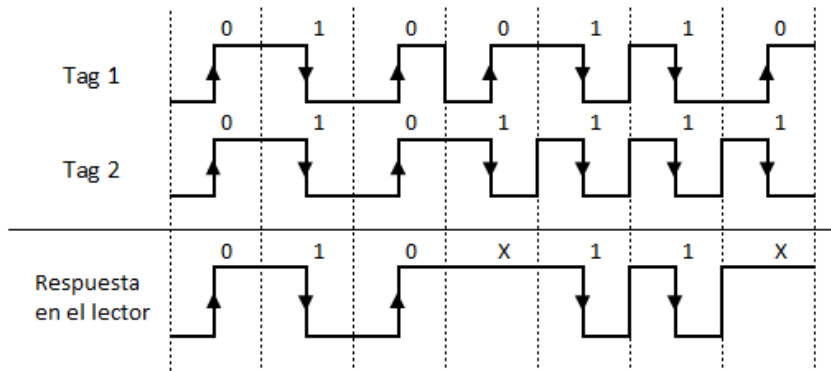


Figura 2.14: Ejemplo de la codificación Manchester

En la Figura 2.14 se muestra un ejemplo de la utilización de la codificación Manchester en la transmisión de dos tags, y de cómo el lector es capaz de conocer el bit específico donde se ha producido la colisión. Dos tags envían sus IDs (0100110; 0101111). La señal recibida por el lector (010X11X) indica con *X* el bit que no cumple las reglas de la codificación Manchester, es decir, el bit que ha colisionado, en este caso el bit 4 y el bit 7.

La tecnología bit-tracking, por tanto, requiere que todos los tags contenidos en la zona de interrogación transmitan sus datos de manera simultánea. Esta asunción resulta razonable ya que el tiempo de propagación es insignificante. La información obtenida en cada colisión permite avanzar de forma más agresiva a la hora de componer los nuevos queries, contribuir a reducir las colisiones adicionales que se pudieran producir y finalmente, disminuir el tiempo de identificación del protocolo. Los principales protocolos de bit-tracking existentes en la literatura son versiones de las categorías TS o QT, a los que se les ha aplicado la tecnología presentada.

2.3 Deterministas o basados en Árbol Binario

- ◇ El protocolo Collision Tree (CT) presentado inicialmente por Zhou et al. en [Zhou 03] y analizado en mayor profundidad por Jia et al. en [Jia 10, Jia 12] es un protocolo en el que se aplica bit-tracking al QT. CT emplea la codificación Manchester para trazar la posición de la colisión. El lector envía un *query* $q_1, q_2 \dots q_L$ de longitud L y aquellos tags cuyo ID coincide responden. El lector recibe la respuesta de los tags $p_1, p_2 \dots p_c \dots p_k$ y la analiza bit a bit, donde k denota la longitud del ID. Si se produce una colisión, una vez conocida la posición (p_c) en la que se ha producido, el lector genera dos queries: $q_1, q_2 \dots q_L, p_1, p_2 \dots p_{c-1}0$ que obtendrá respuesta de uno de los tags colisionados y $q_1, q_2 \dots q_L, p_1, p_2 \dots p_{c-1}1$ que la obtendrá del otro. Todos los queries generados obtendrán respuesta ya que los genera a partir de un bit colisionado, es decir, un bit en el que dos o más tags han transmitido 0 y 1; posibilidades cubiertas con los dos nuevos queries generados. Este protocolo consigue reducir las colisiones a $2n - 1$ de forma determinista, siendo n el número de tags presentes en la zona de interrogación. Y además consigue erradicar por completo los slots libres.

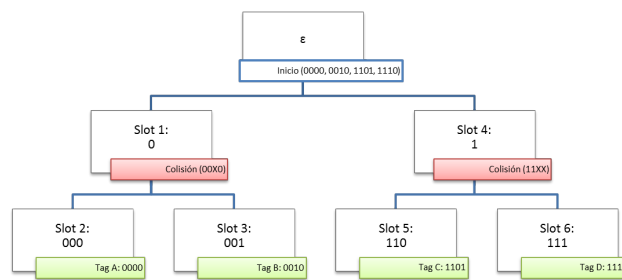


Figura 2.15: Ejemplo de identificación de 4 tags empleando el protocolo CT

En la Figura 2.15 se muestra un ejemplo de identificación de cuatro tags y el procedimiento viene desglosado en la Tabla 2.4. Cabe destacar que el conjunto de tags a identificar coincide con el mostrado en el ejemplo del protocolo QT (ver Figura 2.13, Tabla 2.3). En este caso se emplean 6 slots para identificar los mismos cuatro tags, mientras que el protocolo QT emplea

2. Estado del arte

	Query	Pila Lector	Respuesta Tags	Tipo de Slot
Slot 1	0	[1]	0X	Colisión
Slot 2	000	[001,1]	0	Tag A (0000)
Slot 3	001	[1]	0	Tag B (0010)
Slot 4	1	[-]	1XX	Colisión
Slot 5	110	[111]	1	Tag C (1101)
Slot 6	111	[-]	0	Tag D (1110)

Tabla 2.4: Identificación mediante el protocolo CT

11. Se observan las principales características que ofrece la tecnología bit-tracking: el avance agresivo en la generación de queries y la eliminación completa de slots libres.

- ◇ Enhanced Anti-collision Algorithm (EAA) [Chen 07]: Chen et al. aplican bit-tracking al protocolo TS. Conociendo el bit en el que se produce la colisión, el lector envía un puntero a los tags indicando dicha posición para que únicamente transmitan los bits siguientes. Posteriormente, el resto de bits se identifican empleando el protocolo ABS.
- ◇ New Enhanced Anti-collision Algorithm (NEAA) [Chen 10]: este protocolo es una variación del EAA presentado anteriormente. Las principales diferencias son que el protocolo divide el conjunto de tags en b subconjuntos, en función del número de 1s que disponen sus IDs. Una vez ordenados los tags, el protocolo desarrolla el slot *M-readable* capaz de identificar varios tags en un único slot. Sabiendo que el número de bits 1 en la respuesta es uno, si el lector recibe $X0XX$, las respuestas posibles son 1000, 0010 y 0001. Este protocolo genera un gran número de slots libres, pero reduce las colisiones y los slots satisfactorios.
- ◇ Optimal Query Tracking Tree (OQTT) [Lai 13]: El protocolo presentado por Lai et al. añade un proceso de estimación previa al mismo proceso de identificación del CT. Los tags envían una cadena de bits de la longitud de su ID,

2.3 Deterministas o basados en Árbol Binario

k. Después calculan el número de tags empleando la desigualdad de Chebyshev y el método de Vogt [Vogt 02a], y posteriormente, el lector obtiene el número óptimo de queries necesarios que, una vez generados, almacena en una pila. A continuación realiza la identificación siguiendo las pautas del CT. Este método genera una eficiencia muy alta y reduce el tiempo de detección del conjunto de tags.

Además de la aplicación del bit-tracking a protocolos ya existentes, también existen otro tipo de protocolos que emplean la codificación Manchester para su beneficio, los protocolos Binary Search (BS).

2.3.3.1 Binary Search (BS)

En los protocolos Binary Search (BS) [Finkenzeller 99], el lector envía un número de la misma longitud que el ID de los tags, que éstos comparan con el suyo propio. Aquellos cuyo ID es igual o inferior al número enviado contestan al lector. Éste, usando la codificación Manchester, escucha las respuestas de los tags, y cuando se da una colisión, separa en dos subgrupos los tags colisionados basándose en los bits colisionados.

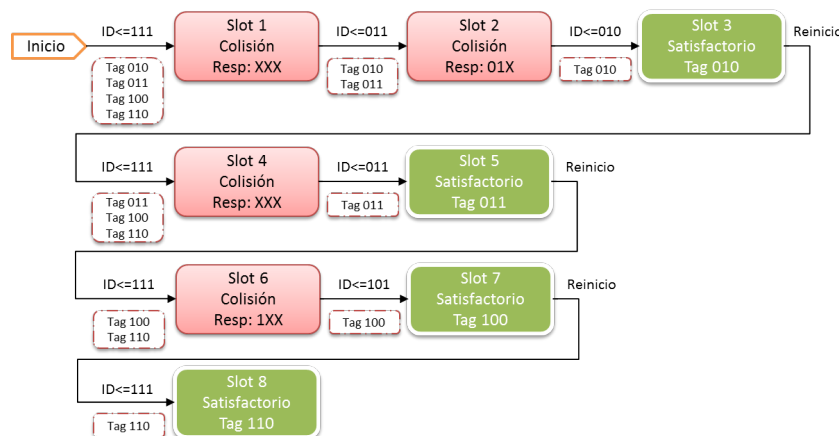


Figura 2.16: Ejemplo de detección de 4 tags mediante el protocolo BS

Como se puede observar en la Figura 2.16, inicialmente el lector comienza leyendo los tags con el mayor valor del ID, 111 en este caso. Los tags con su ID igual o inferior a ese valor contestan, produciéndose una colisión en los 3 bits. El

2. Estado del arte

lector envía otro número sustituyendo el primero de los bits colisionados por 0, es decir 011. La respuesta de los tags obtenida a esta solicitud es 01X, habiéndose producido una colisión en el último bit. Procediendo de la misma manera, el lector sustituye el último bit por 0, quedando 010 e identificando así el primero de los tags. Y seguidamente, reinicia el proceso volviendo a usar el mayor número posible, 111. El proceso se repite hasta que todos los tags quedan identificados.

A continuación, se presentan algunas de las variantes existentes del protocolo en cuestión:

- ◇ Improved Enhanced Binary Search Algorithm (IEBSA): presentada por Liu et al. en [Liu 05]. Este algoritmo identifica dos tags simultáneamente cuando solo se produce la colisión de un bit en las respuestas de los tags al número enviado por el lector.
- ◇ Dynamic Binary Search Algorithm (DBSA) [Finkenzeller 99]: en esta versión del algoritmo, el lector y los tags no emplean la totalidad de la longitud de sus ID, únicamente envían los bits existentes a partir de la posición donde se ha producido la colisión. Es decir, si el lector recibe una respuesta 01X, los tags solo necesitan enviar el último bit de su ID, disminuyendo considerablemente la cantidad de datos intercambiados entre lector y tags.
- ◇ Backtracking Binary Search (BBS) [Jianfang 10]: el protocolo presentado por Wang propone un estado para los tags, *adormecido*. Cuando se produce una colisión, el grupo de tags que colisionó con un 1 quedan en ese estado y continúa la identificación con el otro grupo. Una vez identificados, el protocolo retrocede hasta la anterior colisión para despertarlos y continuar el proceso.

2.3.4 Bitwise arbitration (BTA)

Los protocolos de tipo Bitwise arbitration (BTA) solicitan a los tags que respondan su ID bit a bit, comenzando por el de mayor peso (MSB) hasta llegar al de menor peso (LSB). Partiendo de la premisa de que las respuestas de los tags están sincronizadas, estos protocolos permiten al lector avanzar teniendo en cuenta si se produce o no colisión. En caso de no producirse, el bit recibido forma parte del

2.3 Deterministas o basados en Árbol Binario

ID de todos los tags que han transmitido en ese momento y salta al siguiente. Sin embargo, en caso de colisión, se crea una bifurcación en el árbol para las dos posibilidades 0 y 1. Los principales algoritmos basados en BTA son: el ID-Binary Tree Stack (ID-BTS), el Bit-by-Bit (BBT) y el Bit Query (BQ).

- ◇ ID-Binary Tree Stack (ID-BTS) [Feng 06]: construye un árbol binario con el mismo número de niveles que la longitud del identificador de los tags. Los tags transmiten su identificador bit a bit y en caso de producirse una colisión por envío simultáneo de un bit 0 y 1 produce una bifurcación de dos caminos formando un nodo. De manera que el camino del nodo raíz a un nodo hoja representa el ID de uno de los tags identificados. El lector emplea una pila para almacenar las posiciones del árbol donde se producen las bifurcaciones, pudiendo volver a esa posición cuando ha identificado los tags de la rama adyacente. Paralelamente, los tags implementan un contador de profundidad de la pila del lector, transmitiendo el siguiente bit siempre que ese contador sea 0. En la Figura 2.17 se muestra un ejemplo de ejecución del protocolo. De la misma manera, en la Tabla 2.5 se especifican los valores de las variables de dicho ejemplo.

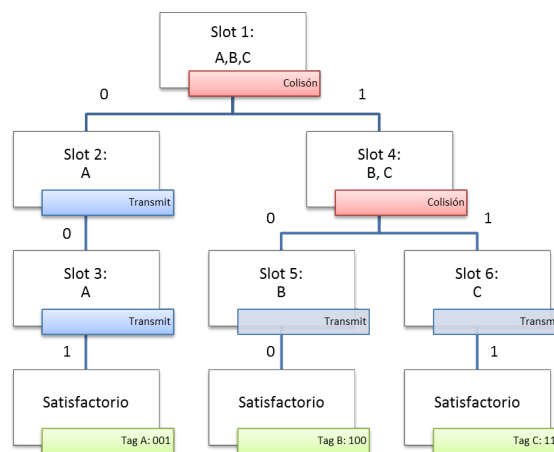


Figura 2.17: Ejemplo del protocolo ID-BTS

Katheeja et al. presentan una variante en [Katheeja 10], en la que el lector solicita a los tags el envío de su ID al completo inicialmente, para a continuación, crear b conjuntos de tags en función de los primeros bits de sus IDs.

2. Estado del arte

Slot	Resp.	Contadores			Pila	Operación
		Tag A	Tag B	Tag C		
1	X	0	0	0	Vacía	Colisión / Tag B + 1 Tag C + 1
2	0	0	1	1	[1]	Transmite
3	1	0	1	1	[1]	Tag A Ident. / Tag B - 1 Tag C - 1
4	X	-	0	0	Vacía	Colisión / Tag C + 1
5	0	-	0	1	[11]	Tag B Ident. / Tag C - 1
6	1	-	-	0	Vacía	Tag C Ident.

Tabla 2.5: Ejemplo del protocolo ID-BTS

Una vez creados, el protocolo identifica los tags empleando el ID-BTS. Esta variación disminuye las colisiones y obtiene buenos resultados cuando el ID es de pequeño tamaño.

- ◇ Bit Query (BQ) [Kim 06, Kim 07]: Kim et al. proponen un protocolo similar al ID-BTS pero empleando la filosofía de las queries del QT. El lector transmite queries formados por: un bit *null*, una cantidad de bits que indican la posición solicitada y un bit que confirma el valor del puntero del tag. Es decir, cada tag se encarga de actualizar un puntero interno que le indica el bit solicitado y es con el que compara el bit-query recibido. El lector solicita así bit a bit el valor del ID de los tags, creando bifurcaciones en el árbol en cada colisión.
- ◇ Bit-by-Bit (BBT): también conocido como *contactless* o protocolo para dispositivos sin contacto [Jacomet 99]. Este protocolo está basado en el TS y, por tanto, es de tipo *reader-driven*. La idea básica es la detección bit a bit del ID de cada tag, por lo que el proceso de arbitraje se realiza tantas veces como bits tiene dicho ID. La principal característica de este protocolo es la modulación empleada, ya que los tags transmiten el bit 0 como una secuencia de 00ZZ (siendo Z alta impedancia o señal sin modulación) y el bit 1 como ZZ00, de forma que el lector puede diferenciar si existen ambas respuestas

en caso de colisión. En [Choi 04], Choi et al. presentan el BBT modificado, sin la modulación anterior y simplemente enviando bits únicos. Además, también presentan el Enhanced Bit by Bit (EBBT), donde el lector solicita primeramente a los tags que contesten con su ID al completo. De aquí el lector obtiene los bits que han colisionado y los que están libres de colisión, y a partir de los resultados, emplea el algoritmo BBT modificado en los bits que han colisionado.

2.3.5 Comparación de los protocolos basados en árbol

Los protocolos basados en árbol, descritos en la sección 2.3, son deterministas y detectan el 100 % de los tags existentes en la zona de interrogación. En este tipo de protocolos, el lector realiza las lecturas en función del ID de cada tag, a excepción de los de tipo TS. Con cada respuesta el lector obtiene una indicación de dónde debe buscar a continuación. Por tanto, dependen directamente de la longitud de dicho ID, ya que afecta directamente a su rendimiento. En la Tabla 2.6 se realiza una comparación entre los diferentes tipos de protocolos basados en árbol, donde n representa el número de tags a identificar y k la longitud de sus IDs.

De la tabla se obtiene la conclusión de que el protocolo QT es el que proporciona el diseño de tag más sencillo. El costo de los tags del TS y QT es bajo, especialmente del QT, lo que posibilita que sean más económicos. Los de tipo QT están diseñados para no necesitar memoria, mientras que los de tipo TS necesitan ser capaces de generar un número aleatorio o pseudo-aleatorio, y almacenar un contador. Esto resulta un problema ya que si el tag se descarga, pierde su valor. Además, dado que el TS ejecuta ese proceso pseudo-aleatorio, existe la posibilidad de que cada tag involucrado en una colisión genere el mismo número aleatorio que los otros implicados, lo que puede llevar a un fenómeno similar al *tag starvation* producido en los protocolos basados en Aloha.

Los protocolos BTA son más complejos ya que además, junto con los de Bit-Tracking, son los únicos que necesitan sincronización entre los mensajes del lector y el tag. Los protocolos de tipo BTA presentan la estructura más compleja, aunque tienen una serie de ventajas como la eliminación de mensajes redundantes que requieren el TS y QT, la evasión de nodos que no poseen ningún tag, ya que si no

2. Estado del arte

	TS	QT	Bit-Track	BTA
Características	División en B subconjuntos	Tags comparan ID con <i>query</i>	Traza la colisión a nivel de bit	Cada tag responde bit a bit
RTF/TTF	RTF	RTF	RTF	RTF
Memoryless	-	Sí	Algunos	-
Codificación	-	-	Manchester	Manchester
Nivel de precisión	100 %	100 %	100 %	100 %
Tiempo requerido	$f(n)$	$f(n)$	$f(n)$	$f(2^k)$
Nº respuestas	$n \log Bn$	$2, 21k \log n + 4, 19k$	$2n - 1$	$f(n(k + 1))$
Costo del tag	Contador Gen. RN	Comparador	Circ. Sincronización	Puntero
	TX/RX		TX/RX	
Eficiencia	34.6 % [Hush 98]	34.6 % [Law 00]	50 % [Jia 12]	-
Costo sistema	Bajo	Muy Bajo	Medio	Alto
Complejidad protocolo	Baja	Muy Baja	Media	Alta

Tabla 2.6: Comparativa de los métodos deterministas

se produce colisión no existe bifurcación, y que en el último nodo no es necesario emplear dos slots más para identificarlo en caso de colisión. Este mismo hecho es lo que consiguen los protocolos de tipo Bit-Tracking. Gracias a que son capaces de trazar la colisión a nivel de bit, consiguen extinguir los slots libres, ya que los nuevos queries generados cubrirán las dos posibilidades y obtendrán respuesta segura.

Excepcionalmente a los protocolos de tipo TS, todo este tipo de protocolos son altamente influenciados por la distribución de los códigos ID de los tags y por su longitud. Así, los protocolos BTA trabajan muy eficientemente con pequeños ta-

maños de ID, pero su rendimiento empeora al aumentar su tamaño. Por contra, a pesar de ser influenciados por ello, los protocolos Bit-Tracking permiten un avance en la generación de queries muy agresivo, lo que reduce las colisiones y el tiempo de detección del conjunto de tags de los protocolos.

2.4 Híbridos

Combinando las técnicas de los algoritmos basados en árbol y Aloha, se han propuesto una serie de protocolos conocidos como híbridos. Existen dos tipos de combinaciones: Aloha en estructuras de árbol, que emplean divisiones aleatorias en cada nodo de un árbol binario; y arboles en cada slot de trama Aloha, desarrollando un árbol con cada colisión producida en un slot de Aloha.

2.4.1 Protocolos Aloha en estructuras de árbol

Los protocolo de tipo Aloha en estructuras de árbol emplean divisiones aleatorias en cada nodo de un árbol binario. Pueden considerarse similares a un protocolo de árbol k -ario, donde se emplean estrategias basadas en Aloha para identificar los tags en cada nodo.

2.4.1.1 Tree Slotted Aloha (TSA)

Es una versión modificada del protocolo FSA empleando una estructura de árbol [Bonuccelli 06, Bonuccelli 07a]. El nodo raíz es un frame idéntico al de FSA en el que transmiten todos los tags escogiendo un slot. Una vez transmitido el frame completo, los tags que han colisionado en cada slot recuerdan su posición para transmitir. En un nuevo frame y con una nueva longitud estimada, transmiten únicamente los tags con los que ha colisionado, de forma que se resuelven las diferentes colisiones solo con los tags que las han provocado. El proceso entero se repite hasta que se resuelven todas las colisiones y por tanto, quedan identificados los tags. Este protocolo requiere memoria en los tags ya que deben recordar la posición que han escogido en el frame anterior y el nivel en el que han transmitido. A continuación, en la Figura 2.18 se muestra un ejemplo de funcionamiento del protocolo TSA.

2. Estado del arte

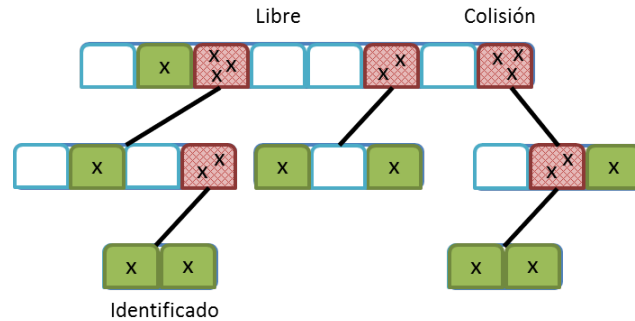


Figura 2.18: Ejemplo de ejecución del algoritmo TSA

El protocolo ejecuta el frame completo, y con la información de colisiones (c), slots libres (i) y satisfactorios (S) aplica el estimador de Vogt [Vogt 02a], y calcula el tamaño de los frames del siguiente nivel.

Como mejora del TSA, Porta et al. presentan el Binary Splitting-TSA (BSTSA) [La Porta 11], en el que se aplica el método TS a lo largo de la primera rama más exterior, hasta que se consigue una identificación. Posteriormente, sobre las ramas que se han ido creando se emplea el método TSA desde las hojas del árbol hasta el nodo raíz.

2.4.1.2 FSA en estructura de árbol QT

Namboodiri y Gao presentan en [Namboodiri 10, Namboodiri 07], tres protocolos de anti colisión híbridos, los cuales emplean una estructura de árbol QT y tramas FSA en cada nodo:

- ◇ Multi-Slotted Scheme (MS): los tags escogen aleatoriamente un slot en un frame concreto para enviar su respuesta en cada nodo del árbol. Si se produce una colisión, el lector crea una bifurcación en el árbol. En una de las ramas contestan de nuevo los tags que coinciden con el q_0 y en la otra con el q_1 , siendo q el *query* enviado por el lector. Este proceso se repite hasta que no se da ninguna colisión en el frame, abandonando esa rama o dando por finalizado el ciclo si no existen más posibles.
- ◇ Multi-Slotted Scheme con Sleep Selectivo (MSS): a la versión de algoritmo MS se le añade la característica de *Muting*. Es decir, que los tags, una vez

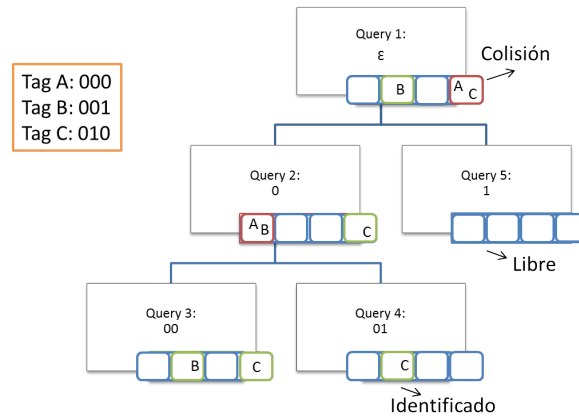


Figura 2.19: Ejemplo de identificación del MS

han sido identificados, permanecen en silencio o en estado *sleep*, hasta que el lector envía un nuevo *query* inicial (ϵ). Esta propiedad contribuye a la reducción de las colisiones en los diferentes nodos del árbol, y hace que el lector requiera de menos *queries* para detectar todos los tags.

- ◇ Multi-Slotted Scheme con asignación de slots (MAS): en los métodos anteriores, una colisión en el frame hace que el lector genere dos nuevos nodos y pruebe todas las posibilidades, como si hubiera existido una colisión en todos los slots, dado que no conoce el prefijo de los tags que han colisionado. Mediante esta modificación, emplea una asignación estructurada de slots en cada nodo. Ahora, los *queries* solo se envían cada $\log_B F$, siendo B el número de bifurcaciones de cada nodo y F el tamaño del frame. Es decir, que el árbol en cuestión deja de ser B -ario para ser N -ario.

2.4.2 Protocolos de árbol en estructuras de Aloha

Los algoritmos presentados a continuación emplean estructuras de Aloha para dividir el conjunto de tags en grupos más reducidos de forma que los algoritmos de tipo árbol produzcan menor número de colisiones y se comporten de forma más eficiente consumiendo menos slots.

2. Estado del arte

2.4.2.1 TS en estructuras FSA

Park et al. presentan en [Park 06, Park 07] un algoritmo denominado FSA con estimación de tags y algoritmo Tree Splitting (EBFSA). Se trata de un algoritmo híbrido que emplea una estructura de FSA con una estimación de frame previa, que resuelve mediante el algoritmo TS las colisiones producidas en cada slot. El proceso de identificación se divide en fase de estimación y fase de identificación. En la fase de estimación el lector usa un frame de longitud fija que va variando hasta que la probabilidad de colisión calculada adquiere un valor inferior a un límite establecido. Una vez estimado el número de tags, comienza la fase de identificación. Con el valor estimado, se establece un tamaño inicial de frame y los tags comienzan a identificarse mediante el método FSA. Cuando se produce una colisión en algún slot, el lector las resuelve empleando el protocolo TS.

Wu et al. presentan en [Wu 13] el Binary Tree Slotted Aloha (BTSA). El algoritmo comienza estableciendo un frame basado en el Slot Counter, y ajusta el parámetro Q en función de las respuestas que obtiene. Es decir, si la primera respuesta es una colisión, aumenta la Q , y si es un slot vacío la disminuye. Si la respuesta obtenida es satisfactoria, emplea ese tamaño de frame (2^Q) y continúa el proceso aplicando el algoritmo TS sobre los slots del frame en los que se produzca una colisión.

Qian et al. presentan en [Qian 13, Qian 10] el Adaptively Splitting-Based Arbitration Protocol (ASAP), en el que combina un método de estimación del número de tags con un método de identificación. El protocolo propuesto por Qian, divide el conjunto de tags en subconjuntos menores y estima las diferentes cantidades de cada subgrupo durante el proceso. El lector lanza un frame de Aloha y los tags escogen un slot. Cuando se produce una colisión, el lector emplea un método de estimación geométrica que requiere funciones hash. Posteriormente, con el valor estimado, el lector establece el tamaño del frame y se lo envía a los tags. De los cuales solo contestan aquellos que colisionaron en el slot anterior.

2.4.2.2 QT en estructura FSA

Siguiendo la filosofía de este tipo de protocolos se presenta el Framed QT (FQT) [Shin 07], un protocolo que divide el conjunto de tags empleando el algoritmo FSA.

El lector envía un tamaño de frame, y cada tag escoge aleatoriamente un slot para transmitir en dicho frame. A continuación aplica el protocolo QT, enviando a los tags participantes de un slot un prefijo para que aquellos coincidentes contesten a la petición. En la Figura 2.20 se muestra un ejemplo de ejecución.

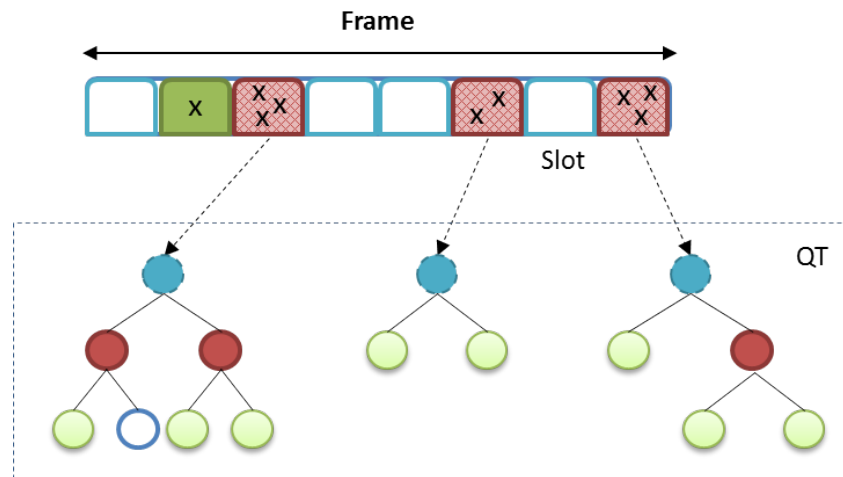


Figura 2.20: Ejemplo de identificación del FQT

El problema que plantean Shin et al., es que no es posible determinar fehacientemente el tamaño del frame desde el principio, ya que el número de tags es desconocido. Para solucionarlo propone usar un First Frame Test (FFT) que consisten en comenzar con un tamaño de frame pequeño y actualizarlo cuando en el primer slot, el número de colisiones supere un determinado valor límite impuesto por los autores, en este caso tres.

2.4.3 Comparación de los algoritmos híbridos

Tanto los protocolos Aloha en estructuras de árbol como los de árbol en estructuras de Aloha tratan de explotar las ventajas de los ya existentes. En el caso de los híbridos con estructura de árbol, tratan de explotar las características del FSA, reduciendo el número de tags por cada frame. Por ello, mediante las propiedades de los algoritmos de árbol, van dividiendo el conjunto de tags en pequeños subconjuntos que son identificados empleando el primer método mencionado. Dado que

2. Estado del arte

los protocolos deterministas de tipo árbol concentran la gran mayoría de sus colisiones en las primeras respuestas de los tags, los algoritmos híbridos con estructura Aloha tratan de diversificar el conjunto de tags para evitar ese número tan elevado de colisiones en un primer momento. En ambos casos las eficiencias de estos métodos superan a las de los métodos origen por separado, pero también aumenta en gran medida la complejidad de los protocolos, haciendo que se requieran diseños mayores y más complejos, tanto en el lector como en los tags, constituyendo su optimización una línea de investigación vigente en el ámbito de RFID. Además de esto, en el caso de los híbridos con estructura de árbol, se debe realizar un ciclo de estimación para que el frame empleado sea lo más ajustado posible al número de tags existentes y así conseguir una mayor optimización y estabilidad del algoritmo.

2.5 Estándares de anti-colisión RFID

Los estándares en RFID definen cómo deben trabajar los sistemas de forma global y detallada: el interfaz del aire, la forma en que lector y tags se comunican, la organización de los datos, etc. Dado que RFID se emplea en gran cantidad de aplicaciones como sistemas de pago o trazabilidad de mercancías, los estándares representan una parte crítica. Durante los últimos años se han desarrollado una gran cantidad de estándares diferentes, para diferentes tipos de sistemas [Violino 05].

La situación de los estándares en RFID se complicó al aparecer la organización EPCglobal, lo que provocó la existencia de dos instituciones de estandarización diferentes: la Organización Internacional de Estandarización (ISO) [ISO 13] y EPCglobal [EPCglobal 13].

- ◇ ISO: esta organización ha definido las especificaciones para la transmisión en el aire de los sistemas de pago, tarjetas *contactless* y de proximidad entre otros. También ha propuesto estándares para la trazabilidad de contenedores y palés.
- ◇ EPCglobal: esta institución desarrolla estándares para las redes en la cadena de suministro internacional, tratando de unificar todos los elementos que participan en ella, desde el producto individual, hasta el nivel más alto de la cadena de control de suministros.

2.5 Estándares de anti-colisión RFID

En la Tabla 2.7 se muestran los protocolos de anti-colisión que emplean los diferentes estándares, tanto ISO como EPCglobal. Además, Philips ha diseñado una serie de hardware con sus propios protocolos, al margen de los estándares presentados, que han ofrecido buenas prestaciones y también son mostrados en la tabla mencionada.

Estándar	Frecuencia	Protocolo
ISO		
18000-3 MODE 1	HF	PA y DFSA
18000-3 MODE 2	HF	FDMA + TDMA (SA)
14443-3 Tipo A	HF	DBSA
14443-3 Tipo B	HF	DFSA
18000-6A	UHF	FSA
18000-6B	UHF	ID-BTS
18000-6C	UHF	DFSA (SC)
EPCglobal		
Class 0	UHF	ID-BTS
Class 1	UHF	ID-BTS
Class 1 Gen 2	UHF	DFSA(SC)
Class 1	HF	BFSA
Philips		
I-Code	HF	DFSA
U-Code	UHF	DFSA(SC)
Mifare	HF	DBSA

Tabla 2.7: Estándares RFID y sus protocolos de anti-colisión

Cabe destacar que, a día de hoy, la última estandarización presentada por EPCglobal [EPCglobal 08] y una de las más empleadas, la EPCglobal Class 1 Gen 2, ha sido incluida en la ISO-18000-6C parte 3. Ambas utilizan una versión de DFSA con el algoritmo Q o *Slot Counter* (SC).

De la tabla mostrada se puede obtener la conclusión de que los protocolos que

2. Estado del arte

trabajan en HF emplean mayoritariamente protocolos basados en Aloha. Mientras que los que trabajan en UHF, emplean de forma indiferente, tanto basados en Aloha como en árbol. Por otro lado, los protocolos con bajo ancho de banda emplean, generalmente, estrategias Aloha.

2.6 Conclusiones

Los protocolos de anti-colisión determinan una parte crítica de un sistema RFID. El problema de las colisiones de tag provoca una serie de inconvenientes que deben ser resueltos, ya que perjudican al sistema en gran medida. Es evidente, por la literatura existente, que se ha realizado una extensa investigación en torno a este problema. Los protocolos diseñados cada vez son más sofisticados y consiguen mejores resultados, pero muchos de ellos no son llevados a sistemas reales. Existen dos tipos principales de protocolos: los probabilísticos y los deterministas. Además, se ha presentado una tercera opción que combina ambas, los híbridos. Pero, ¿es posible destacar alguno de ellos por encima del otro?

Un protocolo probabilístico es aquel en el que las operaciones realizadas y la naturaleza del propio protocolo no garantizan la lectura de la totalidad de los tags existentes en su rango. Aún así, un buen protocolo probabilístico puede maximizar la probabilidad de que todos sean leídos. Como ya se ha visto en la Sección 2.2, los protocolos probabilísticos pueden trabajar como TTF o RTF, aunque los más empleados operan en RTF, permitiendo al lector ser el principal controlador del proceso. Las respuestas de los tags se distribuyen en slots para que todos ellos tengan la oportunidad de ser identificados. Cuanto más uniformemente distribuidas estén sus respuestas, mayor probabilidad hay de que el lector las escuche. Además, una gran ventaja de éstos es que son fácilmente adaptables a las variaciones de tags en el entorno, es decir, a la aparición o desaparición de tags en el rango de lectura. Los protocolos basados en Aloha avanzan en la dirección del DFSA y la estimación de tags. Los estimadores son cada vez más exactos, pero también su complejidad matemática y computacional es mayor, lo cual no favorece a su implementación física. Por ello es imprescindible reducir dicha complejidad.

Un protocolo determinista es aquel capaz de identificar todos los tags de la población presente en el rango del lector. Generalmente, requieren que los tags

permanezcan presentes en el rango a lo largo de la duración del ciclo de lectura completo. El lector controla los diferentes pasos del protocolo y mediante comandos consistentes en órdenes o prefijos de identificadores, va separando el conjunto de tags en pequeños subconjuntos hasta que todos ellos quedan identificados. Cabe destacar que un algoritmo determinista no debe mostrar ningún comportamiento probabilístico. Siendo exigentes en este aspecto, algunos protocolos presentados aquí como el TS y sus variantes, son probabilísticos realmente. Por otro lado, los protocolos basados en árbol se han visto perjudicados por el incremento de los IDs de los tags. Dado que son dependientes, tanto de la distribución de estos IDs, como de su longitud, se han tenido que desarrollar nuevas soluciones para afrontar dicho problema. Los protocolos basados en QT son los que presentan el hardware más sencillo y por tanto, los que a priori van a emplear menos potencia. Pero la elevada cantidad de colisiones que se producen y el exceso de bits transmitidos debido a éstas, hace que su consumo energético y el tiempo de identificación del conjunto de tags se incrementen sustancialmente.

Los protocolos híbridos, por otra parte, combinan las estrategias de Aloha y de árbol. Algunos de los protocolos desarrollados consiguen ciertos avances con respecto a sus protocolos origen. Sin embargo, el costo de tag y lector se incrementa considerablemente.

De los estándares mostrados no se puede obtener un protocolo que destaque por encima del resto. Ya se ha visto que los protocolos de Aloha son más empleados en sistemas RFID de HF, y en los UHF ambos tipos por igual, aunque tienden a los protocolos deterministas.

La principal conclusión es que dado que los sistemas RFID cada vez son más comunes, y el número de elementos que los componen es mayor, el tiempo de identificación de los tags y su consumo se disparan. Los protocolos que muestran mayor capacidad de mejora en este aspecto son los deterministas.

*El asunto es el problema; la forma,
la solución.*

Friedrich Hebbel

3

Colisiones y oportunidades de mejora

Las características de un protocolo RFID obligan a que los tags dispongan de memoria y capacidades computacionales limitadas [Abraham 02]. En la medida en que estas capacidades aumentan, el costo del hardware de los tags y su complejidad aumentan, perjudicando tanto al consumo como al tiempo de identificación de los tags del sistema RFID. Por ello, el protocolo de identificación debe ser simple y demandar la menor actividad posible de los propios tags, por un lado desde el punto de vista económico, para no incrementar el precio de los tags, y por otro, desde el punto de vista del consumo energético. Tanto en el caso de los tags pasivos, que demandan la energía de la propia señal generada por el lector, como en el caso de los activos, que disponen de una batería, se debe tener muy en cuenta el consumo del sistema.

El hardware necesario para implementar los diferentes protocolos determina en gran medida el consumo que éstos van a demandar. Los protocolos basados en Aloha, considerados probabilísticos, y los híbridos emplean sofisticados tags que, al menos, usan generadores numéricos aleatorios o pseudo-aleatorios, y memorias adicionales o variables internas que incrementan el costo de su circuitería (ver Tabla

3. Colisiones y oportunidades de mejora

2.1). Los protocolos basados en árbol, por el contrario, emplean tags de menor complejidad (ver Tabla 2.6). Dentro de este tipo de protocolos, los basados en el TS emplean variables internas, lo que les obliga a disponer de memoria interna donde almacenarlas y un generador de números binarios. Mientras que los protocolos que presentan un hardware más simple son los basados en queries o en el protocolo QT. El hardware del protocolo QT está formado por un comparador, encargado de comprobar si el query recibido es igual que la parte correspondiente del ID del tag, y un transmisor/receptor de señal, común en el resto de protocolos. Este tipo de protocolos, también conocidos como *memoryless*, no necesitan utilizar memoria de almacenamiento de variables necesarias para la identificación. Sino que en cada slot, los tags cuyo ID coincide con el query recibido, responden al lector con su ID al completo. Dado que este último proporciona toda la información necesaria a los tags en cada interrogación, dichos tags no necesitan almacenar información, aparte de sus propios IDs.

3.1 Protocolos basados en queries

Las últimas versiones de los protocolos basados en queries (ver Sección 2.3.2) [Pan 11, Jia 12, Lai 13] mejoran la respuesta eliminando los bits innecesarios recibidos en el query. Es decir, contestan a partir del último bit del prefijo recibido. La ventaja de esta estrategia es que cuando un único tag coincide con el query recibido, éste queda identificado en un único slot. Sin embargo, dado que los queries iniciales empleados son de longitudes pequeñas, las probabilidades de que varios tags coincidan con ese prefijo y contesten son muy elevadas. Por ello, en este tipo de protocolos se produce una gran cantidad de colisiones, especialmente al inicio del protocolo. Debido a este problema, el gasto energético y el número de bits transmitidos entre lector y tag es muy elevado y además, causa el incremento del tiempo de identificación del sistema RFID.

Este problema de exceso de consumo energético y de transmisión de bits es debido a la necesidad por parte de los tags de tener que reenviar sus respuestas al lector. Cada vez que se produce una colisión, el lector debe interrogar a los tags nuevamente, esto conlleva un reenvío del mensaje anterior, incluso con alguna modificación para obtener una respuesta diferente. Por otra parte, dada la simpleza del

3.2 Métodos para la reducción de bits en los mensajes

hardware de los tags y su incapacidad para transmitir y recibir simultáneamente [Abraham 02], los tags no son conscientes de lo que está sucediendo a su alrededor. Por ello, cuando se produce una colisión, independientemente del momento de la transmisión en la que se produzca, el tag transmitirá el mensaje al completo (o el ID restante). El lector, dependiendo de las técnicas o codificaciones del mensaje, podrá enterarse antes de recibirlo completamente o después de producirse la colisión. Pero en ningún momento se lo podrá transmitir al tag para que detenga su envío, ya que la transmisión y recepción simultánea no es posible en los tags [Choi 07]. Las soluciones a estos problemas están enfocadas desde dos puntos diferentes:

- ◇ Reducción de la cantidad de mensajes: Este tipo de soluciones son las comentadas en el Capítulo 2. Todo el conjunto de protocolos recogidos en ese capítulo tratan de reducir el número de slots, o mensajes intercambiados entre tag y lector. En gran medida, el incremento de mensajes se produce por las colisiones, aunque también por los ciclos libres. Así, estos protocolos tratan de reducir esos dos tipos de slots. Al reducir estos mensajes, el tiempo de identificación disminuye (a priori) y el consumo energético desciende. Sin embargo, se dan casos en los que para disminuir los mensajes, el protocolo se ve forzado a aumentar el tamaño de éstos. Esto provoca que el número de bits transmitidos aumente, y pueda llegar a eclipsar la disminución del número de mensajes [Choi 10].
- ◇ Reducción del número de bits por mensaje: La mayoría de los métodos que optan por esta estrategia necesitan emplear memoria interna en los tags. De esta forma, el tag recuerda qué posición del ID transmitió la última vez y el lector no necesita recordárselo en cada slot. Pese a esto, existen otro tipo de estrategias que tratan de reducir el tamaño de los mensajes. Este tipo de métodos se estudian en la sección presentada a continuación.

3.2 Métodos para la reducción de bits en los mensajes

A lo largo de estos últimos años, se han propuesto una serie de métodos para reducir la transmisión de bits en cada mensaje. La gran mayoría de estos métodos se

3. Colisiones y oportunidades de mejora

han planteado desde un punto de vista teórico y han sido comprobados mediante simulación ya que su implementación, o bien no es posible, o las consecuencias que producen son peores que el propio problema. Los métodos son los siguientes: aviso de colisión, corte de portadora, short-long queries y RN16.

3.2.1 Aviso de colisión

Este tipo de método no es realizable y por tanto, todos sus resultados se basan en simulaciones. Ha sido empleado en varios protocolos de anti-colisión y sus resultados son muy positivos [Zhou 03, Zhou 04, Mohammed 11]. Este método trata de evitar la situación provocada por la colisión de la respuesta de varios tags, donde estos continúan enviando su ID al completo hasta el último bit. Si la colisión se produce en el segundo bit enviado, los otros $k - 2$ bits resultan irrelevantes para el lector. Esto, además de ser un gasto innecesario de bits, conlleva unas pérdidas energéticas asociadas a su envío, y la correspondiente pérdida de tiempo. Principalmente requiere dos acciones:

- ◇ Los tags deben emplear una codificación que permita al lector identificar la colisión a nivel de bit. De forma que según vaya recibiendo las respuestas de los tags sea capaz de identificar la colisión antes de que éstos finalicen la transmisión. Además, se requiere que las respuestas de los tags estén sincronizadas.
- ◇ Una vez identificado el bit colisionado, el lector envía un mensaje a los tags para que éstos detengan el envío.

De esta forma se consigue evitar que los tags envíen sus IDs al completo en caso de colisión, aliviando así gran parte del problema antes mencionado. Sin embargo, como ya se ha comentado anteriormente, los tags actuales no son capaces de transmitir en full-duplex, es decir, que no permiten la transmisión y recepción simultánea [Choi 07, Mohammed 11]. La comunicación RFID se produce en half-duplex y por tanto el tag no es capaz de recibir un mensaje del lector hasta que ha completado la transmisión. La viabilidad de esta solución requiere de tags con dos

3.2 Métodos para la reducción de bits en los mensajes

antenas capaces de emitir y recibir en frecuencias distintas. Esto aumenta sobremanera el costo de los tags, abandonando por completo la filosofía de la tecnología RFID, ya que debe preservar la simplicidad de los tags.

3.2.2 Corte de portadora

Con el propósito de solucionar el problema de los bits excedentes en el caso de colisión, Chen plantea en [Chen 10] la metodología del corte de la portadora (*carrier off*). En este caso, el autor propone la utilización de la codificación Manchester, la cuál permite identificar la colisión a nivel de bit y la sincronización perfecta entre las respuestas de los tags. Con ello, propone truncar dichas respuestas una vez se ha detectado la colisión en el lector, deteniendo la señal portadora que los alimenta.

Este tipo de solución presenta algunos inconvenientes:

- ◇ Esta orientada principalmente a los tags pasivos, ya que los activos y semi-pasivos poseen alimentación externa y necesitarían otro tipo de estímulo para indicarles la finalización de la transmisión.
- ◇ En un sistema real se debe hacer un calibrado de la comunicación con los tags al comienzo del proceso de identificación [EPCglobal 08]. Si la portadora se corta, se está forzando a realizar un recalibrado en cada slot. En definitiva, la mejora obtenida no resulta tan rentable ya que se perdería mucho tiempo.
- ◇ Por otro lado, cada vez que la portadora alcanza el tag en cuestión provoca la carga de su condensador interno antes de comenzar con el proceso de recepción y posterior transmisión de su ID. Al quitar la portadora, el condensador se descarga consumiendo un determinado tiempo que se debe sumar al de carga de condensador en el siguiente slot una vez reactivada la portadora. Durante estos tiempos, el tag se encuentra en *stand-by* y no está operativo. Por tanto, se generan una serie de tiempos adicionales que antes no se contemplaban [Recknagel 11]. En la Figura 3.1 se muestra gráficamente el problema de la carga y descarga del condensador de un tag.

3. Colisiones y oportunidades de mejora

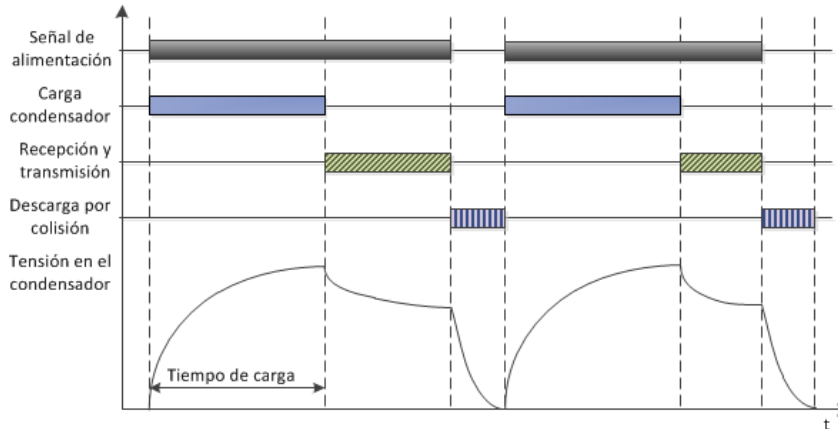


Figura 3.1: Proceso de carga y descarga del condensador de un tag

3.2.3 Short-long queries

De la misma manera que los dos métodos anteriores, el método de *short-long queries* trata de minimizar esa cantidad de bits que no van a ser útiles en una colisión [Law 00]. Para ello implementa dos tipos de queries: *short query* o trama corta y *long query* o trama larga. El primero solo obtendrá respuestas de un bit de los tags mientras que el segundo recibirá el ID completo. De esta forma, el lector únicamente envía un *long query* cuando esté seguro de que solo le va a contestar un tag con su ID completo. El mismo autor añade otra modificación del método en la que propone que los tags empleen memoria interna.

Una variante presentada por el mismo autor es el método *incremental-matching*. Éste requiere que el tag recuerde la posición en la que el prefijo ha sido validado en el último query. Cada tag posee un marcador que indica la posición a partir de la cual se compara el query recibido del lector. Si la trama recibida coincide con la contenida en el tag desde esa posición, incrementa en uno el indicador. Así se consigue que las tramas enviadas sean de menor tamaño que en el protocolo QT, reduciendo la complejidad del lector.

Estas soluciones no indican cómo el lector es capaz de diferenciar varias respuestas similares a un mismo *short query*. Es decir, que si el lector recibe varias transmisiones cortas iguales de los tags no va a ser capaz de detectar la colisión, por tanto, las interpretaría como una respuesta correcta. A continuación enviaría un *long query* y se produciría entonces la colisión. De esta forma, la eficiencia

3.2 Métodos para la reducción de bits en los mensajes

del protocolo se vería muy perjudicada, ya que se encontraría inmerso en un ciclo de comandos *short-long*, resultando muy perjudicial para su rapidez y consumo energético.

3.2.4 RN16

El método QT-RN16 presentado por Choi [Choi 07] reduce el número de bits empleando generadores de números aleatorios de 16 bits (RN16) en los tags. Al inicio del proceso de identificación, cada tag genera un número aleatorio y el lector envía un query que los tags lo comparan con su ID. En caso de que alguno de ellos coincida, en vez de enviar su ID, enviará su RN16. De esta forma se ahorra un total de $k - 16$ bits en cada slot. Cuando se produce una colisión entre varios RN16, el lector añade un 0 y 1 al último prefijo enviado y continúa el proceso de identificación. En caso de recibir correctamente el número aleatorio, el lector responde con un mensaje de tipo ACK y el tag en ese caso envía su ID. Los inconvenientes que presenta este método son:

- ◇ Los tags demandan cierta complejidad ya que, además de necesitar memoria, requieren de un generador aleatorio de números de 16 bits.
- ◇ Se da el problema de que cuando el conjunto de tags a identificar es muy numeroso (por encima de 400 tags aproximadamente), el rendimiento del protocolo desciende de forma abrupta. Esto es debido a que la probabilidad de obtener tags con un RN16 similar es mayor cuanto mayor es el conjunto de tags. Para solucionar este problema Yang en [Yang 11] presenta una modificación del QT-RN16 en el que realiza la operación XOR entre ese RN16 y tuplos de 16 bits del ID del tag. De esta manera consigue solucionar el problema y reduce la cantidad de bits transmitidos por los tags. Como inconveniente cabe destacar que el costo de los tags se incrementa en gran medida con esta modificación.

3. Colisiones y oportunidades de mejora

3.2.5 Comparación de métodos para la reducción de bits en protocolos basados en QT

Los métodos de reducción de bits presentados son soluciones dadas al problema del excedente de transmisión de bits tras una colisión de los protocolos basados en QT. En la Tabla 3.1 se realiza una comparación entre los diferentes métodos, donde n representa el número de tags a identificar y k la longitud de sus IDs.

	Aviso Colisión	Corte Portadora	S-L Queries	RN16
Características	Lector manda un mensaje con colisión	Lector corta CW con colisión	Short Query - Tag 1 bit Long query - Tag k bits	Tags transmiten RN16 en vez de su ID
Codificación	Manchester	Manchester	-	-
Sincronización	Sí	Sí	No	No
Costo sistema	Bajo	Muy Bajo	Medio	Alto
Complejidad solución	Muy Alta	Baja	Media	Alta
Inconvenientes	Transmisión y recepción simultánea	Sincronización de los tags en cada slot	Respuestas cortas iguales	Propenso a colisión con n elevado
Implementable	No	Sí	No	Solo n bajos

Tabla 3.1: Métodos de ahorro de bits en el protocolo QT

Todos los métodos aquí presentados reducen la cantidad de bits malgastados en las colisiones. Los dos primeros, de aviso de colisión y de corte de portadora, lo hacen de forma óptima, ya que no permiten que se transmitan más bits una vez se ha detectado la colisión. El tercero presentado, también da buenos resultados, a pesar de que el método demanda una mayor complejidad. Por último, el RN16 es el único que es perfectamente realizable físicamente, siempre que el número de tags de la zona de interrogación sea pequeño. Aunque también es el método que requiere un mayor costo de tag y lector, así como el que reduce los bits transmitidos por los tags en menor medida.

3.3 Oportunidades de mejora

En esta sección se presenta el principal problema que aborda la presente tesis y las oportunidades de mejora posibles. Como se ha comentado anteriormente, cuando los tags comienzan a transmitir su ID, una vez han comprobado si concuerda con el query recibido, es posible que se produzca una colisión. Dependiendo de la tecnología y codificación empleadas, el lector puede conocer este resultado durante la transmisión o una vez finalizada. Independientemente de ese conocimiento por parte del lector, los tags continuarán transmitiendo hasta que completen la longitud de su ID. Ese exceso de bits que se produce en las colisiones resulta un problema relevante a solucionar ya que, no solo provocan un gasto energético adicional, sino que además ese tiempo que los tags emplean en terminar de transmitir esos bits se podría evitar y el tiempo de identificación del sistema RFID sería menor. Se han presentado una serie de métodos como el de *Aviso de colisión* o *Corte de portadora*, que en simulación dan buenos resultados, pero que son difícilmente realizables por las limitaciones de la tecnología. Para establecer las oportunidades de mejora y desarrollar una solución alternativa al problema existente, se presenta un análisis del protocolo QT.

3.3.1 Análisis del protocolo QT

En este apartado se expone un análisis del protocolo QT basado en las expresiones de Hush y Wood [Hush 98] y Namboodiri y Gao [Namboodiri 10]. Este análisis se realiza para conocer el comportamiento del protocolo en las distintas fases del ciclo de interrogación. Las expresiones mostradas a continuación se basan en distribuciones homogéneas de los IDs de los tags, esto es, la probabilidad de obtener un 0 o un 1 en cada bit del ID del tag es 50 %. Según indican los autores mencionados, la probabilidad de que los prefijos de t tags, de un total de n , coincidan con el prefijo de longitud L enviado por el lector sigue una distribución binomial y viene dado por la expresión 3.1.

$$P(t, n, L) = \binom{n}{t} p^t (1-p)^{n-t}, k > 0 \quad (3.1)$$

3. Colisiones y oportunidades de mejora

Siendo p la probabilidad de que el ID del tag, de longitud k , tenga el mismo prefijo que el query enviado por el lector. En el caso del QT, como es un árbol binario y la longitud del query es L , la probabilidad de que los L primeros bits del ID de un tag coincida con el query enviado por el lector de longitud L es $p = 1/2^L$. Entonces se pueden calcular las probabilidades de: que se de un slot libre (3.2) o P_I^{QT} , un slot satisfactorio (3.3) o P_S^{QT} , y una colisión (3.4) o P_C^{QT} .

$$P_I^{QT} = \mathbb{P}(0, n, L) = (1 - 2^{(-L)})^n, \quad \forall k > 0; L > 0 \quad (3.2)$$

$$P_S^{QT} = \mathbb{P}(1, n, L) = n(2^{(-L)})(1 - 2^{(-L)})^{(n-1)}, \quad \forall k > 0; L > 0 \quad (3.3)$$

$$\begin{aligned} P_C^{QT} &= \mathbb{P>(> 1, n, L) = 1 - (P_S^{QT} + P_I^{QT}) = \\ &= 1 - (1 - 2^{(-L)})^n - n(2^{(-L)})(1 - 2^{(-L)})^{(n-1)}, \quad \forall k > 0; L > 0 \end{aligned} \quad (3.4)$$

Gráficamente las fórmulas anteriores se muestran en la Figura 3.2.

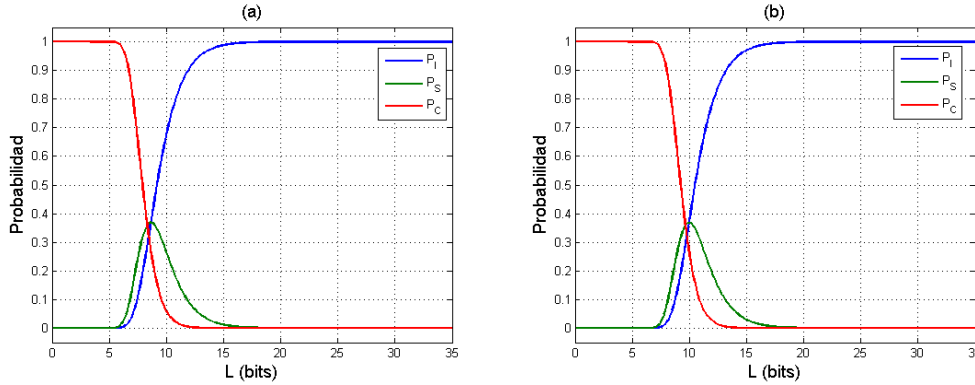


Figura 3.2: Probabilidades del protocolo QT. (a) $n = 400$; (b) $n = 1000$

Analizando estas gráficas se pueden obtener las siguientes conclusiones:

- ◇ La probabilidad de colisión, P_C^{QT} , es muy elevada en la parte inicial del ciclo de interrogación, cuando los queries empleados en la identificación están formados por pocos bits. A medida que el lector va completando los queries y éstos aumentan de tamaño, la probabilidad de colisión permanece constante

hasta una longitud de query concreta, que depende de n , en el que disminuye de manera brusca hasta 0. La principal conclusión que se obtiene, por tanto, es que la mayoría de las colisiones del protocolo se producen en los primeros queries.

- ◇ La probabilidad de slot libre, P_I^{QT} , se comporta de forma contraria a P_C^{QT} . Inicialmente es nula hasta alcanzar el punto de inflexión mencionado anteriormente, donde asciende hasta alcanzar el valor 1. Éste es un comportamiento lógico, ya que con queries de corta longitud es muy probable obtener una respuesta de un tag.
- ◇ La probabilidad de identificación de un tag alcanza su máximo para un tamaño de query, L_{MAX} , que depende de n . La expresión de cómo influye n sobre el tamaño de query donde se produciría el máximo de la función de probabilidad P_S^{QT} para un conjunto homogéneo de tags se ha obtenido de la siguiente forma:

$$\frac{dP_S^{QT}}{dL} = 0 \rightarrow L_{MAX} = \log_2 n \quad (3.5)$$

3.4 Conclusiones

Se ha encontrado un problema relacionado con la transmisión de los IDs de los tags derivado del conocido problema de las colisiones de tags. De acuerdo con el protocolo QT, los tags se ven forzados a enviar su ID en varias ocasiones debido a las colisiones que se producen, antes de completarse su identificación. Cuando se aplicó este protocolo a RFID, las longitudes de los tags no significaban un problema. Actualmente, estas longitudes, k , son elevadas y han de ser tomadas en cuenta, ya que las pérdidas provocadas por el exceso de transmisión de bits son considerables.

Gracias al análisis del protocolo QT, se ha observado que la mayoría de las colisiones se producen al comienzo del ciclo de interrogación. Este hecho hace que el problema sea más crítico aún, ya que las respuestas de los tags a queries de pequeña longitud son mayores (más cercanas a su longitud total, k). Es decir, que el

3. Colisiones y oportunidades de mejora

número de bits que transmiten los tags de forma inútil es mayor al comienzo el ciclo de interrogación, dado el elevado número de colisiones. Además, en una longitud concreta de los queries, dicha probabilidad de colisión comienza a descender de manera abrupta.

En la presente tesis se propone un método para controlar los bits que envían los tags en función de la longitud del query que se está enviando en cada momento. De esta forma, se pretende aliviar el efecto de las colisiones.

Si tu intención es describir la verdad, hazlo con sencillez y la elegancia déjasela al sastre.

Albert Einstein

4

Metodología de la ventana

El problema de las colisiones de tag ha sido durante los últimos años y es, un tema candente en la investigación de RFID [Finkenzeller 99, Klair 10]. Las soluciones presentadas son muy abundantes y se han realizado grandes avances desde que se presentaron los primeros protocolos de multi-acceso aplicados a RFID [Schoute 83, Hush 98]. Los protocolos basados en Aloha [Finkenzeller 99, EPCglobal 08, Namboodiri 12], son considerados probabilísticos y emplean sofisticados tags que al menos usan un generador aleatorio o pseudo-aleatorio de números y memorias adicionales o variables internas que aportan cierto grado de complejidad a su circuitería. Los protocolos basados en árbol [Finkenzeller 99, Hush 98, Law 00, Lai 13], por el contrario, emplean tags de menor complejidad a costa de la transmisión de una mayor cantidad de bits. Dentro de este tipo de protocolos, los de tipo TS [Hush 98] emplean variables internas, lo que les obliga a disponer de memoria no volátil donde almacenarlas. Mientras que los protocolos de tipo QT [Law 00], conocidos por su característica *memoryless*, o la no utilización de memoria para almacenaje de variables necesarias para la identificación, cumplen este tipo de características. Mediante la transmisión de cadenas de bits o *queries*, el lector indica a los tags un prefijo con el que comparar su ID y transmitir, bien su ID completo o la parte restante adyacente al último bit de *query* recibido. La circuitería de

4. Metodología de la ventana

los tags del protocolo QT es más simple que en otros casos ya que, principalmente, con un comparador es suficiente para desarrollar la actividad del protocolo (además de la circuitería correspondiente para la transmisión y recepción).

Para la realización de esta tesis se ha optado por preservar la baja complejidad de los tags, es decir, aquellos que presentan circuiterías poco complejas evitando generadores numéricos aleatorios o memorias de almacenamiento adicionales a la necesaria para el ID de cada tag. Dentro de estas características se engloban los métodos *memoryless* basados en *queries*. Estos métodos provocan una gran cantidad de colisiones que afectan a la latencia del protocolo y a su consumo energético. La disminución de cualquiera de estas medidas se puede llevar a cabo de dos formas: bien reduciendo el número de mensajes entre lector y tags a lo largo del proceso de identificación, o bien reduciendo el número de bits de cada mensaje. La gran mayoría de los protocolos presentados en el Capítulo 2 centran sus objetivos en la reducción del número de slots empleados. El incremento de mensajes viene producido en gran medida por las colisiones y, por tanto, este tipo de métodos tratan de reducirlas. Sin embargo, pocos son conscientes de los bits que están transmitiendo o de la energía que emplean en el proceso [Choi 10], lo que en ocasiones conlleva el empleo de pocos slots formados por un gran número de bits. La segunda opción, por otro lado, trata de reducir el número de bits que se intercambian lector y tags para optimizar las comunicaciones. En la presente tesis se ha optado por la segunda estrategia. Para ello, se presenta una metodología desde un punto de vista teórico, que controla el número de bits que transmiten los tags en cada slot. La metodología de la ventana posibilita que el protocolo que la emplee pueda trabajar como uno basado en *queries* o como uno de tipo BTA ajustando el tamaño de las respuestas de los tags, para reducir el número de bits que transmiten.

En la Tabla 4.1 se indican los términos y la notación que se va a emplear a partir de ahora en el presente documento.

4.1 Aclaraciones previas

En capítulos anteriores se ha hablado del término slot y de las colisiones. En esta sección, se indican algunas consideraciones acerca de estos términos y su aplicación al procedimiento desarrollado en esta tesis.

4.1 Aclaraciones previas

Símbolo	Definición	Unidad
n	Conjunto de tags	Tags
k	Longitud del ID de los tags	Bits
L	Longitud del query	Bits
W	Longitud de la ventana	Bits

Tabla 4.1: Términos empleados y su notación

- ◇ Slot: unidad que inicialmente se definió en los protocolos basados en Aloha, y actualmente se aplica a cualquier tipo de protocolo de anti-colisión. En este caso, un slot engloba un ciclo de interrogación lector-tag. Es decir, una solicitud del lector y una respuesta de los tags que deben responder a esa interrogación.
- ◇ Colisión: en la presente tesis se considera una colisión cuando dos tags transmiten bits diferentes en el mismo slot. Asumiendo un entorno de transmisión ideal, una colisión se produce única y exclusivamente por esa razón, sin intervenir factores como atenuación de señales o *Capture effect*. Para la detección de estas colisiones se emplea el CRC.
- ◇ CRC, *Cyclic Redundancy Check*: procedimiento para comprobar si una cadena de bits transmitida se ha realizado de forma correcta (ver Sección 4.2). Para ello, a la cadena a transmitir se adjunta un conjunto adicional de bits calculados en base al contenido de ésta (CRC). Una vez recibidos en el destino se recalcula el CRC y compara con la cadena transmitida.

A lo largo del presente capítulo se van a realizar una serie de comparativas entre protocolos empleando diversas medidas. Estas explicaciones serán ampliadas en el siguiente capítulo de experimentación, pero como introducción se presentan algunas de ellas:

- ◇ La eficiencia de slots, η : se calcula siguiendo la siguiente eq. 4.1. Sabiendo que n es el conjunto total de tags y τ el número total de slots empleados.

$$\eta = \frac{n}{\tau} \quad (4.1)$$

4. Metodología de la ventana

- ◇ Número total de bits: el cálculo de los bits totales transmitidos por el lector y los tags, toma como referencia la visión del lector. Es decir, en una respuesta de varios tags, se contabilizan los bits recibidos por uno de ellos, ya que al lector le llegan solapados. Por tanto, en cada slot se contabilizan los bits de un único tag, si es que no se produce un slot libre.
- ◇ La latencia: define el tiempo, en segundos, que tarda el lector en identificar todos los tags situados en la zona de interrogación del lector.

4.2 Ventana de bits

El procedimiento de la ventana de bits propuesto controla el número de bits que transmite el tag cuando el lector le solicita su ID. Al marco que engloba la cantidad de bits que un tag transmite en un slot la hemos denominado **ventana**. La idea fundamental de dicha ventana es adaptar su tamaño a las necesidades de cada momento. Así, pequeños tamaños o transmitir un número escaso de bits será adecuado cuando la probabilidad de que se produzca una colisión es elevada; y por el contrario, valores mayores o transmitir una mayor cantidad de bits cuando la probabilidad de colisiones sea menor será más beneficioso. De esta manera, el número de bits transmitidos por el tag se reduce al igual que la transmisión innecesaria de bits. Como se ha visto en la Sección 2.3.4, los métodos de tipo BTA permiten que los tags sean identificados enviando los bits de su ID una única vez, es decir, k bits. Como contrapartida, el lector necesita incrementar los bits transmitidos, lo que incrementa en gran medida el tiempo de identificación y el consumo del protocolo. Por otro lado, los métodos de tipo QT son capaces de identificar un tag en un único slot si es el único que contesta y por tanto, no se produce una colisión. Si ésta ocurre, los bits transmitidos por los tags se entremezclan y el lector no es capaz de interpretarlos, provocando una pérdida de tiempo equivalente a la longitud de su ID enviado, k , además de su correspondiente pérdida energética.

En la Figura 4.1 se muestra la comparativa entre el número medio de bits transmitidos por los protocolos QT [Law 00], y SIBT [Feng 06] con $k = 128$. Los tags del protocolo SIBT, de tipo BTA, transmiten únicamente los k bits de su ID en total, mientras que el protocolo QT sufre el problema que se ha comentado en la Sección

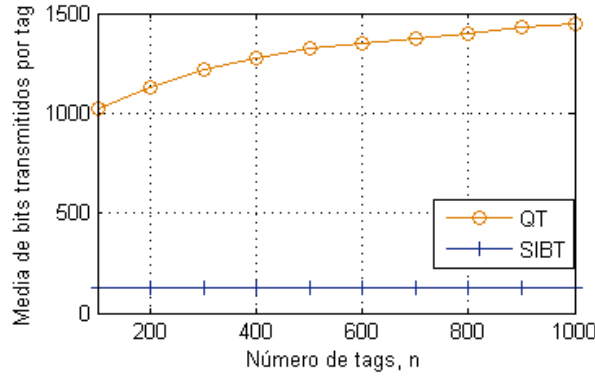


Figura 4.1: Comparativa entre el protocolo QT y el SIBT: número medio de bits transmitidos por tag con $k = 128$

3.3, es decir, que los tags transmiten todos los bits de su ID incluso en los casos que se producen colisiones. Por ello, transmiten un número mucho mayor de bits que los tags del protocolo SIBT. Sin embargo, en las Figuras 4.2.a y b se muestra cómo el protocolo QT supera con creces al SIBT en el número total de slots y bits totales entre lector y tag respectivamente. El protocolo SIBT se ve forzado a incrementar el número de bits que transmite su lector, lo cual, en el computo global de bits, hace que entre lector y tag se intercambien un mayor número de bits que en el protocolo QT. Esto provoca que la latencia y el consumo energético del protocolo SIBT sea mayor, aunque sus tags sean más conservadores que los del protocolo QT.

La ventana de bits busca obtener los beneficios de los métodos de tipo BTA y

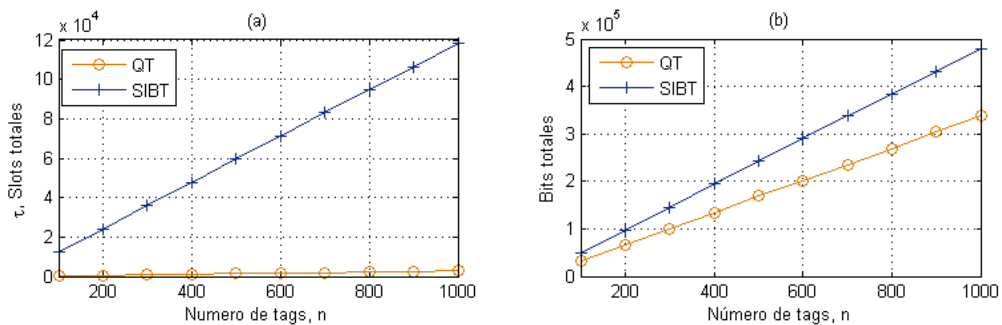


Figura 4.2: Comparativa entre el protocolo QT y el SIBT: (a) número de slots empleados en detectar el conjunto de tags, (b) número total de bits (lector + tags) empleados en el proceso

4. Metodología de la ventana

los del QT evitando por otro lado sus inconvenientes. La ventana, por tanto, trata de ofrecer un comportamiento intermedio al de los protocolos presentados, reduciendo el número de bits transmitidos por los tags, sin incrementar en exceso el número de bits que transmite el lector. Cuando la ventana está formada por una gran cantidad de bits, provoca que el protocolo trabaje de forma similar a los de tipo QT, mientras que ventanas de pequeño tamaño lo asemejan más a un protocolo de tipo BTA en el que las respuestas de los tags se producen bit a bit. Dado que los tags solo contestan el número de bits indicado por la ventana, el lector deberá interrogar a los tags tantas veces como sea necesario para la obtención del ID al completo. Además para que el lector sea capaz de identificar una colisión en el mensaje se añade un conjunto de bits adicionales, *Cyclic Redundancy Check*, de ahora en adelante CRC, empleados para comprobar que la respuesta se interpreta correctamente en el lector. Para el correcto funcionamiento de este protocolo se han asumido una serie de características.

- ◇ El canal de transmisión se considera ideal, al igual que en otros protocolos de la literatura estudiada [Hush 98, Law 00, Jia 12, Lai 13]. De manera que un error de transmisión ocurre únicamente en una situación de colisión, es decir, cuando el lector recibe dos bits de diferente valor y no por un error del canal.
- ◇ Se supone una perfecta sincronización entre las respuestas de los tags. Sincronización que, por otro lado, es razonable ya que el tiempo de propagación es despreciable. Así pues, cuando un conjunto de tags comparan su ID con el *query* recibido y contestan, estas respuestas se producen de forma simultánea. El lector recibe una única trama de bits sobre la que comprueba el CRC para determinar si se han recibido los datos de forma correcta (ver Figura 4.3).

A diferencia de los métodos estudiados en el estado del arte (Sección 2) cuyo canal de transmisión se considera ternario, con tres tipos de respuesta (satisfactoria, colisión, libre), el empleado por la ventana conforma un canal cuaternario. Se pueden dar cuatro tipos de slots: libre, colisión, go-On y satisfactorio.

- ◇ Slot libre (*Idle*): se produce cuando el lector solicita a los tags que envíen su ID mediante un *query* y, al no coincidir con el prefijo de ningún tag, no

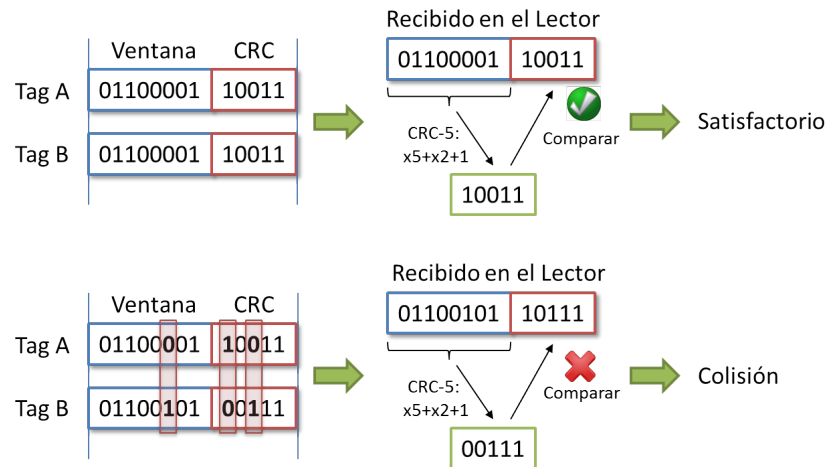


Figura 4.3: Ejemplo de funcionamiento de un CRC-5

obtiene ninguna respuesta. Esta situación acarrea una pérdida de tiempo del lector, ya que espera a obtener una respuesta que no llegará nunca.

- ◇ Slot en colisión (*Collision*): ocurre cuando varios tags forman la ventana con los bits indicados por el lector y sus correspondientes CRCs. Cuando las respuestas de los tags se interpretan en el lector, el CRC resultado recibido no se cumple y por tanto el lector no puede considerar los datos recibidos como correctos concluyendo que se ha producido una colisión en la transmisión (ver Figura 4.3). Las colisiones provocan una pérdida de tiempo y energía, y deterioran el ancho de banda del sistema.
- ◇ Slot *Go-On*: también denominado slot de tipo go-On o go-On slot, este tipo se da cuando uno o varios tags contestan su ventana según el tamaño indicado por el lector. Este último es capaz de interpretar la respuesta correctamente y la verifica mediante el CRC recibido en el mensaje de los tags. Sin embargo, el número de bits recibidos junto con el *query* enviado, no completan el número de bits necesarios para identificar un tag, es decir, que no alcanzan la longitud k del ID ($L + W < k$).
- ◇ Slot satisfactorio (*Success*): es un slot en el que únicamente contesta un tag y por tanto, el lector interpreta la respuesta sin ningún inconveniente. Además,

4. Metodología de la ventana

el tamaño de ventana recibida junto con el *query* enviado definen completamente el ID de un tag, es decir $L + W = k$.

4.2.1 Ventajas e inconvenientes de la ventana

La idea principal de la ventana es controlar el número de bits transmitidos por los tags. Así, las principales ventajas que aporta son:

- ◇ La ventana nace como una posible solución al exceso de bits transmitidos por los bits en los protocolos de anti-colisión basados en *queries*. Por ello, es aplicable a cualquiera de este tipo de protocolos. En esta tesis se aplica a dos (QT y CT) con el fin de justificar esta afirmación.
- ◇ Produce un descenso de los bits desaprovechados en una colisión. El número de bits que transmitiría el tag cuando se produce una colisión se reduce, lo que se refleja en el cómputo global de bits totales transmitidos y en los específicos de cada tag.
- ◇ El número de colisiones se reduce drásticamente, ya que la ventana permite que los tags envíen menor cantidad de información en cada slot y por tanto, la probabilidad de que haya bits diferentes es menor. Además, la ventana contribuye a convertir colisiones en slots satisfactorio-parciales o go-Ons. Es decir, un slot que sería colisión en un protocolo de anti-colisión, con la ventana podría ser uno de tipo go-On.
- ◇ El número de slots libres también sufre un descenso importante. El empleo de tamaños de ventana pequeños aporta información al lector, permitiendo generar nuevos mensajes de interrogación que están más orientados a la identificación de los tags existentes. De esta manera, se evitan ciclos de interrogación en los que no responde ningún tag. Esta ventaja contribuye a disminuir el tiempo de identificación del protocolo.

Alguna de estas ventajas varía ligeramente al aplicarse a cada protocolo específico, aprovechándose de las propias del protocolo. Por otro lado, los principales inconvenientes que presenta la utilización de la ventana son:

- ◇ Una respuesta de un solo tag no garantiza que éste quede completamente identificado. Se puede dar la situación en la cual, un lector recibe una respuesta pero no dispone de toda la información de los k bits de ID, causando una nueva interrogación a ese tag hasta obtener los bits al completo.
- ◇ Como les ocurre a los protocolos de tipo BTA, reducir el tamaño de los mensajes de los tags provoca que el lector deba enviar más información. En el caso de la ventana, se incrementará el número de bits que transmite el lector, y por tanto, el número de slots. Este problema se tratará de mitigar empleando diferentes tipos de ventana.

Así pues, el tamaño de la ventana va a influir de forma significativa en cada momento del ciclo de interrogación. La selección de una ventana de pocos bits al comienzo del proceso de identificación produce un descenso del número de slots libres y de colisiones. En cambio, los tamaños de ventana cercanos a k causarán un mayor número de colisiones y de slots libres al comienzo del ciclo. Sin embargo, en el momento que conteste un solo tag, éste quedará completamente definido e identificado ya que en un tamaño de ventana mayor transmitirá su ID al completo. Esto se estudia con más detalle en las próximas secciones.

4.2.2 Aplicación de la ventana a protocolos basados en *queries*

La ventana ha sido diseñada inicialmente para su aplicación sobre protocolos basado en *queries*. Como ya se ha mencionado en la Sección 4.2, existen dos estrategias contrarias en los protocolos basados en *queries*: de tipo QT (ver Sección 2.3.2) y de tipo BTA (ver Sección 2.3.4). La ventana permite al protocolo sobre el que es aplicada ganar en flexibilidad obteniendo la posibilidad de comportarse, no solo como el propio QT, sino también como uno de tipo BTA en función de las necesidades. Además, añade una serie de características que le permiten sacar provecho de los conjuntos de tags cuyos ID no estén distribuidos de forma homogénea. Las razones por las que se ha aplicado la ventana a los protocolos basados en *queries* son:

- ◇ Preservar la simplicidad de los tags. Este tipo de protocolos emplean tags con una circuitería generalmente simple evitando generadores numéricos aleatorios o memorias de almacenamiento adicionales a la necesaria para el ID de

4. Metodología de la ventana

cada tag. Además del transmisor y receptor, con un comparador que le permita comprobar si el prefijo de su ID es similar al *query* recibido, el tag es capaz de obtener y proveer toda la información necesaria para ser identificado.

- ◇ Característica *memoryless*. Este tipo de protocolos, como el protocolo QT y otros estudiados en el Capítulo 2, no necesitan utilizar memoria para almacenar variables a lo largo del proceso de identificación, simplificando la circuitería de los tags.
- ◇ Los protocolos basados en *queries* presentan una clara capacidad de mejora en cuanto a la cantidad de bits que transmiten los tags. La gran cantidad de colisiones que se producen, sobre todo al inicio del ciclo de interrogación, provocan que los tags envíen su ID al completo o casi al completo sin un resultado satisfactorio. Aunque los tags implementen la funcionalidad de la transmisión de su ID desde el último bit del *query* recibido ($k - L$ bits), el número de bits transmitidos sigue siendo muy elevado, ya que las colisiones ocurren al comienzo del ciclo de interrogación, donde la longitud de los *queries* (L) es pequeña y las respuestas de los tags siguen siendo largas.

La ventana se puede aplicar de forma prácticamente directa a los protocolos basados en *queries*, como el QT, ya que en cada slot se produce una interrogación del lector, seguido de las correspondientes respuestas de los tags. Aprovechando el envío del comando *query* del lector, el tamaño de la ventana deseado W puede ser adjuntado a dicho comando con un tamaño de $\lfloor \log_2 W \rfloor + 1$ bits, donde $\lfloor \cdot \rfloor$ representa el redondeo al entero más cercano hacia $-\infty$. Los tags dividen su ID en la parte a comparar con el *query* recibido, los L primeros bits, y seguidamente, la ventana de bits a transmitir, en caso de que el *query* recibido coincida con los L primeros bits. En la Figura 4.4 se muestra el procedimiento de los comandos empleados al aplicar la ventana a un protocolo basado en *queries*.

A continuación se presenta la aplicación de la ventana sobre dos protocolos basados en *queries*, el QT y el CT. Ambas secciones siguen una estructura similar. Inicialmente se estudia el funcionamiento de los protocolos ante la aplicación de la ventana constante. Una vez obtenidas las conclusiones sobre su influencia en el protocolo, se proponen unas técnicas para dotar a la ventana de capacidad dinámica, y se estudian y comparan las diferentes propuestas.

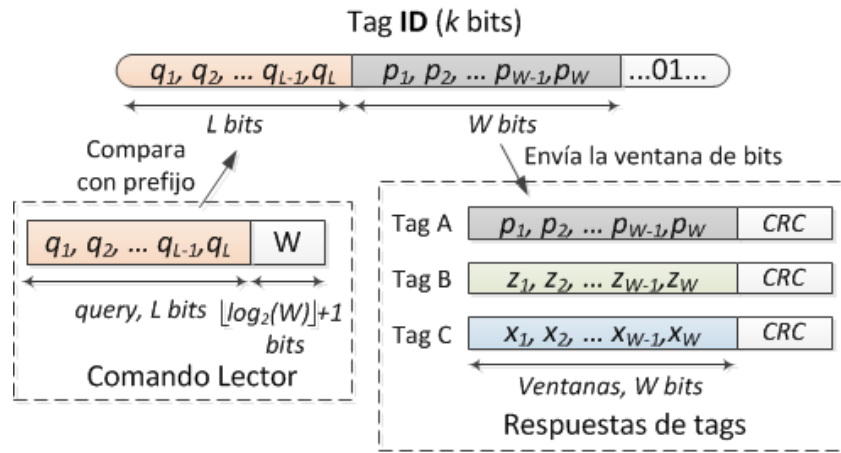


Figura 4.4: Comando del lector y respuestas de los tags en un protocolo basado en queries

4.3 QT con ventana constante

En este apartado se presenta la primera versión del protocolo QT con la ventana propuesta [Landaluce 13a]. El protocolo propuesto añade cierta complejidad a la circuitería de los tags, pero sigue manteniendo la propiedad memoryless del QT. La principal contribución de la aplicación de la ventana al protocolo QT es la clara disminución del número de bits transmitidos por los tags. Este hecho afectará al protocolo disminuyendo el tiempo empleado por los tags a la hora de responder al lector y disminuyendo el consumo energético de éstos. En el caso de tratarse de tags pasivos, se disminuye la complejidad de los tags (bits transmitidos) y el lector tendrá que mantener la CW (ver Sección 1.2) durante menos tiempo, disminuyendo el consumo de éste. Por otro lado, en caso de tratarse de tags activos, la duración de la batería será mayor ya que el número de bits transmitidos disminuye también. En esta versión del protocolo propuesto los tags contestan una cantidad de bits fija. Por ello se dice que la ventana toma un valor constante. En la Figura 4.5 se muestra la estructura que presenta el ID del tag de longitud k bits.

El lector posee una pila LIFO (*Last Input First Output*) para almacenar los queries que va generando (inicialmente '0' y '1'). En cada slot, el lector obtiene un query de L bits (q_1, \dots, q_L) de dicha pila y lo transmite. El tag lo recibe y lo compara con los L primeros bits de su ID de longitud k bits. Si la comparación

4. Metodología de la ventana



Figura 4.5: Estructura del ID del tag

es satisfactoria entonces el tag contesta los próximos W bits adyacentes al query recibido (w_1, \dots, w_W) , en vez del ID restante ($k - L$ bits) más el CRC calculado. En este protocolo propuesto los tags conocen por defecto el tamaño de la ventana a emplear, es decir, el número de bits que deben contestar. Por tanto, el lector únicamente debe transmitir el query calculado en cada slot. En la Figura 4.6 se muestra el esquema de funcionamiento del protocolo. Las respuestas de los tags llegan de forma sincronizada al lector y éste comprueba el CRC para determinar qué tipo de respuesta se ha obtenido (ver Figura 4.3):

- ◇ *Slot libre*: el lector considera un slot libre esperando un determinado tiempo. Si durante ese tiempo no obtiene respuesta, desecha el query enviado y continúa el proceso extrayendo otro query (*POP*) de la pila.
- ◇ *Slot colisionado*: en caso de que la comprobación del CRC no sea correcta, se considera una colisión. En ese caso el lector añade un 0 y un 1 al query enviado previamente, resultando dos nuevos queries $(q_1, \dots, q_L, \mathbf{0}; q_1, \dots, q_L, \mathbf{1})$.
- ◇ *Slot go-On*: si el lector calcula el CRC de la ventana recibida y éste coincide con el CRC incluido en el mensaje, los datos son consistentes y se consideran parte del ID de un tag. Aún así, dado que las respuestas de los tags se producen de forma síncrona, no se puede asegurar de forma absoluta que esa ventana pertenezca únicamente a un tag. Por ello, únicamente se considerará un slot de tipo go-On si se comprueba que la longitud del query enviado sumada al tamaño de ventana recibido es menor que la longitud del ID ($L + W < k$). En tal caso, el lector añade la ventana recibida (w_1, \dots, w_W) al query enviado, resultando un nuevo query, $q_1, \dots, q_L, w_1, \dots, w_W$ siendo $q_i, w_i \in \{0, 1\}$, que se envía a continuación.

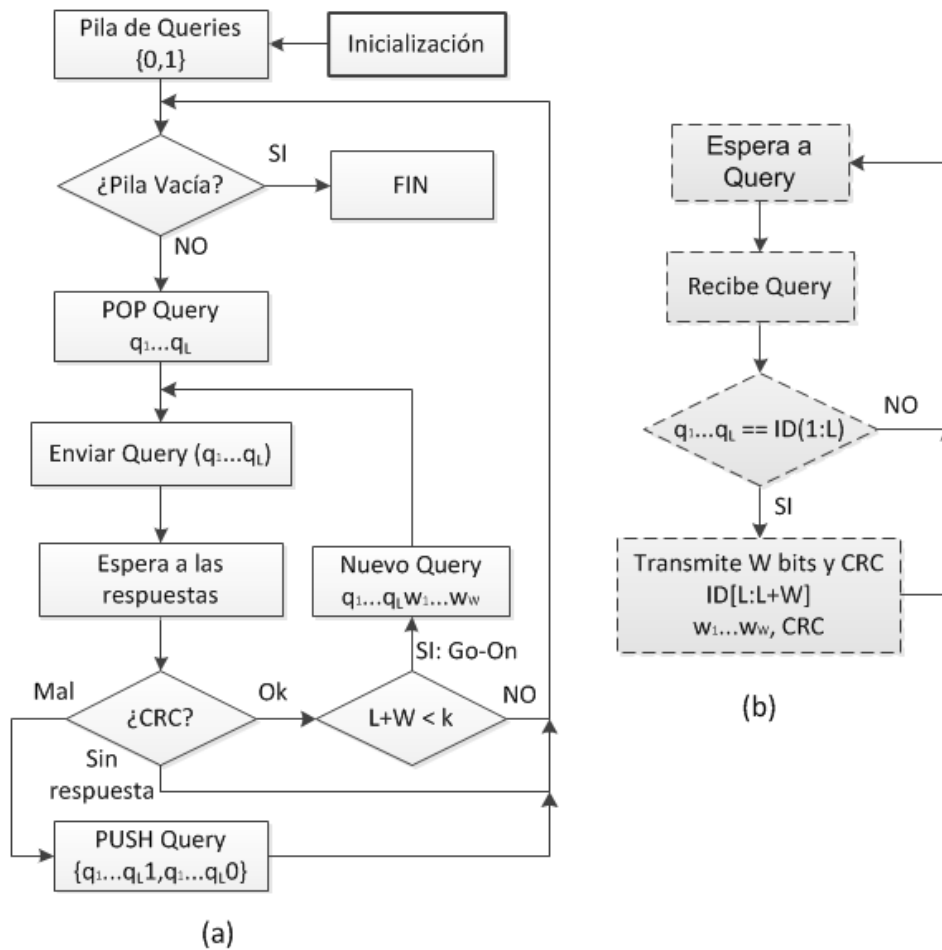


Figura 4.6: Esquema de funcionamiento del protocolo: (a) procedimiento del lector y (b) procedimiento de cada tag

- ◇ *Slot satisfactorio*: por otro lado, si la expresión anterior cumple que es mayor o igual que la longitud del ID, $L + W \geq k$, se sabe de forma absoluta que la respuesta corresponde a un único tag. En consecuencia, se considera un slot satisfactorio y el tag queda completamente identificado. El lector almacena su ID y continúa con el siguiente query de la pila.

Durante el proceso de identificación se puede dar la circunstancia de que el tamaño de ventana constante le indique al tag que debe transmitir más bits de los que le quedan para alcanzar el final de su ID. Por ello se implementa la siguiente función de acotamiento (eq. 4.2), donde i denota el slot actual:

4. Metodología de la ventana

$$W_{i+1} = \begin{cases} W_i, & L_i + W_i \leq k \\ k - L_i, & L_i + W_i > k \end{cases} \quad (4.2)$$

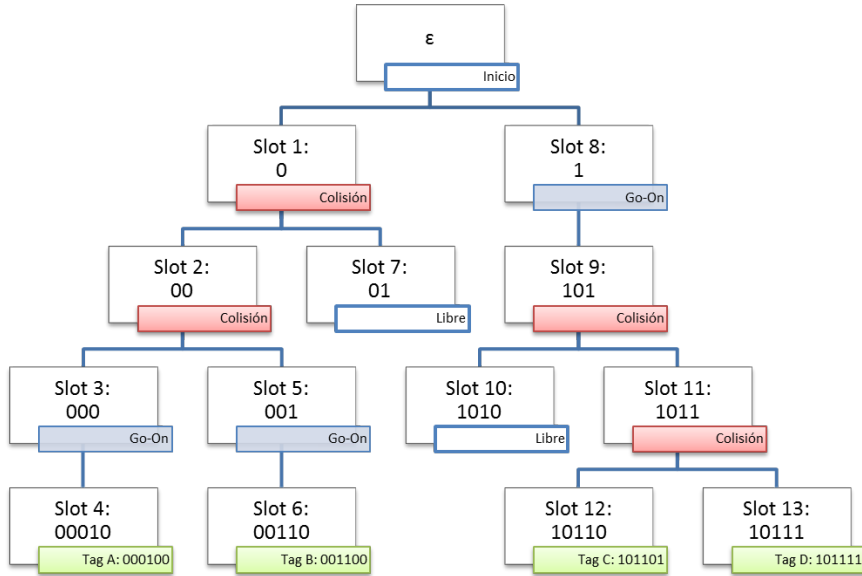


Figura 4.7: Ejemplo de ejecución del protocolo QT con ventana, $W = 2$

A continuación se muestra un ejemplo de ejecución del protocolo propuesto en la Figura 4.7. Además, en la Tabla 4.2 se muestra al detalle el intercambio de información que se produce entre lector y tags para completar el proceso de identificación. Para el ejemplo propuesto se han elegido los parámetros de longitud de ID $k = 6$, y tamaño de la ventana $W = 2$. En el rango de interrogación del lector se han situado cuatro tags (A, B, C y D).

El lector obtiene el query 0 de la pila, previamente inicializada con 0 y 1, y lo transmite. Los tags A y B cumplen la comparación y responden los 2 bits adyacentes ($W = 2$) y sus respectivos CRCs (no se muestran en la Tabla 4.2). Dado que sus ventanas son diferentes, el lector calcula el CRC de la ventana recibida, y al ser éste diferente del CRC adjunto a la ventana, se considera colisión. El lector genera dos nuevos queries, 00 y 01, y continúa el ciclo con el primero de ellos. Tras otra colisión en el slot 2, el lector recibe una ventana correcta en el slot 3. Un único tag responde al query enviado por el lector y por tanto, éste interpreta el CRC de forma correcta. Dado que el query enviado de 3 bits, L , y la ventana recibida de 2, W , no

4.3 QT con ventana constante

Slot	Lector Query	Tag Respuesta	CRC ¿OK?	Tipo de slot
1	0	Tag A: 00 Tag B: 01	No	Colisión
2	00	Tag A: 01 Tag B: 11	No	Colisión
3	000	Tag A: 10	$L + W < k$	Go-On
4	00010	Tag A: 0	$L + W = k$	Tag A 000100
5	001	Tag B: 10	$L + W < k$	Go-On
6	00110	Tag B: 0	$L + W = k$	Tag B 001100
7	01	-	-	Libre
8	1	Tag C: 01 Tag D: 01	$L + W < k$	Go-On
9	101	Tag C: 10 Tag D: 11	No	Colisión
10	1010	-	$L + W < k$	Libre
11	1011	Tag C: 01 Tag D: 11	No	Colisión
12	10110	Tag C: 1	$L + W = k$	Tag C 101101
13	10111	Tag D: 1	$L + W = k$	Tag D 101111

Tabla 4.2: Ejecución detallada del ejemplo mostrado en la Figura 4.7 del QT con $W = 2$

completan los 6 bits de longitud del ID, k , de los tags, el slot se considera go-On, y el lector genera un nuevo query 00010, con el cual el tag resulta identificado. Tras la identificación, el lector rescata un nuevo query de la pila, 001, y continúa el proceso hasta que no le quedan queries en ella, lo que interpreta como el final del proceso de identificación, ya que considera que no existen más tags en su zona de interrogación. A diferencia del slot 3, de tipo go-On, cabe destacar que el slot 8, del mismo tipo, es diferente ya que en ese caso el lector recibe la ventana de los tags C y D simultáneamente. Dado que ambos responden los mismos bits (el CRC es igual) el lector comprueba de forma correcta su cumplimiento y añade la ventana al query enviado, 101. Considerándolo, finalmente, un go-On ya que la suma de L

4. Metodología de la ventana

y W es menor que k .

El protocolo propuesto emplea 13 slots para identificar cuatro tags. Del ejemplo mostrado se pueden observar dos comportamientos diferentes del protocolo en relación a su predecesor, el protocolo QT. En la rama izquierda (query 0), el protocolo propuesto emplea siete slots para identificar los tags A y B. Mientras que si el protocolo QT fuera el encargado de realizar la identificación del mismo conjunto de tags (ver Figura 4.8), éste emplearía cinco slots (obviando el slot inicial ε). Esto es debido a que la ventana impide que los tags sean identificados en el slot 3 y 5 del ejemplo mostrado, forzando a realizar dos slots adicionales para obtener los k bits del ID de los tags, que con el QT no sería necesario. Sin embargo, en el caso de la otra rama (query 1), el protocolo propuesto emplea seis slots, mientras que el QT emplearía 9 (ver Figura 4.8). Esto se debe a que los tags C y D poseen la parte inicial de su ID igual, y el protocolo propuesto saca partido de las zonas comunes entre los tags gracias a los slots de tipo go-On, provocando un avance más agresivo a través de estas zonas. Además, el protocolo QT produciría un mayor número de slots libres. Por otro lado, el número medio de bits transmitidos por cada tag con la ventana es siete, mientras que en el protocolo QT, el número medio sería de 19,5 bits (ver Tabla 4.3).

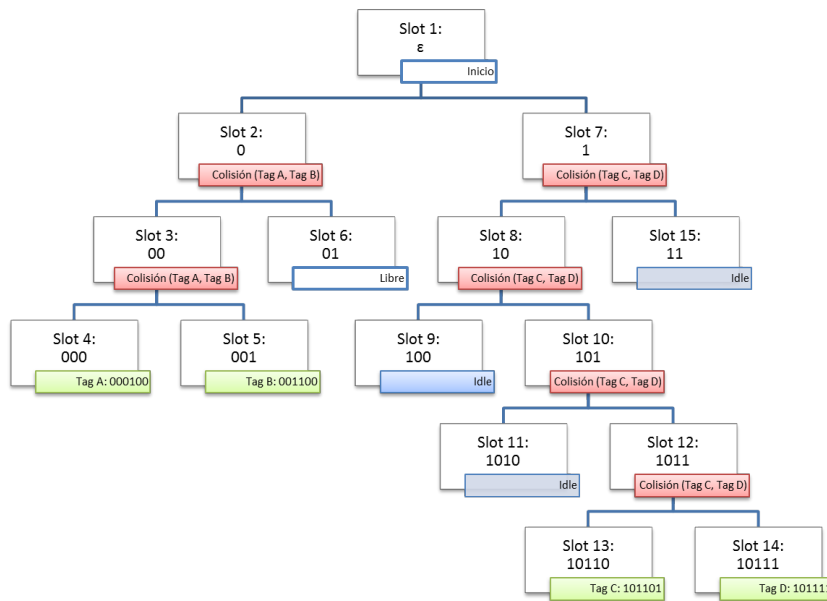


Figura 4.8: Identificación del mismo ejemplo de la Figura 4.7 con el protocolo QT

Slot	Bits transmitidos							
	Tag A		Tag B		Tag C		Tag D	
	QT	W	QT	W	QT	W	QT	W
1	6	2	6	2	6	-	6	-
2	5	2	5	2	-	-	-	-
3	4	2	4	-	-	-	-	-
4	3	1	-	-	-	-	-	-
5	-	-	3	2	-	-	-	-
6	-	-	-	1	-	-	-	-
7	-	-	-	-	5	-	5	-
8	-	-	-	-	4	2	4	2
9	-	-	-	-	-	2	-	2
10	-	-	-	-	-	-	-	-
11	-	-	-	-	3	2	3	2
12	-	-	-	-	-	1	-	-
13	-	-	-	-	2	-	2	1
14	-	-	-	-	1	-	-	-
15	-	-	-	-	-	-	1	-
Total	18	7	18	7	21	7	21	7
Media	$QT = \frac{18+18+21+21}{4} = 19,5$				$W = \frac{7+7+7+7}{4} = 7$			

Tabla 4.3: Bits transmitidos por los tags de las Figuras 4.7 y 4.8

Con el fin de aportar más información y justificar el comportamiento de la aplicación de la ventana sobre el protocolo QT se presenta el siguiente análisis matemático.

4.3.1 Análisis de la ventana aplicada al protocolo QT

En esta sección se realiza un análisis paralelo al realizado en la Sección 3.3.1 con el protocolo QT. Una vez conocido el comportamiento de las probabilidades del protocolo QT, se propone emplear la ventana para reducir el número de bits transmitidos por los tags en los momentos en que la probabilidad de colisión es alta.

4. Metodología de la ventana

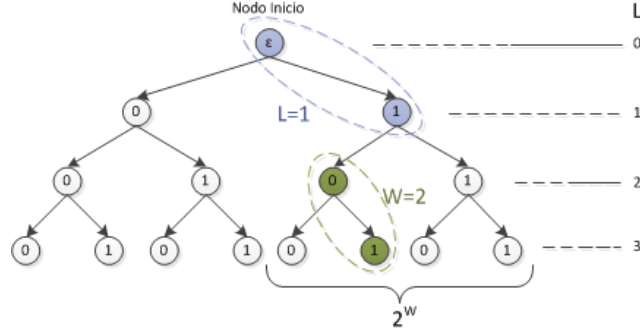


Figura 4.9: Árbol formado por el protocolo QT con ventana $W = 2$

El protocolo propuesto forma un árbol como el mostrado en la Figura 4.9. Sabiendo que el lector envía un query de longitud L a un conjunto de n tags, aquellos cuyo ID coincida con dicho prefijo responden los W bits anexos al query recibido. Se enuncia que la probabilidad de que los prefijos de X tags, de un total de n , coincidan con uno de longitud L sigue una distribución binomial, $X \sim \text{Bin}(\frac{1}{2^L}, n)$ (eq. 3.1). Suponiendo una distribución homogénea de tags se presentan las probabilidades de que se produzca un slot libre, uno satisfactorio o go-On y una colisión.

La probabilidad de que se produzca un slot libre o no conteste ningún tag, P_I^W , no varía con respecto a la expresión del protocolo QT (eq. 3.2), ya que la ventana no interviene en esta situación.

$$P_I^W = P_I^{QT} = \mathbb{P}(X = 0) = (1 - 2^{-L})^n \quad (4.3)$$

A continuación se procede a calcular la probabilidad de que se produzca un slot de tipo go-On, P_G^W . Para ello se dispone de un conjunto de n tags con sus n prefijos o queries en $S = \{s_1, \dots, s_n\}$. Para calcular esta probabilidad se fijan las condiciones de que s_n coincida con los L bits enviados por el lector, que se abrevia en s_n^L , y otra en la que no coincida, $\overline{s_n^L}$. Con estas condiciones se plantea la expresión¹:

¹Para la obtención de las expresiones mostradas se ha utilizado la siguiente ley de probabilidad $P(A) = P(A|B)P(B) + P(A|\overline{B})P(\overline{B})$

$$\begin{aligned}
 P_G^W &= \mathbb{P}(\text{goOn con } n \text{ queries}) = \\
 &= \mathbb{P}(\text{goOn con } n \text{ queries} | s_n^L) \mathbb{P}(s_n^L) + \mathbb{P}(\text{goOn con } n \text{ queries} | \overline{s_n^L}) \mathbb{P}(\overline{s_n^L}) = \quad (4.4) \\
 &= p_n
 \end{aligned}$$

De esta expresión se puede calcular:

$$\mathbb{P}(s_n^L) = \frac{1}{2^L} \quad (4.5)$$

$$\mathbb{P}(\overline{s_n^L}) = 1 - \frac{1}{2^L} \quad (4.6)$$

$$\mathbb{P}(\text{goOn con } n \text{ tags} | \overline{s_n^L}) = \mathbb{P}(\text{goOn con } (n-1) \text{ queries}) = p_{n-1} \quad (4.7)$$

Analizando la situación en la que el query s_n no coincide con el prefijo L , $\overline{s_n^L}$, la ventana de ese tag nunca colisionará con el resto de ventanas de tags, ya que su prefijo no coincidirá con el query recibido. El factor que falta por calcular se puede reescribir de la siguiente manera para facilitar su cálculo:

$$\begin{aligned}
 \mathbb{P}(\text{goOn con } n \text{ queries} | s_n^L) &= \mathbb{P}(s_1, s_2 \dots s_{n-1} \text{ no colisionan con } s_n) = \\
 &= \prod_{k=1}^{n-1} \mathbb{P}(s_k \text{ no colisiona con } s_n) \quad (4.8)
 \end{aligned}$$

Imponiendo como condición que cualquiera de los queries s_k coincida con los L bits del prefijo, se tiene la expresión s_k^L , que a su vez cumple $s_k^L = s_n^L$. Entonces, la eq. 4.8 se puede reescribir empleando la misma propiedad que en 4.4:

$$\begin{aligned}
 &\mathbb{P}(\text{goOn con } n \text{ queries} | s_n^L) = \\
 &= \prod_{k=1}^{n-1} [\mathbb{P}(s_k \text{ no colisiona con } s_n | s_k^L) \mathbb{P}(s_k^L) + \\
 &\quad + \mathbb{P}(s_k \text{ no colisiona con } s_n | \overline{s_k^L}) \mathbb{P}(\overline{s_k^L})] \quad (4.9)
 \end{aligned}$$

De aquí se conocen las siguientes expresiones:

4. Metodología de la ventana

- ◇ La probabilidad de que un query del conjunto S , s_k , coincida con el query de L bits.

$$\mathbb{P}(s_k^L) = \frac{1}{2^L} \quad (4.10)$$

- ◇ Su probabilidad complementaria, es decir, que no coincida.

$$\mathbb{P}(\overline{s_k^L}) = 1 - \frac{1}{2^L} \quad (4.11)$$

- ◇ La probabilidad de que el prefijo s_k del conjunto S no colisione con el prefijo s_n , y coincida con los L bits del query, al igual que el segundo. Es decir, que al menos dos tags responderían al query enviado por el lector, pero solo s_k respondería correctamente.

$$\mathbb{P}(s_k \text{ no colisiona con } s_n | s_k^L) = \frac{1}{2^W} \quad (4.12)$$

- ◇ La probabilidad de que no se produzca colisión entre s_k y s_n , pero s_k no coincida con los L bits del query.

$$\mathbb{P}(s_k \text{ no colisiona con } s_n | \overline{s_k^L}) = 1 \quad (4.13)$$

De esta manera la eq. 4.9 queda de la siguiente forma:

$$\mathbb{P}(\text{goOn con } n \text{ queries} | s_n^L) = \left[1 - \frac{1}{2^L} \left(1 - \frac{1}{2^W} \right) \right]^{n-1} \quad (4.14)$$

Con todo esto, se puede reescribir la expresión de la probabilidad de que se produzca un slot de tipo go-On (4.4) obteniéndose la siguiente ecuación recursiva:

$$P_G^W = p_n = \left[\underbrace{1 - \frac{1}{2^L} \left(1 - \frac{1}{2^W} \right)}_a \right]^{n-1} \frac{1}{2^L} + \underbrace{\left(1 - \frac{1}{2^L} \right)}_b p_{n-1} \quad (4.15)$$

Reescribiendo la expresión obtenida con los términos a y b para su simplificación, se obtiene:

$$\begin{aligned}
 P_G^W &= \frac{1}{2^L} a^{n-1} + b p_{n-1} = \frac{1}{2^L} a^{n-1} + b \left(\frac{1}{2^L} a^{n-2} + b p_{n-2} \right) = \{ \dots \} = \\
 &= \frac{1}{2^L} (a^{n-1} + b a^{n-2} + b^2 a^{n-3} + \dots + b^{n-1}) + \underbrace{b^n p_0}_{p_0=0} = \\
 &= \frac{1}{2^L} \sum_{k=1}^n a^{n-k} b^{k-1}
 \end{aligned} \tag{4.16}$$

Sabiendo que la expresión $a^n - b^n = (a - b) \sum_{k=1}^n a^{n-k} b^{k-1}$ y que $a - b = \frac{1}{2^{L+W}}$, se obtiene la expresión final de la probabilidad de que se produzca un slot de tipo go-On.

$$\begin{aligned}
 P_G^W &= \frac{1}{2^L} \frac{a^n - b^n}{a - b} = 2^W (a^n - b^n) = \\
 &= 2^W \left[\left(1 - \frac{1}{2^L} \left(1 - \frac{1}{2^W} \right) \right)^n - \left(1 - \frac{1}{2^L} \right)^n \right]
 \end{aligned} \tag{4.17}$$

Por último, la probabilidad de que se produzca una colisión se calcula de la misma manera que en el caso del protocolo QT.

$$\begin{aligned}
 P_C^W &= 1 - (P_G^W + P_I^W) = \\
 &= 1 - 2^W (1 - 2^{-L} + 2^{-(L+W)})^n + (2^W - 1)(1 - 2^{-L})^n
 \end{aligned} \tag{4.18}$$

En la Figura 4.10.a se grafican las expresiones obtenidas de la probabilidad de slot libre (P_I^W), probabilidad de go-On (P_G^W) y de colisión (P_C^W), empleando dos valores de ventana diferentes. Se observa como el tamaño de la ventana influye sobre P_G^W y, dada su dependencia, también sobre P_C^W ¹. En la Figura 4.10.b, se completa la información presentando un mayor rango de ventanas (desde 1-20), y se observa como a partir de aproximadamente $W = 5$ permanece constante. Por otro lado, dado que $P_I^W = P_I^{QT}$, las probabilidades de slot libre no se ven influenciadas por el tamaño de ventana.

Estos resultados evidencian que ventanas de valores inferiores pueden reducir el número de colisiones y aumentar el número de slots de tipo go-On. Este tipo de slot resulta crucial en el funcionamiento del protocolo como se observará más adelante.

¹Para la obtención de las gráficas se ha utilizado la expresión mostrada en eq. 4.16

4. Metodología de la ventana

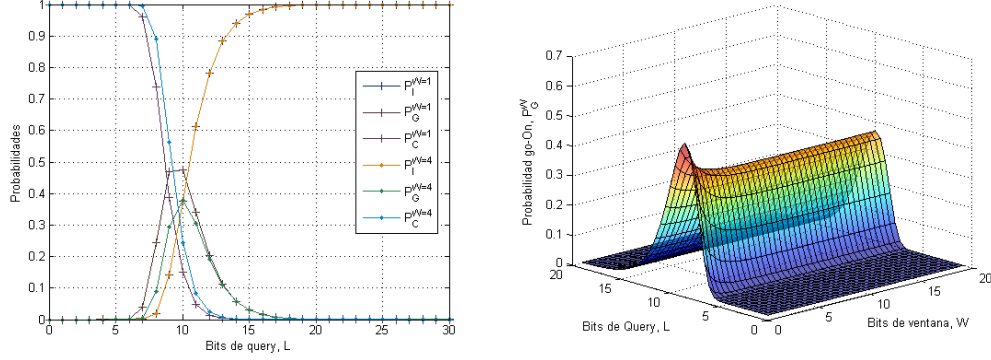


Figura 4.10: Probabilidades del protocolo QwT para un conjunto de tags $n = 1000$: (a) probabilidades para diferentes valores de W ; (b) probabilidad de slot go-On variando L y W

4.3.1.1 Cálculo del número de slots del protocolo QT con ventana de bits

A continuación se presentan las expresiones para el cálculo teórico del número de slots empleados por el protocolo QT con la ventana de bits. En [Hush 98], se especifica que para calcular esta medida es necesaria la probabilidad de visitar un nodo, P_V . Entonces, el cálculo viene dado por:

$$\tau = \sum_{L=0}^{\infty} \sum_{i=0}^{2^L-1} P_V(L_i) = 1 + \sum_{L=1}^{\infty} 2^L P_V(L) \quad (4.19)$$

Donde τ representa el número de slots o nodos visitados, i la rama de un nivel L concreto y $P_V(L)$ la probabilidad de visitar un nodo en el nivel L . A continuación, se procede a calcular P_V .

Para ello, se define un evento $A := \{v \text{ visitado}\}$, que define el acto de visitar el nodo v . En el proceso del protocolo, un nodo cualquiera v se puede visitar de dos formas diferentes (ver Figura 4.11):

- ◇ *Condición 1*, que conforma el *evento B*: si se produce una colisión en el nodo anterior, $v - 1$ y éste es visitado. Al producirse una colisión, se generan dos nuevos queries, que conforman dos hijos nuevos en el árbol, $\{v, 0\}$ y $\{v, 1\}$.
- ◇ *Condición 2*, que equivale al *evento C*: $v - W$ es un nodo visitado satisfactoriamente y v no es un nodo libre, es decir, que se produce un go-On slot

hacia v . En este caso el query enviado para el nodo $v - W$, proporciona una única respuesta, que es la del nodo v .

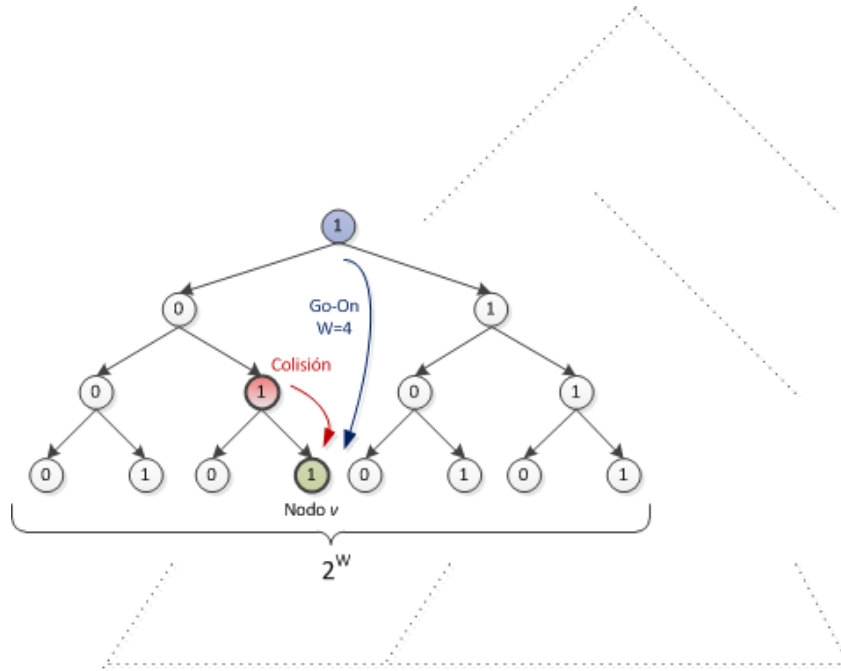


Figura 4.11: Posibilidades de visita del nodo v

Utilizando estas condiciones se puede reescribir el evento A como la unión de dos nuevos eventos:

$$A = B \cup C \tag{4.20}$$

Encontrar la probabilidad exacta de visitar el nodo v no es una tarea sencilla, dado que está fuertemente influenciada por el tipo de distribución de árbol disponible, es decir, por la distribución de los IDs de los tags. Por ello, se puede acotar superiormente la expresión del valor esperado de visitar un nodo.

$$\mathbb{P}(A) \leq \mathbb{P}(B) + \mathbb{P}(C) \tag{4.21}$$

Y lo mismo ocurre con las probabilidades de los eventos B y C :

$$\mathbb{P}(B) \leq P_C^W(v - 1) \tag{4.22}$$

4. Metodología de la ventana

$$\mathbb{P}(C) \leq \mathbb{P}(v - W \text{ go - On hacia } v) = \frac{1}{2^W} P_G^W(v - W) \quad (4.23)$$

Con las expresiones presentadas en eq. 4.21, eq. 4.22 y eq. 4.23, la probabilidad de visitar el nodo v queda restringida superiormente con una expresión que incluye P_G^W y P_C^W ya calculados anteriormente en la eq. 4.17 y la eq. 4.18 respectivamente. Finalmente, la probabilidad de visitar un nodo en el nivel L queda de la siguiente forma:

$$P_V(L) = \begin{cases} 1, & L = 0 \\ P(A), & L > 0 \end{cases} \quad (4.24)$$

Y para el calculo del número de slots se emplea la expresión dada en eq. 4.19.

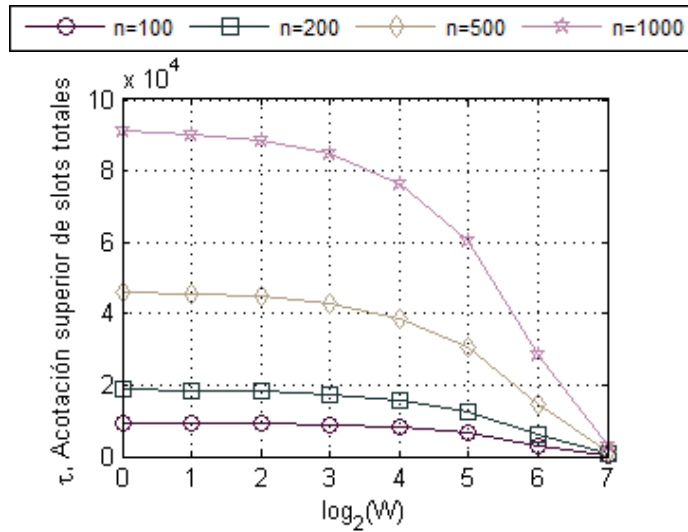


Figura 4.12: Resultados ante diferentes valores de W aplicados sobre el protocolo QT, empleando la eq. 4.19

La Figura 4.12 muestra la acotación superior del cálculo del número de slots que utiliza el protocolo QT con la ventana. Se muestra el decrecimiento del número de slots que provoca el aumento de W para identificar la totalidad de los tags. Para el valor $W = k$, el protocolo propuesto se comporta de forma similar al protocolo QT, ya que los tags contestan todos los bits del ID restantes. Y cuando $W = 1$, entonces su comportamiento se asemeja al de un protocolo de tipo BTA, ya que los

tags contestan únicamente un bit en caso de que el query recibido coincida con el ID del tag. El menor número de slots se consigue para mayores tamaños de W .

Puesto que la definición de la expresión matemática del número de slots resulta un problema que requiere un profundo estudio, y no es el propósito de esta tesis, se ha procedido a realizar una serie de simulaciones que permitan el análisis de la ventana.

4.3.2 Simulaciones para diferentes valores de W

En esta sección se presentan una serie de simulaciones realizadas empleando diferentes valores de W , para observar cómo afecta el tamaño de la ventana al comportamiento del protocolo QT. Para ello, mediante Matlab R2012b, se ha definido una simulación con un lector y un número variable de tags, n . Los ID de los tags propuestos se distribuyen de forma uniforme, es decir, que la probabilidad de definir cada bit de los IDs de los tags con 0 o 1 es del 50 %. Además, para la longitud de dicho ID, k , se ha asumido un valor de 96 bits. Los ID son generados aleatoriamente, empleando una semilla variable, en cada una de las 100 iteraciones que se realiza cada simulación. Los resultados mostrados se han obtenido a su vez, de la media de los resultados de cada una de las iteraciones.

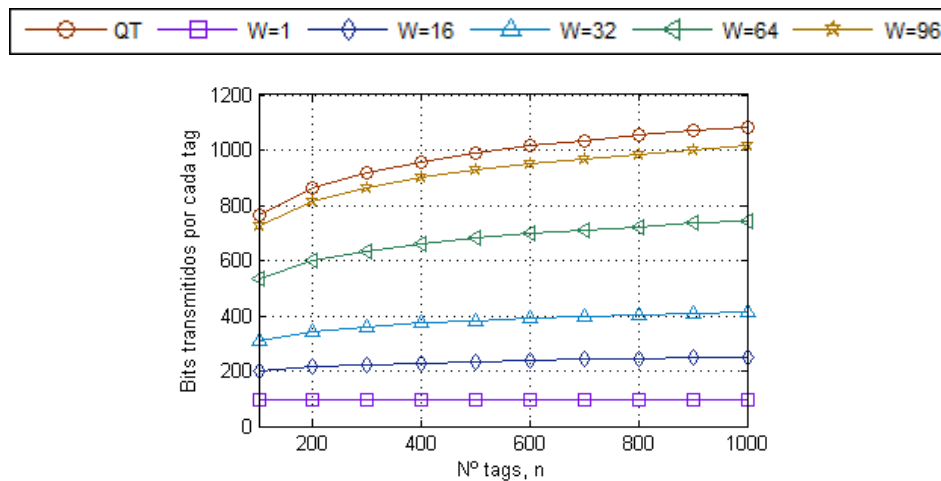


Figura 4.13: Influencia del tamaño de la ventana en los bits transmitidos por cada tag del protocolo QT

4. Metodología de la ventana

El principal beneficio de la aplicación de la ventana al protocolo QT es la clara disminución de los bits que transmiten cada uno de los tags a lo largo del ciclo de interrogación. Como se puede observar en la Figura. 4.13, este número disminuye con el tamaño de la ventana, W . Estos prometedores resultados muestran que para pequeños valores de W , el número de bits que transmite cada tag se acerca mucho al de la longitud de su ID, k , es decir, que transmiten casi únicamente esos bits. Así, se consigue reducir la reiteración de transmisiones de los mismos bits del ID. Por otro lado, en el caso en el que $W = k$, el protocolo con la ventana propuesta se comporta de forma similar al QT y además, el número de bits transmitidos por los tags es ligeramente inferior, ya que los tags de la ventana propuesta no contestan los bits contenidos en el query recibido. Es decir, que para un ID de tag p_1, \dots, p_k siendo $p_i \in \{0, 1\}$, el tag cuyo prefijo p_1, \dots, p_L coincide con el query q_1, \dots, q_L recibido, contesta los bits restantes p_{L+1}, \dots, p_k . Mientras que el protocolo QT puro comparado en la figura responde el ID al completo en cada query.

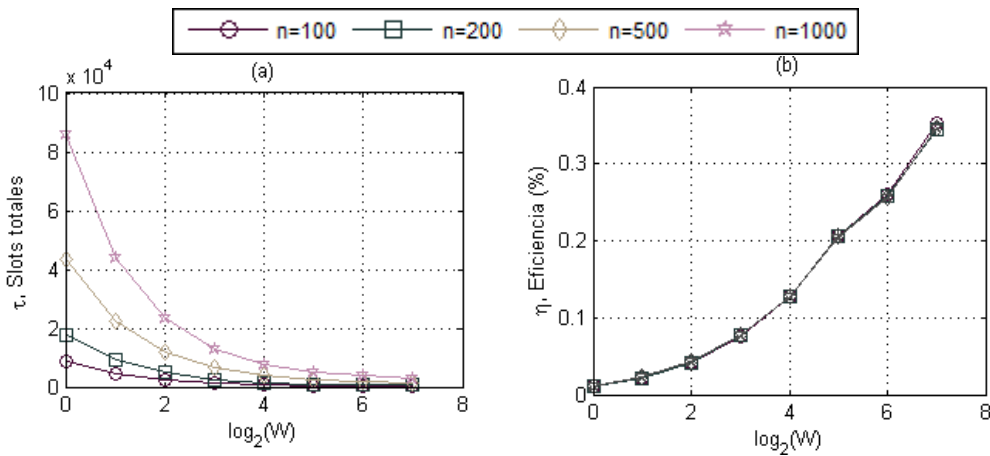


Figura 4.14: Influencia del tamaño de la ventana sobre el protocolo QT: (a) en el número de slots (τ), (b) en la eficiencia de slots

En la Figura 4.14 se muestran el número de slots empleados en la identificación de diferentes conjuntos de tags o número de nodos visitados en el árbol virtual compuesto por los diferentes tipos de nodos (libre, colisión, go-On y satisfactorio) τ , y la eficiencia de slots η , para diferentes tamaños de ventana y conjuntos de tags. En ambas gráficas se observa la influencia de la ventana sobre el protocolo QT.

4.3 QT con ventana constante

Los resultados tienen una tendencia similar a la mostrada por las expresiones desarrolladas en la Sección 4.3.1.1. La simulación muestra que los valores esperados obtenidos son inferiores a los teóricos, ya que se trataba de una acotación superior. La reducción de los slots conforme aumenta el tamaño de ventana, W , se produce por la necesidad del lector de interrogarlos hasta conseguir obtener su ID completo, y, por tanto, a menor tamaño de ventana, mayor número de queries debe transmitir. Valores de W cercanos a $W = k$, producen comportamientos similares al protocolo QT sin ventana, ya que los tags contestan todos los bits del ID restantes. En cuanto a la situación opuesta ($W = 1$) el comportamiento se asemeja al de un protocolo de tipo BTA, ya que los tags contestan únicamente un bit en caso de que el query recibido coincida con el ID del tag. De forma inversa se comporta la eficiencia en slots mostrada en la Figura 4.14.b, calculada empleando la eq. 4.1. En el caso de distribuciones homogéneas de los ID de los tags, la máxima eficiencia de slots que se consigue es la del protocolo QT en el mejor de los casos ($W = k$), aunque no se produce mejoría en el número de bits transmitidos por cada tag (ver Figura 4.13).

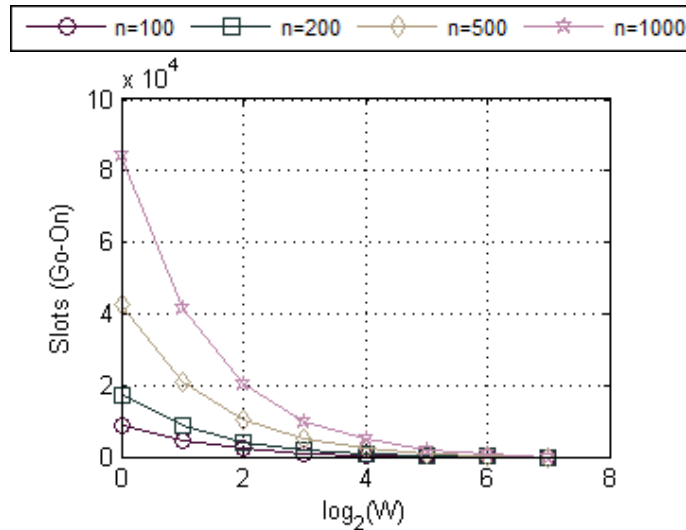


Figura 4.15: Slots go-On generados por la utilización de la ventana

El crecimiento en el número de slots se produce por la aparición del nuevo tipo de slot llamado go-On. Este tipo de slot, causado por la utilización de ventanas de tamaño inferior a la longitud del ID, $W < k$, es necesario para la obtención del ID completo del tag interrogado. En la Figura 4.15 se observa cómo el número de

4. Metodología de la ventana

go-On slots se comporta de forma muy similar a la gráfica del número de slots. Es decir, que el incremento total de slots mostrado en la Figura 4.14 con valores de W inferiores a k viene provocado principalmente por los slots de tipo go-On. De esta manera, se concluye que este tipo de slots resultan críticos para la finalización del ciclo de identificación lo antes posible, o dicho de otro modo, para obtener una latencia mínima. Al emplear tamaños de ventana inferiores a k , el lector necesitará un mínimo de k/W slots más que el protocolo QT para obtener el ID de cada tag. Con el aumento de go-On slots, es muy probable que el protocolo aumente su latencia y resulte en un tiempo de identificación alto.

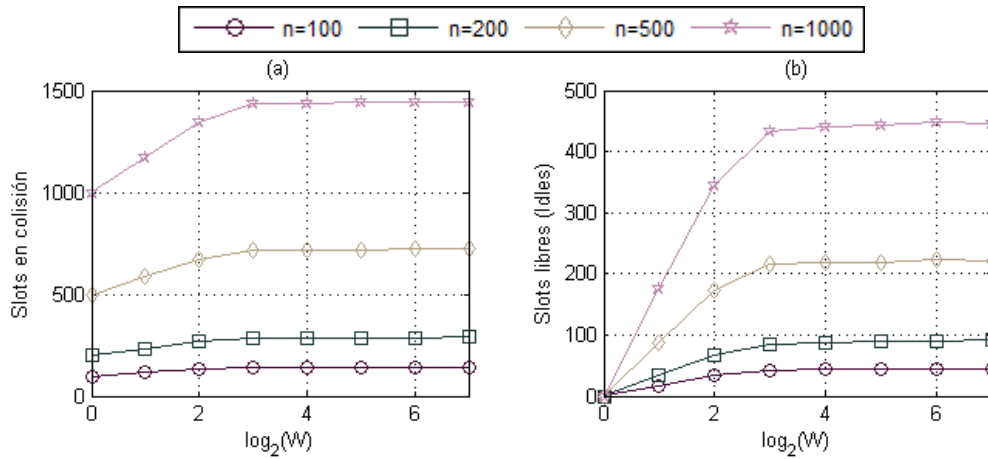


Figura 4.16: Influencia del tamaño de la ventana sobre el protocolo QT: (a) en el número de colisiones, (b) en el de slots libres

Sin embargo, a diferencia del efecto que se produce en los slots de tipo go-On, valores pequeños de W provocan grandes descensos en el número de colisiones y slots libres. En la Figura 4.16 se exponen las gráficas de la influencia del tamaño de ventana sobre: (a) las colisiones producidas, y (b) los slots libres en la identificación de n tags. Especialmente, debe destacarse que para valores de W inferiores a 10 se producen descensos importantes en ambas medidas. Esto es debido a dos razones principalmente:

- ◇ Se ha observado en el análisis teórico de la Sección 4.3.1, que valores bajos de W aumentan la probabilidad de que se produjera un slot de tipo go-On (ver Figura 4.10). Ésta influye directamente sobre la probabilidad de que

se produzca una colisión, reduciéndola, lo que se refleja en un importante descenso de las colisiones cuando se emplean valores de W bajos, y se ha confirmado, también, mediante las simulaciones.

- ◇ Al reducirse el número de colisiones, la probabilidad de que al formar nuevas queries se produzcan slots libres también se reduce y, por ello, el número de slots libres desciende drásticamente con valores bajos de W . Cabe destacar que para el caso de $W = 1$ no se producen slots libres al igual que en los protocolos de tipo BTA.

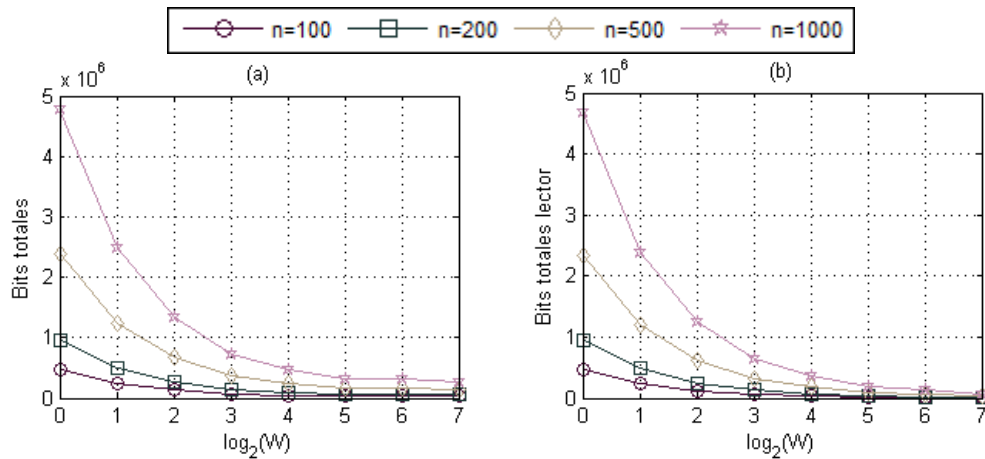


Figura 4.17: Influencia del tamaño de la ventana sobre el protocolo QT: (a) en el número total de bits transmitidos por el lector y los tags, (b) en el número de bits transmitidos por el lector

El mencionado aumento de los slots de tipo go-On cuando $W < k$, provoca un importante aumento en el número total de bits transmitidos, Figura 4.17.a, que está causada principalmente por el aumento del número de bits transmitidos por el lector, mostrado en la Figura 4.17.b. El número de bits totales desciende con el incremento de W . Es decir que se comporta con una tendencia similar a la de los slots go-On. La ventaja que producían los tamaños pequeños de ventana mostrados en la Figura 4.13, se ve ensombrecida por el gran aumento en los bits transmitidos por el lector que se muestra en la Figura 4.17.b. Esto se debe a que el lector necesita enviar un mayor número de queries y más largos (formados por un mayor número

4. Metodología de la ventana

de bits), sobre todo al final de la identificación del tag. Para clarificar la situación con un ejemplo, supongamos que $k = 20$ bits, y $W = 2$ bits. Los tres últimos queries que debe enviar el lector, suponiendo que no se producen colisiones, serían de 14, 16 y 18 bits ya que en cada uno de ellos la respuesta del tag sería de 2 bits. Interpretada por el lector y añadida al query enviado de 16 bits, el lector formaría uno nuevo de 18, lo transmitiría a los tags y así sucesivamente hasta obtener el ID completo.

4.3.3 Conclusiones

Se presenta una nueva metodología que limita el número de bits transmitidos por cada tag. La ventana constante le indica a cada tag el número de bits a transmitir si su ID coincide con el query recibido. Una vez transmitidos los bits, si el lector los recibe correctamente, los adjunta al query transmitido y transmite un nuevo query que obtendrá una respuesta segura. La aplicación de la ventana al protocolo QT sigue manteniendo la característica *memoryless*, gran ventaja del protocolo QT, que permite una mayor simplicidad del hardware de los tags. La ventana aporta las siguientes ventajas/desventajas al protocolo QT:

- ◇ Los tamaños pequeños de ventana (bajos valores de W), proporcionan una buena eficiencia de bits transmitidos por los tags, es decir que se reduce el número de bits transmitidos por éstos, y decrece considerablemente el número de colisiones y ciclos libres.
- ◇ El número de slots que necesita el protocolo aumenta en gran medida cuanto menor es W . Esto se debe a la aparición de los slots go-On, que además provocan el aumento del número de bits que transmite el lector para completar el ciclo de interrogación.
- ◇ Los valores de W cercanos a k hacen que el protocolo con la ventana se comporte de la misma forma que su protocolo origen, el QT en este caso, causando un comportamiento similar. Es decir, que los slots se reducen pero el problema del elevado número de bits que envían los tags no se solventa.

4.4 Query window Tree (QwT)

Las conclusiones obtenidas en el protocolo QT con la aplicación de la ventana constante han permitido que se plantee el desarrollo de un protocolo basado en el QT, en el que los tags envían una cantidad de bits concreta en cada slot dependiendo de las necesidades de cada momento. Es lo que, en la presente tesis, se ha denominado la ventana dinámica. Se ha concluido que los tamaños pequeños de la ventana, W , son beneficiosos para reducir las colisiones y los slots libres, así como para disminuir la cantidad de bits transmitidos por los tags. En cambio, el exceso de slots go-On perjudicaban en gran medida el comportamiento final del protocolo. Por tanto, se pretende que los tags adquieran valores bajos de W cuando se vayan a producir colisiones, pero que se minimicen en la mayor medida posible los slots de tipo go-On. Para ello, el tamaño de la ventana deberá hacerse mayor cuando exista certeza de que no se van a producir colisiones. De esta forma, se conseguirá identificar al tag en menos slots que si la ventana permaneciera con un valor pequeño, W . Se presenta por tanto un protocolo novedoso basado en el QT denominado Query window Tree (QwT) [Landaluce 13b]. Del mismo modo que en el caso anterior con la ventana constante, los beneficios de esta metodología se presentan asumiendo una perfecta sincronización entre las respuestas de los tags, de forma que lleguen los mensajes simultáneamente para que el lector sea capaz de identificar los bits idénticos. Entonces, al confirmar el mensaje recibido con el CRC, pueda considerar que es correcto independientemente del número de respuestas recibidas.

El protocolo propuesto QwT trabaja de forma similar al QT con la ventana constante, con la diferencia de que en este caso el tamaño de la ventana se actualizará de forma dinámica. QwT sigue manteniendo la característica *memoryless* de su protocolo origen (QT), manteniendo la simplicidad en el costo de los tags. Para el correcto desarrollo del protocolo, los tags implementan un puntero que se coloca en el punto indicado por el tamaño de ventana calculado (W) y marca el fin de la transmisión del ID en cada slot. Para la realización de este cálculo de W se desarrolla una función, $f(x)$, que proporciona el tamaño de ventana más adecuado para cada slot. Pueden emplearse varias estrategias a la hora de calcular dicha función. Una posible estrategia es utilizar una estimación de los tags existentes para dar un valor a W . Sabiendo el total de tags que transmiten en la zona de interrogación, el

4. Metodología de la ventana

lector podría realizar los cálculos necesarios para adaptar la ventana a las necesidades de cada momento. Sin embargo, la estimación presenta una serie de desventajas [Wu 13]:

- ◇ Incrementa la complejidad computacional del lector y en ciertas ocasiones también de los tags. Los estimadores convencionales empleados en RFID, Vogt [Vogt 02a], Bayesiano [Wu 10], MAP [Chen 09], buscan el máximo valor en un amplio rango de conjuntos de tags, lo que añade una gran complejidad computacional. Además, algunos de ellos requieren que los tags implementen funciones *hash* ([Qian 10, Zheng 11]), incrementando el costo de los tags.
- ◇ El error de estimación que conlleva el empleo de estimadores perjudica al protocolo provocando un aumento en el número de slots, tanto libres como colisionados.
- ◇ Se necesitarían ciclos de interrogación adicionales para establecer un número de tags estimado, lo que aumentaría el número de bits transmitidos en el proceso, anulando la principal finalidad del empleo de la ventana, como ocurre en el protocolo presentado en [Lai 13].
- ◇ Todos los estimadores necesitan partir de un valor inicial que en la gran mayoría de ellos determina el buen procedimiento de la estimación. Hay estimadores que realizan una pre-estimación de este valor inicial, lo que añade más complejidad al protocolo aumentando el número de bits transmitidos [Wu 13].

Por ello la estimación de tags queda descartada para la tesis presentada en este documento. Para la determinación del tamaño de ventana más adecuado en cada momento se va a optar por emplear datos conocidos y en base a ellos, determinar un valor de W para cada slot. En la Figura 4.18 se muestra un esquema inicial de cómo calcular la ventana en el proceso de lectura, teniendo en cuenta parámetros como la longitud del *query* L , del ID de los tags k , o de la ventana empleada en el slot anterior W_{i-1} , así como un parámetro de ajuste β para adaptar el protocolo a cada situación. El valor calculado deberá transmitirse a los tags junto a cada *query*, para

4.4 Query window Tree (QwT)

que el protocolo siga manteniendo la característica *memoryless*. Los tags recibirán el mensaje, del que obtendrán el *query* para compararlo con su ID y un parámetro adjunto que indicará el tamaño de la ventana (W). Si la comparación es correcta, enviarán el número de bits indicado en W . El conjunto de bits transmitido por cada tag conforman la ventana de bits.

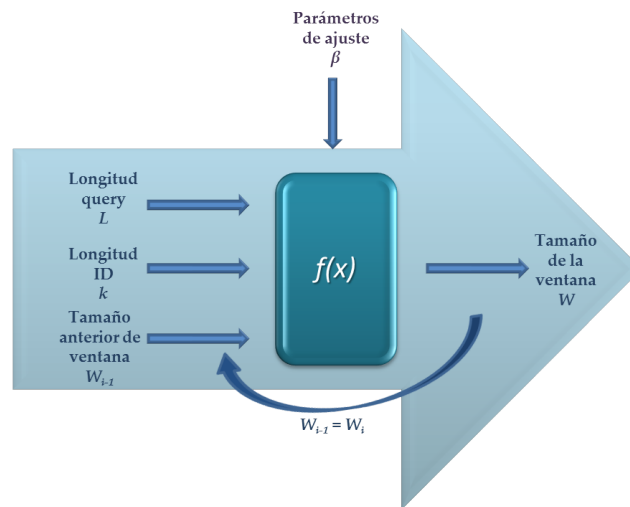


Figura 4.18: Esquema de aplicación de la función en el protocolo QwT

Teniendo en cuenta todos estos parámetros se pretenderá que la función se comporte de la siguiente manera:

- ◇ Para valores pequeños de L , se propondrán valores de W pequeños. Gracias al análisis realizado en la Sección 3.3.1, se ha observado que en las distribuciones de tags homogéneas, la probabilidad de colisión es muy elevada en valores de L pequeños. Acorde al análisis realizado con la inserción de la ventana en el protocolo QT, en la Sección 4.3.1, se ha observado que estableciendo un valor de W pequeño se incrementa la probabilidad de go-On, reduciendo a su vez el número de colisiones. Esto mantendrá una eficiencia de transmisión de bits de los tags más elevada, ya que se desaprovecharán menos, permitiendo un ahorro en tiempo y energía consumida.
- ◇ Para valores de L mayores ocurre el caso contrario. La probabilidad de colisión pasa a ser muy pequeña. Con el fin de reducir lo máximo posible los

4. Metodología de la ventana

slots de tipo go-On, se propondrán valores altos de W para identificar los tags a la mayor brevedad posible. Esta decisión viene motivada por los resultados observados en la ventana constante, que con valores pequeños de W , el número de slots go-On aumentaba considerablemente perjudicando la puesta en marcha del protocolo.

Sin embargo, para los casos en los que se produce una colisión o un slot libre se optará por la no modificación del tamaño de la ventana. Es decir, que la función $f(x)$ solo va a ser aplicada cuando se produzca un slot de tipo go-On. Para ello se presenta la eq. 4.25, contemplando tres casos posibles que dependen de dos variables: L , que representa la longitud del *query* enviado en el slot actual i o en el anterior $i - 1$; y t , que representa el número de respuestas diferentes interpretadas por el lector. Se debe indicar que el lector no es capaz de interpretar el número exacto de respuestas recibidas, aunque sí puede conocer si se ha producido una colisión ($t > 1$), una respuesta satisfactoria ($t = 1$) o un slot libre ($t = 0$).

$$W_i = \begin{cases} f(L_i), & t = 1, L_{i-1} < L_i \\ W_{i-1}, & t \neq 1, L_{i-1} < L_i \\ 1, & t \geq 0, L_{i-1} \geq L_i \end{cases} \quad (4.25)$$

Mediante las dos variables mencionadas, L y t , se definen todos los casos posibles que pueden ocurrir en una identificación de los tags. La comparación de la longitud del *query* actual L_i con el anterior L_{i-1} indica al protocolo si continúa interrogando la misma rama del árbol o si ha finalizado, obteniendo un nuevo *query* de la pila (de menor tamaño). Cuando el protocolo continúa en la misma rama, consulta qué tipo de respuesta ha recibido en el último slot. Si se produce un slot libre o una colisión, aplica la ventana calculada en el slot anterior, W_{i-1} , y las condiciones de que se suceda cualquiera de ambos se puede reescribir de la siguiente manera: $t = 0 \cup t > 1 \Rightarrow t \neq 1$. En caso de que se haya producido un slot de tipo go-On, el lector identifica únicamente una respuesta, $t = 1$, y aplica una función heurística propuesta en la Sección 4.4.1. Por otro lado, si el *query* actual pertenece a otra rama del árbol, su tamaño va a ser igual o inferior al del *query* anterior. En tal caso, el tamaño de ventana debe volver a un valor inicial. Acorde a las pruebas realizadas, se ha observado que un comportamiento beneficioso para el protocolo es que el valor de W pase al mínimo posible ($W = 1$). Así pues, con cada identificación de tag, W

tomará el valor 1 nuevamente para reiniciar el proceso con una nueva identificación de otro tag. Este comportamiento presenta una serie de ventajas y desventajas que afectan al comportamiento global del protocolo:

- ◇ Las posibles colisiones que se puedan producir provocarán una pérdida de tiempo y energía mínima de un bit, y además, los dos nuevos *queries* generados en la colisión obtendrán respuesta nuevamente, evitando slots libres.
- ◇ Si la respuesta es un slot de tipo go-On, la ventana tendrá que crecer de tamaño acorde a la función escogida, $f(L)$. Lo que aumentará el número de slots que conlleva el correspondiente incremento en tiempo y energía consumida.

Al igual que en el protocolo propuesto con la ventana constante, el protocolo QwT también almacena los *queries* generados en una pila LIFO y muestra un procedimiento similar al del QT con la ventana constante. La principal diferencia es que, en el segundo, no era necesario transmitir el tamaño de la ventana a los tags en cada slot. El lector del QwT debe enviar un comando formado por los L bits del *query*, acompañado del tamaño de la ventana, W . Este valor se transmite empleando $\log_2 W$ bits. En la Figura 4.4 mostrada anteriormente se refleja un esquema sobre los comandos que intercambian lector y tags. El tag recibe el *query* y el tamaño de ventana, compara el primero con su ID y actualiza un puntero de ventana con el valor $L + W$. En caso de que la comparativa sea positiva, el tag comienza a transmitir su ID desde el último bit del *query* comparado hasta donde indica el puntero configurado. Es decir, que envía los W bits contiguos al *query* recibido, w_{L+1}, \dots, w_{L+W} . Junto a la ventana de bits se calcula el CRC correspondiente a esos datos, para que el lector sea capaz de identificar el tipo de respuesta recibida. El procedimiento del protocolo se muestra en un nuevo diagrama de flujo, en la Figura 4.19.

Del mismo modo que con la ventana constante, las respuestas de los tags (t) que coincidan con el *query* enviado llegan de forma sincronizada al lector. Para determinar qué tipo de respuesta es, el lector analiza el CRC recibido pudiendo darse los cuatro casos mencionados anteriormente, que determinan el procedimiento de actuación:

4. Metodología de la ventana

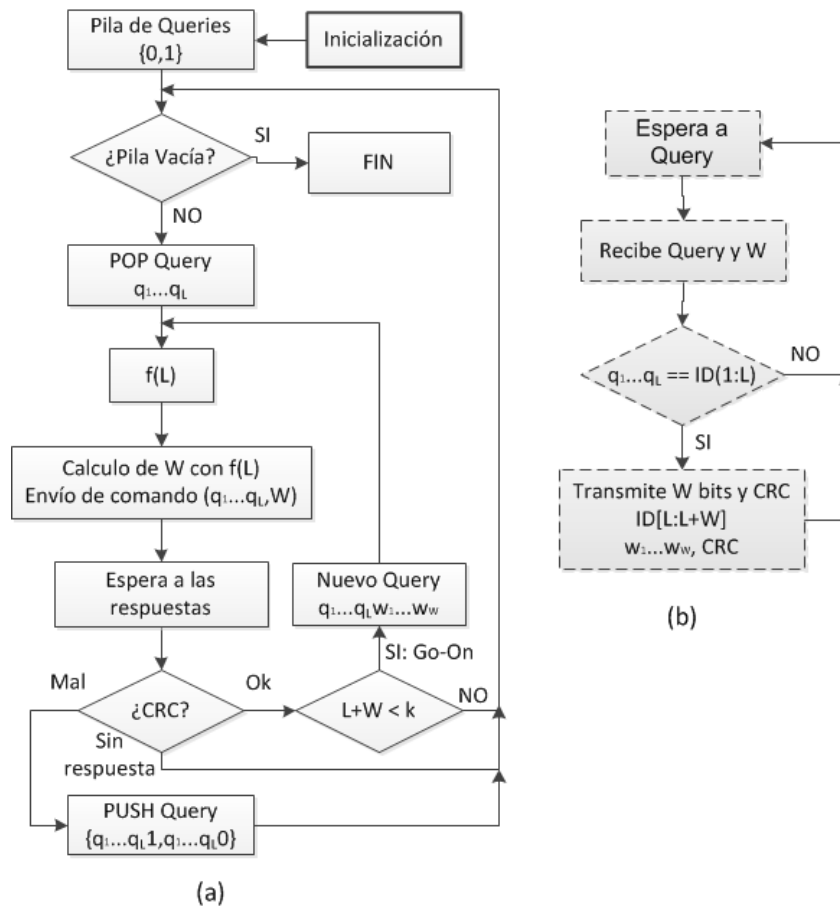


Figura 4.19: Diagrama de flujo del protocolo QwT: (a) lector y (b) tag

- ◇ *Slot libre* ($t = 0$): cuando el lector no recibe respuesta alguna al *query* enviado. El lector desecha el *query* enviado y obtiene otro de la pila LIFO.
- ◇ *Slot en colisión* ($t > 1$): cuando el lector recibe varias respuestas y el CRC no certifica los datos recibidos. En este caso el lector genera dos nuevos *queries* añadiendo un 1 y un 0 al último *query* enviado, obteniendo $q_1, q_2 \dots q_L, \mathbf{1}$ y $q_1, q_2 \dots q_L, \mathbf{0}$.
- ◇ *Slot go-On* ($t = 1$): cuando el lector recibe una o varias respuestas y el CRC certifica los datos recibidos. Para el caso en que $L + W < k$, se considera un slot de tipo go-On, ya que hasta no obtener los k bits del tag, no se le puede dar por identificado. El lector genera un nuevo *query* añadiendo al último

enviado la ventana recibida, $q_1, q_2 \dots q_L, w_1, w_2 \dots w_W$.

- ◇ *Slot satisfactorio* ($t = 1$): cuando contesta un único tag y se cumple que $L + W = k$. El tag queda identificado con el *query* enviado y la ventana recibida, un total de k bits. A continuación, el lector continúa el proceso de identificación con otro *query* de la pila hasta que se vacía.

4.4.1 Funciones de ventana

Se ha comentado anteriormente la intencionalidad de que la ventana dinámica tenga un comportamiento determinado basándose en la evolución de la probabilidad de colisión en las distribuciones de tag homogéneas. Con ello, se pretende que la función determine el tamaño de la ventana en cada momento tomando valores pequeños cuando los *queries* sean reducidos, y valores mayores cuando éstos sean elevados. Para ello, en esta sección se proponen tres funciones heurísticas, $f(x)$, diferentes que sugieren un comportamiento similar al deseado para la actualización del tamaño de ventana. Éstas son: lineal, cuadrática y exponencial. Como se muestra en la figura anterior 4.18, el lector utiliza valores conocidos para generar el tamaño de ventana adecuado, W . En este caso se emplea la dimensión del *query* actual, L , y por ello la función heurística se denomina $f(L)$. Además, se va a emplear un parámetro de ajuste, β , que permite adaptar la función a la distribución de tags situada en la zona de interrogación. Hay que destacar que, además de la función heurística presentada en cada caso, el tamaño de ventana calculado, W , siempre va a ir limitado por la expresión 4.26, donde W_i representa el tamaño de ventana antes de aplicarle la función limitadora, y W_{i+1} , el valor ya limitado, que se transmite finalmente a los tags.

$$W_{i+1} = \begin{cases} W_i, & L_i + W_i \leq k \\ k - L_i, & L_i + W_i > k \end{cases} \quad (4.26)$$

- ◇ **Función Lineal:** el lector utiliza el valor actual del tamaño del *query*, L , para determinar el tamaño de la ventana en el slot, W .

$$f(L) = \left(\frac{k}{k - \beta} \right) \times L \quad (4.27)$$

4.4 Query window Tree (QwT)

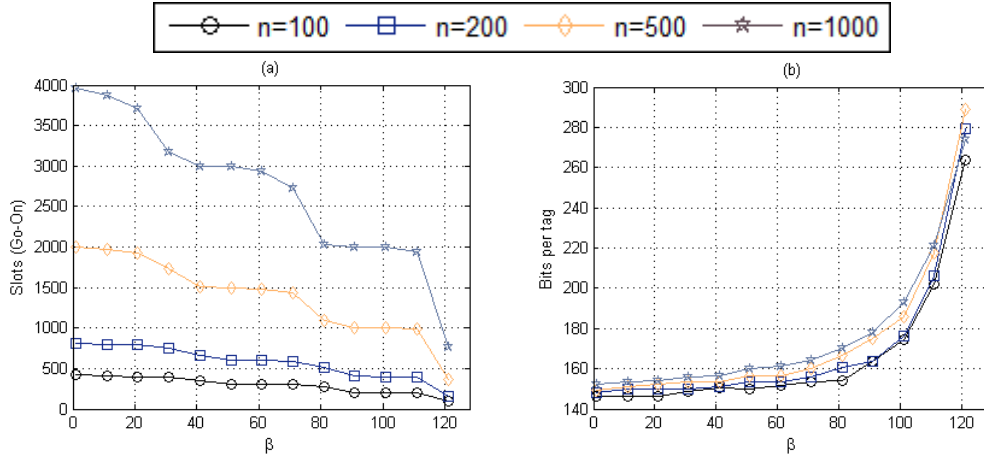


Figura 4.21: Variación del comportamiento del protocolo QwT con función Lineal, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits

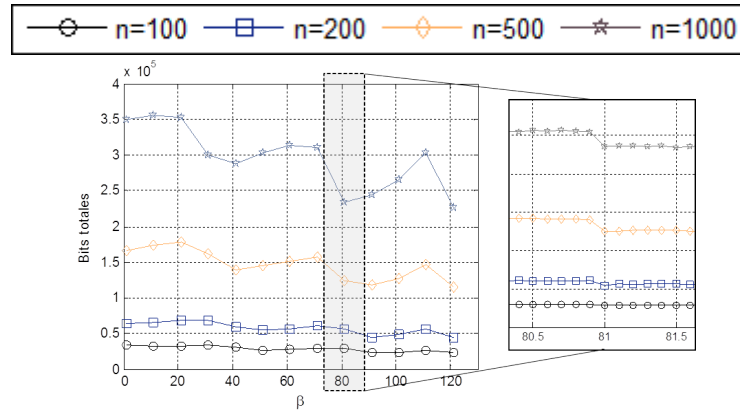


Figura 4.22: Variación del comportamiento del protocolo QwT con ventana lineal para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n .

confirma en la Figura 4.21.b. Por tanto, afinando la simulación en torno a ese rango de valores, el valor final escogido es de $\beta = 81$.

- ◇ Función **Cuadrática**: en esta función, el lector obtiene el valor de W en base al cuadrado del valor de L .

$$f(L) = \frac{k}{(k - \beta)^2} \times L^2 \quad (4.28)$$

4. Metodología de la ventana

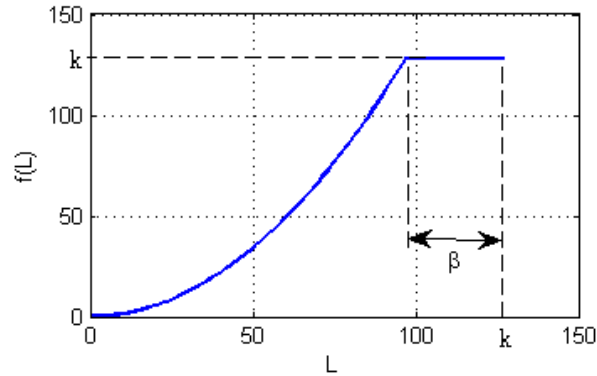


Figura 4.23: Representación de $f(L)$ cuadrática

Del mismo modo que en la función lineal, el valor de β se emplea para determinar qué valor de L se corresponde con el mayor tamaño de ventana posible solicitado a los tags. En la Figura 4.23 se muestra la representación gráfica de la función cuadrática, eq. 4.28, limitada en k bits.

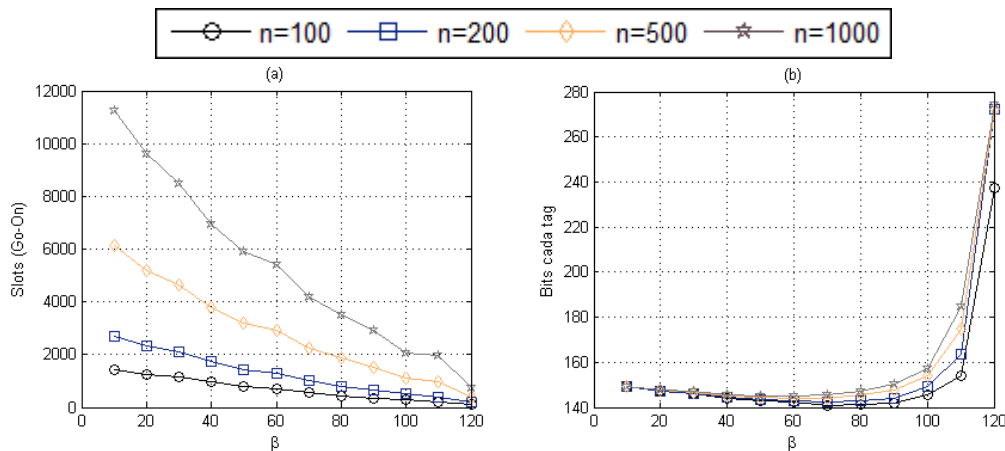


Figura 4.24: Variación del protocolo QwT con función cuadrática, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits

Al igual que en el caso anterior, se sintoniza el valor de β para obtener el mejor comportamiento del protocolo ante una distribución de tags homogénea. En la Figura 4.24.a se muestran los slots de tipo go-On, y en la Figura 4.24.b el número de bits medio que transmite cada tag.

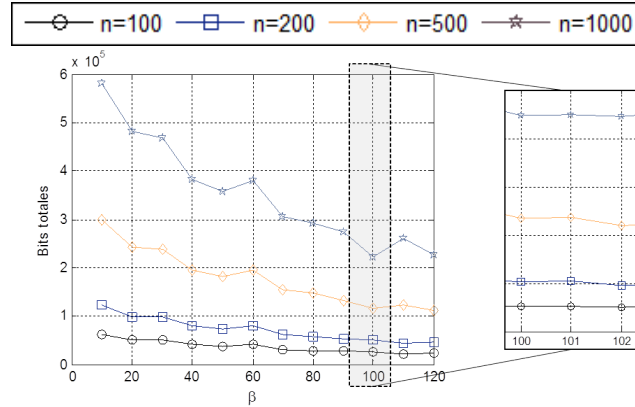


Figura 4.25: Variación del protocolo QwT con ventana cuadrática para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n .

La ventana cuadrática muestra que el número de go-On slots, Figura 4.24.a, descende conforme aumentan los valores de β . Para valores inferiores se produce un mayor consumo de este tipo de slots. Sin embargo, en el caso de los bits enviados por cada tag, Figura 4.24.b, el aumento se produce de forma más brusca en los valores mayores de β de la ventana cuadrática que en los de la lineal. Del mismo modo que en el caso de la ventana lineal, se tratará de buscar un valor de β que reduzca el número de slots de tipo go-On, manteniendo el menor número posible de bits transmitidos por cada tag. Para establecer dicho valor se acude de nuevo a la Figura 4.25, que muestra los bits totales enviados entre lector y tags. Se observa que en el rango de valores de $\beta = [90, 110]$ el número de bits totales es similar al de $\beta = k$, con la diferencia de que el número de go-On slots y de bits por tag es menor. Por tanto, se escoge un valor de $\beta = 102$.

- ◇ **Función Exponencial:** la última de las funciones propuestas de actualización de la ventana es la exponencial [Landaluce 13b]. La función propuesta para este caso, eq. 4.29, es una función exponencial de amplitud k . En ésta, el valor de $1/\beta$ define la constante de tiempo de crecimiento, Figura 4.26.

$$f(L) = k \times (1 - e^{-\beta \times L}) \quad (4.29)$$

4. Metodología de la ventana

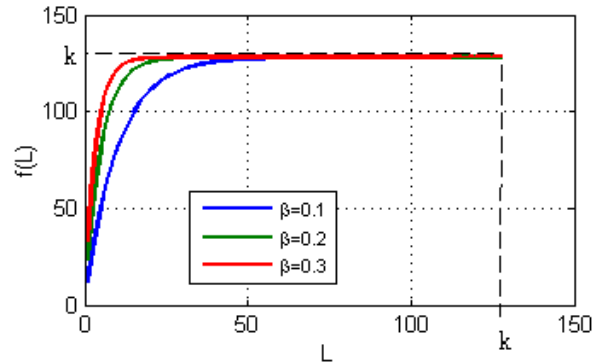


Figura 4.26: Representación de $f(L)$ exponencial

Para ajustar el valor de β se realizan de nuevo las simulaciones variando el valor de β en el rango adecuado. En este caso, el rango de valores empleado varía con respecto a las funciones anteriores. Las mayores variaciones de la $f(L)$ se obtienen en el rango de $\beta = [0, 1]$. Realizando la simulación de la identificación de diferentes conjuntos de tags, n , para los valores de β indicados en el rango se presentan los siguientes resultados: en la Figura 4.27.a se muestran los go-On slots empleados y en la Figura 4.27.b los bits de promedio transmitidos por cada tag.

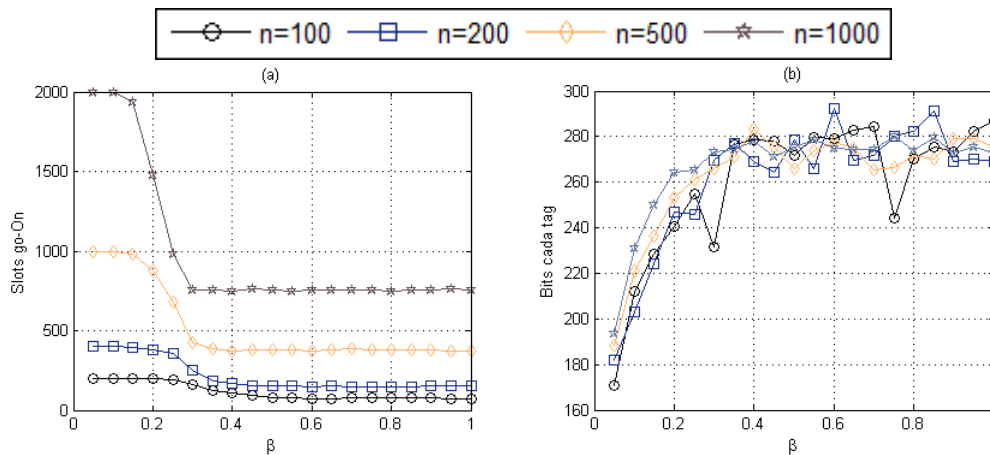


Figura 4.27: Variación del protocolo QwT con función exponencial, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits

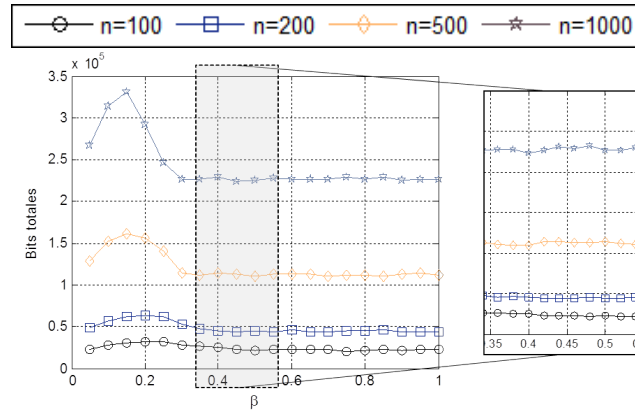


Figura 4.28: Variación del protocolo QwT con ventana cuadrática para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n .

Se observa que el comportamiento del protocolo con la ventana exponencial es similar al de las ventanas anteriores. El número de go-On slots disminuye conforme aumenta el valor de β hasta valores cercanos a $\beta = 0,5$, para luego permanecer constante. De la misma manera, los bits transmitidos por cada tag alcanzan un máximo aproximado a partir de $\beta = 0,6$. De nuevo, para escoger el valor más adecuado de β se acude al número de bits totales empleados en la identificación, Figura 4.28. Esta gráfica muestra una tendencia similar a las anteriores mostradas. A partir de $\beta = 0,4$, el número de bits se estabiliza en valores constantes. Por ello, se escoge como el valor más adecuado el de $\beta = 0,5$. Para conjuntos menores de tags, n , el protocolo trabaja mejor con el valor escogido de β que con el valor inmediatamente inferior. Por ello, y para preservar el número de bits enviados por cada tag en mayor medida, afectando lo menos posible al número de go-On slots, se ha escogido ese valor.

4.4.2 Comparativa entre ventanas

En este apartado se realiza una comparativa entre las tres funciones heurísticas presentadas. Cada una de ellas ha sido sintonizada para trabajar de la forma más adecuada en un entorno de tags con sus ID distribuidos de forma homogénea. Para

4. Metodología de la ventana

realizar la comparativa general entre ellas se van a tener en cuenta, además de las mismas medidas empleadas para sintonizar sus respectivos valores de β , el número de bits empleados por el lector. Esto permite conocer si un descenso en el número de bits enviados por los tags perjudica al lector aumentando su carga en la transmisión. Así pues, en las Figuras 4.29.a y 4.29.b se muestran el número de go-On slots y bits medios transmitidos por cada tag respectivamente. Se observa que la ventana exponencial proporciona el menor número de go-On slots, sin embargo, sus tags envían una mayor cantidad de bits, siendo la ventana lineal y la cuadrática muy parejas en ambas magnitudes.

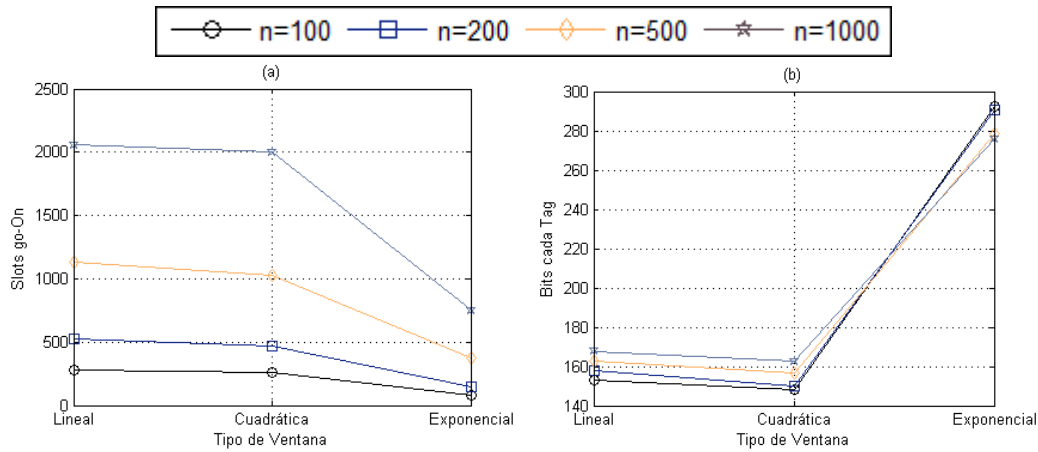


Figura 4.29: Consumos del QwT con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de go-On slots, (b) número medio de bits transmitidos por cada tag.

Pero, ¿cómo tomar una decisión sobre qué ventana es la más adecuada? En las Figuras 4.30.a y 4.30.b se muestran los bits totales transmitidos en la comunicación y los bits transmitidos por el lector respectivamente. Son magnitudes más globales que representan la carga total de comunicación de forma más realista. De la misma manera, mirando directamente a los bits totales, se puede concluir que tanto la ventana cuadrática como la exponencial transmiten una cantidad de bits similares. En cambio, como se ha visto que la ventana exponencial provoca una mayor transmisión de bits de tag, el número de bits transmitidos por el lector es menor en este caso. Por tanto, la ventana cuadrática obtiene los mejores resultados desde el punto de vista de la disminución de bits transmitidos por los tags, que contribuye

4.4 Query window Tree (QwT)

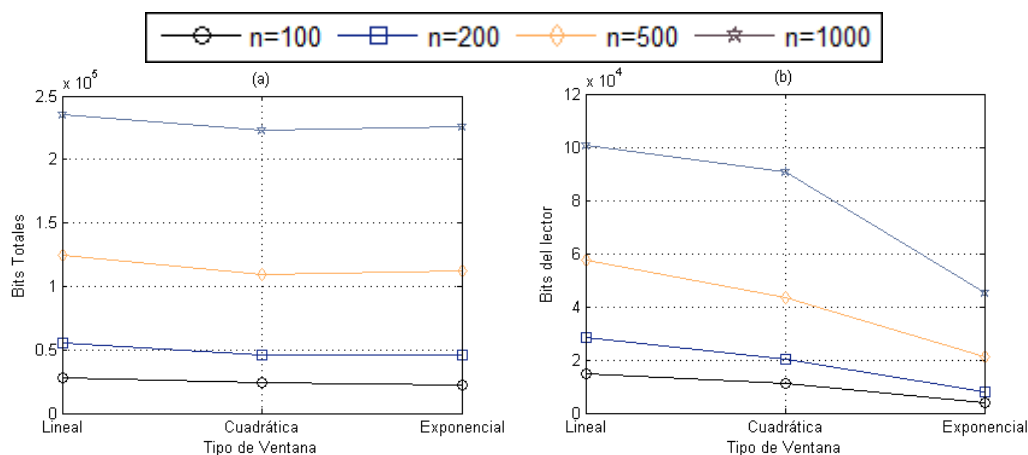


Figura 4.30: Consumos del QwT con diferentes tipos de ventana identificando diferentes grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.

a la disminución global de bits. Por contra, la ventana exponencial consigue unos resultados similares en la transmisión total de bits, ya que el lector implementado consigue la misma finalidad transmitiendo una menor cantidad de bits.

Resumiendo, se han obtenido dos ventanas que ofrecen resultados globales similares. Una de ellas (cuadrática) proporciona unos tags más libres de carga de transmisión, mientras que la otra (exponencial) es al lector al que libera de la carga. Dependiendo del hardware disponible en cada momento, cada una de las ventanas puede resultar más beneficiosa que la otra. Si los tags funcionan empleando baterías o su velocidad de transmisión es muy baja, es interesante que éstos transmitan la menor cantidad posible de bits, acudiendo a una ventana de tipo cuadrática. Por otro lado, en el caso de que los tags puedan conseguir altas velocidades de transmisión, puede resultar más interesante el empleo de una ventana exponencial que consiga menores tiempos de identificación.

4.4.3 Conclusiones

Tras haber realizado un estudio del protocolo QT aplicándole la ventana constante, se han podido observar cuáles son los beneficios y las inconvenientes de ésta. Por ello, en esta sección se ha presentado el protocolo Query window Tree (QwT) en el

4. Metodología de la ventana

que se le dota a la ventana de la posibilidad de variar su tamaño a lo largo del ciclo de interrogación. El protocolo trata de sacar partido de cada situación evitando la transmisión innecesaria de bits por parte de los tags. Además, se consigue mantener la característica *memoryless* de éstos, conservando en la medida de lo posible la sencillez del hardware que los caracteriza.

Para conseguir una modificación correcta de la ventana se ha planteado que ésta tome valores pequeños al comienzo de las ramas del árbol binario, y aumente conforme va subiendo de nivel, L . Así, se han presentado tres funciones heurísticas que proponen comportamientos ligeramente diferentes, pero siguiendo las pautas marcadas por las conclusiones obtenidas del protocolo QT con ventana constante.

De las funciones presentadas, la cuadrática presenta los mejores resultados en bits totales transmitidos y en los bits que transmite cada tag. La función exponencial también ofrece grandes resultados, pero los tags no consiguen reducir de la misma forma los bits transmitidos.

Como ya se ha comentado, el protocolo asume la perfecta sincronización entre las respuestas de los tags, sin imponer restricciones en la codificación empleada. Aún así, conseguir dicha sincronización no es una tarea sencilla en su implementación real. Además, dado que se trata de un diseño teórico no se ha tenido en cuenta el *capture effect*, y las transmisiones de los tags se realizan todas de forma ideal. En la siguiente sección se plantea la utilización de la tecnología bit-tracking para conseguir la sincronización entre los tags, acudiendo a una codificación concreta, la codificación Manchester.

4.5 CT con ventana Constante

En este apartado se aplica la ventana propuesta a otro protocolo basado en *queries* diferente al QT, el Collision Tree (CT), del que se ha hablado en la Sección 2.3.3. Este protocolo es una versión del protocolo QT al que se le ha aplicado la técnica de bit-tracking obteniendo excelentes resultados. Dado que dicha técnica permite rastrear las colisiones hasta el nivel de bit individual, se consigue reducir el número de colisiones y extinguir los ciclos libres, contribuyendo a que el protocolo identifique los tags a mayor velocidad que el QT [Landaluce 14].

4.5.1 Bit-tracking y ventana

Hasta ahora el principal requisito de la ventana era que las transmisiones se produjeran de forma sincronizada, independientemente de la codificación que se usara para transmitir la información. Además, para identificar el tipo de slot, se dotaba a las transmisiones de los tags (ventanas) de un CRC. Se ha comprobado que esta estrategia proporciona una serie de ventajas. A pesar de esto, se han visto, también, una serie de inconvenientes. La ventana reduce el número de bits que transmiten los tags y saca partido de sus transmisiones idénticas, pero, ¿qué ocurre cuando se produce una colisión? Independientemente de la longitud de la ventana enviada por los tags, si el lector interpreta la respuesta como una colisión añade únicamente 0 y 1 al último *query* enviado. Cuando este *query* es de pequeña longitud, el gasto es reducido, pero cuando los *queries* son de mayor tamaño, se desaprovecha un gran número de bits. Además, los tags del protocolo QwT requieren CRC para que el lector sea capaz de identificar si se ha producido una colisión.

Se propone por tanto, el empleo de la técnica bit-tracking al protocolo QwT, o dicho de otro modo adaptar la ventana para trabajar con el protocolo CT. El protocolo resultante seguirá manteniendo la característica principal de los protocolos basados en *queries* y la capacidad de los tags de trabajar sin memoria de almacenamiento o *memoryless*. La ventana propuesta y la técnica bit-tracking son perfectamente complementarias, la primera reduciendo el número de bits transmitidos por los tags y la segunda reduciendo las colisiones y los slots libres. En la Figura 4.31 se representa parte del ciclo de interrogación de tres tags empleando el protocolo CT con un tamaño de ventana de cinco bits. También se realiza la comparación de las respuestas con el protocolo CT sin ventana, para observar la reducción de bits innecesarios transmitidos.

Además la combinación de bit-tracking y ventana contribuye a la transformación de colisiones en identificaciones parciales de los ID de los tags. Las modificaciones que sufre la ventana y el protocolo al aplicar la técnica bit-tracking son:

- ◇ Empleo obligatorio de la codificación Manchester para todas las transmisiones entre lector y tag. La técnica bit-tracking necesita dicha codificación para poder trazar la colisión a nivel de bit. Por otro lado, esta codificación facilita

4. Metodología de la ventana

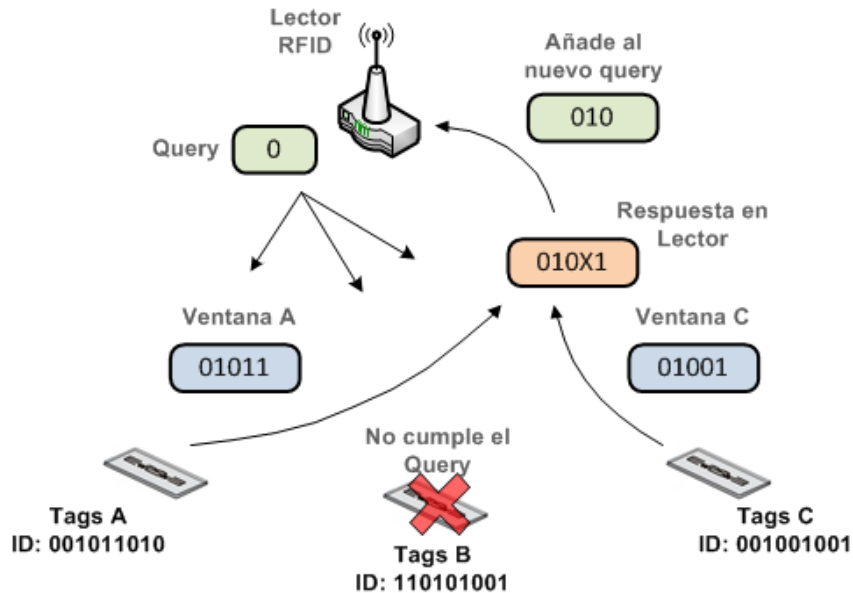


Figura 4.31: Interrogación de 3 tags empleando el protocolo CT con ventana, $W = 5$

la sincronización en las respuestas de los tags, ya que la señal lleva el reloj intrínseco.

- ◇ El CRC ya no es estrictamente necesario para identificar la colisión. Gracias a la codificación Manchester, nuevamente, el lector no necesita valerse del CRC para saber si los datos recibidos han colisionado. Es más, puede saber la posición exacta del bit donde se ha producido la colisión.
- ◇ Se producen tres tipos de slots a diferencia del QwT: colisión, go-On y satisfactorio. Con la diferencia que la actuación en el caso de colisión varía, para asemejarse a la del caso de un go-On slot. Es decir, que el lector puede emplear en el siguiente *query* los bits recibidos correctamente anteriores al bit colisionado. Esto le permite crecer de forma más agresiva y rápida. Además, la aplicación de la técnica bit-tracking consigue extinguir los slots libres.

Al igual que en la aplicación de la ventana al protocolo QT, en este caso, la ventana también hace que el protocolo se pueda comportar como uno de tipo BTA para pequeños valores de W , o de forma similar al CT para valores elevados cercanos a k .

4.5.2 Funcionamiento del protocolo

En la Figura 4.32 se muestra un diagrama de flujo con el funcionamiento del protocolo CT con la ventana propuesta. El flujo presentado es similar al del protocolo QT con la ventana constante, Figura 4.3, exceptuando la diferencia añadida en el caso de colisión. En tal caso, se emplea la estrategia del protocolo CT de trazar en qué bit se ha producido la colisión, v_c y añadir al *query* los bits anteriores, v_1, v_2, \dots, v_{c-1} siendo $v_i \in \{0, 1\}$.

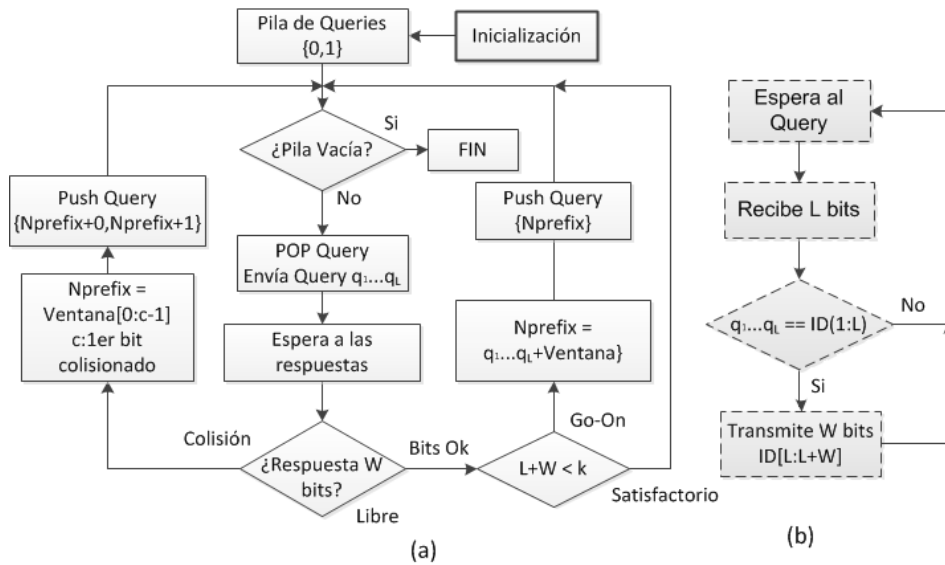


Figura 4.32: Esquema de funcionamiento del protocolo CT con ventana constante: (a) procedimiento del lector y (b) de cada tag

También se presenta un ejemplo de identificación de cuatro tags en la Figura 4.33, acompañada por la Tabla 4.4, donde se muestran, paso por paso, las distintas acciones del protocolo CT con un tamaño de ventana, W , de dos bits y una longitud de ID, k , de siete bits.

Tras inicializar el lector, se introducen dos *queries* iniciales (0, 1) en una pila LIFO y comienza la identificación sacando el primero (POP) y transmitiéndolo a los tags. Mientras el lector permanece a la espera, los tags reciben el *query* y lo comparan con su ID. En caso de coincidir, envía los bits indicados por W alojados a continuación, dos en el caso del ejemplo de la Figura 4.33.

4. Metodología de la ventana

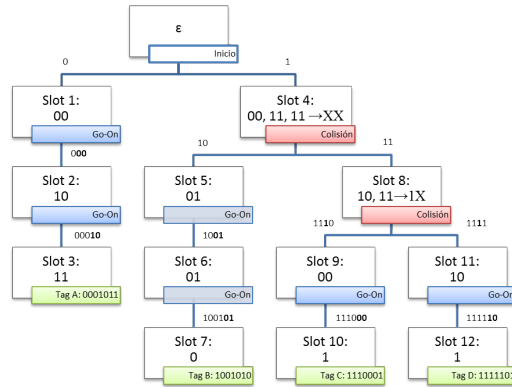


Figura 4.33: Ejemplo de ejecución del protocolo CT con ventana, $W = 2$

Slot	Lector Query	Tag A 0001011	Tag B 1001010	Tag C 1001010	Tag D 1111101	Lector recibe	Tipo de slot
1	0	00	-	-	-	00	Go-On
2	000	10	-	-	-	10	Go-On
3	00010	11	-	-	-	11	Tag A
4	1	-	00	11	11	XX	Colisión
5	10	-	01	-	-	01	Go-On
6	1001	-	01	-	-	01	Go-On
7	100101	-	0	-	-	0	Tag B
8	11	-	-	10	11	1X	Colisión
9	1110	-	-	00	-	00	Go-On
10	111000	-	-	1	-	1	Tag C
11	1111	-	-	-	10	10	Go-On
12	111110	-	-	-	1	1	Tag D

Tabla 4.4: Ejecución detallada del ejemplo mostrado en la Figura 4.33 del CT con $W = 2$

En el primer slot del ejemplo, el lector envía el *query* 0 y, dado que el ID del Tag A coincide, contesta los dos bits adjuntos, 00. El lector recibe correctamente la cadena de bits ya que ningún otro tag contesta y se considera un slot de tipo go-On, pues se cumple que $L + W < k$, $0 + 00$ en el ejemplo. Dado que se trata de un go-On slot, la ventana recibida, 00, se añade al *query* y se reenvía de nuevo a los tags.

Este ciclo se repite hasta que se identifica el Tag A. Cabe destacar, en el slot 4, la situación de colisión que se produce. Tres tags responden al siguiente *query* enviado, 1. Los datos obtenidos por el lector no pueden ser interpretados y, por tanto, el tratamiento de la colisión es igual que en el caso del protocolo QT, generando dos nuevos *queries* añadiendo un 0 y un 1 al *query* enviado. Sin embargo, en la colisión que se produce en el slot 8, dado que el primer bit que envía cada tag coincide, la codificación Manchester permite identificar que el segundo bit enviado es donde se ha producido la colisión. Por ello, el lector genera dos nuevos *queries*, pero esta vez añade la parte interpretada de la respuesta, 1, en ambos *queries* y posteriormente los bits 0 y 1 respectivamente. El proceso de identificación continúa hasta que no queda ningún *query* en la pila.

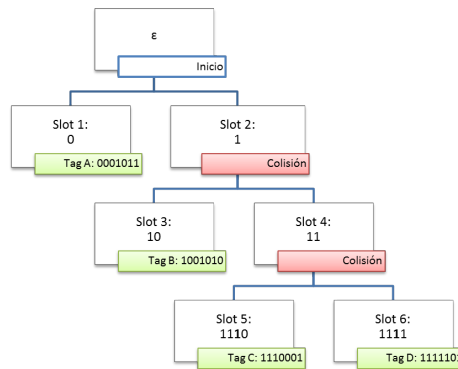


Figura 4.34: Resolución del ejemplo mostrado en la figura 4.33 mediante el protocolo CT.

Al aplicar la ventana al protocolo CT, se produce un incremento de slots similar al producido en el caso del protocolo QT. En la Figura 4.34 se muestra la resolución del ejemplo de la Figura 4.33 mediante el protocolo CT sin emplear la ventana. Se observa cómo el protocolo CT emplea menor número de slots para los mismos tags. En cambio, observando la Tabla 4.5, donde se muestran las transmisiones de bits en ambos protocolos, el número medio de bits transmitidos por cada tag es de 6,75 bits, mientras que si se hubiese empleado el protocolo CT, los tags habrían necesitado una media de 11,25 bits.

A continuación se va a realizar un análisis de la aplicación de la ventana constante al protocolo CT para observar las ventajas y desventajas que ambos se aportan.

4. Metodología de la ventana

Slot	Bits transmitidos							
	Tag A		Tag B		Tag C		Tag D	
	CT	W	CT	W	CT	W	CT	W
1	6	2	-	-	-	-	-	-
2	-	2	6	-	6	-	6	-
3	-	2	5	-	-	-	-	-
4	-	-	-	2	5	2	5	2
5	-	-	-	2	3	-	-	-
6	-	-	-	2	-	-	3	-
7	-	-	-	1	-	-	-	-
8	-	-	-	-	-	2	-	2
9	-	-	-	-	-	2	-	-
10	-	-	-	-	-	1	-	-
11	-	-	-	-	-	-	-	2
12	-	-	-	-	-	-	-	1
Total	6	6	11	7	14	7	14	7
Media	$CT = \frac{6+11+14+14}{4} = 11,25$				$W = \frac{6+7+7+7}{4} = 6,75$			

Tabla 4.5: Bits transmitidos por los tags en los ejemplos de las Figuras 4.33 y 4.34

4.5.3 Análisis de la ventana aplicada al protocolo CT

En esta sección se realiza un análisis paralelo al que se ha visto previamente en la Sección 4.3.1. En cambio, dado que no se ha realizado un análisis del protocolo CT en secciones anteriores, se va a comenzar ofreciendo un breve análisis de éste, para pasar, a continuación, a la aplicación de la ventana.

4.5.3.1 Breve análisis del protocolo CT

Las probabilidades del protocolo CT, se calculan de la misma forma que las del protocolo QT (ver Sección 3.3.1). La principal diferencia entre ambos se produce a la hora de calcular cuándo se va a visitar un nodo. En el caso del protocolo QT, un nodo se visita si se produce una colisión en el nodo anterior. En el protocolo CT, sin embargo, se considera el concepto de bifurcación, ya que en caso de colisión

el lector puede interpretar parte del mensaje recibido de los tags y saltar varios niveles produciéndose la bifurcación en niveles superiores. Por ello, la probabilidad de visitar un nodo v en el protocolo CT, se traduce en la probabilidad de que en v se produzca una bifurcación, es decir, que exista un nodo $\{v, 0\}$ y un nodo $\{v, 1\}$. De esta forma, la probabilidad de que el nodo v sea una bifurcación es:

$$\mathbb{P}(v \text{ bifurcacion}) = \mathbb{P}(\{v, 1\} \text{ no libre} \cap \{v, 0\} \text{ no libre}) \quad (4.30)$$

La expresión se puede reescribir como¹:

$$\begin{aligned} \mathbb{P}(v \text{ bifurcacion}) &= 1 - \mathbb{P}(\{v, 1\} \text{ libre}) - \mathbb{P}(\{v, 0\} \text{ libre}) + \\ &+ \mathbb{P}(\{v, 0\}, \{v, 1\} \text{ libres}) \end{aligned} \quad (4.31)$$

Cabe mencionar que como el nodo v se encuentra en el nivel L y los nodos $\{v, 0\}$ y $\{v, 1\}$ están en el mismo nivel $L + 1$, las probabilidades de que se de un slot libre son similares y dependientes de L . Por tanto la expresión se puede reescribir de la siguiente manera:

$$\mathbb{P}(v \text{ bifurcacion}) = 1 - 2P_I(L + 1) + P_I(L) \quad (4.32)$$

4.5.3.2 Análisis del protocolo CT con la ventana de bits

Conocida la expresión de la probabilidad de visitar un nodo en el protocolo CT, se traslada al problema actual aplicándole la ventana. Al igual que entre los protocolos QT y CT, las probabilidades obtenidas de slot libre, go-On o colisión coinciden en el caso del protocolo QT y CT con ventana, pudiendo emplearse las expresiones obtenidas en la Sección 4.3.1.

Con el objetivo de obtener una expresión que nos permita calcular el número de slots que utiliza el protocolo, τ , se acude de nuevo a la eq. 4.19. Para ello es necesario el cálculo de la probabilidad de visitar un nodo, P_V .

El nodo v únicamente se visitará si su nodo padre es una bifurcación o si existe un nodo de tipo go-On hacia v , W niveles anteriores. Como sucede en el caso

¹Se ha utilizado la propiedad $P(A) = P(\bar{B} \cap \bar{C}) = 1 - P(B \cup C) = 1 - [P(B) + P(C) - P(B \cap C)] = 1 - P(B) - P(C) + P(B \cap C)$

4. Metodología de la ventana

estudiado en la Sección 4.3.1.1, la probabilidad de visitar un nodo está igualmente influenciada por el tipo de distribución de árbol disponible y la distribución de los IDs de los tags. Por ello, se plantea la acotación superior de la expresión adaptando la presentada en la eq. 4.32:

$$P_V = \mathbb{P}(v \text{ visitado}) \leq \mathbb{P}(v - 1 \text{ bifurcacion}) + \frac{1}{2^W} P_G^W(L_{v-W}) \quad (4.33)$$

A continuación se muestra en la Figura 4.35 la representación gráfica de la eq. 4.33.

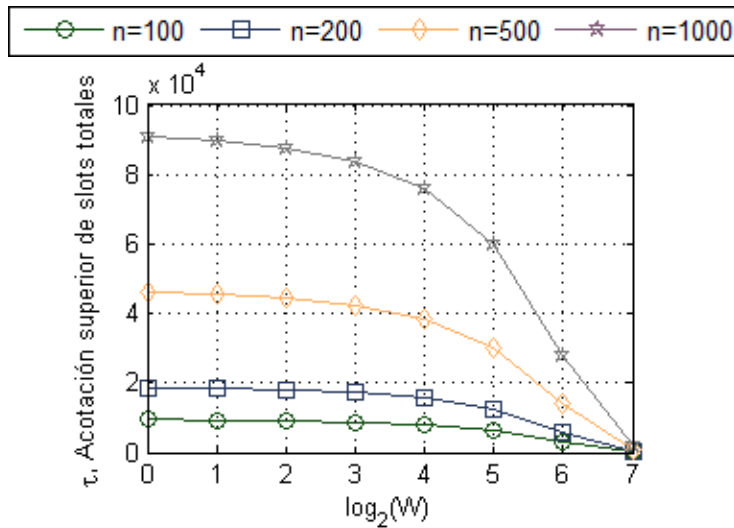


Figura 4.35: Resultados ante diferentes valores de W aplicados sobre el protocolo CT, empleando la eq. 4.19

La influencia que el tamaño de la ventana W ejerce sobre el protocolo CT es similar a la que se produce en el protocolo QT con ventana. Ésta provoca un aumento del número de slots para valores bajos de W , similar al producido en el protocolo QT. Mientras que los valores cercanos al tamaño del ID de los tag, k , produce un comportamiento similar al del protocolo CT sin la ventana.

Para realizar un análisis más profundo de la influencia de la ventana sobre el protocolo CT, se acude a las simulaciones.

4.5.4 Simulaciones para diferentes valores de W

Una vez analizado teóricamente el protocolo, se realiza un proceso paralelo al de la Sección 4.3.2, aplicando diferentes tamaños de ventanas al protocolo CT. De esta manera se puede conocer cómo responde el protocolo a los valores de W y así poder diseñar unas pautas de variación para su comportamiento dinámico en el siguiente apartado. Para la realización de las simulaciones se ha empleado Matlab R2012b, y se ha definido un escenario de simulación formado por un lector y diferentes conjuntos de tags, n . Los IDs de los tags forman una distribución homogénea y su longitud, k , es de 96 bits. Los valores mostrados en cada una de las gráficas son los resultados medios de 100 iteraciones que realizadas en cada experimento. En esta sección los experimentos consisten en variar el tamaño de la ventana ante diferentes conjuntos de tags y ver cómo influye en las características del protocolo.

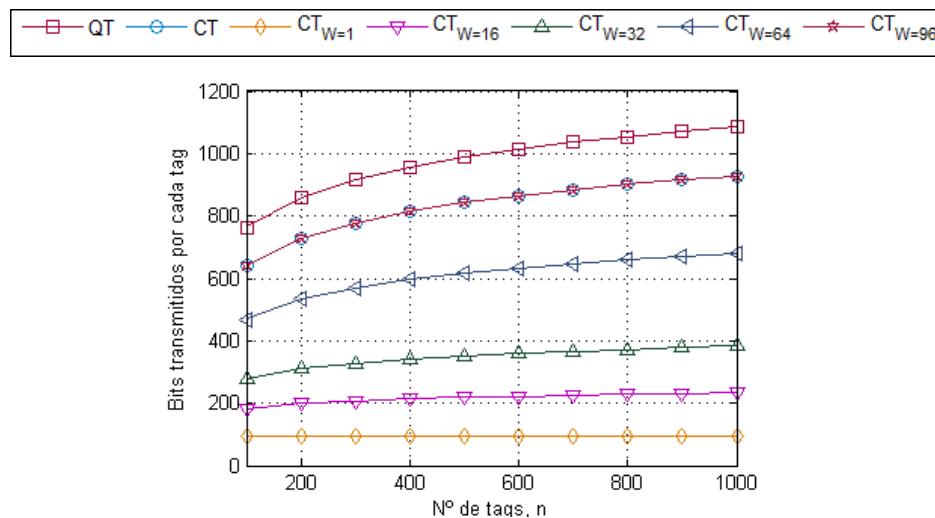


Figura 4.36: Influencia del tamaño de la ventana en los bits transmitidos por cada tag del protocolo CT, con $k = 96$ bits

En la Figura 4.36 se muestran los resultados de la simulación del número medio de bits que transmite cada tag ante diferentes tamaños de ventana. Se realiza una comparación, además, con los resultados del protocolo QT y CT en este aspecto. Al igual que ocurriría con la aplicación de la ventana constante al protocolo QT (ver Figura 4.13), las ventanas de menor tamaño son las que producen los mejores resultados con transmisiones bajas. La ventana de 1 bit hace que el protocolo trabaje

4. Metodología de la ventana

como uno de tipo BTA, y la de 96 bits, k , ofrece los mismos resultados que su protocolo origen, el CT. Se observa también, la reducción de bits entre el protocolo QT y el CT, gracias a la incorporación de la técnica de bit-tracking.

Tras mostrar la simulación variando n , se presentan los resultados obtenidos para una serie de simulaciones realizadas variando el tamaño de la ventana, W , ante diferentes conjuntos de tags. En la Figura 4.37.a y b se muestran el número de slots empleados (τ) en identificar diferentes conjuntos de tags, n , y sus respectivas eficiencias (η) calculadas empleando la eq. 4.1.

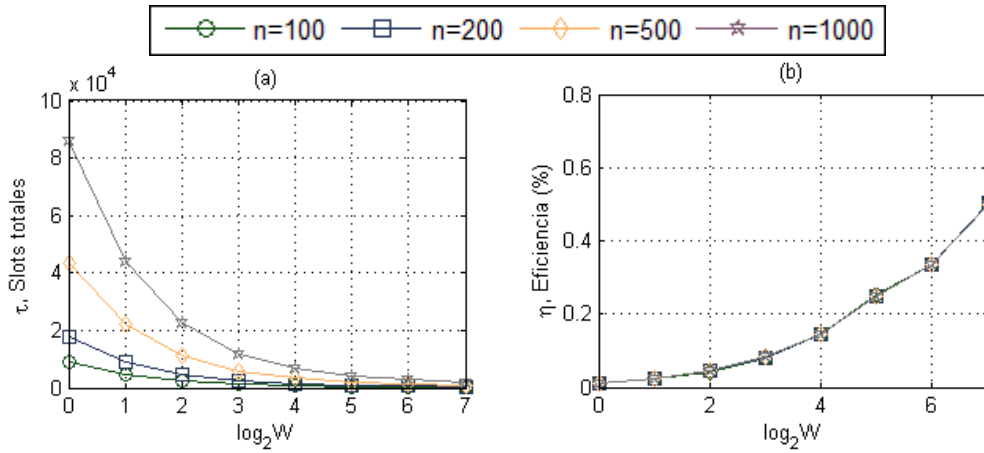


Figura 4.37: Influencia del tamaño de la ventana sobre el protocolo CT: (a) en el número de slots τ , (b) en la eficiencia de slots η

Se observa como las tendencias de las gráficas de la aplicación de la ventana al protocolo CT son similares a las presentadas en la Figura 4.14. Es decir, que la ventana actúa de forma similar tanto en el protocolo QT como en el CT en cuanto a slots empleados. También cabe destacar la tendencia similar a la obtenida según las expresiones teóricas de la Sección 4.5.3.2. Los valores mostrados en la Figura 4.37.a son inferiores a los teóricos de la Figura 4.35, que conformaban una acotación superior pesimista. Por otro lado, se puede destacar que la máxima eficiencia conseguida con el tamaño de ventana máximo ($W = k$) es del 50%. Es decir, que el protocolo CT emplea un menor número de slots que el QT y por ello su eficiencia es mayor.

De nuevo, este incremento en el número de slots viene provocado por la aparición de los slots de tipo go-On demandados por la ventana. Estos slots son nece-

sarios para completar el ID del tag que está siendo interrogado. En la Figura 4.38 se observa la tendencia de este tipo de slots que, evidentemente, influyen directamente sobre el número total de slots. Por tanto, al igual que en el protocolo QwT, la ventana aplicada al CT deberá reducir lo máximo posible este tipo de slots, ya que perjudicarán la puesta en escena del protocolo.

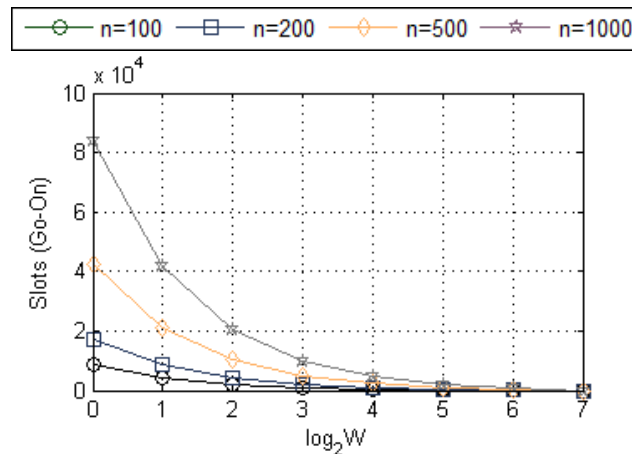


Figura 4.38: Slots go-On generados por la utilización de la ventana

La principal característica del protocolo CT es que siempre se producen el mismo número de slots y colisiones para un determinado número de tags. Aplicando la ventana al protocolo, el número de colisiones permanece constante, no así el número de slots, debido a la aparición de los go-On slots. Del mismo modo, la extinción de los slots libres sigue permaneciendo. En la Figura 4.39 se muestran los resultados de las colisiones y slots libres que se producen en el protocolo ante diferentes ventanas.

De la aparición de los slots de tipo go-On se deducen dos problemas:

- ◇ Por un lado, los tags responden un menor número de bits en cada slot, por las exigencias de la ventana. Esto puede provocar que las respuestas comunes de los tags no abarquen toda su zona común y el lector necesite enviar varios *queries* para obtenerla, mientras que el protocolo CT la obtendría con un solo *query*. Esto causa que los *queries* no se generen tan agresivamente como en el protocolo CT. Se plantea un ejemplo que refleja esta situación: suponiendo que existen dos tags, cuyos IDs son 001101 y 001111 y el tamaño de ventana

4. Metodología de la ventana

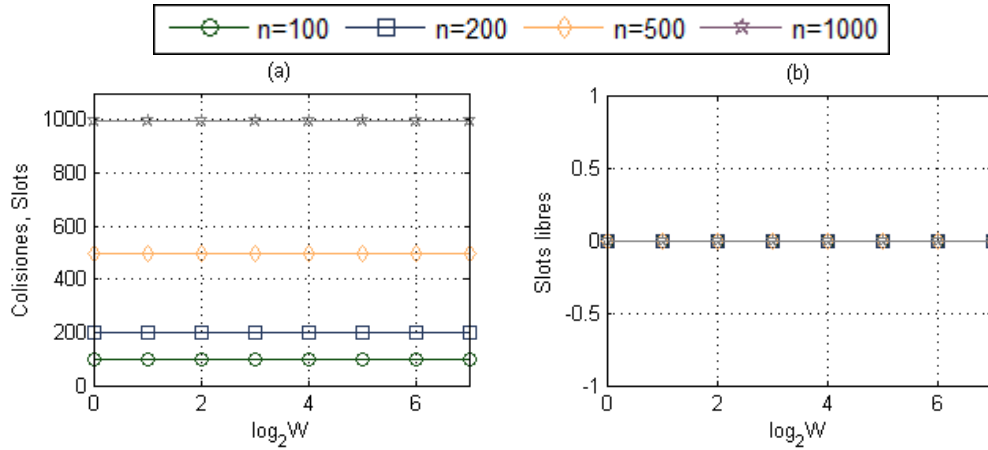


Figura 4.39: Influencia del tamaño de la ventana sobre el protocolo CT: (a) en el número de colisiones, (b) en el de slots libres

es $W = 2$, el lector transmite un *query* 0, y ambos tags contestan 01. El lector añadiría esta respuesta al *query* y continuaría el mismo proceso. Se observa cómo la parte común de sus IDs es mayor que la ventana recibida. Si hubieran transmitido su ID completo, el lector habría ampliado su *query* con 0111, es decir con cuatro bits en lugar de dos.

- ◇ Por otro lado, el lector necesita más *queries* compuestos por una gran cantidad de bits que el protocolo CT origen. Dado que el lector debe interrogar a los tags hasta que éstos envíen sus últimos bits de ID, cuando W es pequeño los *queries* necesarios para obtener los últimos bits deben ser mucho mayores. Dado un tag cuyo ID es 00110011, y $W = 1$, el lector necesita transmitir 00110, 001100 y 0011001 para obtener el ID completo, ya que el tag solo contesta un bit en cada interrogación. Esto aumenta en gran medida los bits transmitidos por el lector.

Como se muestra en las Figuras 4.40.a y b, se produce un aumento considerable de bits al emplear ventanas formadas por pocos bits. Las razones son las explicadas en el apartado anterior. A pesar de que los valores de W bajos reducen el número de bits transmitidos por los tags, los necesitados por el lector superan en cantidad al de los tags. Por eso, el total de bits transmitidos tiene una tendencia ascendente

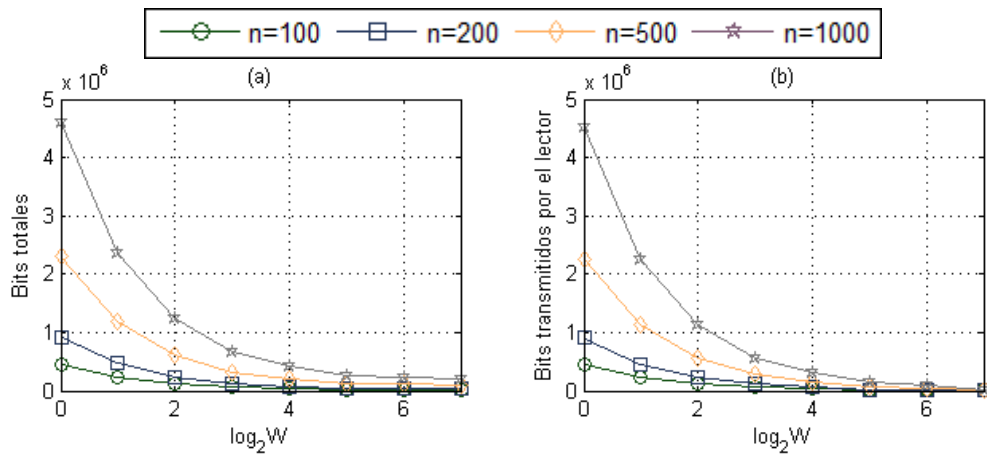


Figura 4.40: Influencia del tamaño de la ventana sobre el protocolo CT: (a) en el número total de bits transmitidos por el lector y los tags, (b) en el número de bits transmitidos por el lector

al descender el valor de W , que es similar a la tendencia de los bits transmitidos por el lector.

4.5.5 Conclusiones

En esta sección se ha presentado un protocolo en el que se aúnan dos técnicas existentes: bit-tracking y la ventana de bits. Juntando estas dos técnicas complementarias se consigue, por un lado disminuir el número de slots gracias a la capacidad del bit-tracking de trazar la colisión a nivel de bit, y por otro lado, reducir el número de bits transmitidos por los tags, gracias a la ventana. Ambas técnicas se complementan ya que una se focaliza en la disminución de los slots, mientras que la otra en la de los bits de los tags.

Además, se ha observado cómo la influencia que provoca la ventana en el protocolo CT (que emplea bit-tracking) es muy similar a la que ejerce sobre el protocolo QT, como se ha presentado en la Sección 4.3. Cuando los valores de W son cercanos a k , el protocolo propuesto se comporta de forma similar al protocolo CT, mientras que en el caso contrario con W cercano a 1, se asemeja al de un protocolo BTA. Cabe destacar, que la ventana no ejerce ninguna influencia en el número de colisiones que se producen, ni tampoco provoca la reaparición de los slots libres.

4. Metodología de la ventana

Sin embargo, al igual que en el caso de la aplicación de la ventana al protocolo QT, aparecen los slots de tipo go-On que pueden ser muy perjudiciales para la latencia y el consumo del protocolo, ya que pueden aumentar el número de bits transmitidos en grandes cantidades. Para evitarlo, se propone dinamizar los valores de la ventana, tratando de aprovechar los beneficios y de evitar los problemas encontrados.

4.6 Collision window Tree (CwT)

En esta sección se presenta un nuevo protocolo, el Collision window Tree (CwT), en el que se aplica la ventana al protocolo CT [Jia 12], dotándola de capacidad dinámica para adoptar diferentes tamaños en función de las necesidades. Gracias a las conclusiones obtenidas en la sección anterior, se puede intuir el comportamiento que debe seguir la ventana dinámica, con el fin de disminuir el número de bits transmitidos por cada tag en distribuciones homogéneas de sus IDs. La ventana dinámica aplicada al protocolo CT produce una sustancial mejora en el número de bits totales transmitidos, gracias al ahorro que se produce en los bits enviados por los tags, causando una disminución de la latencia y del consumo energético del protocolo.

Como ya se ha mencionado, la unión entre la técnica de bit-tracking y la ventana produce beneficios complementarios, lo que se ve reflejado en el resultado final. El protocolo CwT mantiene la característica *memoryless* del protocolo QT y CT, y no necesita emplear el CRC para la detección de las colisiones, ya que la codificación Manchester le proporciona la capacidad de trazar la colisión a nivel de bit. Como las conclusiones obtenidas del estudio de ventana constante sobre el protocolo CT son similares a las obtenidas en el estudio sobre el protocolo QT, se van a emplear las mismas funciones heurísticas que en el caso del QwT: lineal, cuadrática y exponencial.

En la Figura 4.41 se muestra un diagrama de flujo con el funcionamiento del protocolo CwT. Las principales diferencias con respecto al presentado para el QwT son que no se contempla el caso de slots libres ya que no se producen, y la aplicación de la técnica de bit tracking en el caso de una colisión. Además, se observa cómo se realiza el cálculo del tamaño de la ventana previo a la transmisión del *query*, que es transmitido junto a él hacia los tags. El lector almacena los *queries*

4.6 Collision window Tree (CwT)

generados en cada colisión en una pila LIFO y acude a ella cada vez que se identifica un tag.

El tag, además del comparador, incluye un puntero que se actualiza sumándole el parámetro W recibido junto al *query* para saber hasta qué parte de su ID debe transmitir, únicamente si su ID coincide con los L bits del *query* recibido. En este caso, la generación del CRC no es necesaria, ya que la codificación Manchester permitirá conocer al lector la existencia de colisiones.

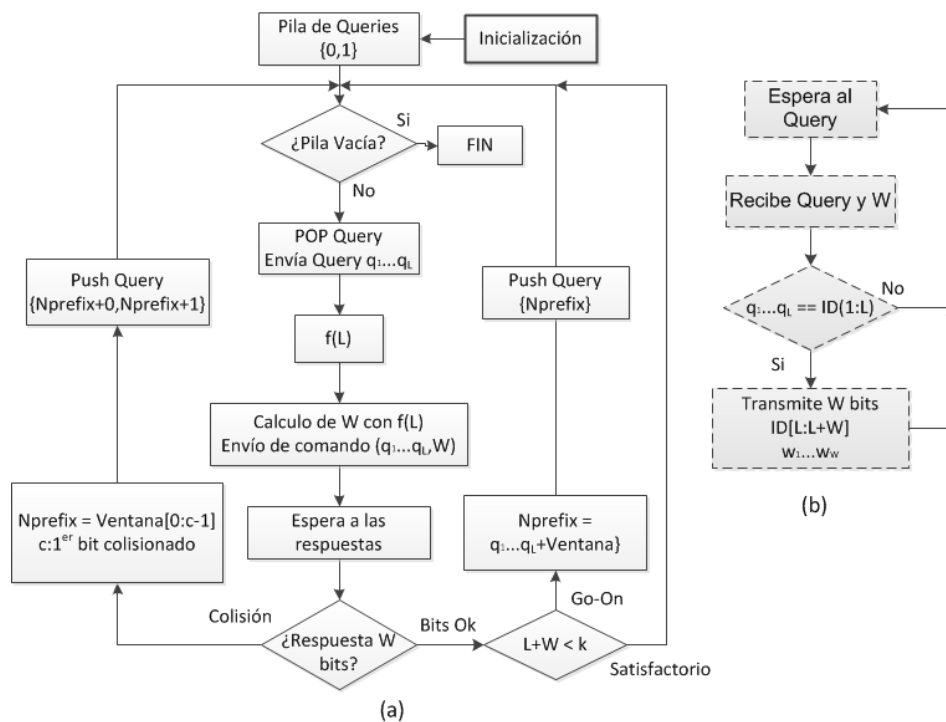


Figura 4.41: Diagrama de flujo del protocolo CwT: (a) lector y (b) tag

La determinación del tamaño de ventana más adecuado en cada momento se realiza de la misma forma que en el caso del protocolo QwT, siguiendo la eq. 4.25. Esta ecuación indica el procedimiento de actuación en cada tipo de slot:

- ◇ Slot go-On ($t = 1$): el protocolo adjunta los bits recibidos al último *query* transmitido, formando un nuevo *query* de $L + W$ bits. El nuevo tamaño de la ventana (W_i) se calcula empleando la función heurística ($f(L_i)$).
- ◇ Slot en colisión ($t > 1$): el protocolo se vale de la técnica de bit-tracking para encontrar el primer bit donde se ha producido la colisión w_c . De esta forma

4. Metodología de la ventana

puede aprovechar la parte anterior a w_c de la ventana recibida en el siguiente *query*.

- ◇ Slot satisfactorio ($t = 1$): el protocolo identifica el tag y obtiene un nuevo *query* de la pila LIFO. Es decir, que cambia la rama del árbol o el tamaño del *query* L . La ventana adquiere el valor 1 nuevamente, para comenzar la identificación de la nueva rama.

Las funciones heurísticas a emplear deben ser ajustadas para proporcionar su mejor rendimiento. Para ello se introduce de nuevo un parámetro de ajuste β .

4.6.1 Funciones de ventana

Las funciones de ventana, $f(L)$, se emplean de la misma forma que en el protocolo QwT. Antes de enviar el *query* correspondiente, el lector empleará la longitud de éste para calcular el tamaño de ventana adecuado, W . Teniendo como objetivo que los tags transmitan pocos bits cuando se produzca una colisión, y un gran número de ellos cuando sea identificable, se proponen tres funciones heurísticas que se adaptan al comportamiento deseado ante distribuciones de tags homogéneas. Éstas son: lineal, cuadrática y exponencial. Además, se emplea un parámetro, β , para realizar el ajuste de las funciones, de forma que se consiga el comportamiento deseado. Es destacable mencionar, que al igual que las funciones aplicadas al protocolo QwT, las aquí presentadas se limitan de igual manera utilizando la expresión 4.26. De forma que la suma entre la longitud del *query* L y el tamaño de la ventana W nunca será mayor que la longitud total del ID k .

- ◇ Función **Lineal**: representada en la Figura 4.20, la función lineal emplea la expresión 4.27 para calcular el valor de W en función de la longitud del *query*, L . Mediante β se establece la región de transición en la que los tags incrementarían el número de bits a transmitir. Esta región es más abrupta para valores mayores o más progresiva empleando valores menores.

Para elegir el valor más adecuado de β se presentan una serie de simulaciones del protocolo CwT empleando la función lineal, y variando β ante diferentes conjuntos de tags (n), cuyos ID son de $k = 128$ bits. Los conjuntos escogidos

4.6 Collision window Tree (CwT)

(100, 200, 500 y 1000) ofrecen una perspectiva suficiente para tomar una decisión sobre cuál es el valor idóneo de β , ya que se observa claramente la tendencia del protocolo.

En la Figura 4.42.a se muestran los slots de tipo go-On que emplea el protocolo, cuyo valor desciende conforme aumenta el valor de β . Sin embargo, el número medio de bits que envía cada tag aumenta de forma contraria, Figura 4.42.b.

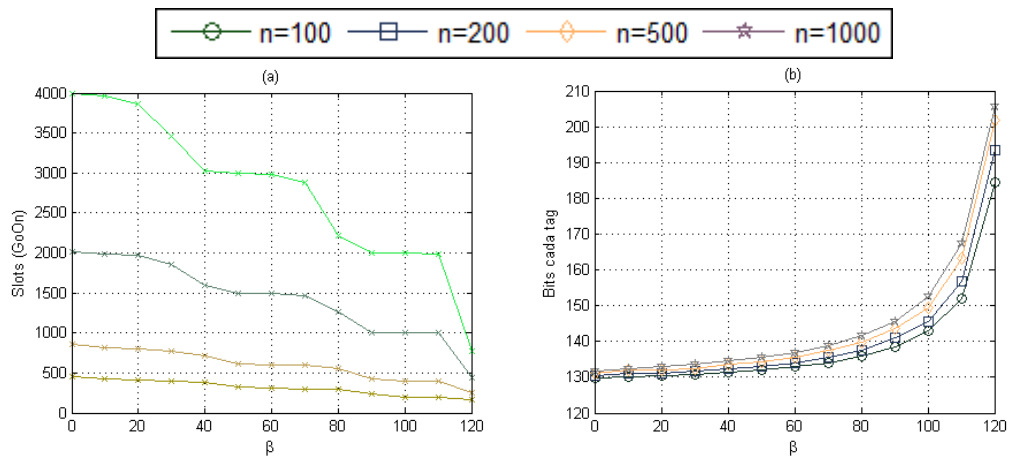


Figura 4.42: Variación del comportamiento del protocolo CwT con función Lineal, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits

Sólo con los datos aportados no se podría tomar una decisión lo suficientemente fiable. Como ya se ha postulado anteriormente, es importante reducir el número de slots de tipo go-On por el incremento en tiempo y energía que producen, aunque también se pretende que los tags envíen la menor cantidad de bits posible. Con esta información resulta complicado encontrar el punto idóneo de trabajo, por lo que se acude al consumo total de bits del protocolo.

La Figura 4.43 muestra los bits totales transmitidos entre lector y tags en la identificación de todo el conjunto. Se puede observar cómo valores de β cercanos a 80 generan un mínimo local de bits transmitidos. Sin embargo, valores de β mayores, cercanos a $\beta = k$, consiguen reducir de forma más clara el número de bits transmitidos, sobre todo para entornos de alta densidad de tags ($n = 1000$). Como se ha mostrado en la Figura 4.42.b, los valores

4. Metodología de la ventana

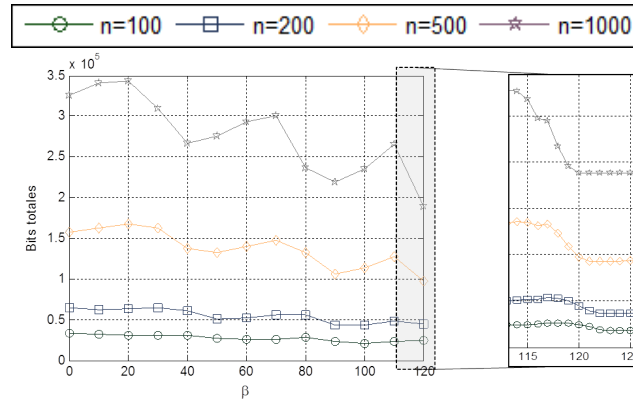


Figura 4.43: Variación del comportamiento del protocolo CwT con ventana lineal para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n

altos de β provocan un mayor número de bits transmitidos por cada tag, pero gracias al descenso que se produce de los slots de tipo go-On (4.42.a), el número total de bits es menor con valores de β mayores que en el mínimo local mencionado. Este fenómeno se produce debido a que en respuestas de tags mayores, el lector puede sacar mayor partido de la técnica bit-tracking, y generar los *queries* adecuados en menor número de slots. Por ello, se opta por el valor $\beta = 123$.

- ◇ **Función Cuadrática:** en esta función, el lector obtiene el valor de W en base al cuadrado del valor de L , según la eq. 4.28 representada en la Figura 4.23. El valor de β sintoniza la función para que trabaje de la forma más adecuada posible para los intereses del protocolo: reducir el número de bits transmitidos por los tags sin provocar un exceso de slots, o un aumento excesivo de los bits transmitidos por el lector. El parámetro β define la pendiente de crecimiento de la función parabólica. Un valor de β alto, provee una ascensión más drástica que un valor bajo.

Seguidamente, se realiza la sintonización del valor de β , como se ha realizado en las anteriores funciones. En la Figura 4.44.a se muestran los slots de tipo go-On, y en la Figura 4.44.b el número de bits medio que envía cada tag.

La primera figura (Figura 4.44.a) muestra cómo el número de slots de tipo

4.6 Collision window Tree (CwT)

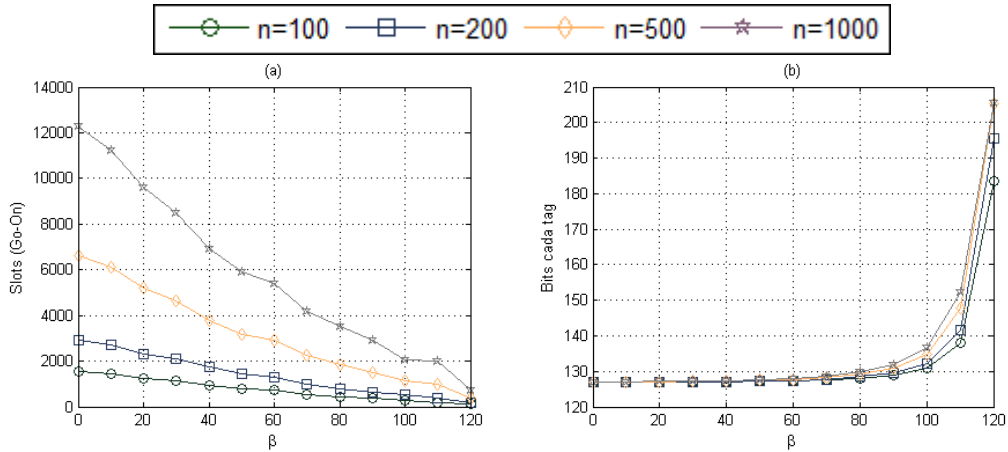


Figura 4.44: Variación del protocolo CwT con función cuadrática, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits

go-On descende conforme aumenta β , y de la misma manera que en el caso de la función lineal, el comportamiento del protocolo en número de bits transmitidos por los tags (Figura 4.44.b) es inverso. Cabe destacar, sin embargo, que el aumento se da de forma mucho más brusca en la ventana cuadrática.

Dado que se busca un punto intermedio en el que el número de slots go-On no aumente demasiado, y el número de bits transmitidos por los tags sea el menor posible, se debe acudir al número total de bits transmitidos. Para esclarecer esta disyuntiva se muestra la Figura 4.45, con los bits totales transmitidos entre lector y tags.

Se observa en este caso que el descenso de bits producido en el rango de valores de $\beta = [90, 110]$ no es tan diferente como el del rango $\beta = [120, 127]$, como sucedía con la función lineal. Ambos intervalos representan buenos comportamientos para el protocolo, pero en entornos de alta densidad de tags, el segundo intervalo ofrece mejores prestaciones. Por otro lado, el primer intervalo resulta más favorable para los tags, ya que transmitirán un menor número de bits, mientras que el número total de bits transmitidos es mejor en el segundo, lo cuál indica que el protocolo ofrecerá un gran comportamiento en latencia y consumo energético. Por tanto, para el primer comportamiento (tag conservador), un valor válido sería $\beta = 104$, mientras que para el

4. Metodología de la ventana

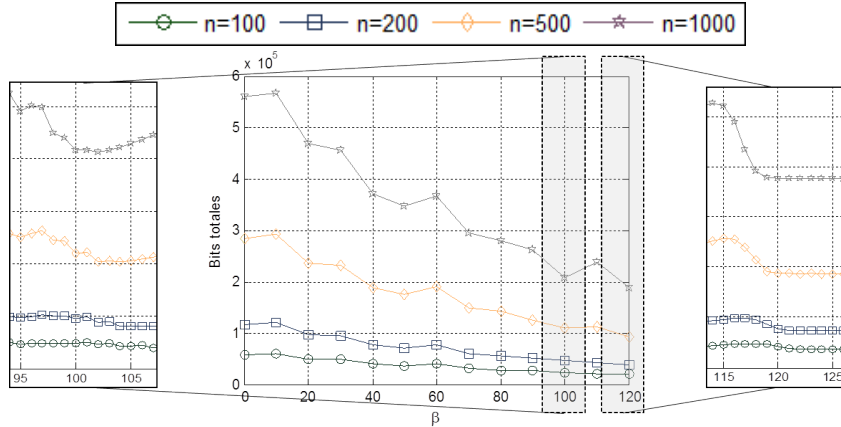


Figura 4.45: Variación del protocolo CwT con ventana cuadrática para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n

segundo (latencia) el valor escogido sería $\beta = 121$.

- ◇ **Función Exponencial:** la última de las funciones es la exponencial. La misma función propuesta en el protocolo QwT, eq. 4.29, de amplitud k representada en la Figura 4.26, se emplea para el protocolo propuesto en esta sección. El valor de $1/\beta$ define la constante de tiempo de crecimiento, por tanto, un valor de β mayor ofrece una tendencia de mayor pendiente que uno menor.

Para ajustar el valor de β se realizan de nuevo las simulaciones variando el valor de β en el rango $[0,1]$, que es el rango que ofrece mayor variabilidad al protocolo. Se presentan los resultados en la Figura 4.46.a, donde se muestran los go-On slots empleados y en la Figura 4.46.b, los bits de promedio transmitidos por cada tag.

En las gráficas mostradas, se produce el mismo efecto al variar β que en el resto de funciones. El número de slots de tipo go-On decrece conforme aumenta β , al contrario que el número de bits transmitidos por cada tag, cuyo valor aumenta conforme aumenta β . Cabe destacar que para valores superiores a $\beta = 0,45$, aproximadamente, el valor de β consigue el mínimo número de slots de tipo go-On. Sin embargo, a partir de valores de β superiores a 0,4, el número de bits transmitidos alcanza un valor máximo estable en torno a 210 bits. En este caso, el punto intermedio entre ambas gráficas lo marca

4.6 Collision window Tree (CwT)

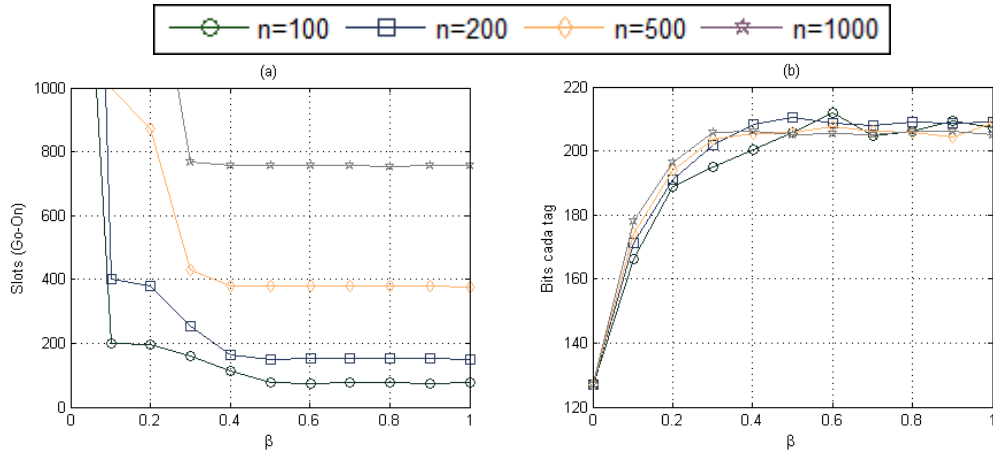


Figura 4.46: Variación del protocolo CwT con función exponencial, para diferentes valores de β en: (a) el número de slots go-On y (b) el número de bits transmitidos por tag, con $k = 128$ bits

el número de go-Ons. Es decir, que es preferible reducir este tipo de slots, al número de bits transmitidos por cada tag, ya que las consecuencias del incremento de los slots son más perjudiciales que las del incremento de bits. A pesar de que ya se intuye que el valor adecuado de $\beta = 0,5$ es el más adecuado, se muestra a continuación el número total de bits empleados por el protocolo.

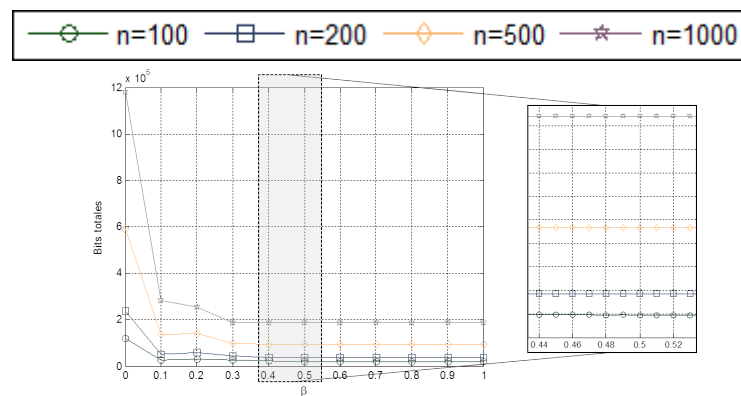


Figura 4.47: Variación del protocolo CwT con ventana exponencial para diferentes valores de β en el número de bits totales transmitidos en la identificación de diferentes grupos de tags, n

4. Metodología de la ventana

En la Figura 4.47 se muestra el número total de bits transmitidos entre lector y tags. Se observa cómo los pequeños conjuntos de tags no reducen el número de bits transmitidos hasta llegar a la franja de valores de $\beta = 0,4$. Por tanto, destaca que la influencia de los slots de tipo go-On en los bits totales es mayor que la de los bits transmitidos por cada tag. Confirmando así que el valor más idóneo es el de $\beta = 0,5$.

4.6.2 Comparativa entre ventanas

Una vez presentadas y sintonizadas las funciones se procede a realizar una comparativa entre ellas. Para ello, se realizan tres simulaciones empleando las tres ventanas presentadas con el valor de β adecuado para conseguir sus máximas prestaciones, y se comparan a posteriori en diferentes aspectos.

Así pues, en las Figuras 4.48.a y 4.48.b se muestran el número de go-On slots y bits medios transmitidos por cada tag, respectivamente. Se observa que la ventana exponencial proporciona el menor número de go-On slots, sin embargo, sus tags envían una mayor cantidad de bits, siendo la ventana lineal y la cuadrática muy parejas en ambas magnitudes.

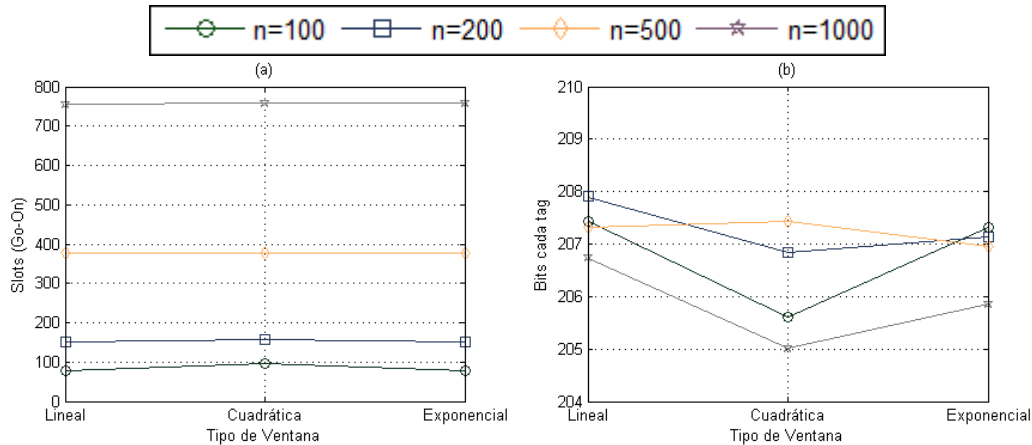


Figura 4.48: Consumos del protocolo CwT con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de go-On slots, (b) número medio de bits transmitidos por cada tag, siendo $k = 128$ bits

Los valores del parámetro β escogidos proporcionaban los mejores resultados en el número total de bits. Es decir, que para conseguir reducir al máximo dicha

4.6 Collision window Tree (CwT)

cantidad, se debían reducir, también, los slots empleados de tipo go-On al máximo. Por contra, el número de bits transmitidos por cada tag, sale perjudicado en esa decisión con respecto al protocolo QwT. Sin embargo, el protocolo ofrecerá mejores prestaciones en términos de latencia y consumo del sistema. Ésto se estudiará con mayor detalle en el capítulo de experimentación (Capítulo 5). En la Figura 4.48.a se observa como el número de go-On slots empleados por las diferentes funciones es similar en entornos de tags densos, mientras que para valores de n menores la ventana lineal consume un número ligeramente menor de slots de este tipo. En cambio, como se puede observar en la Figura 4.48.b es la que emplea un mayor número de bits por cada tag, aunque no es una diferencia muy destacable.

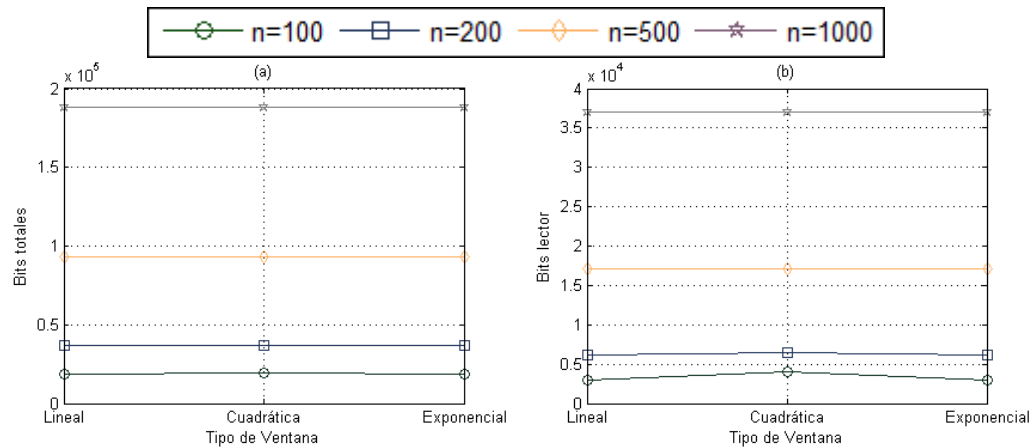


Figura 4.49: Consumos del protocolo CwT con diferentes tipos de ventana identificando diferentes grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.

Para completar esta información, en la Figura 4.49 se muestran dos gráficas: el número de bits totales empleados (a) y el número de bits empleados por el lector (b). En la primera de las gráficas se observa cómo el número de bits empleados por las diferentes ventanas es prácticamente idéntico en todas ellas. Se podría resaltar que la ventana exponencial emplea un número ligeramente superior en todos los conjuntos de tags, aunque es casi inapreciable. No así, en el número de bits transmitidos por el lector, en la siguiente gráfica (4.49.b), donde la ventana exponencial consume mayor número de bits de lector, debido al incremento de slots go-On para menores valores de n .

4. Metodología de la ventana

Destaca que los valores obtenidos en las tres funciones son muy similares. La principal razón para este hecho es que todas las funciones ofrecían las mejores prestaciones acudiendo a ventanas de mayor tamaño, para que la técnica de bit-tracking pudiera ejercer su funcionalidad. Esto es, con respuestas de tags mayores, el lector podría recibir un mayor número de bits comunes previos a la colisión, al contrario que si las ventanas fueran menores, donde se necesitarían más go-ONS para obtener el mismo resultado. Además, el protocolo ha ofrecido su mejor versión cuando las tres funciones empleadas se han sintonizado para ofrecer la mayor pendiente posible en un incremento reducido de L ; en el caso de la lineal y la cuadrática, con valores de β cercanos a k , lo cual producía un comportamiento similar a una función escalón. Ratificando de nuevo, que una vez que el protocolo no detecta colisiones, aumenta el tamaño de ventana de forma vertiginosa, ya que aunque se produzcan colisiones posteriores, no van a ser tan perjudiciales como en el protocolo QwT. La técnica bit-tracking puede obtener más información de ellas, aliviando sus consecuencias. Por contra, no se ha conseguido reducir el número de bits transmitidos por los tags en la misma proporción que en el protocolo QwT. Para paliar este problema se presenta una variante del protocolo.

4.6.3 Collision window Tree plus (CwT+)

Se ha observado que para obtener la mejor versión del protocolo CwT es necesario que éste transmita mayor número de bits por cada tag que los necesarios en el protocolo QwT. En esta sección se presenta una variante del protocolo CwT, en la que al lector se le añade la capacidad de reducir el número de bits transmitidos por los tags, reajustando el tamaño de ventana. A esta modificación del protocolo CwT se le ha denominado Collision window Tree plus (CwT+). El protocolo propuesto en esta sección ofrece una serie de prestaciones similares a las del CwT, estas son sus ventajas y desventajas:

- ◇ Se aprovechan las características de la utilización de la tecnología bit-tracking: mayor destreza en la generación de los nuevos *queries*, al aprovechar los bits recibidos previos al bit colisionado, manteniendo la eliminación de los slots libres.

4.6 Collision window Tree (CwT)

- ◇ Empleando la ventana de bits, consigue disminuir el número de bits transmitidos por los tags sin variar el consumo total de bits transmitidos. Es decir, que involuntariamente se produce un aumento del número de bits que transmite el lector que no perjudica, sin embargo, a la latencia.
- ◇ Se mantiene la característica *memoryless* del protocolo QT, y los tags empleados son iguales que en el protocolo CwT.
- ◇ El aumento del número de bits que transmite el lector, se produce en gran medida por el aumento de slots de tipo go-On. Éste es controlado y no resulta perjudicial para la latencia ni para el consumo del protocolo. Es mas, en caso de trabajar con tags cuya velocidad sea inferior a la del lector, el protocolo trabajaría mejor que el CT y el CwT, en esas mismas circunstancias.

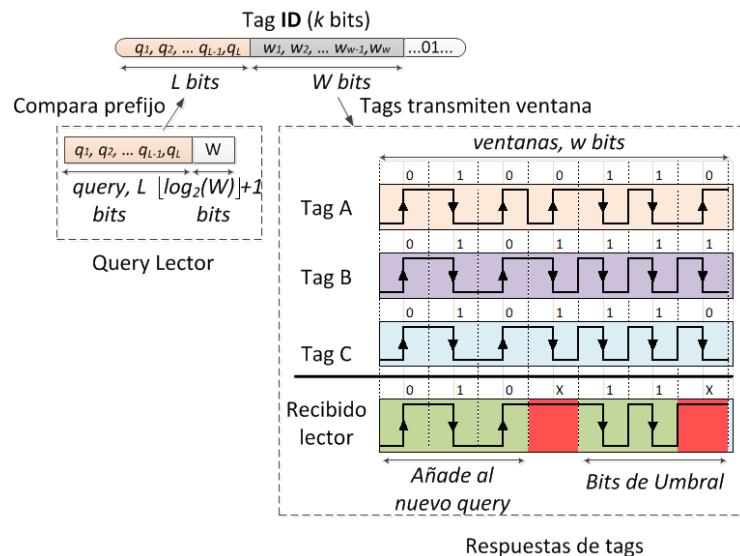


Figura 4.50: Ejemplo de ejecución del protocolo CwT+

Ya se ha comentado que gracias a la técnica de bit-tracking, el lector es capaz de conocer el bit exacto en el que se ha producido la colisión. Y, ¿por qué no emplear también la información de sucesivas colisiones en el mismo mensaje? Esa es la principal idea de esta variante del protocolo CwT. Dado que el lector es capaz de reconocer todas las colisiones que recibe en un mensaje, éste emplea información relativa a las dos primeras colisiones producidas con el fin de determinar un

4. Metodología de la ventana

tamaño más adecuado de la ventana para la posterior interrogación. En la Figura 4.50 se muestra un ejemplo de transmisión del protocolo CwT+. A la cantidad de bits recibida correctamente entre la primera y la segunda colisión, incluyendo la segunda, se le denomina valor *Umbral*. Éste determinará el tamaño de la ventana que deberán emplear los nuevos *queries* generados a partir de dicha colisión. Es decir, que este método permite seguir utilizando la estrategia de generación agresiva de *queries* impuesta por la técnica de bit-tracking y anticipa un valor de la ventana más ajustado a cada situación concreta. De esta forma, se limitan las respuestas de los tags de forma que solo se transmitan los bits adecuados para generar nuevos *queries* válidos, evitando transmisiones innecesarias por parte de los tags.

El protocolo CwT+ trabaja de forma similar al CwT. Primeramente, el lector establece un *query*, un tamaño de ventana inicial y los transmite hacia los tags. El funcionamiento de los tags es exactamente igual al del protocolo CwT y por tanto, si sus respectivos ID coinciden con el *query* recibido, contestan la ventana del tamaño indicado en el parámetro W . Una vez recibidos los datos en el lector, éste comienza a analizar bit a bit empleando la técnica de bit-tracking. En la Figura 4.51 se muestra un ejemplo de ejecución del protocolo ante un slot en colisión, que es donde se produce principalmente, la variación aplicada sobre el protocolo CwT.

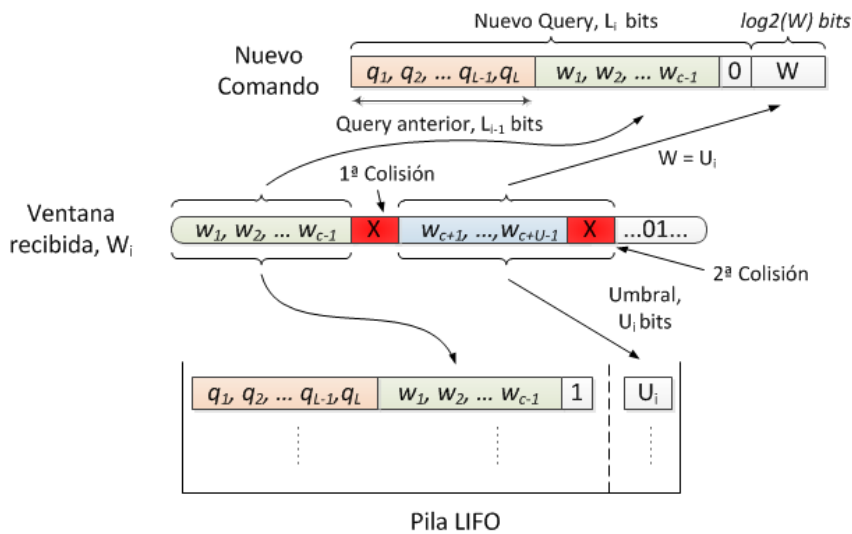


Figura 4.51: Ejemplo de ejecución del protocolo CwT+ en caso de colisión

Se puede observar que en caso de colisión, el lector recibe, w_1, w_2, \dots, w_{c-1} bits previos a la primera colisión, que utiliza para generar dos nuevas *queries* $q_1, q_2 \dots q_L, w_1, w_2, \dots, w_{c-1}, \mathbf{0}$ y $q_1, q_2 \dots q_L, w_1, w_2, \dots, w_{c-1}, \mathbf{1}$, siendo $q_i, w_i \in \{0, 1\}$ y añadiendo un 0 y un 1 a los bits reconocidos antes de la colisión adjuntados al *query* anterior. Posteriormente, se procede a obtener el valor umbral U , para el *query* correspondiente L_i mediante la eq. 4.34.

$$U_{Li} = \begin{cases} b_{c2} - b_{c1} + 1, & b_{c1} > 0, b_{c2} > 0 \\ 1, & b_{c1} = 0 \\ 1, & b_{c2} = 0 \end{cases} \quad (4.34)$$

Los parámetros b_{c1} y b_{c2} representan la posición donde se producen las colisiones del último mensaje recibido. En caso de no producirse ($b_{c1} = 0$ y $b_{c2} = 0$) o que solo se produzca una ($b_{c2} = 0$), el valor umbral se mantiene en 1. Por otro lado, si en una respuesta sí que se producen ambas, se procede a calcular el valor umbral U_{Li} , que se corresponderá con los dos nuevos *queries* generados $qw\mathbf{0}$ y $qw\mathbf{1}$. La razón por la cual se emplea la diferencia más uno, es para evitar que la respuesta de los tags sea un slot de tipo go-On. Sabiendo que los bits entre ambas colisiones se van a recibir correctamente, si se asignara la diferencia entre las posiciones al valor de la ventana se recibirían todos los bits de forma correcta. Esto provocaría que el lector necesitara reenviar el nuevo *query* obtenido al adjuntar la ventana, con el correspondiente incremento de bits y slots. Si, por el contrario, se le añade una posición más, existe la posibilidad de que se produzca una colisión en el último bit recibido de la nueva ventana. Esto no es completamente seguro, ya que no se puede conocer con antelación si la segunda colisión se encontraría dentro de la misma rama de árbol de los tags participantes. De cualquier manera, los beneficios son reseñables por el aprovechamiento de información que se obtiene. Una vez calculado el valor umbral, se asigna al parámetro del tamaño de ventana $W_i = U_{Li}$. Además, como ya se ha mencionado este valor calculado es igualmente válido para los dos *queries* generados, por tanto, también es almacenado en la pila LIFO junto al *query* que permanece a la espera. Así, el lector se asegura de que los bits que va a recibir en respuesta a los *queries* generados van a ser los justos para generar, nuevamente, dos nuevos *queries*. Estos últimos, con altas probabilidades de obtener respuesta y ajustados de forma que todos los bits recibidos sean útiles. Si en el proceso de

4. Metodología de la ventana

detección se hubiera empleado el protocolo CwT, se recalcularía la ventana y se transmitiría el primero de esos queries generados, almacenando el otro en la pila LIFO. Dándose el caso de una ventana calculada de mayor tamaño que el marcado por el umbral de bits, se produciría una transmisión excesiva de $W_{i+1} - U_i$ bits más de los necesarios, es decir, la diferencia entre el tamaño de ventana de bits del siguiente slot y el valor umbral.

En la Figura 4.52 se presenta el diagrama de flujo del protocolo propuesto (CwT+). En ella se refleja el procedimiento que sigue el protocolo ante la colisión explicada previamente y el resto de tipos de slot.

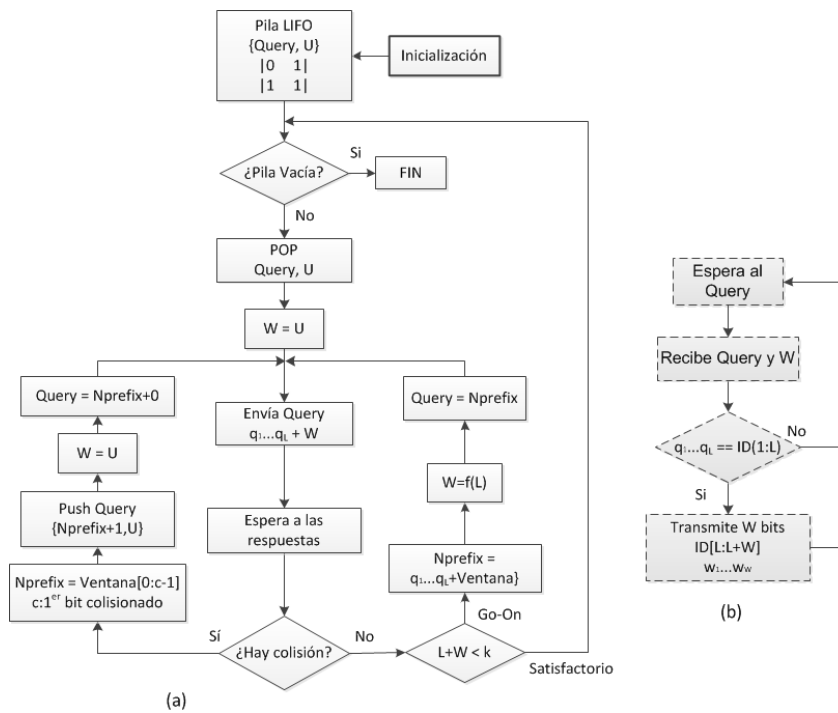


Figura 4.52: Diagrama de flujo del protocolo CwT+: (a) lector y (b) tag

Con el objetivo de discernir si el lector debe emplear esta funcionalidad, la identificación del tipo de slot es una tarea imprescindible para el protocolo. Para ello, se observa si se ha producido colisión ($c = 1$) o no ($c = 0$) y seguidamente, se procede a comprobar la longitud de la dupla L, W , obteniendo tres tipos de slots diferentes:

- ◊ *Slot en colisión* ($c = 1$): ocurre cuando el lector recibe varias respuestas y se

produce una colisión en alguno de los bits recibidos. Como se ha comentado, el lector genera dos nuevos *queries* empleando los bits recibidos anteriores a la colisión, y añadiendo un 0 y un 1. Además, calculará el valor umbral teniendo en cuenta los bits recibidos correctamente entre las dos primeras colisiones.

- ◇ *Slot go-On* ($c = 0$ y $L + W < k$): sucede cuando el lector recibe una o varias respuestas, pero entre los bits recibidos no se produce ninguna colisión. Para considerarse un slots de este tipo se debe cumplir que $L + W < k$. El lector, en este caso, genera un nuevo *query* añadiendo al último enviado la ventana recibida, $q_1, q_2 \dots q_L w_1, w_2 \dots w_W$.
- ◇ *Slot satisfactorio* ($c = 0$ y $L + W = k$): si el lector no encuentra colisión en los bits y se cumple que $L + W = k$. El tag queda identificado con el *query* enviado y la ventana recibida, un total de k bits. Identificado el tag, el lector obtiene un nuevo *query* de la pila y su correspondiente valor umbral para adjudicárselo a la ventana.

Del mismo modo que en los protocolos QwT y CwT, el CwT+ también emplea la función $f(L)$ para determinar el tamaño de la ventana en los casos que se produzca un slot de tipo go-On. Para establecer el valor de ventana en cada uno de los slots comentados se presenta la eq. 4.35:

$$W_i = \begin{cases} f(L_i), & c = 0, L_{i-1} < L_i \\ U_{L_i}, & c = 1, L_{i-1} < L_i \\ U_{L_i}, & L_{i-1} \geq L_i \end{cases} \quad (4.35)$$

La ecuación mostrada contempla tres casos diferenciados por la longitud del *query*, L , empleada por el lector y por si se ha producido colisión en la última respuesta de los tags, $c = 1$. Primeramente, el lector determina si el *query* actual, L_i es mayor que el anterior, L_{i-1} , es decir, si la nueva ventana a generar se destina a una interrogación en la misma rama del árbol. En caso afirmativo, comprueba si ha habido colisión, para en caso de no producirse ($c = 0$), asegurarse de que se trata de un slot go-On recurriendo a la función heurística ($f(L_i)$) para calcular el tamaño de ventana actual (W_i). En el resto de casos es donde se producen las principales diferencias con los protocolos QwT y CwT y su eq. 4.25. Si, por el

4. Metodología de la ventana

contrario, se ha producido al menos una colisión ($c = 1$), el lector asigna el valor umbral calculado para el *query* actual (U_{Li}), al tamaño de ventana (W_i), en lugar de permanecer con el tamaño de ventana constante. Por otro lado, si se produce una terminación de rama por identificación de tag u otra razón, el *query* actual (L_i) pasa a ser inferior al *query* anterior (L_{i-1}). En tal caso, en vez de asignarle un valor fijo de 1 a la ventana, el protocolo CwT+ le asigna de nuevo el valor umbral almacenado en la pila LIFO junto al *query* obtenido (que pasa a ser el actual), U_{Li} . Esta variación genera respuestas de tag ajustadas, de mayor número de bits si se van a recibir correctamente, o de un bit si la siguiente respuesta se sabe que va a ser una colisión. Esto aumenta la eficiencia de bits de los mensajes transmitidos, ya que evita las transmisiones innecesarias y, por tanto, reduce el número de bits transmitidos por cada tag.

4.6.3.1 Funciones de ventana

Dado que el protocolo CwT+ presentado en esta sección es una variación del CwT, las funciones de ventana propuestas son las mismas que las del protocolo en el que se ha basado. Es más, incluso los valores de sintonización de β son idénticos. En este caso, como puede resultar evidente, el protocolo va a trabajar mejor con las pendientes más elevadas de las funciones. Gracias a que el protocolo CwT+ trata de anticiparse a la respuesta que va a recibir en el siguiente slot, limita las ventanas de forma muy acertada y cuando consigue una respuesta correcta, tratará de aumentar el tamaño de la ventana lo más rápido posible (de ahí la gran pendiente de las funciones) para obtener el ID completo del tag lo antes posible. En caso de producirse una colisión en ese momento, las consecuencias no van a ser tan perjudiciales como en el protocolo QwT o el CwT, ya que la técnica de bit-tracking aprovechará los bits recibidos anteriores a la primera colisión, y el valor umbral determinará la próxima ventana.

A continuación, se realiza un análisis más detallado para confirmar las afirmaciones aquí presentadas. Las funciones empleadas son las ya conocidas lineal (eq. 4.27), cuadrática (eq. 4.28) y exponencial (eq. 4.29) y en la Figura 4.53 se muestran los diferentes mínimos de slots de tipo Go-On de cada una de las funciones. Se han realizado las simulaciones para los mismos conjuntos de tags, n , que permiten conocer la tendencia del protocolo.

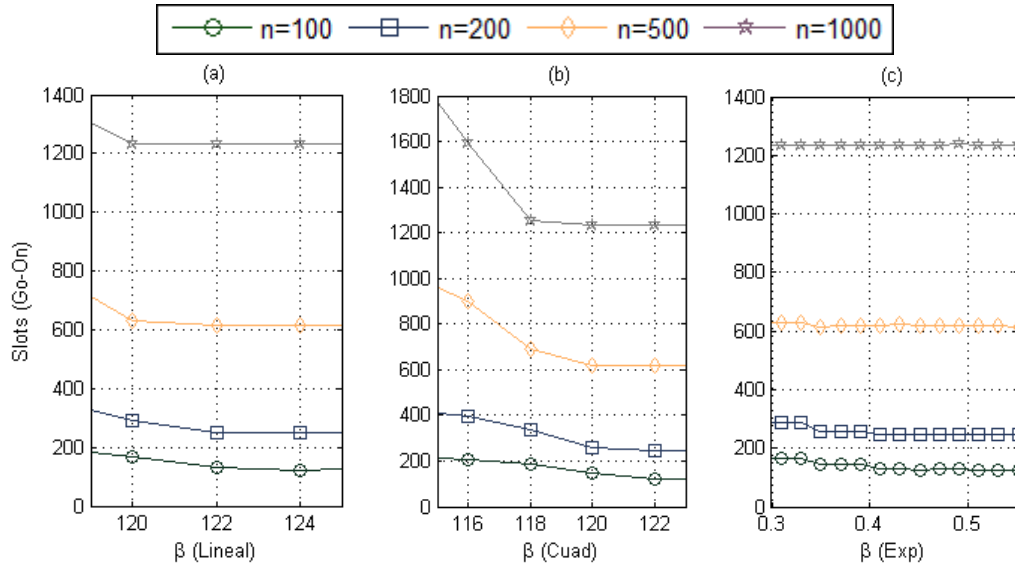


Figura 4.53: Míminos valores de slots de tipo go-On variando β en: (a) función Lineal, (b) función Cuadrática y (c) función Exponencial

La evolución de las gráficas es muy similar a las del protocolo CwT y además, los valores de β para los que se obtienen los mínimos consumos de slots de tipo go-On son los mismos que en su protocolo origen: $\beta = 123$ para la función lineal (Figura 4.53.a), $\beta = 121$ para la función cuadrática (Figura 4.53.b) y $\beta = 0,5$ para la función exponencial (Figura 4.53.c). Cabe destacar que los valores de slots go-On mínimos obtenidos son muy similares en las tres funciones. De nuevo, los valores de β producen funciones con un crecimiento tan drástico que casi la propia forma de la función, que le proporciona su crecimiento característico, resulta indiferente. En el caso de los bits transmitidos por cada tag (Figura 4.54), el resultado es también similar en las tres gráficas mostradas, en los valores escogidos de β . Todas ellas obtienen valores inferiores a 195 bits, siendo $k = 128$ bits. En la siguiente sección, se realiza una comparativa entre las tres funciones aplicadas al protocolo CwT+, y el protocolo CwT.

4.6.3.2 Comparativa de las funciones

En este apartado se realiza la comparativa entre el protocolo CwT+ empleando cada una de las tres funciones mencionadas en la anterior sección, y el protocolo CwT

4. Metodología de la ventana

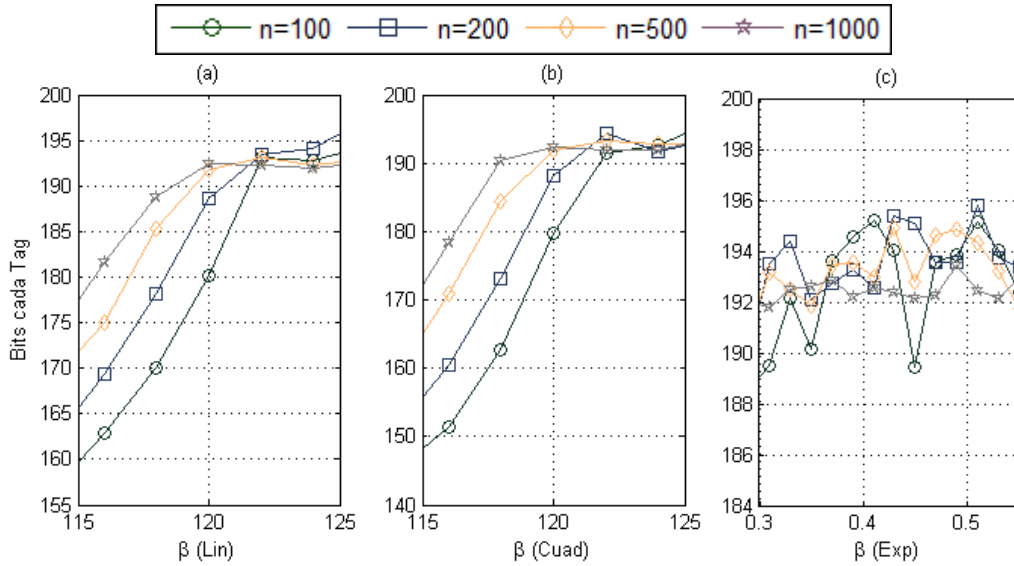


Figura 4.54: Valores de los bits transmitidos por los tags alrededor de los valores de β seleccionados en: (a) función Lineal, (b) función Cuadrática y (c) función Exponencial sabiendo que $k = 128$ bits

utilizando la función lineal, cuyos resultados en términos de consumo total de bits y de slots de tipo go-On son los más resaltables de entre el resto de funciones. De esta forma se trata de, además de discernir qué función es la que más le conviene al protocolo CwT+, establecer cuáles son las mejoras o inconvenientes de la variación realizada al protocolo CwT (CwT+).

Para la comparativa se han realizado tres simulaciones empleando las tres funciones de ventana presentadas con el valor de β adecuado para conseguir sus máximas prestaciones (punto intermedio entre menor número de go-Ons y bits transmitidos por cada tag, y mínimo número de bits totales), y otra simulación adicional del protocolo CwT con la ventana lineal y su valor idóneo $\beta = 123$. De estas simulaciones se obtienen los resultados de los diferentes consumos de los protocolos.

En las Figuras 4.55.a y 4.55.b se muestran el número de go-On slots y bits medios transmitidos por cada tag respectivamente.

Se observa en las gráficas presentadas cómo el protocolo CwT+ no consigue reducir el número de go-On slots de la misma manera que el protocolo CwT. Sin embargo, en la Figura 4.55.b, se observa que el objetivo perseguido por el protocolo propuesto en esta sección es conseguido. Es decir, que el protocolo CwT+ consi-

4.6 Collision window Tree (CwT)

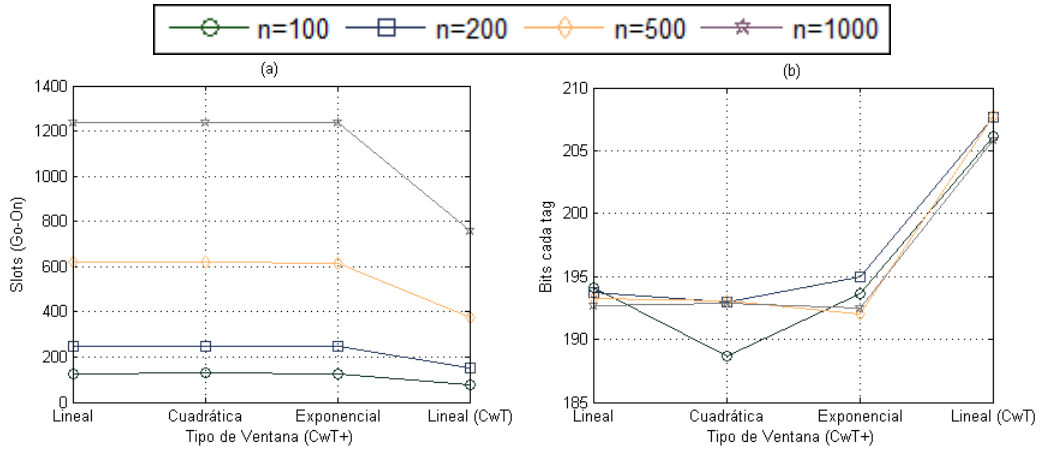


Figura 4.55: Consumos del protocolo CwT+ con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de go-On slots, (b) número medio de bits transmitidos por cada tag.

que reducir el número de bits transmitidos por cada tag empleando cualquiera de las funciones de ventana. Además, como se observa en la Figura 4.56.a, el número total de bits transmitidos entre lector y tags permanece muy similar al de su protocolo origen, el CwT. Dicho de otro modo, el CwT+ consigue reducir el número de bits transmitidos por los tags a costa de un ligero aumento en el número de bits transmitidos por el lector (ver Figura 4.56.b). Este aumento no es reseñable en el caso en el que la velocidad de transmisión de los tags y el lector sea la misma. Sin embargo, cuando los tags no pueden afrontar las altas velocidades impuestas por el lector, el CwT+ ofrecerá mejores prestaciones en latencia. Dado que el CwT+ reduce el número de bits transmitidos por cada tag, el tiempo que el protocolo emplea en esa tarea será menor y, por tanto, el tiempo total que el protocolo destina a identificar el conjunto de tags será igualmente menor.

Por otro lado, en las Figuras 4.57.a y 4.57.b se presentan el número de colisiones y de slots libres empleados por los protocolos CwT+ y CwT, respectivamente. Gracias a la técnica de bit-tracking el número de colisiones permanece constante en ambos protocolos, independientemente del tipo de ventana empleada. Además, la utilización del valor umbral no influye en la completa eliminación de los slots libres, como se puede observar en la Figura 4.57.b.

4. Metodología de la ventana

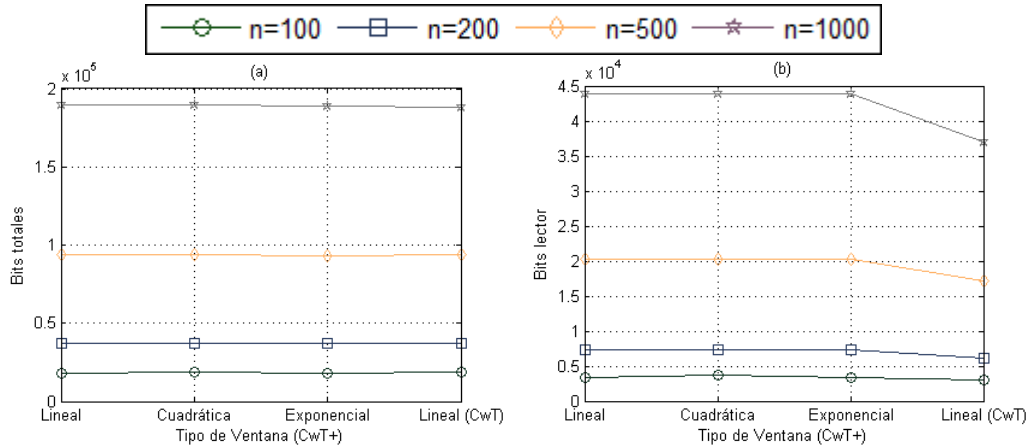


Figura 4.56: Consumos del protocolo CwT+ con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.

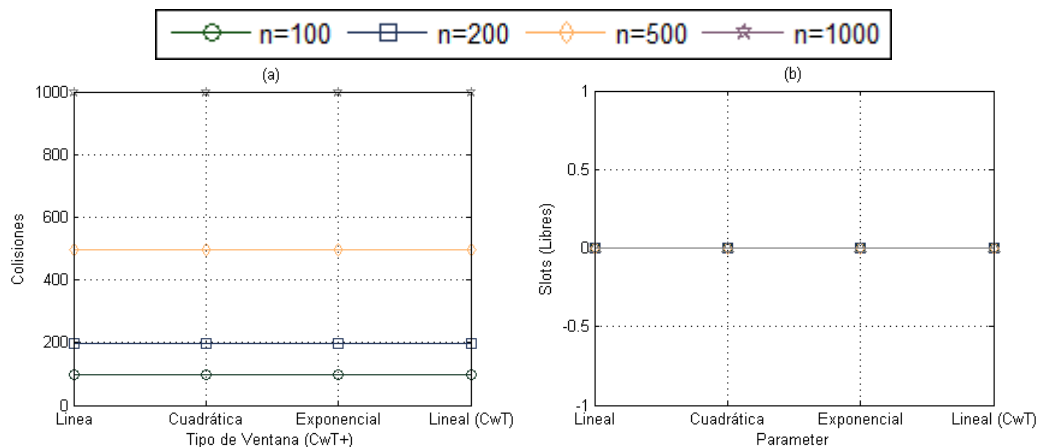


Figura 4.57: Consumos del protocolo CwT+ con diferentes tipos de ventana identificando varios grupos de tags, n , de: (a) número de bits totales transmitidos, (b) número de bits transmitidos por el lector.

4.6.4 Conclusiones

En esta sección se ha presentado un protocolo, el Collision window Tree (CwT), el cuál emplea dos técnicas conocidas y ya presentadas anteriormente: la técnica de bit-tracking y la ventana de bits. Mediante la combinación de estas dos estrategias, las cuales presentan gran compatibilidad, se pretende mejorar dos aspectos cruciales de los protocolos de anti-colisión. Por un lado, el bit-tracking reduce el número

de colisiones que se producen en el protocolo y aprovecha posibles bits comunes previos a la colisión, gracias a la capacidad que dota esta técnica para trazar la colisión a nivel de bit, produciendo una reducción del número de slots totales. Por otro lado, la ventana de bits permite reducir el número de bits transmitidos por los tags evitando la transmisión innecesaria de éstos. De esta manera se consigue un protocolo robusto que centrándose en disminuir el número total de bits transmitidos, consigue grandes resultados en latencia y consumo energético como se verá en el próximo capítulo.

Para definir el comportamiento de la ventana de bits se han aplicado las mismas tres funciones heurísticas que se aplicaron en el protocolo QwT. Los resultados obtenidos evidencian un comportamiento ligeramente diferente al protocolo anterior. Gracias a que la técnica de bit-tracking es capaz de aprovechar los bits comunes recibidos previamente a la primera colisión de bits, el protocolo ofrece mejores resultados incrementando la ventana de forma drástica en cuanto recibe los primeros slots de tipo go-On. Por ello, los valores de β escogidos describen funciones de crecimiento muy abrupto. Esto ofrece grandes resultados en términos de bits totales intercambiados entre lector y tags, ya que disminuye el número de go-On slots. Sin embargo, produce un mayor aumento de los bits transmitidos por lo tags. Los resultados obtenidos presentan unos tags transmitiendo un mayor número de bits en comparación con los obtenidos en el protocolo QwT. Para paliar este efecto secundario de la combinación entre bit-tracking y ventana de bits se ha presentado una modificación del protocolo CwT, el Collision window Tree plus (CwT+).

El CwT+ trabaja de forma similar al protocolo CwT, pero aprovecha la combinación de la tecnología bit-tracking y la ventana en un nivel superior. Además de trazar la colisión hasta el primer bit, el lector tiene en cuenta el número de bits hasta el siguiente y emplea dicha información. De esta forma, conociendo cuantos bits deben transmitir los tags al nuevo *query* ensamblado, se puede evitar ese excedente de bits transmitidos que se produce en el CwT. De nuevo se han aplicado las tres funciones heurísticas aplicadas sobre el QwT y el CwT, y se ha observado como la sintonización del parámetro β es similar a la del CwT. Los resultados que ofrece el protocolo en términos de bits totales transmitidos entre lector y tags también son similares. Este resultado resalta que si los bits totales no varían y los bits transmitidos por los tags descienden, entonces los bits transmitidos por el lector aumentan.

4. Metodología de la ventana

Y efectivamente es así. Esta variante del protocolo produce un ligero aumento de slots de tipo go-On, lo que a su vez genera la necesidad del lector de transmitir un mayor número de bits. De cualquier manera, los resultados globales del CwT y el CwT+ son muy similares, con la diferencia de que las aplicaciones en las que los tags no puedan alcanzar la velocidad de transmisión del lector saldrán beneficiadas si emplean la variante CwT+.

El experimentador que no sabe lo que está buscando no comprenderá lo que encuentra.

Claude Bernard

5

Experimentación y resultados

En el presente capítulo se realiza la experimentación necesaria para validar la hipótesis planteada. Para ello, se ha desarrollado un simulador capaz de obtener los resultados del comportamiento del protocolo diseñado en esta tesis y de otros protocolos existentes en la literatura.

La tecnología RFID presenta sistemas complejos que son influenciados por una gran cantidad de variables externas. En publicaciones realizadas experimentando con un lector y tags reales se ha podido observar como las simulaciones realizadas se acercan a los resultados medidos, pero presentan ciertos errores. En [Landaluce 12a, Landaluce 12b] se diseñó un simulador del protocolo EPC C1G2 [EPCglobal 08] y se comparó con los resultados medidos. Las principales conclusiones obtenidas fueron que, a pesar de que el simulador ofrecía unos datos fiables, existía un error entre ambos resultados. Las pérdidas de propagación de la señal y las interferencias producidas entre los propios tags provocaban un aumento del tiempo de identificación de la simulación. Sin embargo, este error podía ser emulado en la simulación gracias a su repetibilidad en los diferentes experimentos. Por tanto, para la realización de la comparativa entre los diferentes protocolos se ha

5. Experimentación y resultados

optado por simplificar el escenario eliminando este tipo de factores externos que no aportan información extra para dicha comparativa.

El capítulo se va a dividir en varias secciones. Primeramente se procede a describir cómo va a ser el proceso de validación de la hipótesis y el escenario de la simulación, teniendo en cuenta todas las asunciones que se llevan a cabo para todos los protocolos de la comparativa. Seguidamente, se especifican las medidas más importantes que se van a emplear para determinar las ventajas y desventajas de cada uno de los pretendientes. Posteriormente, se presentan los resultados obtenidos de las diferentes simulaciones. Dentro de los resultados presentados, éstos se han clasificado teniendo en cuenta sus diferentes naturalezas. Así pues, en la primera comparativa entran en juego los protocolos basados en árbol (con y sin memoria) en los que se engloba el protocolo propuesto QwT. Tras ello, se añade la técnica de bit-tracking, con los principales protocolos que hacen uso de ella. En esa comparativa entra el protocolo propuesto CwT y la mejora de éste, el CwT+. Finalmente, una vez presentados los resultados de todas las versiones propuestas, se establecen una serie de resultados donde se pueden comparar las propuestas de esta tesis entre sí.

5.1 Estrategia de validación de la hipótesis

El objetivo principal de la experimentación es la verificación del cumplimiento de la hipótesis. En el caso de la presente tesis, se considera por las razones anteriormente mencionadas, que la opción de validación más adecuada es la simulación de los diferentes protocolos, evitando gran parte de los factores externos que dificultan la experimentación de un sistema RFID. La simulación aporta un control total y absoluto sobre el sistema RFID, pudiendo influir en cualquier punto del proceso de identificación y conocer su información interna en todo momento. Dado que el interés de esta tesis reside en la transmisión de bits de los protocolos, las simulaciones describen el comportamiento de los diferentes protocolos, evitando factores anteriormente mencionados.

Para definir cómo se va a proceder a la experimentación, se muestra nuevamente la hipótesis planteada en el Capítulo 1, alrededor de la cual gira la presente tesis.

5.1 Estrategia de validación de la hipótesis

«Es posible diseñar un procedimiento que, aplicado sobre un protocolo de anti-colisión RFID basado en *queries* ya existente, consiga reducir el número de bits transmitidos por los tags, mejorando el comportamiento del protocolo en latencia y consumo energético en entornos de tags pasivos.»

La hipótesis presentada se puede desgranar en dos ideas principales. Por un lado, ésta enuncia que se pretende desarrollar un procedimiento aplicable a protocolos basados en *queries*, por tanto, debe demostrarse su aplicabilidad a este tipo de protocolos. En el Capítulo 4 ya se ha visto cómo la ventana se ha aplicado a dos protocolos de anti-colisión basados en *queries*. En este capítulo, además, se pretende demostrar que el acoplamiento de este procedimiento resulta beneficioso.

Con esta idea se acude a la segunda parte de la hipótesis, que sostiene que dicho procedimiento, además, mejorará el comportamiento en latencia y consumo energético, del protocolo conformado en entornos de tags pasivos. Para ello, se procederá a realizar la comparativa entre el protocolo de anti-colisión origen y el conformado empleando la ventana. Con la intención de obtener los resultados de latencia y consumo energético, y de encontrar las respuestas a los posibles comportamientos que muestren los protocolos, se deberán obtener, previamente, otras medidas que faciliten la obtención de las mencionadas. Además de todo esto, se procederá también a realizar dichas comparativas con otros protocolos existentes en la literatura reciente, demostrando que los protocolos conformados ofrecen un resultado destacable dentro del estado del arte actual.

Se pretende, por tanto, conocer los resultados de los protocolos propuestos, y compararlos con algunos de los existentes más recientes. Para ello, no solo es necesaria la simulación de los propios protocolos diseñados, sino que también se debe proceder a simular los protocolos participantes en la comparativa. Esto es necesario además, ya que la comparativa debe realizarse en el mismo entorno, empleando las mismas condiciones y parámetros para cada uno de los protocolos. De esta forma, se realiza una comparativa lo más justa y equitativa posible. Las simulaciones llevadas a cabo se engloban dentro de un banco de pruebas desarrollado en esta tesis.

5. Experimentación y resultados

5.2 Banco de pruebas

Se ha realizado un banco de pruebas empleando Matlab R2012b. Mediante éste se pretenden simular los comportamientos de los protocolos diseñados en este documento y los ya existentes en la literatura. La simulación de un protocolo consiste en describir su comportamiento y emular el proceso que seguiría la identificación de un conjunto de tags n , empleando su correspondiente estrategia de arbitraje entre el lector y los propios tags. Cada una de las simulaciones se realiza un total de 100 iteraciones, y cada simulación no finaliza hasta que se produce la identificación del conjunto completo de tags.

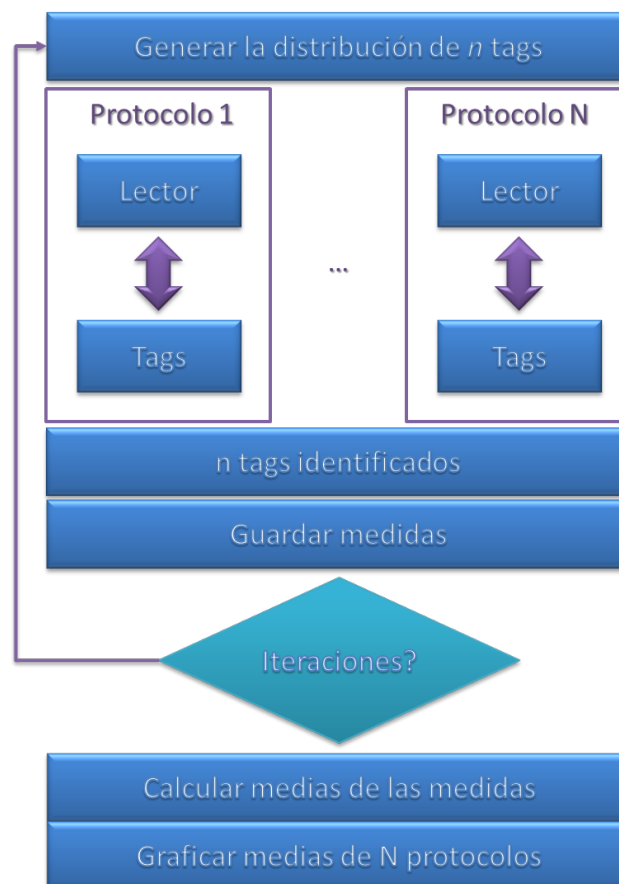


Figura 5.1: Esquema del banco de pruebas empleado

Todas las simulaciones se realizan en las mismas condiciones, variando únicamente los IDs del conjunto de tags. La naturaleza probabilística de los protocolos

basados en Aloha o la dependencia de los protocolos basados en árbol de los ID de sus respectivos tags, son algunas de las razones que requieren la realización de la simulación un gran número de iteraciones. Los datos obtenidos en cada una de las simulaciones se almacenan y se computan con los obtenidos en el resto de iteraciones para calcular los valores medios, máximos y mínimos. Finalmente, y a efectos prácticos, se realiza el graficado de las medias obtenidas en el paso anterior, de forma que se pueda realizar la comparativa entre los diversos protocolos. En la Figura 5.1 se muestra un esquema del banco de pruebas empleado. La realización de un banco de pruebas ha permitido la variación de diversos parámetros que afectan directamente a los protocolos. De esta manera, se han podido observar las influencias de dichos parámetros y cómo el procedimiento de la ventana se ve afectado por ellos.

5.2.1 Presentación del escenario

En esta sección se presenta el escenario que se ha empleado en las simulaciones. Dado que se va a estudiar el problema de las colisiones de tag, se plantea un escenario en el que únicamente existe un lector RFID. Con la finalidad de evitar la influencia de la distancia en la interrogación de los tags y el *capture effect*, el lector se sitúa en el centro junto a una antena omnidireccional. El conjunto de tags se disponen alrededor del lector y a la misma distancia entre sí. En la Figura 5.2 se muestra un esquema de la situación propuesta.

Un sistema RFID se ve expuesto a una gran cantidad de factores externos que modifican su comportamiento. Dado que en esta tesis se está estudiando una solución para el problema de las colisiones de tags, se pretende aislar este problema y darle una solución evitando este tipo de agentes externos que modificarían, no solo los resultados, sino también el propio problema. Para ello, se han asumido los siguientes puntos:

- ◇ El tiempo se subdivide en slots. Como ya se ha comentado anteriormente, un slot determina un espacio de tiempo en el que se produce la comunicación de un subciclo entre lector y tag. Este subciclo puede estar compuesto por un comando del lector y una respuesta de tag, aunque en algunos métodos como los basados en Aloha y los de tipo TS, se complementan con otra respuesta

5. Experimentación y resultados

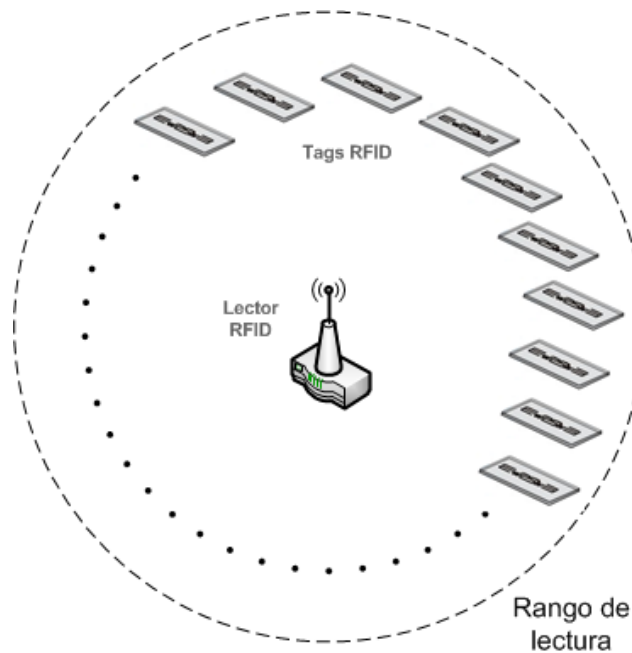


Figura 5.2: Escenario de las simulaciones

del lector a modo de confirmación. En el caso de protocolos cuyos tags emplean seudónimos (Aloha, ver Sección 2.2 o QTRN16, ver Sección 3.2.4), un mismo slot satisfactorio incluirá, además, otra solicitud del lector, en este caso de su ID; el envío del ID por parte del tag; y la correspondiente confirmación del lector. Es decir, que dependiendo del protocolo, los slots pueden ser de mayor o menor duración. En caso de ocurrir una colisión en la primera interrogación, terminaría el slot, pero si el lector interpreta la respuesta, entonces solicita el resto de la información en el propio slot.

- ◇ El canal de transmisión (aire) se asume perfecto. Ningún error de transmisión de señal ocurre en él. Las colisiones se producen por diferencias de bits, y no por degradación de las señales. Al considerarse perfecto, tampoco se tiene en cuenta el *capture effect*, mediante el cual si la potencia de una señal es muy superior a otra en el mismo canal, la primera eclipsa a la de menor potencia y el lector solo detecta la señal fuerte.
- ◇ Las señales transmitidas por los tags en el canal se realizan de forma simultánea y sincronizada. De esta forma, las respuestas de los tags alcanzan el

lector y sus bits se solapan perfectamente. Esta asunción puede considerarse razonable ya que situando todos los tags a la misma distancia del lector, el tiempo de propagación es similar y cercano a 0.

- ◇ El canal de transmisión puede considerarse ternario, es decir, que la interrogación de los tags puede dar como resultado tres estados: presencia de cero, uno o varios tags. Sin embargo, en el caso particular de la ventana, el canal de transmisión resulta ser cuaternario. Dado que los tags no envían su ID completo pueden darse situaciones en que la respuesta es correcta pero la ventana no abarca todo el ID del tag, en ese caso la respuesta es incompleta y se le denomina go-On slot. En tal caso, se dan cuatro estados: cero tags responden, un tag responde completando su ID, uno o varios tags responden de forma correcta sin completar su ID, y varios tags responden colisionando.
- ◇ La distribución de los ID de los tags es homogénea. Es decir, que cuando se forma la población de tags, la probabilidad de obtener un 0 o un 1 en cualquier bit del identificador es $1/2$.

El número de tags que se sitúan alrededor del lector variará en función del tipo de simulación propuesta. Por otro lado, el ID de cada tag se considera de una longitud $k = 128$ bits. Estos identificadores se generan de forma dinámica en cada iteración de la simulación. Para ello se emplea una semilla diferente en cada generación, asegurando poblaciones diferentes pero homogéneas. Por tanto, se utilizan diferentes poblaciones de tags, todas ellas cumpliendo la restricción de distribución homogénea.

5.2.2 Medidas del protocolo

La gran mayoría de los protocolos presentados en el Capítulo 2 emplean la medida *slot* para determinar la mejora del protocolo. A priori puede parecer que un protocolo que emplea un menor número de slots tendrá un mejor comportamiento que otro que le supere en ese aspecto. Esta afirmación no es totalmente cierta. La variable o medida de slot es flexible en tiempo, es decir, que dependiendo del protocolo, un slot está compuesto por varios comandos de lector y respuestas de tags. Hay protocolos que emplearán menos slots para identificar los tags, pero éstos estarán

5. Experimentación y resultados

compuestos por un mayor número de bits y su duración será mayor. En computo global, ambos protocolos pueden emplear el mismo tiempo en identificar los tags [Choi 10]. Por ello, en esta tesis no se considera el número de slots como una medida cuyas comparaciones sean relevantes, sino que deberá ser completada con otras, cuya aportación puede ser más importante. Las medidas escogidas para las comparativas son las siguientes:

- ◇ El **número de slots**. Ya se ha comentado lo que engloba un slot, y cómo se ha empleado en numerosos artículos de la literatura para evaluar los protocolos diseñados. En esta tesis se van a presentar estos datos, pero también se va a esclarecer que no necesariamente son relevantes ya que otras medidas lo demuestran.
- ◇ El **número de bits** empleados en la detección de los tags es una magnitud que resulta ser más fiable que el número de slots. Dado que un slot está compuesto de bits y puede considerarse una subdivisión de éste, esta magnitud ofrecerá unas medidas más exactas y permitirá una comparativa más ajustada a la realidad. Para calcular el número de bits totales transmitidos se deben tener en cuenta: los bits que transmite el lector en cada uno de los comandos enviados a los tags, B_R , y la superposición de las respuestas de los tags recibidas en el lector, B_T [Yan 08, Yan 10]. En la eq. 5.1 se muestra el cálculo de B_R , que se realiza en cada slot i desde el comienzo del ciclo de interrogación hasta el final de éste, cuando se han identificado todos los tags en t slots y la variable B_{Rslot} representa el número de bits que transmite el lector en cada slot. A continuación, se especifica la expresión para el protocolo QwT propuesto en esta tesis, donde L_i representa la longitud del *query* enviado en el slot i . Los bits necesarios para enviar el tamaño de ventana calculado para ese slot, se calculan mediante la expresión $\lfloor \log_2 W_i \rfloor + 1$, donde $\lfloor \cdot \rfloor$ representa el redondeo al entero más cercano hacia $-\infty$.

$$B_R(bits) = \sum_{i=1}^t B_{Rslot} = \sum_{i=1}^t (L_i + \lfloor \log_2 W_i \rfloor + 1) \quad (5.1)$$

Para el cálculo de los bits transmitidos por los tags, visto desde el lector, se emplea la eq. 5.2, donde B_{Tslot} representa los bits transmitidos por cada tag

en un slot. Especificando la expresión para el protocolo QwT y dado que las respuestas de los tags llegan de forma simultánea al lector, el total de bits de tag se calcula como la suma de los bits recibidos en cada slot, es decir, el valor de la ventana, W y los bits de CRC, si es que emplea como en el caso del QwT.

$$B_T(bits) = \sum_{i=1}^t B_{Tslot} = \sum_{i=1}^t (W_i + CRC_{bits}) \quad (5.2)$$

Por otro lado, también se presentan los datos del número medio de bits transmitidos por cada tag a lo largo del ciclo de interrogación. Para calcular esta medida, el simulador almacena los bits que ha transmitido cada tag para ser identificado, es decir, la suma de los B_{Tslot} bits cada vez que transmite, hasta un total de y ciclos y, posteriormente, los divide entre los n tags para dar un número de bits promedio. En particular, para el protocolo QwT, cada tag transmite W_{ij} bits, el CRC correspondiente, y realiza la media entre todos los datos obtenidos (ver eq. 5.3) entre el conjunto de tags, n .

$$\bar{b}_T(bits) = \frac{\sum_{i=1}^n \sum_{j=1}^{y_i} B_{Tslot}}{n} = \frac{\sum_{i=1}^n \sum_{j=1}^{y_i} (W_{ij} + CRC_{bits})}{n} \quad (5.3)$$

- ◇ En la medida del número de bits no se contemplan parámetros como la codificación empleada o la velocidad del sistema RFID, que pueden variar su comportamiento de forma clara. Además, los slots libres tampoco aparecen en la expresión del cálculo. Éstos pueden ser una fuente de incremento del tiempo de identificación y, por eso, se tiene en cuenta la medida de **latencia** del protocolo, D . Por otro lado, es una medida con la que se puede realizar una comparativa más exhaustiva con otros protocolos, aunque también está influida por un mayor número de parámetros externos. Estableciendo la velocidad de transmisión de bits (afectada por la codificación empleada), se puede conocer el tiempo que se tarda en transmitir una determinada cantidad de bits, T (ver 5.4), donde B es el número de bits transmitidos y V la velocidad o *data-rate* a la que son transmitidos.

5. Experimentación y resultados

$$T(s) = \frac{B(bits)}{V(bps)} \quad (5.4)$$

La eq. 5.5, permite conocer el tiempo que emplea el protocolo en identificar los tags contenidos en la zona de interrogación, donde T_R y T_T representan el tiempo que lector y tags, respectivamente, tardan en transmitir los bits necesarios para completar el ciclo de interrogación. Estos tiempos, a su vez, dependen del número de bits que lector B_R , y tags B_T , transmiten, y la velocidad o *data rate* a la que son transmitidos del lector a los tags V_R , y viceversa V_T . Aparte de esto, como se ha comentado anteriormente, se debe considerar el tiempo en el que el protocolo ha sufrido slots libres, T_I . Este tiempo puede calcularse asignando el equivalente a un número de bits que el lector esperará antes de considerar la falta de respuesta y continuar con el proceso. A la suma de estos bits ficticios, que el lector espera recibir y nunca llegan se les ha denominado B_{idle} , y el tiempo empleado en esperarlos es B_{idle}/V_R . Por otro lado, en la presente tesis, los tiempos de procesamiento y propagación se consideran despreciables.

$$D(s) = T_R + T_I + T_T = \frac{B_R}{V_R} + \frac{B_{idle}}{V_R} + \frac{B_T}{V_T} \quad (5.5)$$

- ◇ Al igual que la latencia, la **energía consumida** en el proceso de identificación resulta una magnitud fiable para la comparativa. La energía consumida, E , es una función del tiempo de escaneado, D , (ver eq. 5.6), que a su vez depende del número de bits que se transmiten en la comunicación de lector y tags, y la potencia de transmisión, P . Los tiempos empleados en otras tareas diferentes a la transmisión y recepción se consideran despreciables, para dotar de mayor simplicidad al modelo [Klair 07a, Namboodiri 10].

$$E = P(W) \times D(s) \quad (5.6)$$

Dependiendo de si se consideran tags pasivos o activos, se debe o no considerar la potencia del tag respectivamente. En caso de considerarse activos, por un lado, se debe tener en cuenta la energía consumida por el lector en

5.2 Banco de pruebas

la transmisión y en la recepción, E_{Rtx} , E_{Rrx} eq. 5.7, siendo P_{Rtx} la potencia empleada en la transmisión y P_{Rrx} en la recepción, y T_{Rtx} y T_{Rrx} los tiempos empleados en el proceso, respectivamente [Namboodiri 10].

$$E_R = P_{Rtx} \times T_{Rtx} + P_{Rrx} \times T_{Rrx} \quad (5.7)$$

A continuación, la energía media consumida por cada tag, \bar{e}_T en la eq. 5.8, donde P_{Ttx} es la potencia empleada por un tag en la transmisión, \bar{t}_{Ttx} su tiempo medio de transmisión de sus bits del identificador, y P_{Trx} la potencia empleada en la recepción y \bar{t}_{Trx} el tiempo medio empleado en recibir los bits.

$$\bar{e}_T = P_{Ttx} \times \bar{t}_{Ttx} + P_{Trx} \times \bar{t}_{Trx} \quad (5.8)$$

Por otro lado, si los tags se consideran pasivos, el lector debe transmitir la señal CW para alimentar a los tags permitiéndoles generar la respuesta. Por ello, se considera la potencia de transmisión del lector, P_{Rtx} , cuando envía los bits hacia los tags, T_R , y tanto la potencia de recepción, P_{Rrx} , como la de transmisión, P_{Rtx} , cuando recibe los bits de los tags durante T_T . Así pues, la eq. 5.9 muestra cómo se realiza el cálculo de la energía empleada por un lector de un sistema RFID con tags pasivos.

$$E = P_{Rtx} \times T_R + (P_{Rtx} + P_{Rrx}) \times T_T \quad (5.9)$$

Para la obtención de valores numéricos en las simulaciones se ha optado por emplear los datos de dos dispositivos comerciales, el lector *Philips MF RC530 Reader IC* y para tags activos el *Infineon TDA5250* empleado en [Namboodiri 10]. Estos datos son mostrados en la Tabla 5.1.

Lector		Tags Activos	
P_{Rtx}	P_{Rrx}	P_{Ttx}	P_{Trx}
825 mW	125 mW	35 mW	28 mW

Tabla 5.1: Potencias empleadas por un sistema RFID comercial

5. Experimentación y resultados

- ◇ Como una de las finalidades principales de la ventana diseñada es la de reducir el número de bits transmitidos por los tags, es necesaria una magnitud que permita comparar esta característica con otros protocolos de cualquier naturaleza de una forma clara y razonable. En la eq. 4.1 se presentaba la eficiencia de slots. Si se emplea esa misma estrategia enfocada en la transmisión de bits se propone la **eficiencia de bits**, U_B . Esta medida muestra de forma clara el aprovechamiento que se produce de los bits transmitidos en cada slot y ofrece una clara visión de la proporción de bits que se desechan de forma innecesaria. Además, ofrece información sobre el aprovechamiento que se produce de los slots. La eq. 5.10 indica el procedimiento de cálculo de la eficiencia de bits.

$$U_B = \frac{k(bits)}{b_T(bits)} \quad (5.10)$$

5.3 Simulaciones y resultados de QwT

En esta sección se presentan los resultados de las simulaciones realizadas del protocolo QwT. Primeramente se realiza una comparativa entre el protocolo QT origen, y las versiones presentadas del QwT, es decir, con las tres ventanas propuestas (lineal, cuadrática y exponencial). De esta forma se pueden observar claramente los cambios que produce la ventana sobre el protocolo origen. Para las primeras comparativas cabe destacar que se han empleado CRCs de dos tamaños diferentes: un bit o bit de paridad (QwT-CRC1) y cinco bits (QwT-CRC5). Además, posteriormente se realiza un análisis del protocolo QwT ante diferentes valores del CRC.

Para la realización de las simulaciones, se especifican en la Tabla 5.2 los parámetros empleados necesarios para definir todas las incógnitas. La elección de estos valores se ha realizado acorde a datos de dispositivos físicos existentes, como los especificados en la Sección 5.2.2 o empleados en artículos de la literatura existente. Estos valores son consistentes con sistemas RFID que trabajan en el rango UHF de frecuencias.

Como se ha propuesto en la hipótesis, se pretende diseñar una metodología que permita reducir el número de bits transmitidos por los tags. La Figura 5.3, muestra

5.3 Simulaciones y resultados de QwT

Símbolo	Definición	Valor
k	Longitud del ID de los tags	128bits
V_R	Data rate (velocidad de transmisión) del lector	160kbps
V_T	Data rate (velocidad de transmisión) de los tags	80kbps
P_{Rtx}	Potencia del lector en la transmisión	825mW
P_{Rrx}	Potencia del lector en la recepción	125mW
P_{Ttx}	Potencia del tag activo en la transmisión	35mW
P_{Trx}	Potencia del tag activo en la recepción	28mW
b_{libre}	Equivalencia en bits del tiempo de espera en caso de un slot vacío	$k/4$
C	Número de bits que componen un comando	2bits
Q_c	Query en el protocolo Q	22bits
Q_{rep}	Comando del protocolo Q que no varía Q	4bits
Q_{adj}	Comando del protocolo Q que varía Q	9bits
T_{ext}	Bits de Preámbulo en el protocolo Q	10bits
t	Numero de respuestas en un slot	-
n	Conjunto de tags en el rango de identificación	-

Tabla 5.2: Valores de las distintas variables empleadas en la simulación

los resultados de las simulaciones del número promedio de bits transmitidos por cada tag a lo largo del ciclo de interrogación del protocolo QwT propuesto en esta tesis, comparado con su protocolo origen, el QT. En ella se observan los beneficios de las diferentes ventanas aplicadas sobre dicho protocolo. Para el cálculo de los resultados presentados se han empleado las expresiones mostradas en la Tabla 5.3, en la que se indican los bits que transmite cada tag en un slot, para luego aplicar la eq. 5.3.

Protocolo	Bits/slot cada tag, B_{Tslot}
QwT	$W + CRC$
QT	$k - L$

Tabla 5.3: Tag bits/slot de los protocolos QwT y QT

5. Experimentación y resultados

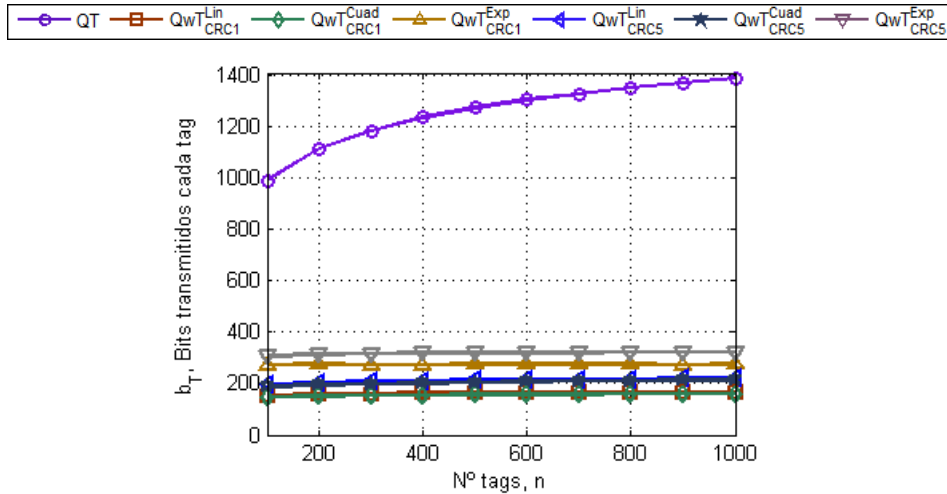


Figura 5.3: Promedio de bits transmitidos por cada tag de los protocolos QT y QwT empleando los tres tipos de ventanas propuestas y dos tamaños de CRC.

El promedio de bits transmitidos por cada tag se reduce de forma muy sustancial al aplicar la ventana sobre el protocolo QT, probando su efectividad en ese aspecto. Cualquiera de los tres tipos de ventanas produce una gran reducción del número de bits transmitidos por cada tag, siendo la más destacada la ventana cuadrática con un CRC de un bit. Se debe mencionar que los tags que emplean un CRC de menor tamaño (un bit), transmiten en general un menor número de bits. Por otra parte, de las tres ventanas empleadas, la cuadrática es la que mejor comportamiento muestra si se compara con las otras ventanas de la misma longitud de CRC. Por tanto, al emplear un CRC de menos bits, el número de bits que transmite el tag es menor, aunque la fiabilidad de las respuestas se reduce. En la sección 5.3.1 se realiza un estudio más extenso sobre la influencia del CRC en el protocolo.

La reducción que produce la ventana en el número de bits transmitidos por los tags queda patente con los resultados mostrados, sin embargo, como ya se ha observado en el estudio realizado sobre la ventana en el Capítulo 4, la aparición del slot de tipo go-On provoca el aumento de los bits transmitidos por el lector. En la Figura 5.4 se muestran los bits transmitidos únicamente por el lector y el total de bits intercambiados entre lector y tags, vistos desde el propio lector. Además, en la Tabla 5.4 aparecen indicados los bits que transmite y recibe el lector en cada slot para cada uno de los dos protocolos comparados. Los resultados mostrados se han

5.3 Simulaciones y resultados de QwT

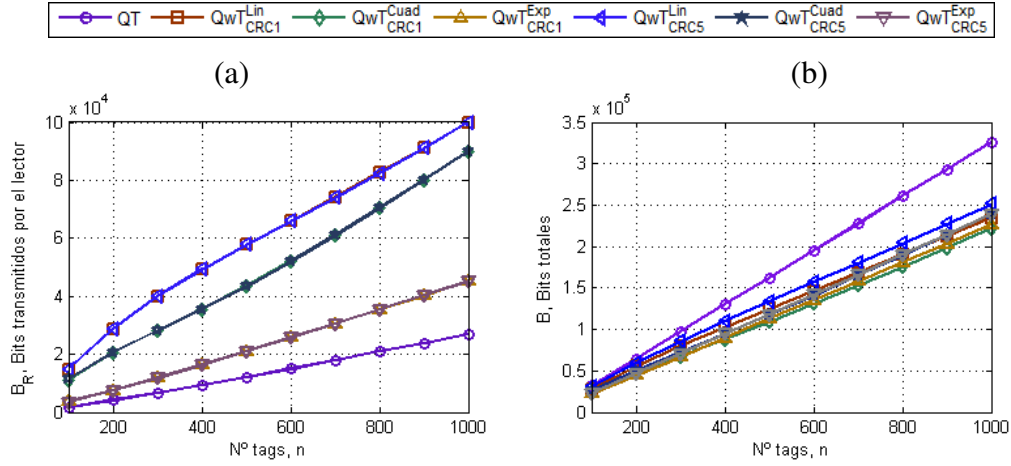


Figura 5.4: Comparativa de QwT con QT en: (a) bits transmitidos por el lector, (b) bits totales transmitidos entre lector y tags

Protocolo	Bits transmitidos en cada slot		
	Lector, B_{Rslot}		Tags, B_{Tslot}
QwT	$L + \lfloor \log_2 W \rfloor + 1$	+	$W + CRC$
QT	L	+	$k - L$

Tabla 5.4: Bits/slot de los protocolos QwT y QT

calculado teniendo en cuenta la eq. 5.1, 5.2 y la suma de ambas.

La aparición del slot de tipo go-On provoca un aumento considerable de los bits transmitidos por el lector con respecto al protocolo QT, siendo la ventana exponencial la que provee el mejor comportamiento de entre todos los tipos de ventana. A pesar de esta consecuencia, el número total de bits transmitidos permanece inferior al del QT. Es decir, que la reducción de bits transmitidos por los tags que produce la utilización de la ventana, compensa el aumento de éstos en el lector. De hecho, el mejor comportamiento lo presenta la ventana cuadrática con un CRC de un bit, que es la ventana que presentaba el mejor comportamiento en la Figura 5.3 del promedio de bits transmitidos por cada tag. A ésta le sigue de cerca la ventana exponencial de la misma longitud de CRC.

La reducción que se produce en el número total de bits transmitidos se ve reflejada en que la latencia del protocolo se reduce, como se puede observar en la Figura

5. Experimentación y resultados

5.5. Los datos empleados de *data rate* o velocidades de tag y lector se muestran en la Tabla 5.2. En la Sección 5.4.1 se analiza la influencia de dicho parámetro sobre el protocolo QwT, y cómo afecta a la latencia, comparándolo con otros existentes en la literatura. En la Tabla 5.5 se indican las expresiones empleadas para el cálculo de la latencia en cada slot, que junto con la eq. 5.5 han sido empleadas para obtener los resultados presentados.

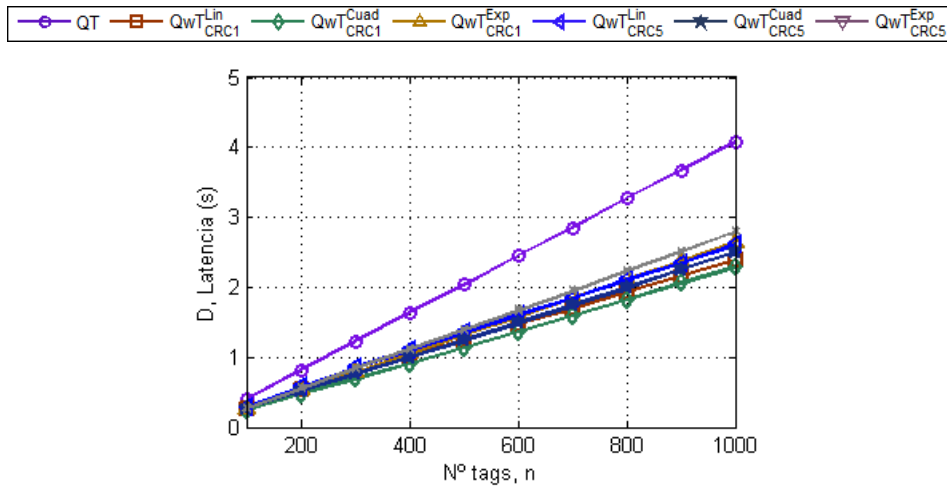


Figura 5.5: Latencias de los protocolos QwT y QT

Protocolo	Latencia de cada slot	
	Lector	Tags
QwT	$\frac{L + \lceil \log_2 W \rceil + 1}{V_R}$	$+$ $\frac{W + CRC}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$
QT	$\frac{L}{V_R}$	$+$ $\frac{k - L}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$

Tabla 5.5: Latencia/slot de los protocolos QwT y QT

Los resultados de la latencia exhibidos resultan ser satisfactorios, presentando un protocolo que ofrece grandes prestaciones en velocidad de identificación de los tags. La ventana cuadrática, de un bit de longitud de CRC presenta los mejores resultados, mejorando los tiempos del protocolo QT y del resto de ventanas. Por tanto, se demuestra que mediante la reducción de los bits transmitidos por los tags, el protocolo propuesto QwT consigue reducir el tiempo total necesario para identificar el conjunto de tags (latencia), lo que se traduce en un incremento en la ve-

5.3 Simulaciones y resultados de QwT

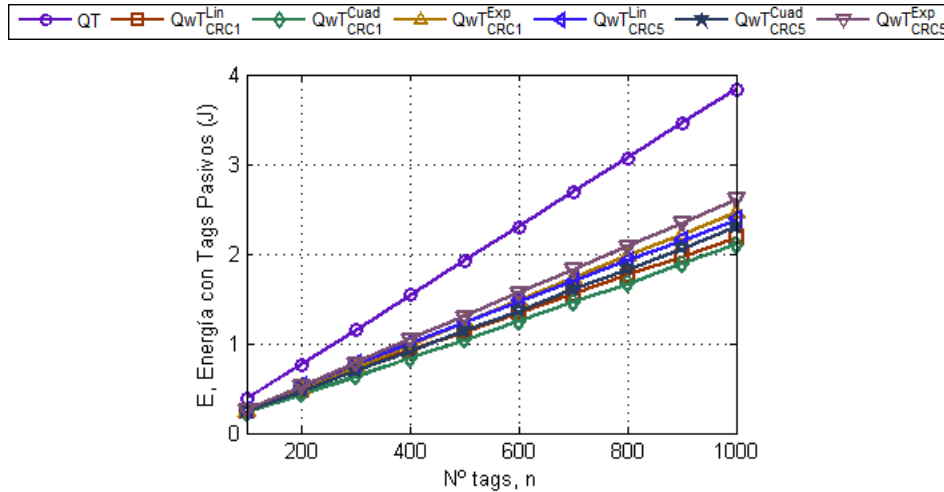


Figura 5.6: Energía consumida por los protocolos QwT y QT considerando tags pasivos

locidad de identificación de los tags situados en la zona de interrogación. Además, cuanto mayor es el conjunto de tags, el protocolo obtiene mejores resultados de tiempo en comparación con el resto, llegando a un ahorro de casi dos segundos para un conjunto de tags $n = 1000$ entre el QwT y su protocolo origen sin la ventana de bits, QT.

En cuanto al consumo energético del protocolo se presentan los casos en que los tags se consideran pasivos o activos:

- ◇ Tags pasivos: en este caso es el lector quien alimenta a los tags mediante la transmisión de la onda continua (CW) y, por tanto, solo se considera el consumo de éste. Por ello, dado que debe alimentar a los tags mientras éstos responden, se debe tener en cuenta la potencia de transmisión y de recepción, ya que transmitirá la CW e interpretará las respuestas de los tags simultáneamente. En la Figura 5.6 se muestra la energía consumida por el lector identificando tags pasivos. Para el cálculo de estas gráficas se ha empleado la expresión presentada en la eq. 5.9 para el cálculo energético de un sistema RFID con tags pasivos.

Los resultados obtenidos en la gráfica presentada son muy similares a los mostrados en la de latencias (Figura 5.5), siendo de nuevo el protocolo que

5. Experimentación y resultados

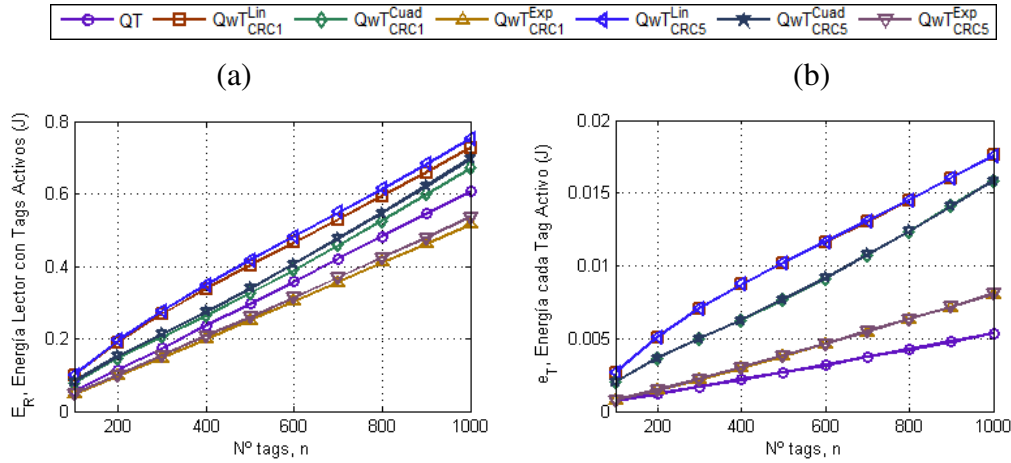


Figura 5.7: Energía consumida por los protocolos QwT y QT considerando tags activos

mejor consumo presenta el QwT con ventana cuadrática y una longitud de CRC de un bit. Dado que la energía se calcula multiplicando los tiempos por las potencias, los resultados obtenidos son aproximadamente proporcionales dichos tiempos. De hecho, el protocolo propuesto sale beneficiado en este análisis, ya que la energía necesaria para recibir los tags se calcula empleando la potencia de transmisión (P_{Rtx}) y de recepción (P_{Rrx}). Es decir, que se requiere más potencia en un aspecto en el que el protocolo propuesto es muy efectivo, como son los bits transmitidos por los tags. De ahí que el ahorro energético es todavía mayor que en su protocolo origen, QT.

- ◇ Tags activos: como se ha explicado anteriormente, en este caso el lector únicamente se ocupa de transmitir sus bits y de recibir los de los tags, sin necesidad de usar la CW . Dado que los tags son activos, poseen alimentación autónoma o batería. Así, se considera el consumo energético del lector y el de los tags por separado. Usando la expresión de la eq. 5.7 se calcula la energía consumida por el lector de un sistema RFID con tags activos y con la eq. 5.8, la energía promedio consumida por cada tag activo de dicho sistema. En la Figura 5.7 se exponen los resultados de la energía calculada empleando esas ecuaciones.

Acorde a los resultados obtenidos, la aplicación de la ventana no resulta ren-

table en caso de disponer de tags activos. Para el cálculo de la energía en un sistema activo tanto lector como tags deben tener en cuenta la potencia empleada en recibir los bits correspondientes, y en el caso de los tags, la potencia empleada en la recepción es mayor que en la transmisión. Este factor no es favorable para la ventana, ya que su utilización provoca el efecto contrario al deseado en este caso. Esto es, aumenta los bits recibidos por los tags, mientras disminuye los transmitidos. Esto se ve claramente reflejado en la gráfica 5.7.b donde el protocolo QT es el que presenta mejores resultados y las ventanas que menor consumo proveen son las exponenciales, dado que son las que menor número de bits de lector demandan. Además, se observa que no existe apenas diferencia en los consumos de las ventanas idénticas con diferente longitud de CRC. De nuevo, esto es debido a la gran influencia que tiene la recepción de bits en el consumo de los tags activos. Dado que cuando se emplean este tipo de tags, un factor determinante es un bajo consumo que garantice la vida más larga posible para estos dispositivos, la ventana no resulta rentable en este tipo de sistemas. Por otro lado, la ventana de bits no perjudica solo a los tags activos, sino que el lector también sale perjudicado en la comparativa. Las ventanas exponenciales consiguen superar, otra vez, al protocolo QT por ser las ventanas que menor uso hacen de bits de lector, y junto con la reducción de bits recibidos por los tags consiguen reducir el consumo por debajo del presentado por el CT. Sin embargo, el resto de ventanas elevan el consumo energético por encima del presentado por su protocolo origen.

Una vez analizado el protocolo QwT con las diferentes ventanas y comparándolo con el QT como referencia, se puede afirmar que la ventana cuadrática es la más adecuada para realizar una comparativa más exhaustiva con otros protocolos de la literatura actual. En primer lugar, porque es el protocolo que mejor refleja la estrategia fundamental de la ventana, que es el reducir el número de bits transmitidos por los tags. Y en segundo, porque esta mejora se ve reflejada en la latencia del protocolo, reduciendo de forma considerable el tiempo requerido en la identificación del conjunto de tags.

5. Experimentación y resultados

A continuación, se realiza un análisis del CRC sobre la ventana cuadrática para conocer los efectos que tiene la longitud de éste sobre el protocolo.

5.3.1 Influencia del CRC en la ventana

Como ya se ha comentado en la sección 4.2, el protocolo QwT necesita emplear un CRC para identificar el tipo de slot recibido. Cuando se recibe una respuesta en el lector, éste calcula el CRC de la ventana recibida, y lo compara con el CRC adjunto recibido. Si los CRCs no coinciden, la respuesta es interpretada como colisión, y si por el contrario son iguales, puede tratarse de una respuesta de tipo go-On o satisfactoria.

En la Figura 5.8 se muestran dos gráficas de cómo afecta el CRC al protocolo QwT en el número de bits transmitidos por tag y en latencia (5.8.b).

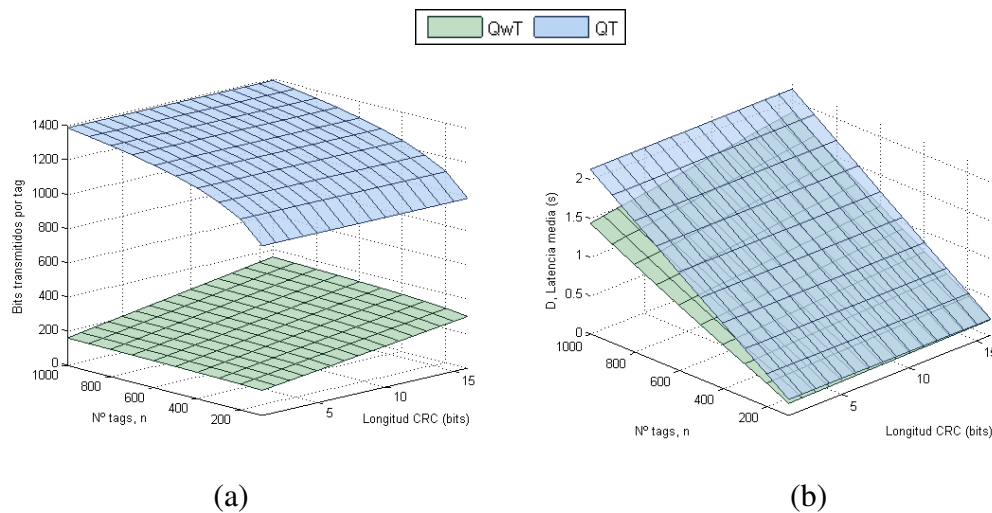


Figura 5.8: Influencia de la variación de la longitud del CRC sobre el protocolo QwT comparado con el QT en: (a) el número de bits transmitidos y (b) latencia

En la primera de las gráficas (5.8.a), el número de bits transmitidos por cada tag del protocolo QwT aumenta conforme aumenta la longitud del CRC. Sin embargo, la ventana reduce este número en tal cantidad que el protocolo QT transmite mayor número en todos los puntos de la gráfica. Del mismo modo, en la gráfica que muestra la latencia (5.8.b) sucede algo similar. La latencia del protocolo QT es mayor que la del protocolo QwT para cualquier valor de longitud del CRC.

En el punto más crítico de los simulados, cuando éste emplea su máximo valor ($CRC = 16 \text{ bits}$), a pesar de que las latencias son muy parecidas ante un grupo reducido de tags ($n = 100$), las diferencias se incrementan cuando crece dicho grupo. Por lo tanto, el QwT y la ventana presentan un gran margen frente al protocolo QT.

El funcionamiento adoptado por la ventana produce una gran cantidad de respuestas de tags de un único bit, con lo cual, utilizar CRCs de gran longitud resulta contraproducente para la propia finalidad del protocolo, a pesar de seguir mostrando resultados prometedores. Son preferibles, por tanto, valores de CRC de entre uno y cinco bits. El problema del CRC de un bit o bit de paridad es que si se corrompen dos bits del mensaje, el resultado del cálculo del CRC va a ser el mismo que el del mensaje correcto (ver Figura 5.9).

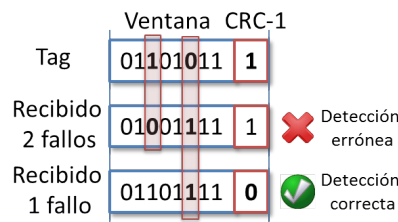


Figura 5.9: Ejemplo de error empleando un CRC de 1 bit, o bit de paridad

El CRC de 5 bits no ofrece resultados tan destacables en términos de bits transmitidos por cada tag como la anterior opción, aunque asegura una mayor fiabilidad de las transmisiones sin resultar perjudicado en exceso. Es más, mantiene unos buenos resultados de latencia y número de bits transmitidos por cada tag.

5.3.2 Comparativa del QwT con ventana cuadrática

En esta sección, se presenta la comparativa del protocolo QwT con algunos de los protocolos más destacados de la literatura presentados en el estado del arte (Capítulo 2). Se han elegido para la comparativa protocolos de tipo Aloha y árbol. El protocolo EPC Class 1 Gen 2 y su algoritmo Q de tipo Aloha (ver Sección 2.2.4.1) cuyos tags transmiten números aleatorios de 16 bits (RN16); los protocolos TS y su mejora FTTP (ver Sección 2.3.1) de tipo árbol dentro de la variante cuyos

5. Experimentación y resultados

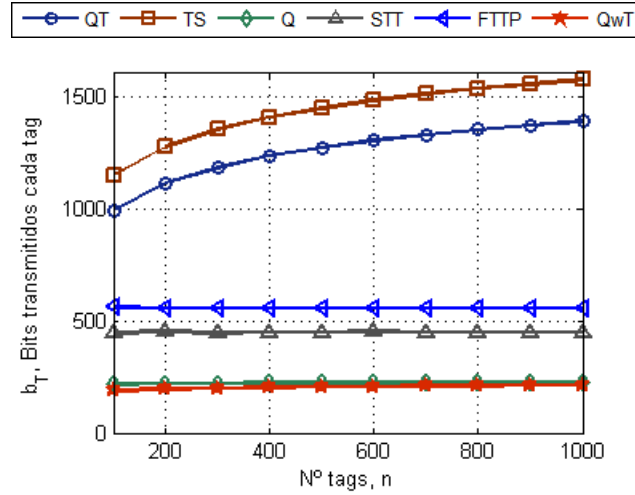


Figura 5.10: Número promedio de bits transmitidos por cada tag en un ciclo completo de interrogación del protocolo QwT y diversos protocolos de la literatura existente

tags emplean memoria y generan un número aleatorio en caso de colisión; y por último, los protocolos memoryless QT y su mejora STT (ver Sección 2.3.2) de tipo árbol también. Tanto FTTP como STT se han presentado como los protocolos más veloces (de menor latencia) de la literatura actual. En el caso del protocolo propuesto en esta tesis, el QwT, se ha optado por emplear la ventana cuadrática con un CRC de cinco bits de longitud acorde a las conclusiones obtenidas en las anteriores secciones.

En la Figura 5.10 se muestran los resultados de la experimentación en el número-

Protocolo	Bits/slot cada tag, B_{Tslot}
QwT	$W + CRC$
QT	$k - L$
TS	k
Q	$T_{ext} + RN16$ ($+T_{ext} + k$ si $t = 1$)
STT	$k - L$
FTTP	k

Tabla 5.6: Tag bits/slot de los protocolos comparados en la Figura 5.10

5.3 Simulaciones y resultados de QwT

ro promedio de bits transmitidos por cada tag. Estos cálculos se han realizado empleando la eq. 5.3 teniendo en cuenta los datos aportados en la Tabla 5.6, donde se indican los bits específicos que transmiten los tags de cada protocolo en un slot del proceso de identificación.

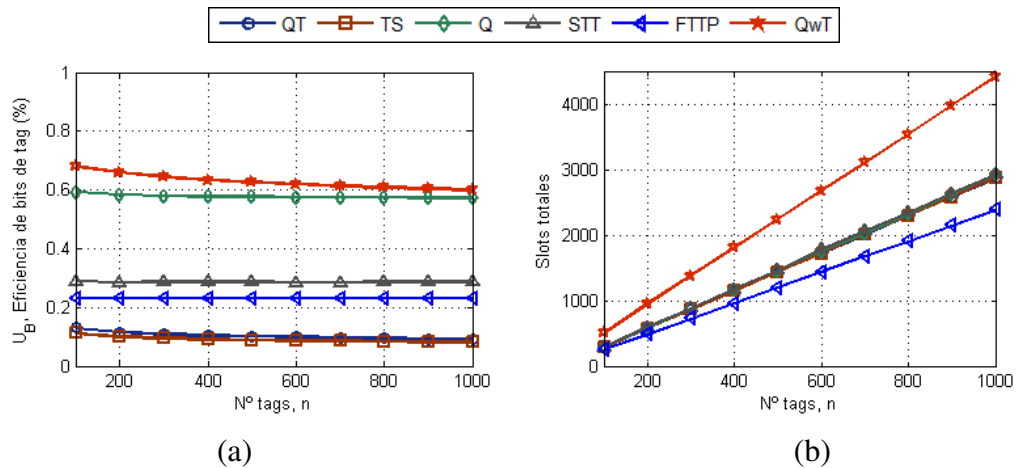


Figura 5.11: Comparativa del protocolo QwT con diversos protocolos de la literatura existente en: (a) eficiencia de bits de los tags y (b) slots totales necesarios para identificar la totalidad del conjunto de tags

Observando los resultados mostrados se confirma que la ventana aplicada sobre el protocolo QT (QwT) presenta el mejor comportamiento. Ofrece el menor número de bits transmitidos por los tags, seguida muy de cerca por el algoritmo Q de tipo Aloha, cuyos tags transmiten mensajes de 16 bits en slots aleatorios (RN16), hasta que uno de ellos es identificado para entonces, transmitir los k bits de su ID. Los tags del protocolo TS presentan los peores resultados, seguidos por su variante memoryless, QT. Los tags de las mejoras de estos protocolos (FTTP y el STT respectivamente) transmiten los mismos bits por cada slot, sin embargo, los resultados mostrados indican una mejora frente a sus predecesores. Esto es debido principalmente a que la finalidad de estos protocolos es la de reducir el número de slots y, por ello, sus tags transmiten un menor número de ocasiones. En la Figura 5.11.b se presenta el número de slots que necesita cada protocolo para identificar el conjunto de tags. Destaca como el protocolo QwT necesita un mayor número de mensajes debido, en gran parte, a los slots de tipo go-On. El protocolo FTTP exhibe su for-

5. Experimentación y resultados

taleza en este aspecto, y de ahí la diferencia con respecto a su predecesor TS en el número de bits transmitidos por los tags.

Por otro lado, se muestran las eficiencias de bits de los tags en la Figura 5.11.a. En esta gráfica se observa más claramente como el protocolo QwT aprovecha mejor los bits que transmiten sus tags que el algoritmo Q, aunque para entornos más densos de tags esta ventaja se reduce. También se confirma la afirmación anterior en la que STT y FTTP realizan menos transmisiones de $k-L$ y k bits respectivamente, y por ello, su eficiencia de bits es superior a la del QT y TS.

Una vez mostrada la principal característica de la ventana, se procede a exponer en la Figura 5.12 los bits que transmiten los lectores de cada uno de los protocolos y el cómputo global de bits transmitidos. En la Tabla 5.7 se especifican los bits que se transmiten en cada slot en cada uno de los protocolos simulados, que han sido empleados para calcular los resultados finales de cada gráfica mediante las eq. 5.1 y 5.2.

Los protocolos cuyos lectores transmiten un menor número de bits son el TS y el FTTP, ya que sus tags disponen de memoria y el lector no necesita indicarles quiénes o qué parte de su ID deben transmitir. Mediante un comando de 2 bits (C), indican el tipo de slot detectado para que los tags actúen en consecuencia. El resto de protocolos permanecen en la zona media de la gráfica, mientras que el protocolo

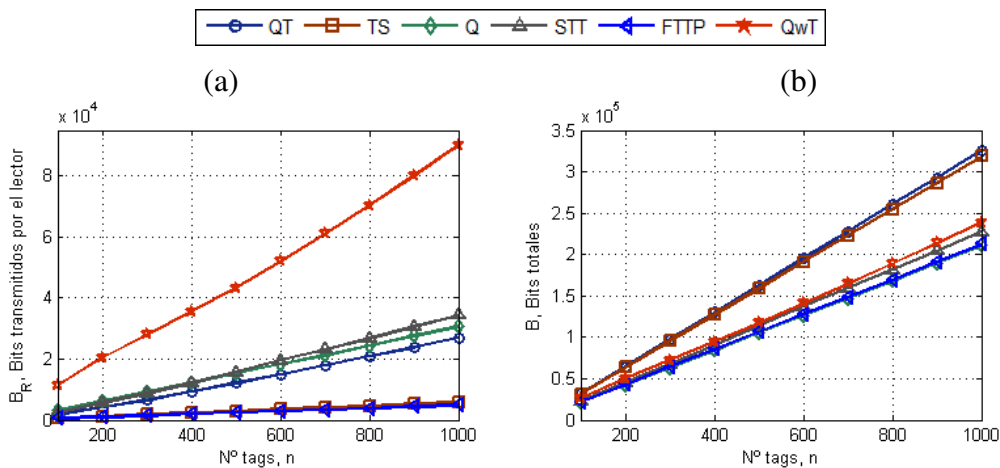


Figura 5.12: Comparativa del protocolo QwT y diversos protocolos de la literatura existente en: (a) los bits transmitidos por su correspondiente lector, y (b) los bits totales transmitidos entre lector y tags

5.3 Simulaciones y resultados de QwT

Protocolo	Bits transmitidos en cada slot		
	Lector, B_{Rslot}		Tags, B_{Tslot}
QwT	$L + \lceil \log_2 W \rceil + 1$	+	$W + CRC$
QT	L	+	$k - L$
TS	C	+	k
Q	$Q_{adj} \circ Q_{rep}$ si $Q_{i-1} = Q_i$	+	$T_{ext} + RN16$ ($+T_{ext} + k$ si $t = 1$)
STT	L	+	$k - L$
FTTP	C	+	k

Tabla 5.7: Bits/slot de los protocolos comparados en la Figura 5.12

propuesto, QwT, se muestra como el que más carga aporta a su lector. La aparición del slot de tipo go-On provoca un aumento considerable de los bits transmitidos por el lector, debido a la necesidad de interrogar a los tags en varias ocasiones hasta obtener su ID al completo. A pesar de esta consecuencia, el número total de bits transmitidos se acerca mucho a los mínimos presentados por los protocolos FTTP y Q. El primero gracias a la clara disminución de slots que consigue y el segundo por el empleo del RN16 en cada respuesta de tag y su carga moderada en el lector.

En cuanto a la latencia que presentan los protocolos, en la Figura 5.13 se exhiben las gráficas de las latencias de cada uno de ellos. Los *data rates* considerados son los mostrados en la Tabla 5.2. Además, para el cálculo de las latencias se ha empleado la eq. 5.5 teniendo en cuenta los bits por slot que transmite cada protocolo, indicado en la Tabla 5.8.

Los resultados presentados muestran cómo la aplicación de la ventana de bits en el protocolo QwT se refleja en la reducción de la latencia. STT y FTTP superan a sus respectivos protocolos origen QT y TS, y el protocolo Q consigue unos resultados similares a los del FTTP. Los resultados presentados por el QwT superan al resto de protocolos de la figura gracias a que la reducción que produce en los bits transmitidos por cada tag compensa su incremento en los bits enviados por el lector. Observando la Figura 5.12, se comprueba que el protocolo Q muestra uno de los mejores resultados en bits totales transmitidos y, en la Figura 5.10, en número

5. Experimentación y resultados

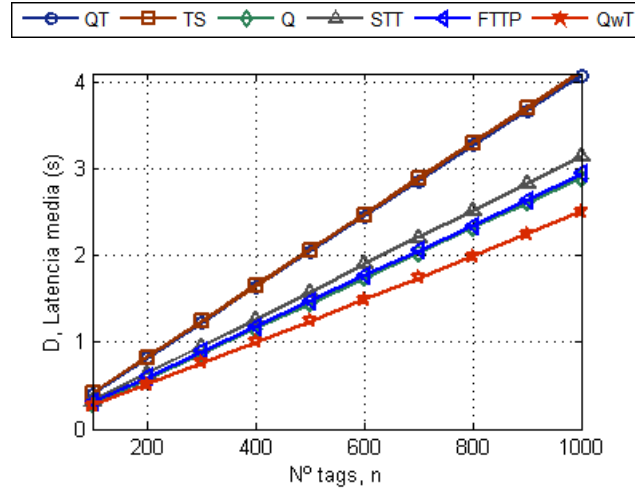


Figura 5.13: Comparativa de las latencias medias del protocolo QwT y diversos protocolos de la literatura existente

Protocolo	Latencias de cada slot		
	Lector		Tags
QwT	$\frac{L + \lceil \log_2 W \rceil + 1}{V_R}$	+	$\frac{W + CRC}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$
QT	$\frac{L}{V_R}$	+	$\frac{k-L}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$
TS	$\frac{C}{V_R}$	+	$\frac{k}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$
Q	$\frac{Q_{adj}}{V_R}$ o $\frac{Q_{rep}}{V_R}$ si $Q_{i-1} = Q_i$	+	$\frac{T_{ext} + RN16}{V_T}$ (+ $\frac{T_{ext} + k}{V_T}$ si $t = 1$)
STT	$\frac{L}{V_R}$	+	$\frac{k-L}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$
FTTP	$\frac{C}{V_R}$	+	$\frac{k}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$

Tabla 5.8: Latencia de los protocolos comparados en la Figura 5.13

de bits promedio de cada tag. A pesar de encontrarse muy cerca de los resultados del QwT, la diferencia existente en el número de slots libres con respecto al protocolo diseñado decanta la balanza de la latencia a favor del QwT. La naturaleza probabilística del algoritmo Q, de tipo Aloha, genera una gran cantidad de slots libres como se puede observar en la Figura 5.14.

Debido a que la ventana del protocolo QwT consigue reducir los slots libres, permite reducir también la latencia del protocolo. En este caso se ha considerado un espacio equivalente a la transmisión de $k/4$ bits, para que se considere un slot

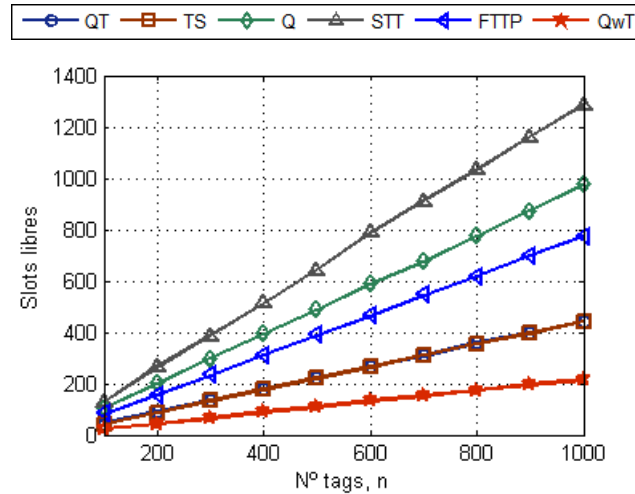


Figura 5.14: Número de slots libres del protocolo QwT comparado con diversos protocolos de la literatura existente

como libre. Si este valor es mayor, las diferencias de latencia entre el QwT y el resto de protocolos serán todavía mayores, dotando al QwT de mayor ventaja.

Este número de slots vacíos también afecta al consumo energético. Ya se ha visto en la comparativa anterior (Figura 5.7), que el protocolo QwT no presenta buenos resultados empleando tags activos y, por tanto, la comparativa se realiza únicamente con tags pasivos. En la Figura 5.15 se exponen los resultados de la simulación energética de los diferentes protocolos comparados.

Los resultados energéticos, calculados empleando la eq. 5.6, ofrecen un comportamiento destacable del protocolo QwT. La reducción de bits transmitidos por los tags y de slots libres favorecen que el lector no transmita su CW durante tanto tiempo, y por ello el consumo energético del lector disminuye frente a su principal perseguidor, el protocolo FTTP.

5.3.3 Conclusiones

En esta sección se han presentado los resultados de las simulaciones del protocolo QwT, donde se ha aplicado la ventana de bits sobre el protocolo QT con el fin de validar la hipótesis planteada. Se ha realizado una primera comparativa para observar las modificaciones que la ventana realiza sobre el protocolo origen. Aquí se

5. Experimentación y resultados

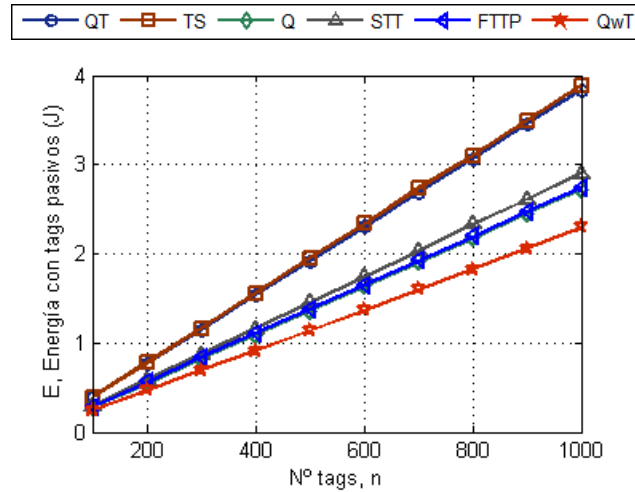


Figura 5.15: Energía consumida por el protocolo QwT comparado con diversos protocolos de la literatura existente empleando tags pasivos

constata la reducción del número de bits transmitidos por los tags sobre el protocolo QT, y el incremento de bits transmitidos por el lector. Aunque finalmente, gracias a la disminución que produce la ventana en slots libres y colisiones, la latencia y el consumo energético en tags pasivos se reducen en el protocolo propuesto, especialmente en el caso de la ventana cuadrática. Por ello, se escoge esta ventana para la posterior comparativa con algunos de los protocolos de anti-colisión más recientes de la literatura. De nuevo, los resultados son igualmente favorables en latencia y consumo energético para protocolo diseñado QwT en sistemas que emplean tags pasivos, superando incluso a protocolos como el FTTP que emplean tags con un hardware de mayor complejidad. Esto confirma la hipótesis planteada para el caso de un único protocolo basado en *queries*, el QT.

En la próxima sección, se muestra la experimentación realizada sobre otro protocolo basado en *queries*, el CT, que forma el protocolo CwT al aplicarle la ventana de bits.

5.4 Simulaciones y resultados de CwT

Una vez presentados los resultados del QwT y comparados con otros protocolos de la literatura existente, se procede a incluir la técnica del bit-tracking en la ecua-

5.4 Simulaciones y resultados de CwT

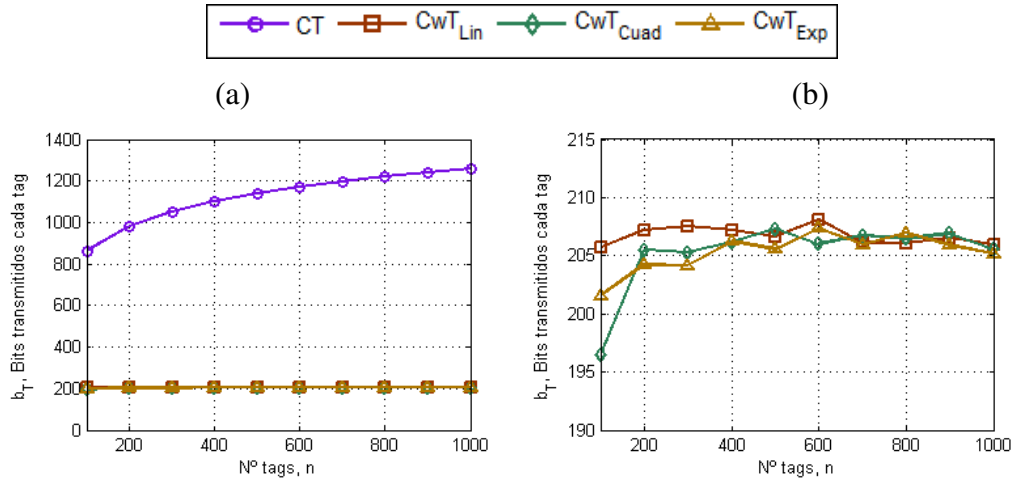


Figura 5.16: Promedio de bits transmitidos por cada tag de los protocolos CT y CwT empleando los tres tipos de ventanas propuestas. En la figura (b) se muestra el rango de bits [190-215] ampliado

ción. Para ello, se presentan los resultados de las simulaciones del protocolo CwT comparándolo con otros protocolos que, igualmente, emplean bit-tracking. Continuando con la estructura de la anterior sección, primeramente se procede a realizar una comparativa entre el protocolo propuesto CwT y los diferentes tipos de ventanas empleadas con su protocolo origen, el CT. Una vez realizado, se escoge la ventana que mejores resultados obtiene y se compara con otros protocolos existentes en la literatura. Por último, se presentan los resultados de la mejora CwT+ para disminuir el número de bits por tag y se compara con el CwT. Para completar este análisis, los resultados obtenidos de las simulaciones del CwT se emplean junto con los ya obtenidos del QwT para analizar la influencia de la variación del *data rate* sobre estos protocolos.

Los datos empleados en las simulaciones son los mostrados en la Tabla 5.2. Sin embargo en este caso, el valor del *data rate* de los tags se incrementa hasta *160 kbps*. Dado que los resultados obtenidos con el CwT muestran mejores latencias que los del QwT, el protocolo sigue mostrando excelentes resultados con tags que emplean velocidades elevadas.

Al aplicar la técnica de bit-tracking, los protocolos de anti-colisión obtienen una gran ventaja en cuanto a que son capaces de trazar la colisión a nivel de bit,

5. Experimentación y resultados

Protocolo	Bits/slot cada tag, B_{Tslot}
CwT	W
CT	$k - L$

Tabla 5.9: Tag bits/slot de los protocolos CwT y CT

y por tanto, aprovechar constructivamente los bits recibidos anteriores a la colisión. Aplicando la ventana sobre el protocolo CT (CwT) que emplea bit-tracking, se consigue un efecto similar al aplicado sobre el protocolo QT. El número de bits transmitidos por los tags se reduce de forma drástica frente a los datos obtenidos por el protocolo sin ventana, CT. La Figura 5.16 muestra el número medio de bits transmitidos por cada tag a lo largo del ciclo de interrogación. La gráfica muestra la comparativa entre el protocolo CT y el CwT con los tres tipos de ventana propuestos: lineal, cuadrático y exponencial. También se presentan los bits por slot que transmiten los tags de cada protocolo en la Tabla 5.9. Para el cálculo del número promedio de bits mostrados en la figura se usa la eq. 5.3.

Acorde a los resultados mostrados en la Figura 5.16, se observa que la ventana ejerce el mismo efecto que al aplicarse sobre el protocolo QT. El considerable descenso del número de bits transmitidos por los tags, fruto del empleo de la ventana en el protocolo CT, se muestra en la Figura 5.16.a. Para observar las pequeñas diferencias existentes entre las tres ventanas comparadas, se muestra en la Figura 5.16.b la región ampliada de los resultados de éstas. No se puede afirmar que ninguna de las ventanas destaque sobre otra, pero la ventana exponencial ofrece resultados ligeramente mejores sobre el resto, identificando pequeños conjuntos de tags ($n < 200$). Esta gráfica demuestra que la ventana aplicada a un protocolo diferente al QT y basado en *queries* ofrece una mejora similar en el número promedio de bits transmitidos por los tags a la conseguida en éste.

A continuación, se analiza el efecto que produce la ventana en el número total de bits transmitidos entre lector y tags y los específicamente enviados por el lector. Los resultados de estas dos medidas se muestran en la Figura 5.17, donde se observa el incremento en el número de bits transmitidos por el lector debido a la aparición del slot de tipo go-On. En la Tabla 5.10 se especifican los bits transmitidos en cada slot, y empleando las eq. 5.1 y 5.2 se pueden calcular los resultados

5.4 Simulaciones y resultados de CwT

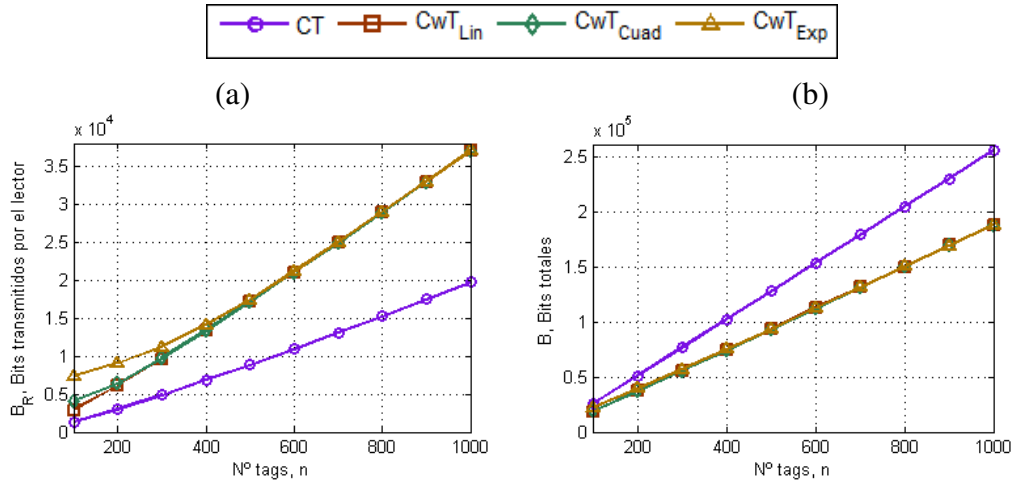


Figura 5.17: Comparativa de CwT con CT en: (a) bits transmitidos por el lector, (b) bits totales transmitidos entre lector y tags

Protocolo	Bits transmitidos en cada slot		
	Lector, B_{Rslot}		Tags, B_{Tslot}
CwT	$L + \lfloor \log_2 W \rfloor + 1$	+	W
CT	L	+	$k - L$

Tabla 5.10: Bits/slot de los protocolos CwT y CT

mostrados en las gráficas.

Los resultados mostrados son congruentes con los exhibidos anteriormente en la Figura 5.16. Las ventanas empleadas sobre el protocolo CT provocan el aumento del número de bits de lector (Figura 5.17.a) para conseguir reducir los bits que transmiten los tags, incrementando el número de bits transmitidos por el lector del protocolo CT. Por otro lado, comparando las ventanas entre sí, se había mostrado cómo la ventana exponencial aplicada sobre el protocolo CT empleaba un promedio de bits por tag menor que el resto de ventanas. Para conseguir una mayor reducción de este número de bits, el lector se ve forzado a aumentar el número de go-On slots en los que se solicitaban ventanas menores, y por ello el número de bits transmitidos por los tags es menor.

En cuanto al cómputo total de bits expuesto en la Figura 5.17.b, la aplicación de la ventana resulta muy positiva sobre el protocolo CT. El aumento producido en

5. Experimentación y resultados

el número de bits del lector es una consecuencia que no se ve reflejada en la transmisión total, ya que la gran reducción que se produce en los bits transmitidos por los tags compensa el aumento mencionado. En este caso, las tres ventanas muestran resultados similares, reduciendo todas ellas el número total de bits transmitidos con respecto al protocolo CT.

Para obtener la latencia media de los protocolos se han empleado las expresiones mostradas en la Tabla 5.11 acerca del tiempo que tarda cada slot, y se ha realizado el cómputo global acorde a la eq. 5.5. Los resultados se exhiben en la Figura 5.18.

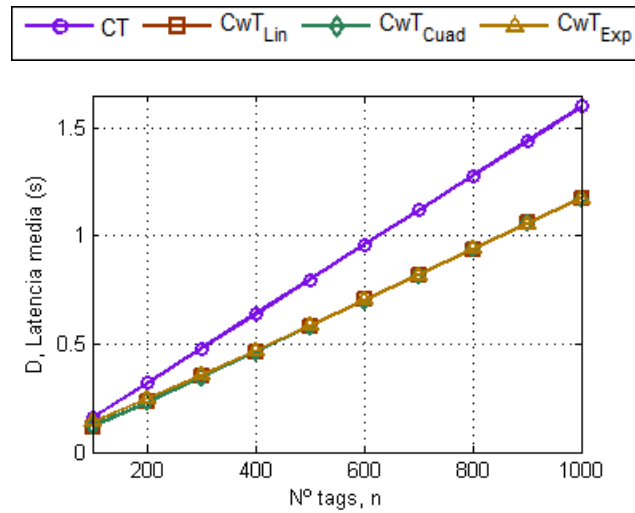


Figura 5.18: Latencias de los protocolos CwT y CT

Protocolo	Latencias de cada slot		
	Lector		Tags
CwT	$\frac{L + \lceil \log_2 W \rceil + 1}{V_R}$	+	$\frac{w}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$
CT	$\frac{L}{V_R}$	+	$\frac{k-L}{V_T}, t > 0$ o $\frac{b_{libre}}{V_T}, t = 0$

Tabla 5.11: Latencia/slot de los protocolos CwT y CT

En este caso, los resultados mostrados por la latencia son muy similares a los mostrados en el número de bits totales. Las diferencias que muestra la latencia frente a la medida del número de bits totales están principalmente en las velocidades

5.4 Simulaciones y resultados de CwT

de lector y tags, que en este caso son idénticas (160kbps), y en el número de slots libres producidos. Dado que la tecnología bit-tracking consigue extinguir este tipo de slots ($b_{libre} = 0$), no añade diferencias a la latencia con respecto a los bits totales, salvo la variación proporcional de los valores al dividirse entre las velocidades. Por lo tanto, la aplicación de la ventana resulta en una mejora sustancial de la latencia del protocolo al que se le ha aplicado, QwT, sobre el protocolo original, QT.

Seguidamente, se pasa a presentar los resultados energéticos del protocolo propuesto en caso de considerarse tags pasivos o activos.

- ◇ Tags pasivos: en este caso se deben tener en cuenta únicamente las potencias del lector, tanto de transmisión como de recepción. Así, la potencia de transmisión se debe considerar en ambos casos, ya que el lector necesita transmitir la onda CW para alimentar a los tags y que éstos emitan sus respuestas mientras tanto, consumiendo en transmisión y recepción simultáneamente. En la Figura 5.19 se muestra la energía consumida por el lector identificando tags pasivos. Estos resultados han sido calculados empleando la eq. 5.9 para el cálculo energético de un sistema RFID con tags pasivos.

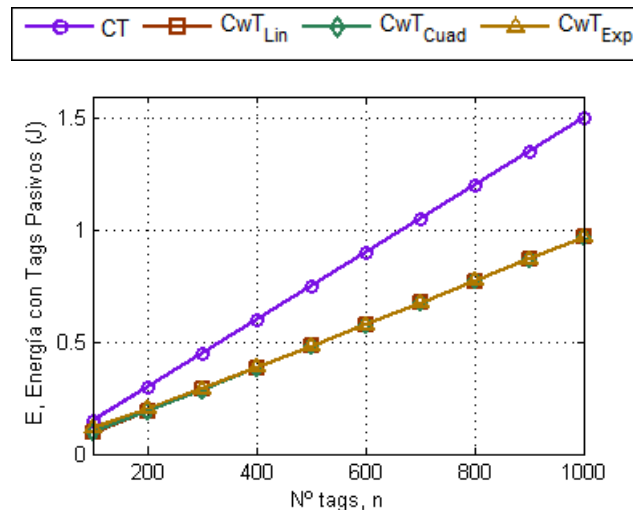


Figura 5.19: Energía consumida por los protocolos CwT y CT considerando tags pasivos

Los resultados obtenidos muestran que el lector del protocolo propuesto proporciona un consumo menor que el CT. Gracias a la reducción que se produce

5. Experimentación y resultados

en la latencia por el empleo de la ventana, el consumo del protocolo se reduce en proporciones similares. Gracias a la importante reducción de los bits transmitidos por los tags, el tiempo que el lector consume por transmisión y recepción simultáneamente se reduce, y por ello el consumo empleando tags pasivos presenta buenos resultados.

- ◇ Tags activos: para el caso en el que se emplean tags activos, el lector no necesita transmitir la señal CW y, por tanto, no se produce la situación de consumo simultáneo de transmisión y recepción. Los resultados presentados en la Figura 5.20 han sido calculados empleando las ecuaciones eq. 5.7 para el cálculo de la energía consumida por un lector con tags activos, y eq. 5.8 para el cálculo del promedio energético consumido por un tag activo.

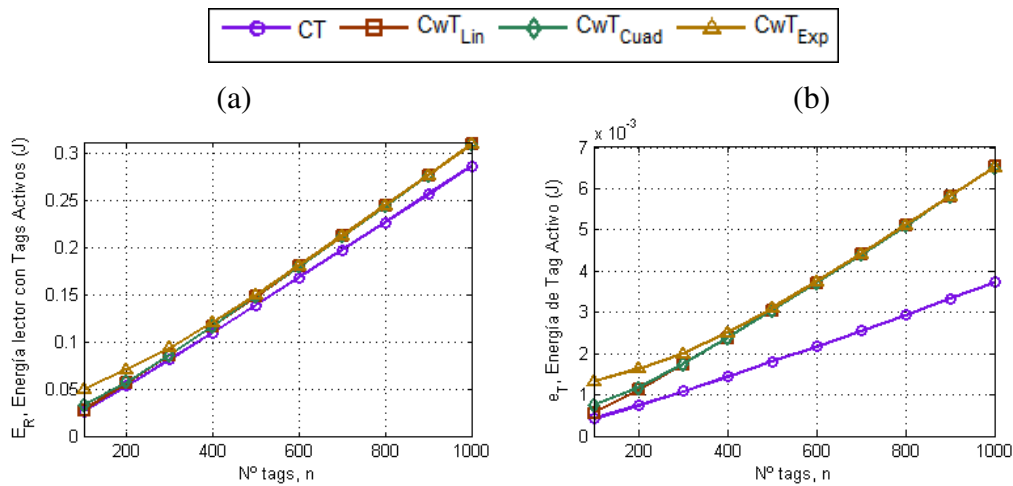


Figura 5.20: Energía consumida por los protocolos CwT y CT considerando tags activos

A pesar de que los resultados obtenidos son mejores que los presentados en la Figura 5.7 para el protocolo QwT, la estrategia de la ventana no resulta adecuada para la utilización de tags activos, ya que aumenta el consumo tanto del lector, como de los tags. En la Figura 5.20.a, los resultados muestran a las ventanas lineal y cuadrática con consumos similares al protocolo CT ante conjuntos de tags reducidos ($n < 300$). Sin embargo, a partir de ese momento, la ventana exponencial que mostraba un consumo mayor en ese rango, se solapa con los consumos de las otras dos ventanas y sobrepasan el

del protocolo CT, aumentándose la diferencia lentamente conforme aumenta n . En el caso de los tags activos, los resultados obtenidos muestran que la ventana produce un consumo mayor para todo el rango de n analizado. Esto se debe en gran medida a que la potencia necesaria de los tags para leer la información recibida del lector es mayor que la de transmisión, y dado que la ventana provoca un aumento de los bits transmitidos por el lector, la disminución de bits a transmitir por éstos no es suficiente para llegar a reducir el consumo. Además, el hecho de que la ventana obligue al tag a enviar su ID en tramos de W bits provoca que deba recibir más mensajes que el protocolo CT y, por tanto, más bits del lector, aumentando su consumo. Dado que la ventana exponencial era la que más bits de lector empleaba, es también la que mayor consumo por tag activo muestra, especialmente en entornos reducidos de tags.

Los resultados ofrecidos muestran una mejora considerable del protocolo CT al incluir la ventana, formando el CwT, en los bits transmitidos por los tags. Esto se refleja en la baja latencia necesaria del protocolo para identificar el conjunto de tags y el bajo consumo energético del lector en caso de tags pasivos. Por contra, el número de bits transmitidos por el lector aumenta, aunque se ve compensado por la reducción de los bits de cada tag, no siendo perjudicial para la latencia. Se observa que las tres ventanas estudiadas muestran resultados muy similares. Esto es debido en gran parte a que las sintonizaciones de las ventanas se han realizado empleando pendientes de función tan elevadas que su comportamiento se asemeja al de un elemento binario, es decir, que la ventana emplea mayormente dos valores posibles. El cambio de tamaño de la ventana se produce en un intervalo muy pequeño de valores de L , actuando principalmente los valores: 1 o $k - L$. Esto resulta muy beneficioso para los protocolos que emplean bit-tracking, ya que el lector puede aprovechar los bits recibidos anteriores al bit colisionado. De esta forma, los bits extra transmitidos por los tags no resultan tan perjudiciales como en el caso del protocolo QwT.

Dado que las ventanas presentan resultados muy similares en todas las medidas simuladas, se opta por escoger la ventana lineal. Ésta presenta unos resultados

5. Experimentación y resultados

uniformes en bits transmitidos por los tags y en latencia, a diferencia de la ventana exponencial. Como factor determinante, la implementación de la función lineal es la más sencilla de las ventanas propuestas. La función lineal consiste en una multiplicación, frente a operaciones exponenciales, en el caso de la cuadrática y la exponencial. A continuación se realiza un análisis de la influencia del *data rate* sobre los protocolos QwT y CwT, el primero empleando la ventana cuadrática y el segundo la lineal.

5.4.1 Influencia de la relación del *data rate* entre lector y tags

La velocidad a la que el lector y los tags transmiten su información se denomina *data rate*, y se mide en bits por segundo (bps). La velocidad influye en el tiempo que demandan cada uno de los miembros del sistema RFID para transmitir la información necesaria en cada slot. Por tanto, resulta un parámetro que influye en gran medida en diferentes aspectos del sistema RFID. En este apartado se analiza cómo afecta el *data rate* a la latencia del protocolo.

Para la realización del análisis se presenta el parámetro θ definido en la eq. 5.11. Éste relaciona la velocidad de transmisión de los tags con la del lector. De esta forma se puede observar el comportamiento de los protocolos variando la proporción entre las dos velocidades, es decir, se puede observar cómo influyen las diferentes velocidades de los tags manteniendo fija la velocidad del lector.

$$\theta = \frac{V_T}{V_R} \quad (5.11)$$

El *data rate* se ve influenciado por elementos externos como la codificación empleada o la frecuencia de banda base del protocolo [EPCglobal 08]. Esto provoca que el tag emplee velocidades, generalmente, inferiores a la empleada por el lector. De esta forma, la cantidad de información que envían los tags puede resultar determinante, dado que este hecho puede aumentar considerablemente la latencia. Por otro lado, los tags, tanto activos como pasivos, tratan de evitar un consumo muy elevado, y una de las posibles alternativas para conseguirlo es la disminución de la velocidad. A continuación se muestran dos gráficas en la Figura 5.21, en la

5.4 Simulaciones y resultados de CwT

que se comparan los protocolos propuestos QwT y CwT, con sus respectivos protocolos origen QT y CT, variando, además del conjunto de tags existente en la zona de interrogación n , la relación entre las velocidades de lector y tags, θ .

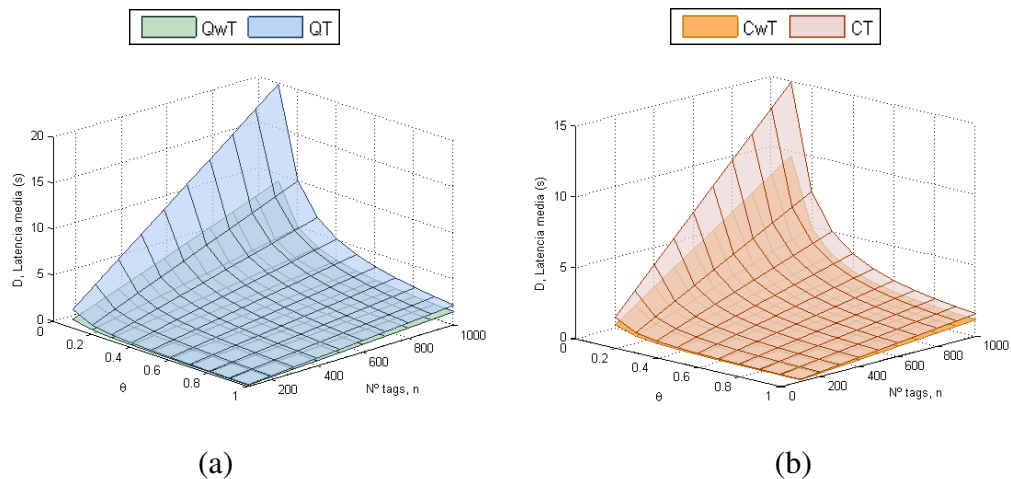


Figura 5.21: Latencias medias de los protocolos variando el conjunto de tags existente en la zona de interrogación y el ratio entre la velocidad del lector y los tags: (a) QwT comparado con su predecesor QT, (b) CwT comparado con CT

En el primer caso, Figura 5.21.a, el protocolo QwT presenta valores de latencia inferiores al QT en todo el rango de la comparativa. Anteriormente se había realizado esta misma comparativa con el parámetro $\theta = 0,5$ en la Figura 5.5, y estos resultados se completan en la gráfica presentada aquí. La ventana consigue disminuir la latencia del protocolo propuesto en todo el rango de la comparativa, y además, los resultados mejoran el comportamiento del QT en mayor medida cuanto menor es el valor de θ . Es decir, que a bajas velocidades de tag, el QwT necesita menos tiempo para completar la identificación en gran medida, gracias a la reducción del número de bits transmitidos por los tags lo que reduce la penalización en latencia causada por la baja velocidad de transmisión de éstos.

Para el caso de la comparativa entre el CwT y el CT (Figura 5.21.b), se da una situación similar. Gracias a la utilización de la técnica de bit-tracking, las latencias disminuyen sustancialmente en comparación a los protocolos QwT y QT. El comportamiento de los protocolos comparados muestra el mismo resultado, la ventana se beneficia de la disminución de velocidad de los tags, y el aumento que se produ-

5. Experimentación y resultados

ce en la latencia es menor que el sufrido por el CT. Además este ahorro en latencia se acentúa cuanto menor es la velocidad de transmisión de los tags.

Como se ha observado en las gráficas mostradas anteriormente, los protocolos propuestos superan ampliamente a sus predecesores en términos de latencia. Por ello, a continuación se presenta la misma comparativa frente a los protocolos FTTP, y OQTT respectivamente. El FTTP es el protocolo con el que se han realizado las comparaciones del QwT en la Sección 5.3.2, y el OQTT emplea la técnica de bit-tracking y consigue reducir los slots en gran medida gracias a una estimación de tags previa al proceso de identificación. Éste se empleará también en la sección 5.4.2, donde se realizará una comparación más completa de otras medidas. Ambos protocolos consiguen reducir la latencia de forma considerable presentándose como los protocolos más veloces de la literatura reciente. Además, han demostrado ser unos duros competidores mostrando excelentes resultados de latencia. En la Figura 5.22 se comparan entonces los protocolos mencionados variando el conjunto de tags existente en la zona de interrogación n y la relación entre las velocidades de lector y tags, θ .

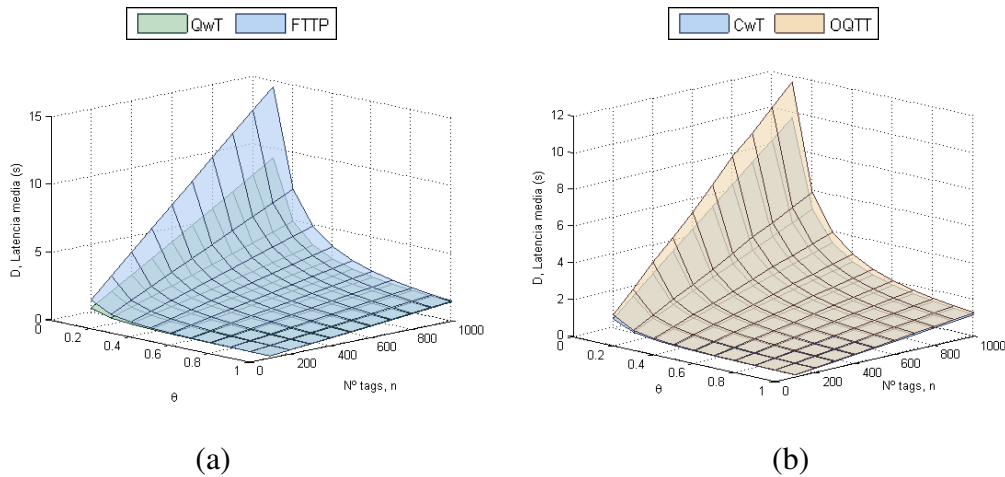


Figura 5.22: Latencias medias de los protocolos: (a) QwT comparado con el FTTP, (b) CwT comparado con el OQTT; variando el conjunto de tags existente en la zona de interrogación y el ratio entre la velocidad del lector y los tags

En este caso, las diferencias entre los protocolos comparados no son tan destacadas como en la comparativa anterior. Sin embargo, ambos protocolos propuestos, QwT y CwT, consiguen ofrecer latencias inferiores a FTTP y OQTT en todo

el rango de la comparativa. La comparativa de la primera de las gráficas (Figura 5.22.a) muestra el protocolo FTTP que ofrece mejores resultados que el QT, y sus resultados se acercan a los del QwT cuando ($\theta = 1$). Sin embargo, las ventajas de la ventana se acentúan cuando θ desciende, aumentando la latencia de protocolo FTTP en mayor proporción que en el QwT. En el caso de la segunda gráfica (Figura 5.22.a), los resultados mejoran los de la gráfica anterior. La latencia del protocolo CwT permanece inferior en todo el rango mostrado a la del protocolo OQTT. Destaca que conforme decrece θ , las diferencias entre ambos protocolos no se acentúan tanto como en el resto de gráficas mostradas. La principal causa de esto se debe a que el protocolo OQTT reduce drásticamente el número de slots y no necesita interrogar a los tags en tantas ocasiones como el CwT, por ello no se ve influenciado tanto como el FTTP, el QT o el CT.

Se ha demostrado que tanto el protocolo QwT como el CwT soportan de forma más favorable las velocidades reducidas de los tags. A continuación se realiza una comparativa del protocolo CwT y algunos de los protocolos más recientes de la literatura existente, tomando el caso más crítico mostrado en la gráfica ($\theta = 1$).

5.4.2 Comparativa del CwT con ventana lineal

Se han presentado los resultados del protocolo CwT comparándolos con su protocolo origen (CT), para determinar los principales beneficios de la aplicación de la ventana sobre dicho protocolo. En esta sección se compara el protocolo CwT empleando la ventana lineal con una serie de protocolos de la literatura reciente. La elección de la ventana lineal se debe principalmente a la sencillez de cálculo frente a la cuadrática y la exponencial, ya que los resultados obtenidos de cada una de las simulaciones eran similares. Por otro lado, se han escogido protocolos de la literatura de tipo Aloha, árbol, y dentro de estos últimos algunos de ellos empleando la técnica de bit-tracking. Así pues, se propone el protocolo EPC Class 1 Gen 2 y su algoritmo Q (ver Sección 2.2.4.1) de tipo Aloha y sus respuestas de tags RN16, los protocolos QT y su mejora STT (ver Sección 2.3.2) de tipo árbol sin bit-tracking y memoryless, y los protocolos CT y OQTT (ver Sección 2.3.3), que también mantienen la característica memoryless y además incluyen la técnica de bit-tracking. Además se añade a la comparativa el otro protocolo propuesto QwT simulado y

5. Experimentación y resultados

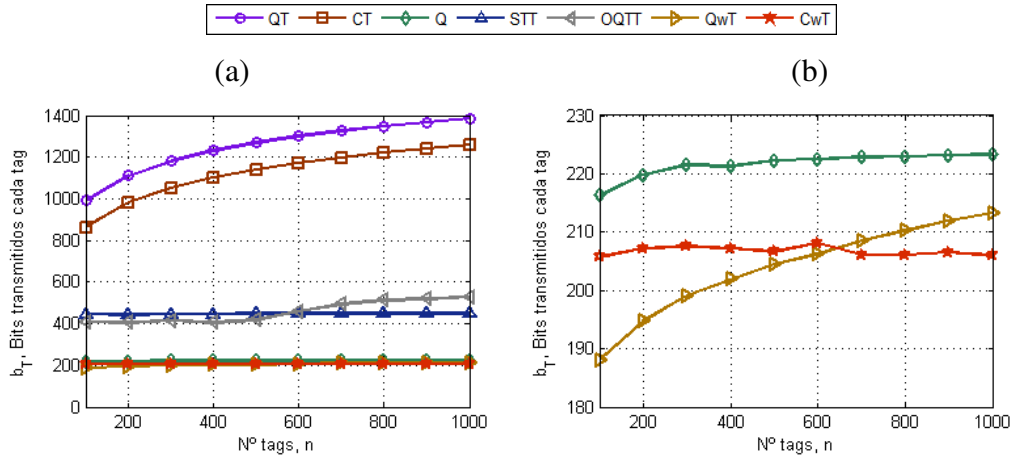


Figura 5.23: Número promedio de bits transmitidos por cada tag en un ciclo completo de interrogación del protocolo CwT y diversos protocolos de la literatura existente. Se muestra en (b) el rango de bits [180-230] ampliado

expuesto en la anterior Sección 5.3. De forma que se puedan determinar los beneficios de la conjunción ventana - bit-tracking frente al de solo ventana, QwT. De entre todos los protocolos propuestos para la comparativa, se pueden distinguir dos tipos de estrategias de identificación: la seguida por la ventana en la que se produce un mayor número de lecturas de los tags, pero estos contestan un menor número de bits, frente a la de los protocolos QT, CT, STT y OQTT, que reducen el número de slots, pero los tags contestan su ID casi al completo cada vez que son interrogados. Ambas son analizadas en profundidad ahora.

Inicialmente se presenta el número promedio de bits transmitidos por los tags en la Figura 5.23. La Tabla 5.12 muestra los bits que transmite cada uno de los tags en cada slot del ciclo de identificación. El valor promedio mostrado se ha calculado con la eq. 5.3.

Los resultados mostrados en la Figura 5.23 confirman la reducción del número de bits transmitidos por los tags que consigue la ventana en los protocolos CwT y QwT. Sin embargo, la combinación de ventana - bit-tracking provoca un ligero aumento del número de bits transmitidos, especialmente en conjuntos de tags reducidos. Este aumento se debe principalmente a que la sintonización del valor de β permite a los tags alcanzar antes el valor de $W = k$, con el que transmiten su ID al completo. Además de esto, el protocolo Q que emplea respuestas de 16 bits (RN16)

Protocolo	Bits/slot cada tag, B_{Tslot}
CwT	W
QwT	$W + CRC$
QT	$k - L$
CT	$k - L$
Q	$T_{ext} + RN16$ ($+T_{ext} + k$ si $t = 1$)
STT	$k - L$
OQTT	k 1 ^{er} slot, $k - L$ resto

Tabla 5.12: Tag bits/slot de los protocolos comparados en la Figura 5.23

para reducir este número ofrece buenos resultados, pero no consigue alcanzar a los protocolos que emplean la ventana, como se puede observar en la ampliación de la Figura 5.23.b. Por otro lado, el resto de protocolos simulados, centrados en la reducción del número de slots, no presentan unos resultados tan destacables. El STT y el OQTT, que reducen los slots con respecto al QT y CT, mejoran sus resultados ya que los tags responden en menor número de ocasiones, pero el gasto de bits es mayor. Estos resultados se confirman con las gráficas de eficiencia de bits transmitidos por los tags y de slots totales empleados en la identificación de la Figura 5.24.a y 5.24.b respectivamente.

Las eficiencias calculadas con la eq. 5.10 muestran que la ventana consigue los mejores resultados, previniendo a los tags de enviar sus bits de ID en exceso. Las dos estrategias empleadas por los protocolos comparados muestran que la empleada por la ventana proporciona mejores eficiencias de bits que la propuesta por la estrategia contraria de los protocolos STT, OQTT, CT y QT. Por otro lado, a pesar de que el CwT se basa en la primera de las estrategias mencionadas, es decir, que se centra en reducir el número de bits de los tags sin importar el número de slots, consigue disminuir esto último gracias al empleo de la técnica de bit-tracking superando a los protocolos STT y Q. Este hecho demuestra que la técnica de bit-tracking es muy efectiva para la reducción de slots, ya que los usuarios de ésta como el CwT, el CT o el OQTT obtienen los mejores resultados, siendo este último el que emplea un menor número. El QwT, por contra, continúa mostrándose como

5. Experimentación y resultados

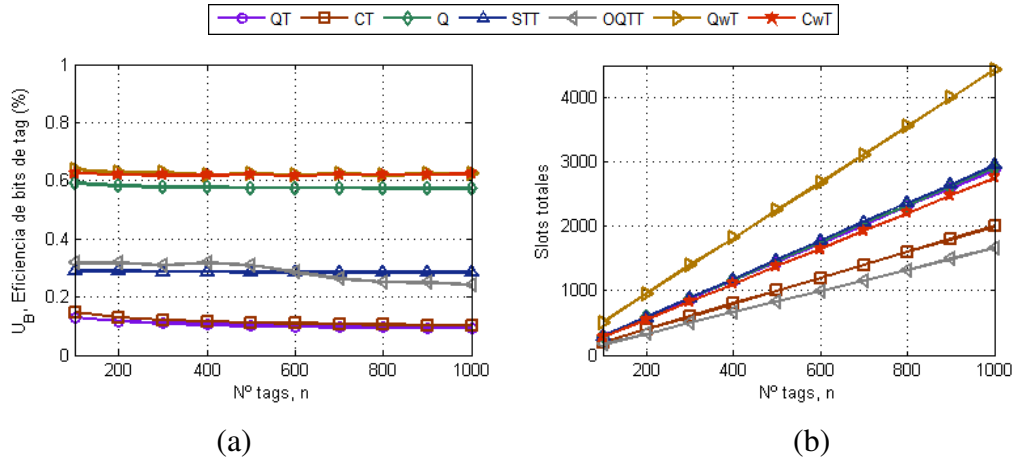


Figura 5.24: Comparativa del protocolo CwT con diversos protocolos de la literatura existente en: (a) eficiencia de bits de los tags y (b) slots totales necesarios para identificar la totalidad del conjunto de tags

Protocolo	Bits transmitidos en cada slot	
	Lector, B_{Rslot}	Tags, B_{Tslot}
CwT	$L + \lfloor \log_2 W \rfloor + 1$	$+$ W
QwT	$L + \lfloor \log_2 W \rfloor + 1$	$+$ $W + CRC$
QT	L	$+$ $k - L$
CT	L	$+$ $k - L$
Q	Q_{adj} o Q_{rep} si $Q_{i-1} = Q_i$	$+$ $T_{ext} + RN16$ $(+T_{ext} + k$ si $t = 1)$
STT	L	$+$ $k - L$
OQTT	l 1 ^{er} slot, L resto	$+$ k 1 ^{er} slot, $k - L$ resto

Tabla 5.13: Bits/slot de los protocolos comparados en la Figura 5.25

el protocolo que más ciclos necesita para identificar los tags.

A continuación se procede a analizar el comportamiento de los protocolos comparados en el número de bits transmitidos por sus respectivos lectores y el computo global de bits transmitidos en la Figura 5.25. Los bits transmitidos en cada slot se indican en la Tabla 5.13, y los cálculos totales se han realizado empleando las eq. 5.1 y 5.2.

5.4 Simulaciones y resultados de CwT

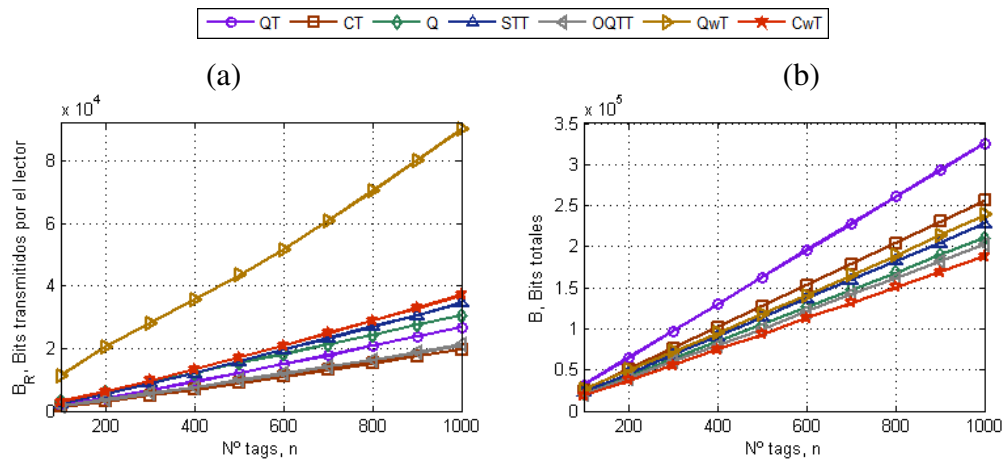


Figura 5.25: Comparativa del protocolo CwT y diversos protocolos de la literatura existente en: (a) los bits transmitidos por su correspondiente lector, y (b) los bits totales transmitidos entre lector y tags

Los resultados obtenidos en la Figura 5.25.a de los bits transmitidos por el lector permiten observar cómo el protocolo CwT, que utiliza la combinación ventana - bit-tracking, reduce considerablemente este número de bits frente al QwT, que solo emplea la ventana. Esto se debe a dos razones principalmente:

- ◇ Por un lado, la tecnología bit-tracking produce un importante descenso del número de slots necesarios para identificar el conjunto de tags como se ha presentado en la Figura 5.24.b. Esto implica una disminución de los mensajes que transmite el lector, y por tanto del número de bits que éste transmite.
- ◇ Por otro lado, la sintonización del parámetro β escogida fomenta que los tags incrementen el tamaño de la ventana drásticamente de un slot a otro. Esto beneficia la identificación de los tags cuando no se producen colisiones, pero por otro lado, puede aumentar el número de bits desechados. La técnica de bit-tracking ayuda al protocolo a que, en caso de colisión, no se desechen todos los bits transmitidos, pudiendo aprovechar los anteriores a la colisión. Por ello, el resultado final es positivo y mejora el del QwT.

Los protocolos CT y OQTT son los que menor número de bits de lector necesitan, gracias a la utilización de bit-tracking. El protocolo OQTT reduce exhaustivamente el número de mensajes empleados por la utilización de una estimación

5. Experimentación y resultados

previa de los tags existentes. Sin embargo, esa técnica le penaliza aumentando el número de bits que transmite el lector de inicio (l bits) y la respuesta de los tags (k bits). De esta forma, no presenta unos resultados excesivamente mejores que los de su predecesor, el CT, en los bits transmitidos por el lector. En el cálculo global de bits (Figura 5.25.b), es el CwT el que ofrece los mejores resultados, seguido del protocolo OQTT y el Q. Las dos estrategias diferenciadas anteriormente ofrecen resultados similares, aunque la empleada por la ventana atisba una mejoría frente a la de reducción de slots. El QwT, por otro lado, consigue superar al CT que utiliza bit-tracking pero no al OQTT que, gracias a la estimación, reduce el número de mensajes y, por tanto, el total de bits transmitidos.

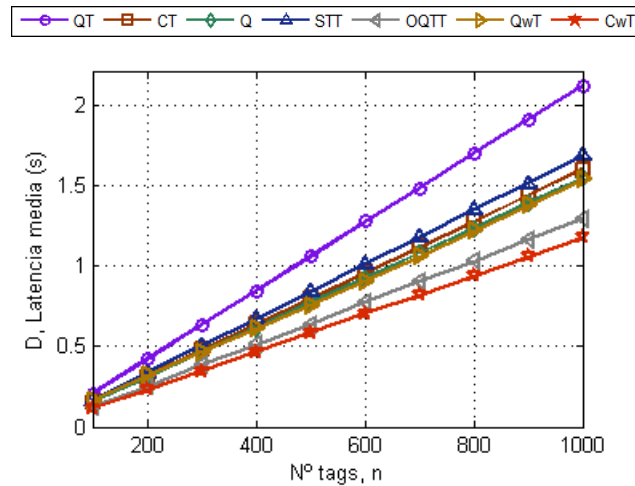


Figura 5.26: Comparativa de las latencias medias del protocolo CwT y diversos protocolos de la literatura existente

Los resultados de la latencia correspondiente a los protocolos comparados se muestran en la Figura 5.26. Dadas las latencias escogidas, se muestran los datos más críticos para la ventana, en la que el *data rate* de lector (V_R) y tags (V_T) es el mismo, es decir, el caso que corresponde a $\theta = 1$ en la comparativa realizada en la Figura 5.22.b. Los cálculos realizados se muestran en la Tabla 5.14, dónde se indica la latencia de cada slot, y la eq. 5.5 para el cálculo de la latencia media total.

Los resultados de latencias obtenidas muestran que el protocolo CwT propuesto obtiene los mejores resultados, seguido de cerca por el protocolo OQTT, aunque esta diferencia se acentúa conforme aumenta el conjunto de tags. La ventana del

5.4 Simulaciones y resultados de CwT

Protocolo	Latencias de cada slot	
	Lector	Tags
CwT	$\frac{L + \lfloor \log_2 W \rfloor + 1}{V_R}$	$+$ $\frac{W}{V_T}$ o $\frac{b_{libre}}{V_T}$ si $t = 0$
QwT	$\frac{L + \lfloor \log_2 W \rfloor + 1}{V_R}$	$+$ $\frac{W + CRC}{V_T}$, $t > 0$ o $\frac{b_{libre}}{V_T}$, $t = 0$
QT	$\frac{L}{V_R}$	$+$ $\frac{k-L}{V_T}$, $t > 0$ o $\frac{b_{libre}}{V_T}$, $t = 0$
CT	$\frac{L}{V_R}$	$+$ $\frac{k-L}{V_T}$, $t > 0$ o $\frac{b_{libre}}{V_T}$, $t = 0$
Q	$\frac{Q_{adj}}{V_R}$ o $\frac{Q_{rep}}{V_R}$ si $Q_{i-1} = Q_i$	$+$ $\frac{T_{ext} + RN16}{V_T}$ ($+$ $\frac{T_{ext} + k}{V_T}$ si $t = 1$)
STT	$\frac{L}{V_R}$	$+$ $\frac{k-L}{V_T}$, $t > 0$ o $\frac{b_{libre}}{V_T}$, $t = 0$
OQTT	$\frac{L}{V_R}$ 1 ^{er} slot, $\frac{L}{V_R}$ resto	$+$ $\frac{k}{V_T}$ 1 ^{er} slot, $\frac{k-L}{V_T}$ o $\frac{b_{libre}}{V_T}$ si $t = 0$ resto

Tabla 5.14: Latencia de los protocolos comparados en la Figura 5.26

QwT también ofrece resultados ligeramente mejores que el protocolo CT, es decir, que la ventana proporciona una ligera mejoría sobre el bit-tracking, y ambas combinadas en el CwT, obtienen los mejores resultados.

Los protocolos que emplean bit-tracking consiguen eliminar por completo los slots libres, como se puede observar en la Figura 5.27. En cambio, el protocolo OQTT que implementa una técnica de estimación de tags junto con bit-tracking provoca la reaparición de este tipo de slot, lo que le perjudica también en el cálculo de latencia. Aún así, las diferencias existentes entre CwT y OQTT no se acentúan demasiado con respecto a las del total de bits consumidos, ya que el número de slots libres tampoco es muy elevado.

El CT y el CwT, por otro lado, muestran cómo el bit-tracking elimina por completo este tipo de slot. Por ello, los cálculos de sus latencias son proporcionales a los del total de bits transmitidos. Cabe mencionar que el protocolo STT trata de generar *queries* de una forma más agresiva, lo que aumenta de forma considerable el número de slots libres penalizando su latencia también. Por último, el protocolo Q de origen probabilístico también muestra un elevado número de slots libres, provocados por el desajuste existente entre el frame calculado y el número de tags que responden a los comandos del lector.

Teniendo en cuenta las latencias y los slots libres, se exhiben los resultados de la energía consumida por el lector empleando tags pasivos en la Figura 5.28.

5. Experimentación y resultados

Éstos se han calculado empleando la eq. 5.6 que tiene en cuenta las potencias de transmisión y recepción durante las respuestas de los tags ya que el lector debe excitarlos transmitiendo la señal CW .

Los resultados energéticos resultan muy destacables para el protocolo CwT, que despunta sobre el resto ofreciendo un consumo inferior. Factores como la disminución del número de bits transmitidos por los tags, la extinción de slots, y la reducción de la latencia permiten al protocolo propuesto reducir el consumo del protocolo CT. El QwT, que se ha visto anteriormente cómo ofrecía grandes resultados energéticos, se ve superado por el OQTT, que gracias a la reducción de mensajes que consigue, ahorra energía de forma ligeramente más acusada. Como también se ha visto en la Figura 5.15, el QT es el protocolo que presenta mayor consumo.

5.4.3 Comparativas del protocolo CwT+

Se ha observado en la Figura 5.23.b que el protocolo CwT ha aumentado la cantidad de bits transmitida por cada tag frente al QwT. En términos de latencia, el CwT ofrece mejores prestaciones, pero sin embargo, los tags van a necesitar mayor tiempo y energía para transmitir los bits. Por ello, se presenta la modificación CwT+, con la que se pretende reducir el número de bits que transmiten los tags del

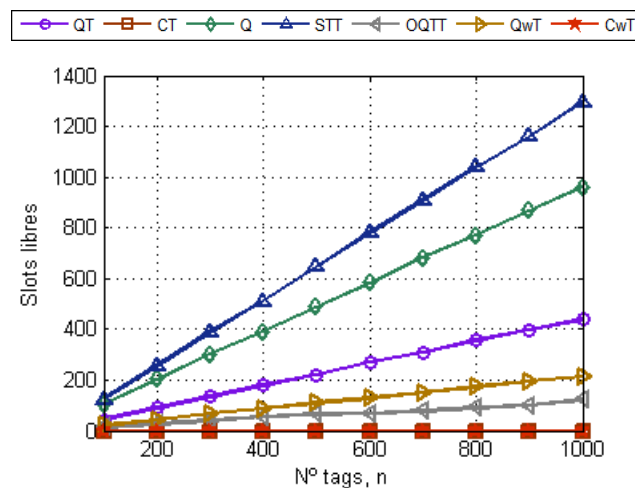


Figura 5.27: Número de slots libres del protocolo CwT comparado con diversos protocolos de la literatura existente

5.4 Simulaciones y resultados de CwT

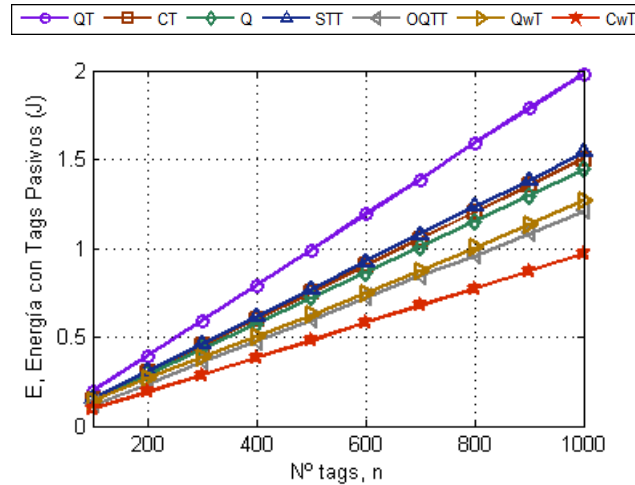


Figura 5.28: Energía consumida por el protocolo CwT comparado con diversos protocolos de la literatura existente empleando tags pasivos

protocolo. Con el objetivo de demostrar sus efectos, se han realizado una serie de simulaciones comparando el protocolo CwT+ con el CwT y el QwT, que usan la ventana, y el CT y OQTT que usan bit-tracking. De esta manera se puede observar qué ventajas e inconvenientes aporta esta nueva modificación.

En la Figura 5.29 se exhiben los resultados de la simulación del número de bits promedio transmitidos por los tags.

Los resultados obtenidos muestran que los protocolos que utilizan la ventana de bits son los que mejor resultado ofrecen en este ámbito. De entre estos protocolos, el CwT+ destaca gracias al valor umbral que utiliza. Este protocolo consigue reducir el número de bits transmitidos por los tags, ya que traza los bits de las dos primeras colisiones ocurridas. Con ello, puede calcular valores de ventana más exactos reduciendo así el número de bits malgastados y, por tanto, consiguiendo una mejora de la eficiencia de bits (Figura 5.30). Solo el QwT es capaz de hacerle frente para conjuntos de tags reducidos ($n < 100$), pero con el aumento de n , el número de bits transmitidos por los tags crece excesivamente y la eficiencia disminuye de forma inversa.

Se había comentado en la Sección 4.6.3, que el protocolo CwT+ conseguía reducir el número de bits transmitidos por los tags, sin perjudicar la latencia del CwT. En la Figura 5.31 se muestran los resultados de la latencia calculada acorde

5. Experimentación y resultados

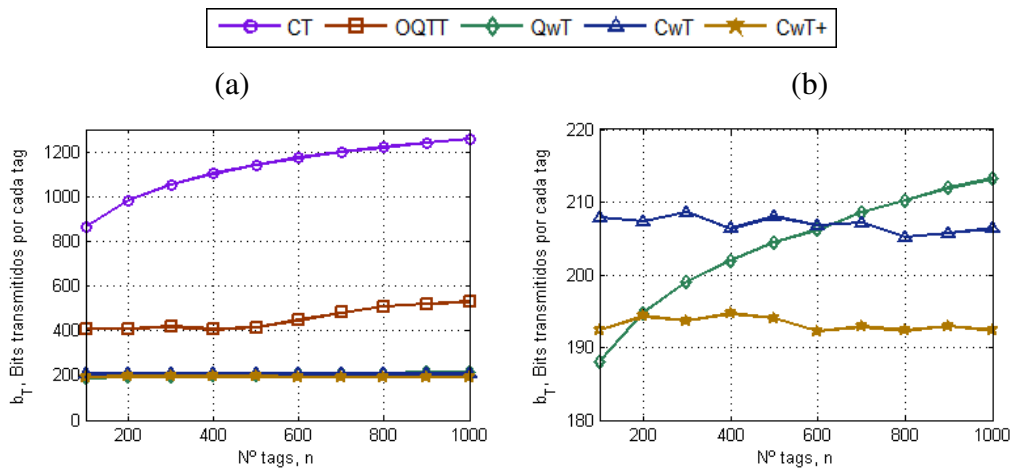


Figura 5.29: Comparativa del protocolo CwT+ con el CwT y diversos protocolos de la literatura existente en número promedio de bits transmitidos por cada tag en un ciclo completo de interrogación. Se muestra en (b) el rango de bits [180-220] ampliado

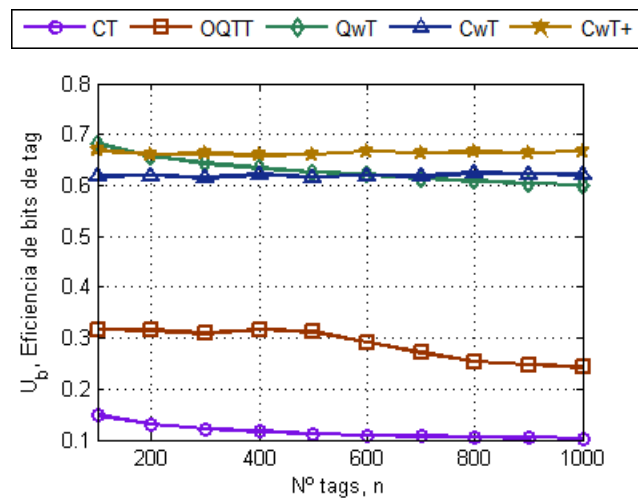


Figura 5.30: Comparativa del protocolo CwT+ con el CwT y diversos protocolos de la literatura existente en eficiencia de bits transmitidos por los tags

a las directrices de la eq. 5.5.

Los resultados obtenidos muestran las latencias de los protocolos CwT y CwT+ muy parejas, ofreciendo menores latencias que su competidor más cercano, el protocolo OQTT. Esto demuestra la principal intención de la modificación propuesta para el protocolo CwT (CwT+), en la que se consigue reducir el número de bits

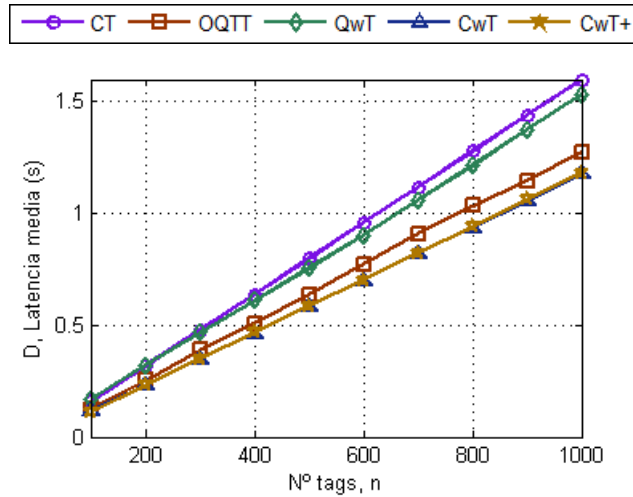


Figura 5.31: Comparativa del protocolo CwT+ con el CwT y diversos protocolos de la literatura existente en latencia

transmitidos por tag, superando incluso los buenos resultados ofrecidos por el protocolo QwT, y manteniendo la prometedora latencia conseguida por el protocolo CwT.

5.4.4 Conclusiones

Se ha diseñado un protocolo aplicando la ventana de bits sobre el protocolo CT. La característica principal de este último es la utilización de la tecnología bit-tracking, con lo que el protocolo propuesto finalmente combina dos técnicas diferentes que, a su vez, resultan perfectamente acopladas. Se han presentado los resultados del protocolo CwT comparado con una serie de protocolos existentes en la literatura, algunos de ellos muy recientes, con el objetivo de validar la hipótesis planteada.

Primeramente, se ha observado que las características flexibles del procedimiento de la ventana le permiten aplicarse tanto sobre el protocolo QT, como el CT, conformando el QwT y el CwT. Es más, la tecnología bit-tracking empleada en el protocolo CT permite a la ventana evitar el uso del CRC, conformando un procedimiento más liviano para su inclusión en el sistema RFID.

Por otro lado, los resultados obtenidos demuestran que la ventana de bits produce una importante reducción del número de bits transmitidos por los tags. A

5. Experimentación y resultados

pesar de que esta reducción ha de compensarse con un aumento del número de bits transmitidos por el lector, este aumento no es lo suficientemente elevado para perjudicar el comportamiento final del protocolo. Los protocolos diseñados, CwT y su inmediata mejora, CwT+, exhiben un reducido empleo de bits en su totalidad que, finalmente, se refleja en una baja latencia y consumo energético utilizando tags pasivos. Esto confirma la hipótesis planteada, en la que se justifica que la aplicación de la ventana sobre un protocolo basado en *queries* consigue reducir la latencia y el consumo energético del sistema RFID.

La conclusión es que sabemos muy poco y sin embargo, es asombroso lo mucho que conocemos.

Bertrand Russell

6

Conclusiones

A lo largo de los capítulos presentados hasta el momento, se han ido describiendo, de forma detallada, las actividades realizadas durante el desarrollo de la tesis doctoral. Una vez situado el entorno de la tecnología RFID, se ha abordado el problema de las colisiones de tag y las consecuencias que esta problemática conlleva. Tras realizar el análisis de la literatura se han encontrado diversas oportunidades de mejora relacionadas con la cantidad de bits transmitidos en los protocolos basados en *queries*. Para dar solución a este problema se ha desarrollado un procedimiento, denominado la ventana de bits, el cual ha sido aplicado a dos protocolos de anti-colisión, conformando finalmente tres nuevos protocolos (QwT, CwT y CwT+). Posteriormente, se ha llevado a cabo la experimentación correspondiente, que ha justificado o validado la hipótesis planteada al inicio de la tesis, y además, ha permitido dar una visión detallada de los beneficios e inconvenientes que presenta la aplicación de la ventana a un protocolo de anti-colisión basado en *queries*.

En este capítulo se ofrece, inicialmente, una percepción global de las cuestiones que se han tratado en este documento, haciendo hincapié en las contribuciones principales. Seguidamente, se expondrán los resultados de la experimentación que han permitido llevar a cabo la validación de la hipótesis, destacando los beneficios

6. Conclusiones

y las limitaciones que presenta el procedimiento diseñado. Para finalizar, se proponen las líneas futuras de trabajo, que proponen una serie de mejoras que podrían realizarse en torno a esta tesis y unas consideraciones finales.

6.1 Visión general del trabajo

La tecnología de identificación por radio frecuencia (RFID), una de las más conocidas técnicas de auto identificación, se emplea para detectar el código almacenado en los diversos tags situados en las cercanías de un dispositivo lector empleando ondas de radio frecuencia. Asociar un tag a cada objeto es aportarle una nueva funcionalidad que puede ser empleada de múltiples formas. Desde un conocimiento más exhaustivo de la situación de cada objeto, es decir, facilidad para conocer la trazabilidad de cada tag asociado al objeto, hasta aplicaciones de localización, control de acceso y múltiples beneficios en la cadena de suministro. Todo esto, sin la necesidad de que exista una línea de visión directa entre lector y tags, lo que facilita aplicaciones muy poco intrusivas con el usuario y permite la adaptabilidad al paradigma de computación ubicua o al *Internet de las cosas*.

RFID se presenta como una tecnología de gran flexibilidad, en la que múltiples tags y lectores tienen cabida. Lo cual, por otro lado, provoca una serie de inconvenientes. El principal, y tratado en esta tesis, es el conocido como *problema de las colisiones de tag*. Dado que el canal de transmisión de esta tecnología es el aire, cuando un lector trata de identificar un conjunto de tags puede darse la posibilidad de que transmita más de un tag simultáneamente y que sus ondas de radio frecuencia se interfieran mutuamente, causando una modificación de los datos recibidos por el lector. Este fenómeno, conocido como colisión, es el causante del incremento del tiempo necesario para identificar los tags presentes, ya que éstos se ven forzados a retransmitir sus mensajes hasta que el lector los reciba correctamente. Este hecho, produce un incremento del consumo energético y una reducción del ancho de banda del sistema.

Dada la envergadura del problema mencionado, numerosas soluciones se han planteado para mitigarlo. Las colisiones son un problema de sobra conocido en la teoría de redes, luego el camino más directo para solucionar el problema era aplicar las técnicas de multi-acceso. Sin embargo, a causa de las restricciones impuestas

por la tecnología RFID, los métodos de multi-acceso no eran aplicables directamente y han tenido que ser adaptados a tales imposiciones, pasando a denominarse **protocolos de anti-colisión**. El diseño de un buen protocolo continúa siendo un tema candente de investigación, en el que se ha avanzado mucho y existe abundante literatura.

Todos los protocolos de anti-colisión presentados en la literatura, pueden englobarse en tres grupos principalmente: los de tipo Aloha, de tipo árbol e híbridos.

- ◇ Los protocolos basados en Aloha son considerados probabilísticos, ya que basan el arbitraje de las respuestas de los tags en la aleatoriedad del momento en que un tag transmite, no garantizando la lectura de la totalidad de los tags existentes en el rango de lectura. Sin embargo, un buen protocolo probabilístico puede maximizar la probabilidad de que todos sean leídos. Las respuestas de los tags se distribuyen en slots, y cuanto más uniforme sea el espaciado entre sus respuestas, mayor es la probabilidad de que sean identificados. La investigación en protocolos basados en Aloha se centra principalmente en la mejora del tipo DFSA y la estimación de tags. Los estimadores son cada vez más exactos, pero también la complejidad matemática y computacional necesarias aumentan, lo cual demanda un hardware más sofisticado y costoso en su implementación.
- ◇ Los protocolos basados en árbol, considerados deterministas por identificar el conjunto completo de tags, requieren tags de menor complejidad a costa de una mayor transmisión de información entre lector y tags. El lector controla los diferentes pasos del protocolo y mediante comandos consistentes en órdenes o prefijos de identificadores va separando el conjunto de tags en pequeños subconjuntos hasta que todos ellos quedan identificados. Pueden diferenciarse dentro de esta opción tags con y sin memoria interna. La principal diferencia es que cuando los tags no requieren de memoria interna, el lector le indica en cada interrogación la información requerida, incrementando la cantidad de datos transmitidos. Sin embargo, los requerimientos hardware de este tipo de tags son menores.

6. Conclusiones

- ◇ Por último, los protocolos híbridos combinan ambas estrategias con el objetivo de atenuar algunos de sus inconvenientes, todo ello a costa de un gran incremento en la complejidad del hardware de lector y tags.

Es interesante que la complejidad del sistema no sea muy elevada, ya que una de las principales características de la tecnología RFID es que los dispositivos tags pasivos no poseen alimentación interna, sino que el lector, a través de la señal de interrogación, provee esta alimentación a los tags. Por lo tanto, una elevada complejidad electrónica de los tags puede provocar un incremento del consumo del lector, que necesitará más potencia de transmisión para excitar sus circuiterías. Y como estas potencias están controladas por las regulaciones de cada país, pueden requerir un aumento del tiempo de excitación, afectando directamente a la latencia del sistema. Además, la complejidad de la electrónica de éstos es un factor clave que influye directamente en el precio de manufacturado de los tags. Un incremento muy elevado en su precio no resulta rentable para la viabilidad del sistema RFID, que busca una similitud con los códigos de barras en este aspecto.

Se ha observado que los protocolos más simples, que demandan la actividad menos sofisticada posible de los tags, son los protocolos de tipo árbol basados en *queries*, como el QT. Los tags que trabajan bajo las directrices de este protocolo requieren únicamente de un comparador, además del correspondiente transmisor y receptor de señal. El tag recibe el *query* transmitido por el lector, lo compara con su ID, y si coincide, éste transmite el ID completo o la parte restante del prefijo recibido hasta completar el ID. Cuando Law et al. aplicaron este protocolo a RFID, las longitudes de los tags no eran tan elevadas como hoy en día, donde una transmisión del ID fallida puede suponer 128 bits desperdiciados. Este comportamiento genera un aumento excesivo de los bits de transmisión de lector y tags, debido al elevado número de colisiones que se producen a lo largo del ciclo de interrogación. Los beneficios proporcionados por la simplicidad de la circuitería de los tags quedan empañados, ya que, en cada colisión, los tags transmiten su ID completo, o casi al completo. Es decir, que se incrementa en gran medida el número de bits transmitidos con respuestas que podrían evitarse, provocando un incremento de la latencia y del consumo energético del sistema.

Con el objetivo de preservar las características mencionadas anteriormente, deseables para un protocolo de anti-colisión, se ha optado por aquellos cuyos tags demandan baja complejidad electrónica, dicho de otro modo, aquellos que presentan circuiterías poco complejas, evitando generadores numéricos aleatorios o memorias de almacenamiento adicionales a la necesaria para el ID de cada tag. Dentro de estas características se engloban los métodos *memoryless* basados en *queries*, los cuales presentan una oportunidad clara de mejora en la cantidad de información que intercambian lector y tags.

A raíz del análisis del estado del arte, en esta tesis se planteó como objetivo general desarrollar un procedimiento que modificase los protocolos basados en *queries* existentes, conformando así nuevos protocolos. Mediante este procedimiento se ha pretendido mejorar las características de latencia y consumo energético del protocolo al que se le aplica (protocolo origen). A continuación, se exponen los detalles que demuestran que se ha cumplido el objetivo de manera íntegra.

6.1.1 Contribuciones principales

La principal contribución de esta tesis es el procedimiento denominado ventana de bits. A la cantidad de bits que un tag transmite en un slot la hemos denominado **ventana**. Así, el lector decide en cada slot el tamaño que debe adquirir en función del tipo de respuesta de los tags y se lo indica a éstos en cada interrogación. Los tipos de respuestas recibidas de los tags conforman cuatro tipos de slot: libre, en caso de no recibir respuesta alguna; colisión, si las ventanas recibidas de los tags son diferentes; go-On, cuando la o las ventanas de los tags recibidas son idénticas pero la longitud de la respuesta no alcanza la longitud de los IDs de los tags; y finalmente, satisfactorio, al recibir una única respuesta que completa los bits recibidos con una longitud igual a la de los IDs de los tags.

La idea fundamental del procedimiento es la de adaptar el tamaño de dicha ventana en función de las necesidades del sistema. Pequeños tamaños serán requeridos cuando la probabilidad de colisión sea mayor, al contrario que cuando un único tag responda, que será más beneficioso que transmita una ventana mayor. De esta manera, el número de bits transmitidos por el tag se reduce al igual que la transmisión innecesaria de bits. Dada la dificultad en la predicción de cuándo se va a producir

6. Conclusiones

una colisión, el proceso que se ha seguido para definir el comportamiento de la ventana es el siguiente:

- ◇ Inicialmente se ha realizado un análisis del protocolo aplicándole diferentes tamaños de ventana de valor constante. De esta forma, se pretende conocer de qué forma afecta cada tamaño de ventana al protocolo. Para, con las conclusiones obtenidas, definir un procedimiento dinámico que reduzca el problema encontrado del exceso de bits transmitidos.
- ◇ A continuación, se plantea un comportamiento para la ventana, teniendo en cuenta las conclusiones obtenidas del caso anterior de la ventana constante. Para la definición de dicho comportamiento se recurre a funciones simples (lineal, cuadrática, exponencial en este caso), que son sintonizadas para adaptarse a un tipo de distribución de IDs de tags.

Dado que este tipo de comportamiento se acopla de forma inmediata con el empleado por los protocolos basados en *queries* como el QT, esta tesis se ha centrado en la aplicación del procedimiento sobre este tipo de protocolos. Se han escogido los protocolos: QT, por ser el protocolo que provee el hardware más sencillo y con mayor capacidad de mejora; y el protocolo CT, que implementa la técnica bit-tracking y se encuentra a la vanguardia de los protocolos de anti-colisión.

- ◇ Protocolo **QwT**: en este caso la ventana se aplica sobre el protocolo QT. Cuando el lector transmite el *query* que indica a los tags si deben transmitir o no, el procedimiento diseñado añade el dato que indica el tamaño de la ventana. El tag recibe el mensaje y compara el *query* con su ID. En caso de coincidir, éste rellena la ventana del tamaño indicado utilizando los bits de su ID situados a continuación del *query* recibido y añade un CRC que permita la identificación de la colisión en el lector. Una vez transmitida la respuesta, el lector recibe la información y chequea si el CRC de la respuesta recibida es coherente. Las principales ventajas que aporta la ventana sobre el protocolo QT son:
 - Descenso del número de bits desaprovechados en una colisión y por tanto, decremento del número de bits transmitidos por los tags.

- Reducción del número de colisiones, ya que las ventanas de pequeño tamaño ($W < 10$) disminuyen la probabilidad de colisión.
- Del mismo modo, ventanas de pequeño tamaño aumentan la probabilidad de obtener respuesta, bien colisión o bien de tipo go-On, disminuyendo la probabilidad de que se produzcan slots libres y reduciendo también su cantidad.

Cabe destacar que el lector debe comprobar si la suma del tamaño del *query* y la ventana recibida es menor o igual que la longitud del ID, ya que en caso de no serlo, se considera un slot de tipo go-On. Esto provoca que el lector deba actualizar el último *query* enviado con la ventana recibida al completo y retransmitirla como un nuevo *query* para obtener el resto del ID.

El proceso de implantación de la ventana en el protocolo comienza, tal como se ha indicado, realizando un estudio ante valores fijos de la ventana, W . Las conclusiones obtenidas al aplicar la ventana constante sobre el protocolo QT son:

- Bajos valores de W proporcionan una buena eficiencia de bits transmitidos por los tags y generan un decrecimiento del número de colisiones y ciclos libres. Sin embargo, produce un aumento del número de slots necesarios por la aparición del slot de tipo go-On, lo que a su vez aumenta el número de bits transmitidos por el lector.
- Valores de W cercanos a la longitud del ID de los tags, k , provocan un comportamiento similar al del protocolo origen, QT en este caso. Por tanto, los slots se reducen con respecto al caso anterior, pero persiste el problema del elevado número de bits transmitidos por los tags malgastados.

Con las conclusiones obtenidas, se plantean tres funciones heurísticas (lineal, cuadrática y exponencial), para dotar de dinamismo a la ventana y aprovechar sus beneficios mitigando los inconvenientes observados, como el exceso de slots de tipo go-On. Estas funciones permiten a la ventana adoptar valores pequeños cuando la probabilidad de colisión es elevada, provocando que los

6. Conclusiones

tags transmitan cantidades menores de bits. O por otro lado, aumentar el valor W con la longitud del *query*, permitiendo que los tags aumenten la cantidad de éstos y se reduzca así el número de slots de tipo go-On.

Dado que este protocolo asume la perfecta sincronización entre las respuestas de los tags, se propone a continuación, aplicar la ventana sobre el protocolo CT, que emplea bit-tracking facilitando esta tarea.

- ◇ Protocolo **CwT**: aplica la ventana sobre el protocolo CT. Con esto, se presenta el reto de la convivencia entre dos técnicas: la propia ventana y la técnica de bit-tracking. Las principales ventajas que se consiguen con esta combinación son las siguientes:
 - La técnica de bit-tracking facilita la omisión del CRC empleado para identificar el tipo de colisión, ya que la codificación Manchester permite identificar el bit exacto colisionado.
 - La sincronización entre las respuestas de los tags. La codificación Manchester, que lleva la señal de reloj incorporada, facilita la sincronización de las respuestas.
 - El procedimiento empleado en el CT facilita la extinción de los slots libres, y el empleo de la ventana lo mantiene en el CwT.

Inicialmente se ha aplicado la ventana constante sobre el protocolo CT, con el objetivo de observar el comportamiento del protocolo ante diferentes tamaños de ventana. Los resultados obtenidos denotan similitud con respecto al protocolo QT, en la manera que valores de W cercanos a la longitud del ID de los tags provocan un comportamiento similar al del protocolo origen, y valores inferiores reducen el número de bits transmitidos por los tags, aumentando el número de slots y bits transmitidos por el lector. Cabe destacar, que en este caso la ventana no ejerce ninguna influencia en el número de colisiones que se producen, ni tampoco provoca la reaparición de los slots libres.

A continuación, al proponer el dinamismo de la ventana, se procede a aplicar las tres funciones heurísticas ya mencionadas (lineal, cuadrática y exponencial) con el objetivo de mantener las ventajas comentadas y reducir los inconvenientes. Los comportamientos que producen estas funciones sobre el protocolo son, sin embargo, diferentes a los que producen en el protocolo QwT. Las sintonizaciones que mejor resultado han dado, producían funciones de forma muy abrupta, es decir, que un incremento de longitud de *query* generaba un incremento muy grande de ventana. La principal razón para este comportamiento es que la técnica de bit-tracking es capaz de aprovechar los bits comunes recibidos previamente a la primera colisión de bits y, por tanto, el protocolo da mejores resultados ante ventanas de mayor tamaño, puesto que puede aprovechar bits anteriores a una posible colisión. Esto, por otro lado, genera un problema derivado que es el incremento en bits transmitidos por los tags con respecto al QwT. Con el fin de mitigar este efecto secundario de la combinación entre bit-tracking y ventana de bits se ha presentado una modificación del protocolo CwT, el CwT+.

- ◇ Protocolo **CwT+**: este protocolo es una modificación del CwT. Se trata del mismo caso de la combinación de la ventana con la técnica de bit-tracking, en la que se da un paso más. El protocolo propuesto estima el número de bits que los tags necesitan transmitir para evitar el exceso de bits desechados. Para ello se introduce el concepto del *valor umbral*. Gracias a la capacidad de la técnica de bit-tracking para trazar la colisión a nivel de bit, se propone obtener también el bit dónde ocurre la segunda colisión en las ventanas recibidas, en caso de que ésta se produzca. El valor umbral adquiere el valor de la diferencia de bits entre las dos colisiones más uno y el protocolo genera, de la misma forma que el CwT, dos nuevos *queries* (tras la colisión). Dicho valor umbral es aplicable a ambos *queries* generados y, por ello, este valor se almacena en la pila LIFO junto a uno de los *queries*, además de que se asigna al tamaño de ventana a solicitar junto al otro *query*. De esta manera, las respuestas de los tags, a ambos *queries*, se reducen a la transmisión establecida entre dos colisiones, evitando bits innecesarios y consiguiendo reducir

6. Conclusiones

el número de bits transmitidos por los tags, sin perjudicar a la latencia o el consumo energético del CwT.

Al requerir un comportamiento de ventana diferente, se resintonizan las tres funciones heurísticas propuestas. Sin embargo, la forma de las funciones adquiere un aspecto similar al del CwT, donde resulta más interesante el rápido incremento del tamaño de la función en cuanto se produce un slot de tipo go-On. Por un lado, para aprovechar los bits anteriores al bit colisionado, y por otro, para aprovechar el valor umbral en mayor medida, ya que en ventanas mayores la probabilidad de producirse dos colisiones es mayor.

En la posterior sección, se muestran las conclusiones de la experimentación, es decir, de la simulación y comparación de los protocolos diseñados con sus protocolos origen y con otros pertenecientes a la literatura actual.

6.2 Conclusiones de la experimentación

La experimentación realizada se ha llevado a cabo de manera que se pueda validar la hipótesis. Por tanto, para ofrecer las conclusiones de dicha experimentación, se procede primeramente a recordarla (Sección 1.4.1).

«Es posible diseñar un procedimiento que, aplicado sobre un protocolo de anti-colisión RFID basado en *queries* ya existente, consiga reducir el número de bits transmitidos por los tags, mejorando el comportamiento del protocolo en latencia y consumo energético en entornos de tags pasivos.»

Para validar la hipótesis se ha desarrollado un banco de pruebas que ha permitido la simulación de los protocolos diseñados, y su comparación con otros protocolos de la literatura existente. Este banco también ha permitido la variación de diversos parámetros que afectan al comportamiento de dichos protocolos, de manera que se ha podido observar la influencia que ejercía la ventana sobre aquellos a los que se ha aplicado.

Inicialmente y siguiendo el orden establecido en esta tesis, el primer método sometido a examen ha sido el **QwT**. En primer lugar, se han comparado las diferentes versiones de éste, es decir, bajo la influencia de las tres funciones propuestas

6.2 Conclusiones de la experimentación

(lineal, cuadrática y exponencial), con el protocolo QT únicamente, a fin de observar los beneficios e inconvenientes que provocan sobre el protocolo. La primera conclusión clara obtenida es que la ventana produce una drástica reducción del número de bits transmitidos por los tags. De la comparativa realizada cabe destacar que la función cuadrática (parábola) presenta los mejores resultados en bits totales transmitidos y en los bits que transmite cada tag. La función exponencial también ofrece grandes resultados, pero los tags no consiguen reducir de la misma forma los bits transmitidos. La reducción de bits transmitidos por los tags provoca, en todos los diseños planteados, que el lector se vea forzado a incrementar el número de mensajes que intercambia con ellos, lo que aumenta el número de bits transmitidos por dicho lector. A pesar de este incremento, todas las versiones del protocolo QwT presentadas ofrecen una latencia inferior a la de su protocolo origen. Para el caso considerado en la presente tesis y reflejado en la hipótesis de partida, empleando tags pasivos, el consumo energético se reduce gracias a que la reducción de bits transmitidos por los tags se refleja en la latencia empleada por el protocolo. Es decir, que la reducción de la transmisión de bits proporcionada por la ventana, se ve reflejada en el total de bits transmitidos entre lector y tags y de la misma manera en la latencia total del sistema RFID. Posteriormente, considerando los consumos del lector ante las diferentes situaciones del proceso de lectura se concluye que la ventana produce un descenso del consumo energético del lector interrogando tags pasivos, siendo la función cuadrática la que asigna los valores más correctos y produciendo los mejores resultados. Una vez escogido el mejor comportamiento de la ventana, se ha realizado la comparativa del protocolo QwT, y la función cuadrática, con una serie de protocolos existentes en la literatura. Se ha confirmado con esto, que el QwT ofrece grandes resultados en latencia y en el consumo energético de sistemas que emplean tags pasivos, superando incluso a protocolos como el FTTP que emplean tags con un hardware más costoso y complejo, validándose de nuevo la hipótesis planteada.

A continuación, se ha procedido a realizar la experimentación correspondiente de la ventana sobre el protocolo CT, en el protocolo diseñado **CwT**. La característica principal de este último es la utilización de la tecnología bit-tracking, con lo que el protocolo propuesto finalmente combina dos técnicas diferentes, que a

6. Conclusiones

su vez resultan perfectamente acopladas, la ventana de bits junto con la mencionada bit-tracking. Al igual que en la anterior experimentación, se ha comenzado comparando las tres versiones del CwT propuestas con el protocolo CT. Con estas comparativas se ha demostrado que la ventana de bits produce una importante reducción del número de bits transmitidos por los tags. A pesar de que esta reducción ha de compensarse con un aumento del número de bits transmitidos por el lector, este excedente no es lo suficientemente elevado para perjudicar el comportamiento final del protocolo. Así pues, el CwT exhibe un reducido empleo de bits en su totalidad, que finalmente se refleja en una baja latencia y consumo energético utilizando tags pasivos, siendo la función lineal la que ha ofrecido mejores resultados, especialmente en el número de bits transmitidos por los tags, ya que en el resto de medidas los resultados han sido muy parejos.

Las pruebas realizadas confirman la complementariedad de ambas técnicas, y en las simulaciones mostradas en el Capítulo 5 se presentan los excelentes resultados que se consiguen con dicha combinación. Por un lado, el bit-tracking reduce el número de colisiones que se producen en el protocolo y aprovecha posibles bits comunes previos a la colisión. Por otro lado, la ventana permite reducir el número de bits transmitidos por los tags evitando la transmisión innecesaria de éstos. La compaginación de ambas técnicas proporciona, también, una disminución del número de slots en relación al protocolo QwT y a otros protocolos de la literatura reciente. En esta comparativa se ha incluido el protocolo OQTT, que emplea bit-tracking, y muestra unos excelentes resultados en cuanto a número de slots empleados. Sin embargo, el protocolo propuesto CwT con la función de ventana lineal ofrece mejores resultados en latencia y consumo energético, mostrando nuevamente la validación de la hipótesis. Cabe destacar que a pesar de los excelentes resultados ofrecidos por el CwT, el número de bits transmitidos por los tags es mayor que el ofrecido por el protocolo QwT. Es decir, que se ha conseguido un protocolo que mejora la latencia con respecto éste, a costa de un ligero incremento en esta medida. Para solventar este contratiempo o efecto secundario, se presenta la mejora CwT+.

El protocolo **CwT+** propone trazar no solo la primera colisión, sino también la segunda. Este número de bits existentes entre ambas colisiones, denominado valor umbral, se emplea para establecer un tamaño para la ventana más preciso que en el caso del CwT, consiguiendo reducir el número de bits malgastados por los tags, y

reduciendo el total transmitido por éstos. Debe mencionarse que al no variar apenas el número total de bits transmitidos, reduciendo los enviados por los tags, son los bits de lector los que aumentan. Esta variante del protocolo produce un ligero aumento de slots de tipo go-On, lo que a su vez genera la necesidad del lector de transmitir un mayor número de bits. De cualquier manera, los resultados globales del CwT y el CwT+ son muy similares, con la diferencia de que las aplicaciones que requieran tags más conservadores energéticamente o aplicaciones en las que los tags no puedan alcanzar la velocidad de transmisión del lector saldrán beneficiadas si emplean la variante CwT+.

Cabe destacar que se han diseñado tres protocolos de anti-colisión aplicando la estrategia de la ventana a dos protocolos basados en *queries* dadas sus flexibles características, validando nuevamente la hipótesis planteada en la que se proponía la aplicabilidad de la ventana a este tipo de protocolos.

Como conclusión final, se ha observado que el principal efecto de la aplicación de la ventana es la disminución del número de bits transmitidos por los tags y su consecuencia más directa, el incremento en los bits transmitidos por el lector. Este aumento, sin embargo, se ha conseguido controlar, en el sentido en que afecta lo menos posible tanto a la latencia como al consumo energético. Y es que, el incremento de bits que se produce en la transmisión del lector no es lo bastante elevado como para ocultar la reducción producida en el número de bits transmitidos por los tags, viéndose reflejado en la latencia y el consumo energético principalmente. Todo esto, empleando un procedimiento sencillo que no causa un gran incremento de la complejidad del sistema, ya que los protocolos diseñados continúan siendo *memoryless* y los tags únicamente requieren un contador adicional refrescado en cada slot con el valor del tamaño de la ventana y un circuito simple de cálculo de CRC (en el caso del QwT).

6.3 Limitaciones identificadas

En esta sección se recogen las limitaciones identificadas en el procedimiento diseñado en esta tesis. Se han ido observando algunas de ellas a lo largo del documento, aunque en este apartado se recopilan todas ellas.

6. Conclusiones

El principal inconveniente que provoca la aplicación del procedimiento de la ventana es el incremento de slots que se produce, mayoritariamente de slots de tipo go-On. Este incremento ha sido mitigado en gran parte ajustando el parámetro β de las funciones heurísticas, y se han observado excelentes resultados en latencia y energía consumida. A pesar de ello, se debe seguir trabajando para tratar de eliminar en lo posible este exceso de slots, ya que los protocolos presentados podrían ofrecer mejores resultados reduciendo este problema.

Por otro lado, se ha observado que la aplicación de la ventana ha sido mucho más efectiva sobre el protocolo QT que sobre el CT. Esto sucede ya que la capacidad de mejora del primero es mayor que la del bit-tracking. Sin embargo, los resultados finales han mostrado que el segundo se comporta mejor en latencia y consumo energético. Esta capacidad de mejora inferior se debe a un inconveniente que introduce la ventana en el protocolo CT y es preferible evitar, aunque sus efectos no son destacables. Cuando el tamaño calculado de la ventana produce una respuesta de tipo go-On, es decir, no se produce ninguna colisión en la respuesta, no significa que no puedan producirse colisiones en posteriores interrogaciones del mismo tag. Una ventana del protocolo CT recibiría todas las respuesta de los tags identificando en el primer slot el bit colisionado, pudiendo emplear los bits anteriores a la colisión para la generación de únicamente dos *queries* de forma más agresiva. Mientras que en la ventana, podría darse el caso de no recibir colisiones en las primeras interrogaciones. Por tanto, el CwT necesitaría transmitir más *queries* que su protocolo origen para averiguar la posición de la primera de las colisiones, con el correspondiente incremento de slots de tipo go-On y de bits transmitidos por el lector. Este problema no ha resultado de extrema relevancia ya que no se da de forma habitual, pero puede producir aumentos de latencia que se deben solucionar con incrementos de ventana más agresivos, corriendo el riesgo de incrementar los bits transmitidos por los tags.

Se ha comentado la consecuencia del incremento de bits del lector. Este incremento ha sido suficientemente mitigado para ofrecer buenos resultados de latencia y de consumo energético empleando tags pasivos. Sin embargo, también se ha realizado parte de la experimentación con tags activos. En tal caso, al considerar la potencia que emplean los tags en la recepción de los bits del lector, y considerando que ésta es mayor que la de transmisión, la ventana ha resultado contraproducente

en este caso. Bien es cierto que los tags activos son más complejos electrónicamente y que por tanto puede no ser muy efectivo que empleen un protocolo basado en *queries*, ya que es más factible que éstos dispongan de memoria interna u otro tipo de funcionalidades que faciliten esta tarea. Por ello, el procedimiento presentado en esta tesis no se ha pensado para este tipo de tags.

Las conclusiones obtenidas de los análisis realizados con la ventana constante, se han empleado para definir las funciones heurísticas en distribuciones de tags homogéneas. Sin embargo, no se ha realizado experimentación bajo otro tipo de distribuciones de tags, considerando ciertas zonas de su ID comunes u otro tipo. La realidad es que las funciones presentadas no se prevee que vayan a ofrecer buenos resultados ante estas circunstancias y debieran ser reajustadas.

Por último, se debe mencionar que los protocolos presentados en esta tesis se han experimentado en entornos teóricamente ideales, donde no se producen fenómenos de atenuación de señal, ni *capture-effect* u otro tipo de problemas relacionados con la transmisión y recepción de señales, así como tiempos de ejecución del hardware, de ahí que no se hayan considerado tiempos entre mensajes o tiempos de calibración. Sin embargo, en un sistema real se deberán tener en cuenta todos estos factores que pueden influir sobre el protocolo incrementando previsiblemente las medidas presentadas en este trabajo, ya que cambiarían las condiciones de las simulaciones.

6.4 Líneas futuras de trabajo

Dadas las limitaciones presentadas, la capacidad de mejora de las soluciones presentadas en este documento son claras. Por ello, se proponen las siguientes líneas para continuar la investigación:

- ◇ **Estandarización de los tamaños de ventana:** dado que uno de los principales problemas presentados es el elevado número de bits transmitidos por el lector, se propone la estandarización de los tamaños de ventana a valores que sean potencias de 2. De esta forma, en vez de enviar $\log_2 W$ bits junto a cada *query*, se enviarían siempre 3 bits con los que se puede representar hasta una ventana de $2^7 = 128$ bits (tamaño habitual del ID de los tags actuales)

6. Conclusiones

facilitando también la tarea de los tags para separar el *query* del parámetro de ventana, ya que los últimos tres bits recibidos siempre corresponderán al tamaño de ventana.

- ◇ **Tags con memoria:** plantear la aplicación de la ventana a un protocolo con memoria en los tags. De esta manera se tratará de reducir el número de bits transmitidos por el lector, ya que los tags poseerán la capacidad de recordar qué zonas de su ID deben transmitir en cada momento, a costa de un incremento en la complejidad de su hardware.
- ◇ **Funciones adaptativas:** se ha demostrado que las funciones empleadas dan buen resultado ante distribuciones homogéneas de tags. Pero, ¿qué ocurre cuando la distribución de los tags posee un prefijo, sufijo o interfijo común? En algunos casos, la ventana puede reaccionar de forma correcta, pero en otros puede resultar contraproducente, ya que la propia heurística no contempla esa posibilidad. De ahí que se plantee la posibilidad de adaptar la función que rige el tamaño de ventana teniendo en cuenta la distribución de los tags. Esta información la obtendrá el lector de los tipos de slot recibidos, y se irá completando a lo largo del ciclo de interrogación. Es decir que se pretende, o bien que se produzca un ajuste dinámico del valor de β en función del tipo de distribución de tags, o una modificación de los valores de la propia función teniendo en cuenta zonas de los ID comunes como prefijos, interfijos o sufijos.
- ◇ **Lógica difusa:** dado que el cálculo del tamaño de la ventana puede resultar poco exacto, se plantea la posibilidad de aplicar la lógica difusa para, a partir de la realimentación obtenida de las respuestas de los tags, calcular una respuesta empleando diferentes criterios teniendo en cuenta las probabilidades de colisión, slot libre o satisfactorio.
- ◇ Aplicación de la ventana a **protocolos Aloha:** también se puede plantear la posibilidad de aplicar la ventana a protocolos basados en Aloha. Generalmente este tipo de protocolos emplean números pseudo-aleatorios, los llamados *RNI6*. En cambio, se puede tratar de cambiar estos números aleatorios por fragmentos de sus ID definidos por el tamaño de ventana, generando

un envío de ventanas espaciadas en el tiempo. De esta forma el lector recibiría fragmentos del ID que podría ir rellenando de la misma manera que un rompecabezas.

6.5 Consideraciones finales

Desde que se comenzó a presentar la tecnología RFID en 1948 hasta hoy en día, el cambio producido ha sido considerable, especialmente a lo largo de los últimos 10 años, donde la investigación se ha incrementado en este campo. Una de las principales razones de este hecho es la adaptabilidad característica de esta tecnología a multitud de aplicaciones muy comunes hoy en día, como pueden ser: el control de acceso, control de inventario, trazabilidad de mercancías, detección de presencia o localización de bienes. Aún así, la razón de mayor peso es su aplicabilidad a uno de los temas más recurrentes de investigación de los últimos años, la computación ubicua.

Destaca el gran poder que aporta esta tecnología, la cual puede permitir la localización de artículos en un almacén para identificar aquellos que deban reabastecerse. O en el caso de consumidores, suplir información detallada de productos y eliminar así la necesidad de colas para pagar sus compras, deduciendo la compra automáticamente de su correspondiente tarjeta de crédito. Además de esto, los tags son capaces de incorporar sensores que envíen sus medidas una vez sean identificados por un lector. La tecnología presentada, por tanto, provee un abanico muy extenso de posibles aplicaciones, gracias a su flexibilidad y versatilidad.

Sin embargo, debe mencionarse que se trata de una tecnología que, a pesar de que se está implantando de forma gradual, todavía debe resolver algunos aspectos que pueden resultar perjudiciales a la larga, como: la influencia sobre el medio ambiente de los tags desechados, las posibles interferencias de las ondas de radio en las comunicaciones lector-tags, los entornos metálicos o líquidos cerca de los tags que dificultan su lectura o la vulnerabilidad que puede suponer un tag que contenga datos personales que no quieran ser divulgados. Por ello, hoy día la tecnología RFID continúa siendo un tema protagonista de investigación en diferentes campos. Desde el presentado en esta tesis, donde se trata de mejorar el consumo y la

6. Conclusiones

latencia de las lecturas de los tags, hasta el diseño de tags biodegradables o de protocolos que se centran en controlar la privacidad del sistema RFID. Finalmente, lo que está fomentando el rápido crecimiento de la tecnología y su esparcimiento en la industria, comercio y servicios es la estandarización. Unificando las reglas y el funcionamiento de RFID bajo un mismo estándar se pueden conseguir objetivos como reducir costes de manufacturado, mejorar la interoperabilidad entre diferentes empresas o sincronización global de datos entre productores y compradores. Prever el mundo de mañana es cada vez más difícil por la velocidad a la que avanza, pero se puede esbozar una idea que puede ajustarse a la realidad más inmediata.



Conclusions

All along this document the proposed work has been exposed. Once introduced the overview of the RFID technology, the tag collision problem and its consequences have been presented. The related literature has been analysed and an improvement opportunity has been encountered in query based protocols. Query based readers need to transmit prefixes and they obtain tag's full ID responses, which increase the total number of bits transmitted by the tags. A procedure has been developed to solve this problem, called the bit window, and has been tested in two anti-collision protocols, producing three new protocols (QwT, CwT and CwT+). Furthermore, a large number of experiments have been developed concerning the validation of the hypothesis presented at the beginning of the thesis. Simulations performed have provided a great amount of information in order to explain the benefits and drawbacks the application of the window adds to the tested anti-collision protocols. All these conclusions are gathered in this part of the thesis.

This chapter is organized as follows. Firstly, a global view of the issues discussed in this document, stressing the main contributions. Secondly, the conclusions of the experimentation are shown, highlighting the validation of the hypothesis. And finally, limitations, some future work, and final considerations are presented.

A.1 General overview of the thesis

The RFID technology is one of the most known auto identification techniques. It is used to identify codes stored into small devices, called tags, using radio frequency. Tags are attached to different objects providing them with new functionalities. Location, traceability, and remote identification of the objects are the most common applications of RFID. The main characteristic of this technology is that it does not need any direct vision between the reader and the tags to perform all the mentioned applications. RFID is a low intrusive technology which can be easily adapted to the *Internet of things* and the paradigm of the *Ubiquitous Computation*.

The great flexibility of the RFID technology allows to use several readers and tags at the same time. This advantage causes some problems and the main studied in this thesis is **the tag collision problem**. Since tags share the same communications channel (the air), they can respond simultaneously to the same interrogation command of the reader interfering and garbling their waveforms. That leads to a retransmission of the tag messages until the reader receives them correctly, causing an increase in the time to identify the tags and the energy consumed during that process.

Numerous solutions have been given to mitigate that mentioned problem. Collisions are a well-known issue controlled by multi-access protocols in the classic networks theory. The first solutions proposed to solve the tag collision problem were inspired in those multi-access protocols. However, these protocols cannot be applied directly to RFID, since it imposes some restrictions. Multi-access protocols had to be adapted, producing **anti-collision protocols**. The design of these is still an interesting research topic including a lot of existing literature. The main anti-collision protocols presented in the literature can be classified in three groups: Aloha based, tree based, and hybrid protocols.

- ◇ Aloha based protocols are probabilistic, since tags responses are organized randomly. Thus, the identification of all the tags is not guaranteed. However, a good anti-collision protocol maximizes the probability of identifying all the tags in the range of the antenna. Tags responses are distributed among slots and the more uniform the space between transmissions the bigger the probability of success. Research in Aloha based protocols is focused on DFSA

improvement. Estimators are increasingly accurate as well as mathematical and computational complexity. Therefore, these protocols demand high cost tags to be implemented.

- ◇ Tree based protocols identify the total amount of tags in the interrogation zone and are deterministic. The total amount of information transmitted between their readers and tags is usually higher than the transmitted by Aloha based protocols. However, the hardware of the tags is usually simpler in tree based protocols. The reader controls every step of the protocol using commands or prefixes (called queries) to separate tags responses into smaller groups until it obtains unique answers to identify each one of the them. There are two types of tags: with or without memory. The main difference between them is that tags not using memory need the reader to transmit a query prefix, and only matching tags will respond to that; whilst tags with memory usually use a counter to determine the allowance of transmission. The need of the tags not using memory of transmitting those queries increases the amount of information transmitted between the reader and the tags. In contrast, tag hardware requirements are the smallest ones.
- ◇ Finally, hybrid protocols combine both previous strategies in order to decrease some of their drawbacks. Even so, an increase in the whole system's hardware complexity is needed.

Given that passive tags obtain power from the reader signal and do not use batteries, the simplicity of the tags is very important. Otherwise, the high cost of the tags can cause an increase in the reader's power consumption because it will need more energy to transmit messages and to power up its circuitry. And in the same way, the power of the signals transmitted by the reader is controlled by country regulations, which could be lower than expected. This fact leads to an increase of the tag's exposure time, affecting the latency of the system. Additionally, the price of manufacturing tags is a key factor and is completely influenced by the electronic complexity of them. An increase in the manufacturing price of the tags could be fatal for the viability of the RFID technology, which aims to substitute bar codes.

A. Conclusions

Literature presents that the simplest protocols, demanding the less sophisticated activity from the tags, are the query based protocols such as the QT. Apart from a transceiver and a receiver tags, working under this protocol conditions require just a comparator. Tags receive the query transmitted by the reader and compares with its own ID. Then, matching tags transmit their complete ID, or the remaining portion to complete the ID. When Law et al. designed the QT for RFID, the length of the tags' ID were not as long as nowadays. A collision, these days, can cause a common loss of 128 bits. Although QT tags are electronically simple, the excessive number of collisions produced by the QT generates an enormous increase in the information transmitted between the reader and the tags. Each collision produces a high loss of information that leads to an increase in the latency and power consumption of the RFID system.

With the main purpose of preserving the mentioned features of the query based protocols, a low tag complexity protocol is preferred. The *memoryless* protocols based on queries fit those specifications and their tags avoid random number generators or additional storing memories apart from the ID memory. This kind of protocols show an important improvement opportunity related to the quantity of information transmitted between the reader and the tags.

As a result of the state of the art analysis, a new procedure is proposed to improve query based protocols. This is applied to this type of protocol to improve their latency and energy consumption features, by means of the amount of information interchanged between the reader and the tags. Below the main contributions are exposed in order to prove the achievement of the objectives.

A.1.1 Main contributions

The main contribution of this thesis is the procedure called the bits window. The **window** is defined as a frame which contains the bits a tag must transmit in a slot. The size of the window or the number of bits the tag should transmit is commanded by the reader and attached to the query on each slot. Four types of slots can occur depending on the answer of the tags: idle, if no tag response is received; collision, if windows received are different; go-On, whether the windows responses are

identical but the length of the response is not enough to define the tag ID completely; and finally a success, if only one window is received and completes the length of the tag ID.

The main idea of the proposed procedure is to adapt the size of the window according to the demands of the protocol. In this manner, small window values will be required when the probability of collision is high, and on the contrary, a larger window size will be more beneficial when only one tag responds. Thus, the number of bits transmitted by the tags and the unnecessary bits transmitted are likewise reduced. Since it is difficult to predict a collision and define the window behaviour accordingly, this process has been followed:

- ◇ In the beginning, the protocol has been analysed under constant window value conditions. The influence of each window value can, therefore, be known, so that a dynamic window behaviour, beneficial for the protocol, could be established.
- ◇ Besides, a new window behaviour is suggested with the conclusions obtained previously. With that purpose, three defining functions are presented (linear, quadratic and exponential), which are tuned to be adapted to the tag ID distribution.

The proposed procedure fits directly in the query based protocols, such as the QT protocol. This thesis is therefore focused on the applications of this procedure on these kinds of protocols. Two protocols have been chosen in this work to be modified with the window: the QT, which provides the simplest hardware and the highest improvement capacity; the CT, that uses bit-tracking and is one of the latest proposed anti-collision protocols.

The first presented protocol was the **QwT**, which applied the window to the QT protocol. Whenever the reader transmitted a query to indicate the tags whether to transmit or not, the designed procedure added a window size parameter in the message. Tags received the command (composed of the query and the window size) and compared the query with their ID. Matching tags filled in the window frame, which was defined by the window size parameter, with their following ID bits next to the last query bit received. Apart from this, a CRC was calculated, attached to

A. Conclusions

the window and sent back to the reader. Once transmitted, the reader received all the messages and checked for the consistency of the CRC. The main advantages the window added to the QT protocol were:

- ◇ A decrease in the number of wasted bits per collision. And also, a decrease in the number of bits transmitted by the tags.
- ◇ A reduction in the number of collisions, since a small window size ($W < 10$) decreases the probability of collision.
- ◇ Similarly, a small window size increase the probability of obtaining any kind of tag answers (collision, go-On, success), which at the same time decreases the number of idle slots.

It should be noticed that the reader needed to compare the sum of the query and window lengths to the ID length. If the result was smaller than the ID length, the response was considered a go-On slot. Thus, the reader had to update the query with the received window. Then, it retransmitted the new one in order to obtain the full length of the ID and defined the identified tag completely.

As previously mentioned, the implantation process of the window began with an analysis of the protocol under fixed window size value W conditions. These are the main conclusions obtained:

- ◇ Low W values ($W < 10$) provided an excellent efficiency of the bits transmitted by the tags (near 100%) and a decrease in the number of collisions and idle slots. However, the number of go-On slots needed to identify the full length of the ID was high, which caused the increase in the total number of slots and bits transmitted by the reader.
- ◇ W values near the length of the tag ID provided a similar behaviour to the source protocol (QT in this case). The number of slots were therefore reduced, but the previously mentioned problem of the excessive number of bits transmitted by the tags persisted.

Considering these results, three heuristic functions were presented (linear, quadratic and exponential) in order to provide dynamism to the window size. These three functions allowed the reader to choose small window size values when the probability of collision was prone to increase, which provided a decrease in the number of tag bits wasted. On the contrary, these functions offered larger W values, when the length of the query increased (the probability of collision decreased) allowing the tags to transmit a larger amount of bits, which also contributed to decrease the number of go-On slots.

Assuming a perfect synchronization between tag responses, the application of the window to the CT protocol was proposed, which used bit-tracking to improve its performance.

The second proposed protocol was the **CwT**, which applied the window to the CT protocol, causing the problem of the coexistence of two techniques: the proposed window and the bit-tracking. The main advantages of this combination were:

- ◇ The bit-tracking technology facilitated the avoiding of the CRC, used in the QwT to identify the type of slot. As a matter of fact, Manchester codification allowed the reader to identify the exact collided bit.
- ◇ Synchronization between tags' responses. The Manchester coding, which incorporated the clock signal with the data, simplified the drawback of synchronization.
- ◇ The combination of both techniques kept the ability to extinguish idle slots in the proposed CwT protocol.

Following the procedure done for the QwT, the CT protocol was tested under constant window values. As a result, similar conclusions were also obtained. The nearer the W value was to the length of the tag's ID, the more similar the behaviour was to its source protocol. And on the other hand, small W decrease the number of bits transmitted by the tags causing an increase in the number of go-On slots and bits transmitted by the reader. It is important to note also that there was no influence of the window in the number of collisions, which remained the same, and the idle slots kept extinguished.

A. Conclusions

Since results obtained from applying the window to both QT and CT pointed in the same direction, the dynamic window was presented with the same (linear, quadratic and exponential) heuristic functions. The produced behaviour in the CwT protocol was, however, different from the QwT. These functions provided a bigger increase of the window size with a small change of the length of the query than the QwT. The main reason for this, was that the bit-tracking technology was able to take advantage of the received bits previous to the first collided bit of the window, and used them in the next query generation. The protocol, therefore, provided better results under bigger window size values. In contrast, this behaviour caused an increase of the number of bits transmitted by the tags in accordance with the QwT. A new modification of the CwT was proposed to mitigate this drawback, the CwT+.

The last proposed protocol was the **CwT+**, which was an improvement of the CwT and thus, it also used a combination of the bit-tracking and window techniques. This last proposed protocol tried to guess the exact number of bits the tags needed to transmit avoiding the wasting transmissions. With that purpose the *threshold value* was introduced. The bit-tracking technique was able to trace not only the first collided bit, but also the second, in case it happened. The threshold value was calculated as the difference between the position of the second and the first collisions. The protocol continued like the CwT generating two new queries (after the collision). In this manner, the calculated threshold value was valid for the two generated queries. One of them was stored in the LIFO stack, together with the calculated value. And the remaining query was transmitted to the tags, with the window size value. This was updated with the threshold value too, so that the received response could comprise the number of bits established between the previous calculated collisions. Therefore, the number of bits transmitted by the tags decreased not varying the latency or energy consumption of the CwT protocol.

The CwT+ was presented as a modification of the CwT requiring a different behaviour for the dynamism of the window, and thus, retuning the same three window functions. As a matter of fact, experimentation showed that the shape of the functions were pretty similar to the CwT shapes, which provided a faster increase of the window size value than the QwT functions. On the one hand, the reader took advantage of the bits received before the first collision. And on the other hand, it

A.2 Conclusions of the experimentation

increased the probability of collision in more than one bit, since a large window was prone to collide easier than a small one.

In the next section the conclusions of the experimentation are exposed. Simulations and their results are compared between them, considering the designed protocols and the latest ones in the literature.

A.2 Conclusions of the experimentation

In order to validate the hypothesis experimentation was achieved. Initially, before presenting the conclusions obtained, the hypothesis is remembered (Section 1.4.1).

It is possible to design a procedure, which applied to an existing query based anti-collision protocol, manages to reduce the number of bits transmitted by the tags. Apart from this, the procedure will improve the latency and energy consumption of its father protocol in a passive tags environment.

With the aim of validating the hypothesis a testing workbench was developed. Simulations of the designed protocols were performed in this workbench, and compared to others existing in the literature. This testing workbench also allowed to vary diverse parameters related to the protocols tested, in order to observe their behaviours. In addition, simulations also showed the influence of the window on the chosen protocols.

Following the order established in this thesis, the first protocol examined is the **QwT**. At first, three versions of the protocol were compared with the QT under the three functions mentioned before (linear, quadratic and exponential). The main and most clear conclusion obtained was the reduction of the number of bits transmitted by the tag in all the simulated versions of the QwT, especially the one using the quadratic function. The exponential function also presented great results but it did not manage to decrease tag bits as the quadratic did. This mentioned reduction caused the increase of the number of reader messages in all versions, and the number of bits transmitted by the reader were, therefore, increased regarding the QT. Despite this fact, all proposed versions of the QwT obtained better latencies than

A. Conclusions

their source protocol (QT). Using passive tags, as it has been mentioned in the hypothesis, energy consumption was also reduced in all the proposed versions, thanks to the decrease in latency and total transmitted bits. Namely, the reduction of bits provided by the window was reflected on the latency and the power consumption of the RFID system. Furthermore, after the simulations performed under different tag circumstances with all the QwT versions, it was concluded that the proposed window produced a decrease in the energy consumption of the RFID system considering passive tags. From all the presented results, the quadratic function showed the best behaviour in latency and energy consumption. Once these conclusions were obtained, a comparison between the QwT protocol with the quadratic function and some of the latest protocols in the literature was performed. These simulations confirmed the great latency and energy consumption results of the QwT using passive tags, which outperformed even the FTTP that used more sophisticated tags. In this manner, the hypothesis was again validated.

The next tested protocol was the **CwT** protocol, which was composed of the window applied to the CT protocol. The main feature of this protocol was the combination of the bit-tracking and the window procedure. Both techniques could be perfectly coupled and that was shown in the results of the experimentation. This was started comparing the three versions of the CwT with the CT. The first simulation demonstrated the improvement that the window produced on the CT protocol decreasing the number of bits transmitted by the tags. Despite the fact that this reduction was compensated with an increase on the number of reader bits, this increment was not enough to increase the total number of bits. Thus, CwT performed the identification with a low number of bits, which, in the end, was reflected on a low latency and energy consumption identification using passive tags. From all the simulated versions, the linear function presented the best results, especially in the number of bits transmitted by the tags.

The performed tests confirmed that both techniques were fairly complemented and simulations showed in Chapter 5 exhibited the excellent results using this combination. On the one hand, the bit-tracking technology decreased the number of collisions and used the bits received before the first collided bit. On the other hand, the window procedure reduced the number of bits transmitted by the tags avoiding the unnecessary bits transmission. The combination of both techniques brought also

A.2 Conclusions of the experimentation

a reduction of the number of slots in accordance with the QwT protocol and other novel protocols in the literature. The OQTT protocol, which used bit-tracking too, was included in the comparison. Despite the fact that it showed excellent results in the number of slots used, the proposed CwT with the linear function performed with better results in latency and energy consumption than the OQTT, validating the hypothesis again. Although these were good results, it should also be noticed that the number of bits transmitted by the tags of the CwT was higher than the QwT. With the aim of solving this drawback, the CwT+ improvement was presented.

The **CwT+** protocol traced not only the first collision of the received window but also the second. The number of bits existing between both collisions was called threshold value, and it was used to establish a more precise window size than the CwT protocol. This reduced the number of wasted bits by the tags and decreased the total number of bits transmitted by them. Besides, the total number of bits presented no variation due to a little increase in the number of bits transmitted by the reader. The limitation of the window size with the threshold value forced the reader to increase the number of go-On slots, which produced that increase in the number of bits transmitted by the reader. Final results of the CwT and CwT+ protocols were pretty similar, in any case. The main difference was that applications requiring less consuming or low speed tags would work better with the CwT+ than with the CwT, producing lower latency and energy consumption results.

To sum up, three anti-collision protocols have been developed applying the proposed window procedure to two anti-collision protocols. In order to validate the applicability of the procedure to any *query* based protocol, two protocols have been chosen.

In conclusion, the main benefit of the application of the window to a query based protocol was the decrease of the number of bits transmitted by the tags. And in contrast, that caused the increase of the number of bits transmitted by the reader due to the go-On slot appearance. However, the consequences of the go-On slot were mitigated, since it did hardly affect latency and energy consumption. Namely, the increase in the number of bits transmitted by the reader was not high enough to overwhelm the reduction of the number of bits transmitted by the tags, which was reflected on the latency and energy consumption of the RFID system. The main objectives of the hypothesis have been achieved with a low cost procedure for the

A. Conclusions

tags. Keeping the *memoryless* feature, tags used an additional pointer and a simple CRC calculator (only for the QwT).

A.3 Identified limitations

This section presents a collection of all the limitations encountered during the design and experimentation process. Some of them have been already mentioned, but in this part of the document all of them are gathered.

The main problem caused by the window is the increase in the number of slots, mainly caused by the increase in the number of go-On slots. That has been controlled or mitigated adjusting the β parameter of the three heuristic functions, and excellent results in latency and energy consumption have been obtained. However, this behaviour can be improved and therefore, more work should be done to decrease the number of go-On slots.

Moreover, it has been observed that the window is much more effective in the QT protocol than in the CT. Although the improvement capacity of the first one is larger than the other one, which uses bit-tracking, CT latency and energy consumption results are better than QT results. The main reason for the small improvement capacity of the CT is the go-On slot again. The window size can produce go-On slot that in the CT would have been a collision. And therefore, the QwT will need more slots than the CT to reach that collision, increasing the number of go-On slots and reader transmitted bits. This is not a relevant problem since it is not very common, however, it can cause an increase in latency or energy consumption. More aggressive increments of the window size can solve this drawback, but in contrast, the risk of increasing the number of bits transmitted by the tags rises.

Latency and energy consumption results obtained with passive tags have been remarkable. However, experimentation using active tags has not been as favourable as with passives. Considering that the reception power of the bits transmitted by the reader in the active tags is larger than the transmission power of the tags, the window is counter-productive, since it manages to decrease transmission regardless of reception. Undoubtedly, the cost of active tags is usually higher than passive tags, with memories and many additional functionalities. Consequently, it will

not be very effective that active tags use a query based protocol. The presented procedure, therefore, is not suitable for this kind of tags.

Conclusions obtained with the analysis of the window under constant values have been used to define the heuristic functions in homogeneous tags distributions. But in fact, no experimentation has been done with non-homogeneous tag ID distributions. It is believed that proposed heuristic functions are not going to provide a good behaviour for the protocols and they should be retuned.

At last, it should be mentioned that all these protocols have been tested in a theoretically ideal environment, where there are no attenuation phenomena, nor *capture-effect* or other kinds of transmission or reception problems. Besides, hardware execution time, time between messages or calibration time have not been taken into account. A real system will require all these factors, and the proposed protocols will predictably increase the obtained measures in this work since simulation conditions would have changed.

A.4 Future work

After presenting the limitations of this thesis, the improvement capacity of the proposed solutions is clear. Some proposals are shown in this section:

- ◇ **Standardization of window sizes:** it has been shown that one of the problems is the excessive number of bits transmitted by the reader. A standardization of window sizes is proposed to use multiples of 2. In this manner, instead of sending $\log_2 W$ bits attached to a query, only 3 bits will always be transmitted to describe a maximum of $2^7 = 128$ bits (common value of the actual tag IDs). This will facilitates the tags to identify the window size since the last three bits will always be the value sought.
- ◇ **Tags with memory:** applying the window to tags with internal memory. This will try to decrease the number of bits transmitted by the reader avoiding the need of transmitting queries. Tags will remember which part of their ID they should transmit on a slot at the expense of an increase of the cost of tags.

A. Conclusions

- ◇ **Adaptive functions:** it has been shown that the heuristic functions used work properly under homogeneous tag ID distributions. But, what does it happen when the distributions is not homogeneous? In some of the cases, the window can work properly, but as a matter of fact, it can be counter-producing since heuristics are not considering these possibilities. Adaptive functions are proposed to be modified according to the feedback of the tag responses. A dynamic adjustment of the β parameter or a modification of function values are suggested considering the type of the ID distribution.
- ◇ **Fuzzy logic:** since the window size calculation is not a precise calculation, it is suggested to apply fuzzy logic instead the heuristic function, using the feedback of the tags (collision, idle, go-On, success) to generate the window size.
- ◇ Window applied to **Aloha protocols:** the application of the window to Aloha based protocols is also suggested. Generally, these kind of protocols use pseudo-random numbers, called *RN16*. However, instead of transmitting those RN16, some parts of these can be transmitted and controlled by the window size. Besides, a separation of windows in time is also suggested. Then the reader will have to order the received messages to complete the received IDs.

A.5 Concluding remarks

Since the beginning of RFID's history in 1948 to nowadays, there has been a significant change. Specially during the last 10 years, research has been increased in this field. The main features of this technology make it attractive for a great number of very common applications such as: access or stock control, traceability or localization of goods and detection of presence. But mainly, because RFID can be applied to one of the most interesting research topics in the last few years, Ubiquitous Computing.

The great power of this technology should be noticed. It allows the localization of objects in a storage room in order to identify which should be restocked. Or in case of the consumers, to obtain detailed information of the products or avoid

queries in the supermarkets, deducing the payment from their credit cards automatically. Additionally, tags can include sensors, whose measures can be transmitted when queried. RFID technology, therefore, provides an enormous range of possible applications, thanks to its flexibility and versatility.

Although RFID is being gradually implemented, it should still solve some aspects that can be damaging in the long term such as: possible radio frequency interferences in reader-tag communications, metallic or liquid environments, vulnerability of a tag containing personal data or nature issues like the influence on the environment of wasted tags. Nowadays, RFID technology is still a trending topic in different fields of research. Since the one presented in this work, where latency and energy consumption has been improved, to the design of biodegradable tags or privacy aware protocols. At the same time, standardization is promoting a fast growth of this technology and is being widespread on the industry, commerce and services. Should rules and functioning be unified under a standard, great targets will be achieved, such as: a decrease of manufacturing costs, an improvement of interoperability between different companies or a global synchronization between sellers and buyers data.

To foresee the world of tomorrow is increasingly difficult due to its fast growing speed. However, we can sketch an idea that can meet the most immediate reality.

*«We shall require a substantially new manner of thinking
if mankind is to survive.»*

Albert Einstein

Bibliografía

- [Abraham 02] Cherian Abraham, Vinay Ahuja, Arnab Kumar Ghosh, Praveen Pakanati *et al.* *Inventory management using passive RFID tags: a survey*. Department of Computer Science, The University of Texas at Dallas, Richardson, pp. 1–16, 2002. 12, 13, 25, 40, 65, 67
- [Abramson 70] Norman Abramson. *THE ALOHA SYSTEM: another alternative for computer communications*. In Proceedings of the November 17-19, 1970, fall joint computer conference, pp. 281–285. ACM, 1970. 12, 22, 27
- [Barthel 06] Henri Barthel. *Regulatory status for using RFID in the UHF spectrum*. EPC global, 2006. 9
- [Bertsekas 92] Dimitri P Bertsekas, Robert G Gallager and Pierre Humblet. *Data networks*, volume 2. Prentice-Hall International, 1992. 38
- [Bhandari 06] Naval Bhandari, Anirudha Sahoo and Sridhar Iyer. *Intelligent query tree (IQT) protocol to improve RFID tag read efficiency*. In Information Technology, 2006. ICIT'06. 9th International Conference on, pp. 46–51. IEEE, 2006. 45
- [Bin 05] ZHEN Bin, Mamoru Kobayashi and Masashi Shimizu. *Framed ALOHA for multiple RFID objects identification*. IEICE Transactions on Communications, Vol. 88, No. 3, pp. 991–999, 2005. 32

BIBLIOGRAFÍA

- [Bonuccelli 06] Maurizio A Bonuccelli, Francesca Lonetti and Francesca Martelli. *Tree slotted ALOHA: a new protocol for tag identification in RFID networks*. In Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks, pp. 603–608. IEEE Computer Society, 2006. 55
- [Bonuccelli 07a] Maurizio A Bonuccelli, Francesca Lonetti and Francesca Martelli. *Instant collision resolution for tag identification in RFID networks*. Ad Hoc Networks, Vol. 5, No. 8, pp. 1220–1232, 2007. 55
- [Bonuccelli 07b] Maurizio Angelo Bonuccelli, Francesca Lonetti and Francesca Martelli. *Exploiting id knowledge for tag identification in rfid networks*. In Proceedings of the 4th ACM workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks, pp. 70–77. ACM, 2007. 45
- [Bueno-Delgado 09] MV Bueno-Delgado, J Vales-Alonso and FJ Gonzalez-Castao. *Analysis of DFSA anti-collision protocols in passive RFID environments*. In Industrial Electronics, 2009. IECON'09. 35th Annual Conference of IEEE, pp. 2610–2617. IEEE, 2009. 33
- [Burdet 04] Luc Andre Burdet. *RFID multiple access methods*. In Smart Environments Seminar, Zürich, URL: http://www.vs.inf.ethz.ch/edu/SS2004/DS/reports/06_rfid-mac_report.pdf [cited March 2004], 2004. 27, 28, 29
- [Capetanakis 79] John Capetanakis. *Tree algorithms for packet broadcast channels*. Information Theory, IEEE Transactions on, Vol. 25, No. 5, pp. 505–515, 1979. 22, 37
- [Center 03] Auto-ID Center. *13.56 MHz ISM Band Class 1 Radio Frequency Identification Tag Interface Specification: Candida-*

te Recommendation, Version 1.0. 0. Auto-ID Center, 2003. 34

- [Cha 05] Jae-Ryong Cha and Jae-Hyun Kim. *Novel anti-collision algorithms for fast object identification in RFID system.* In Parallel and Distributed Systems, 2005. Proceedings. 11th International Conference on, volume 2, pp. 63–67. IEEE, 2005. 30, 31, 32
- [Cha 06] Jae-Ryong Cha and Jae-Hyun Kim. *Dynamic framed slotted ALOHA algorithms using fast tag estimation method for RFID system.* In Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE, volume 2, pp. 768–772. IEEE, 2006. 32
- [Chen 07] Wei-Chih Chen, Shi-Jinn Horng and Pingzhi Fan. *An enhanced anti-collision algorithm in RFID based on counter and stack.* In Systems and Networks Communications, 2007. ICSNC 2007. Second International Conference on, pp. 21–21. IEEE, 2007. 48
- [Chen 09] Wen-Tzu Chen. *An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length ALOHA.* Automation Science and Engineering, IEEE Transactions on, Vol. 6, No. 1, pp. 9–15, 2009. 108
- [Chen 10] Yuan-Hsin Chen, Shi-Jinn Horng, Ray-Shine Run, Jui-Lin Lai, Rong-Jian Chen, Wei-Chih Chen, Yi Pan and Terano Takao. *A novel anti-collision algorithm in RFID systems for identifying passive tags.* Industrial Informatics, IEEE Transactions on, Vol. 6, No. 1, pp. 105–121, 2010. 48, 69
- [Chen 11] Yung-Chun Chen, Kuo-Hui Yeh, NaiWei Lo, Yingjiu Li and Enrico Winata. *Adaptive collision resolution for efficient RFID tag identification.* EURASIP Journal on Wireless

BIBLIOGRAFÍA

- Communications and Networking, Vol. 2011, No. 1, pp. 1–14, 2011. 44
- [Cheng 07] Tao Cheng and Li Jin. *Analysis and simulation of RFID anti-collision algorithms*. In Advanced Communication Technology, The 9th International Conference on, volume 1, pp. 697–701. IEEE, 2007. 30
- [Choi 04] Ho-Seung Choi, Jae-Ryon Cha and Jae-Hyun Kim. *Fast wireless anti-collision algorithm in ubiquitous ID system*. In Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th, volume 6, pp. 4589–4592. IEEE, 2004. 53
- [Choi 07] Ji Hwan Choi, Dongwook Lee and Hyuckjae Lee. *Query tree-based reservation for efficient RFID tag anti-collision*. Communications Letters, IEEE, Vol. 11, No. 1, pp. 85–87, 2007. 67, 68, 71
- [Choi 10] Jihoon Choi, Inkyu Lee, Ding-Zhu Du and Wonjun Lee. *FTTP: A fast tree traversal protocol for efficient tag identification in RFID networks*. Communications Letters, IEEE, Vol. 14, No. 8, pp. 713–715, 2010. 40, 67, 78, 166
- [Dobkin 12] Daniel M Dobkin. *The rf in rfid: Uhf rfid in practice*. Newnes, 2012. 9
- [EPC-Global 11] GS1 EPC-Global. *GS1 EPC Tag Data Standard 1.6*. 2011. 44
- [EPCglobal 08] EPC EPCglobal. *Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz–960 MHz Version 1.2.0*. 2008. 32, 34, 61, 69, 77, 159, 194
- [EPCglobal 13] GS1 EPCglobal. *Electronic Product Code*. In <http://www.gs1.org/epcglobal>, 2013. 60

- [Fan 08a] Xiao Fan, InChan Song and KyungHi Chang. *Gen2-based hybrid tag anti-collision Q algorithm using Chebyshev's inequality for passive RFID systems*. In Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on, pp. 1–5. IEEE, 2008. 34
- [Fan 08b] Xiao Fan, InChan Song, KyungHi Chang, D Shin, H Lee, C Pyo and J Chae. *Gen2-based tag anti-collision algorithms using Chebyshev's inequality and adjustable frame size*. ETRI journal, Vol. 30, No. 5, pp. 653–662, 2008. 34
- [Feng 06] Bo Feng, Jin-Tao Li, Jun-Bo Guo and Zhen-Hua Ding. *ID-binary tree stack anticollision algorithm for RFID*. In Computers and Communications, 2006. ISCC'06. Proceedings. 11th IEEE Symposium on, pp. 207–212. IEEE, 2006. 51, 80
- [Finkenzeller 99] Klaus Finkenzeller and Rachel Waddington. *Rfid handbook: Radio-frequency identification fundamentals and applications*. Wiley New York, 1999. 2, 4, 7, 9, 12, 23, 25, 26, 36, 38, 46, 49, 50, 77
- [Floerkemeier 06a] Christian Floerkemeier. *Transmission control scheme for fast RFID object identification*. In Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on, pp. 6–pp. IEEE, 2006. 34
- [Floerkemeier 06b] Christian Floerkemeier and Matthias Wille. *Comparison of transmission schemes for framed ALOHA based RFID protocols*. In Applications and the Internet Workshops, 2006. SAINT Workshops 2006. International Symposium on, pp. 4–pp. IEEE, 2006. 32

BIBLIOGRAFÍA

- [Floerkemeier 07] Christian Floerkemeier. *Bayesian transmission strategy for framed ALOHA based RFID protocols*. In RFID, 2007. IEEE International Conference on, pp. 228–235. IEEE, 2007. 34
- [Geng 08] Shu-qin Geng, Da-ming Gao, Chao Zhu, Ming He and Wuchen Wu. *An improved dynamic framed slotted ALOHA algorithm for RFID anti-collision*. In Signal Processing, 2008. ICSP 2008. 9th International Conference on, pp. 2934–2937. IEEE, 2008. 24
- [Geng 10] Shu-qin Geng, WC Wu, LG Hou and Wang Zhang. *Anti-collision algorithms for multi-tag rfid. radio frequency identification fundamentals and applications, bringing research to practice (chapter 7)*. 2010. 25
- [Héctor Ramos 06] Francisco Maciá Héctor Ramos J. Vicente Berná. *Computación ubicua mediante dispositivos rfid (pág. 59–71 de 'servicios electrónicos para la sociedad de la información. desarrollo de grandes aplicaciones distribuidas sobre internet')*. Servicio de Publicaciones de la Universidad de Alicante, 2006. 4
- [Helmut 09] Woellik Helmut. *A simulation tool for RFID anti-collision algorithms based on ALOHA*. In Systems, Signals and Image Processing, 2009. IWSSIP 2009. 16th International Conference on, pp. 1–4. IEEE, 2009. 33
- [Hush 98] Don R Hush and Cliff Wood. *Analysis of tree algorithms for RFID arbitration*. In Information Theory, 1998. Proceedings. 1998 IEEE International Symposium on, p. 107. IEEE, 1998. 38, 40, 54, 73, 77, 82, 98
- [Hwang 06] Tae-Wook Hwang, Byong-Gyo Lee, Young Soo Kim, Doug Young Suh and Jin Sang Kim. *Improved anti-collision*

- scheme for high speed identification in RFID system.* In Innovative Computing, Information and Control, 2006. ICI-CIC'06. First International Conference on, volume 2, pp. 449–452. IEEE, 2006. 30
- [ISO 13] ISO. *RFID ISO Standards*. In <http://www.iso.org/iso/>, 2013. 60
- [Jacomet 99] Marcel Jacomet, Adrian Ehram, Urs Gehrig *et al.* *Contactless identification device with anticollision algorithm*. In Proceedings of IEEE Conference on Circuits, System, Computers and Communications, pp. 269–273, 1999. 52
- [Jia 10] Xiaolin Jia, Quanyuan Feng and Chengzhen Ma. *An efficient anti-collision protocol for RFID tag identification*. Communications Letters, IEEE, Vol. 14, No. 11, pp. 1014–1016, 2010. 47
- [Jia 12] Xiaolin Jia, Quanyuan Feng and Lishan Yu. *Stability Analysis of an Efficient Anti-Collision Protocol for RFID Tag Identification*. Communications, IEEE Transactions on, Vol. 60, No. 8, pp. 2285–2294, 2012. 47, 54, 66, 82, 136
- [Jianfang 10] Wang Jianfang. *A Novel Anti-collision Backtracking Algorithm Based on Binary-tree Search in UHF*. In Proceedings of the Third International Symposium on Computer Science and Computational Technology (ISCST '10), pp. 212–214. ACADEMY PUBLISHER, 2010. 50
- [Kaplan 85] M Kaplan and Eugene Gulko. *Analytic properties of multiple-access trees*. Information Theory, IEEE Transactions on, Vol. 31, No. 2, pp. 255–263, 1985. 22, 37
- [Katheerja 10] M Katheerja, P Sheik and M Munir. *Analysis of Bit Grouping Algorithm for collision resolution in passive RFID tags*. International Journal of Engineering Science and Technology, Vol. 2, No. 9, pp. 4192–4204, 2010. 51

BIBLIOGRAFÍA

- [Khandelwal 07] Girish Khandelwal, Kyoungwan Lee, Aylin Yener and Semih Serbetli. *ASAP: a MAC protocol for dense and time-constrained RFID systems*. EURASIP journal on Wireless Communications and Networking, Vol. 2007, No. 2, pp. 3–3, 2007. 32
- [Kim 06] Sung Hyun Kim, Min Kyo Shin and PooGyeon Park. *A New Tree-based Tag Anti-collision Protocol for RFID Systems*. In Communications in Computing, pp. 83–86, 2006. 52
- [Kim 07] Sung Hyun Kim and PooGyeon Park. *An efficient tree-based tag anti-collision protocol for RFID systems*. Communications Letters, IEEE, Vol. 11, No. 5, pp. 449–451, 2007. 52
- [Kim 11] Soo-Cheol Kim and Sung Kwon Kim. *An Enhanced Anti-collision Algorithm for EPC Gen2 RFID System*. In Multimedia and Ubiquitous Engineering (MUE), 2011 5th FTRA International Conference on, pp. 293–296, 2011. 34
- [Klair 07a] Dheeraj K Klair, Kwan-Wu Chin and Raad Raad. *An Investigation into the Energy Efficiency of Pure and Slotted Aloha Based RFID Anti-Collision Protocols*. In World of Wireless, Mobile and Multimedia Networks, 2007. WoW-MoM 2007. IEEE International Symposium on a, pp. 1–4. IEEE, 2007. 27, 28, 29, 37, 168
- [Klair 07b] Dheeraj K Klair, Kwan-Wu Chin and Raad Raad. *On the accuracy of RFID tag estimation functions*. In Communications and Information Technologies, 2007. ISCIT'07. International Symposium on, pp. 1401–1406. IEEE, 2007. 32
- [Klair 07c] Dheeraj K Klair, Kwan-Wu Chin and Raad Raad. *On the suitability of framed slotted Aloha based RFID anti-collision protocols for use in RFID-enhanced WSNs*. In Computer Communications and Networks, 2007. ICCCN

2007. Proceedings of 16th International Conference on, pp. 583–590. IEEE, 2007. 37
- [Klair 10] Dheeraj K Klair, Kwan-Wu Chin and Raad Raad. *A survey and tutorial of RFID anti-collision protocols*. Communications Surveys & Tutorials, IEEE, Vol. 12, No. 3, pp. 400–421, 2010. 23, 25, 26, 32, 37, 38, 77
- [Kodialam 06] Murali Kodialam and Thyaga Nandagopal. *Fast and reliable estimation schemes in RFID systems*. In Proceedings of the 12th annual international conference on Mobile computing and networking, pp. 322–333. ACM, 2006. 32
- [Koh 08] Hyunho Koh, Sangki Yun and Hyogon Kim. *Sidewalk: a RFID tag anti-collision algorithm exploiting sequential arrangements of tags*. In Communications, 2008. ICC'08. IEEE International Conference on, pp. 2597–2601. IEEE, 2008. 45
- [La Porta 11] Thomas F La Porta, Gaia Maselli and Chiara Petrioli. *Anti-collision protocols for single-reader rfid systems: Temporal analysis and optimization*. Mobile Computing, IEEE Transactions on, Vol. 10, No. 2, pp. 267–279, 2011. 56
- [Lai 13] Yuan-Cheng Lai, Ling-Yen Hsiao, Hong-Jie Chen, Ching-Neng Lai and Jian-Wei Lin. *A Novel Query Tree Protocol with Bit Tracking in RFID Tag Identification*. IEEE Transactions on Mobile Computing, Vol. 12, No. 10, pp. 2063–2075, 2013. 48, 66, 77, 82, 108
- [Landaluce 12a] H. Landaluce, A. Perallos and I. Angulo. *A simulation tool for RFID EPC Gen2 protocol*. In Iberian Conference on Information Systems and Technologies, CISTI, 2012. 159

BIBLIOGRAFÍA

- [Landaluce 12b] H. Landaluce, A. Perallos, I.J.G. Zuazola and I. Angulo. *Performance analysis of the Slot Counter algorithm in comparison with the real performance of a commercial RFID reader supporting the EPC Class 1 Generation 2 protocol*. In 2012 20th International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2012, 2012. 159
- [Landaluce 13a] H. Landaluce, A. Perallos and I. Angulo. *Influence of managing the number of tag bits transmitted on the query tree RFID collision resolution protocol*. Journal of Communications Software and Systems, Vol. 9, No. 1, pp. 35–43, 2013. 87
- [Landaluce 13b] H. Landaluce, A. Perallos and I.J.G. Zuazola. *A Fast RFID Identification Protocol with Low Tag Complexity*. IEEE Communications Letters, Vol. 17, No. 9, pp. 1704–1706, 2013. 107, 117
- [Landaluce 14] H. Landaluce, A. Perallos and I. Angulo. *Managing the Number of Tag Bits Transmitted in a Bit-Tracking RFID Collision Resolution Protocol*. Sensors (Switzerland), Vol. ACCEPTED FOR PUBLICATION, p. pp, 2014. 122
- [Landt 05] Jeremy Landt. *The history of RFID*. Potentials, IEEE, Vol. 24, No. 4, pp. 8–11, 2005. 6
- [Law 00] Ching Law, Kayi Lee and Kai-Yeung Siu. *Efficient memory-less protocol for tag identification*. In Proceedings of the 4th international workshop on Discrete algorithms and methods for mobile computing and communications, pp. 75–84. ACM, 2000. 40, 41, 54, 70, 77, 80, 82
- [Lee 05] Su-Ryun Lee, Sung-Don Joo and Chae-Woo Lee. *An enhanced dynamic framed slotted ALOHA algorithm for RFID tag*

- identification*. In Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005. The Second Annual International Conference on, pp. 166–172. IEEE, 2005. 34
- [Lee 07] Donghwan Lee, Kyungkyu Kim and Wonjun Lee. *Q+ algorithm: An enhanced RFID tag collision arbitration algorithm*. In Ubiquitous Intelligence and Computing, pp. 23–32. Springer, 2007. 33
- [Liu 05] Leian Liu, Zhenhua Xie, Jingtian Xi and Shengli Lai. *An improved anti-collision algorithm in RFID system*. In Mobile Technology, Applications and Systems, 2005 2nd International Conference on, pp. 5–pp. IEEE, 2005. 50
- [Liu 09] Hsin-Chin Liu and Jhen-Peng Ciou. *Performance analysis of multi-carrier RFID systems*. In Performance Evaluation of Computer & Telecommunication Systems, 2009. SPECTS 2009. International Symposium on, volume 41, pp. 112–116. IEEE, 2009. 24
- [Liu 10] Hsin-Chin Liu. The approaches in solving passive rfid tag collision problems. radio frequency identification fundamentals and applications, bringing research to practice (chapter 3). 2010. 23
- [Loeffler 10] Andreas Loeffler, Fabian Schuh and Heinz Gerhaeuser. *Realization of a CDMA-based RFID System Using a Semi-active UHF Transponder*. In Wireless and Mobile Communications (ICWMC), 2010 6th International Conference on, pp. 5–10. IEEE, 2010. 24
- [Loeffler 11] Andreas Loeffler. Using cdma as anti-collision method for rfid - research & applications. current trends and challenges in rfid (chapter 15). InTech, 2011. 24

BIBLIOGRAFÍA

[Massey 80] James L Massey. *Collision-resolution algorithms and random-access communications*. California Univ., Los Angeles Report, Vol. 1, 1980. 12, 22, 37

[Microelectronics 11] E Microelectronics. *Supertag category protocols. Datasheet*. Rapport technique, <http://www.gaw.ru/doc/EM-Marin/P4022.PDF>, 2011. 27

[Mohammed 11] Usama S Mohammed and Mostafa Salah. *Tag Anti-collision Algorithm for RFID Systems with Minimum Overhead Information in the Identification Process*. RADIOENGINEERING, Vol. 20, No. 1, p. 61, 2011. 68

[Mutti 08] Carlo Mutti and Christian Floerkemeier. *CDMA-based RFID systems in dense scenarios: Concepts and challenges*. In *RFID, 2008 IEEE International Conference on*, pp. 215–222. IEEE, 2008. 24

[Myung 06a] Jihoon Myung and Wonjun Lee. *Adaptive binary splitting: a RFID tag collision arbitration protocol for tag identification*. *Mobile networks and applications*, Vol. 11, No. 5, pp. 711–722, 2006. 39

[Myung 06b] Jihoon Myung and Wonjun Lee. *Adaptive splitting protocols for RFID tag collision arbitration*. In *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, pp. 202–213. ACM, 2006. 39, 42

[Myung 06c] Jihoon Myung, Wonjun Lee and Timothy K Shih. *An adaptive memoryless protocol for RFID tag collision arbitration*. *Multimedia, IEEE Transactions on*, Vol. 8, No. 5, pp. 1096–1101, 2006. 42

[Myung 07] Jihoon Myung, Wonjun Lee, Jaideep Srivastava and Timothy K Shih. *Tag-splitting: adaptive collision arbitration*

- protocols for RFID tag identification*. Parallel and Distributed Systems, IEEE Transactions on, Vol. 18, No. 6, pp. 763–775, 2007. 39, 42
- [Namboodiri 07] Vinod Namboodiri and Lixin Gao. *Energy-Aware Tag Anti-Collision Protocols for RFID Systems*. 2013 IEEE International Conference on Pervasive Computing and Communications (PerCom), Vol. 0, pp. 23–36, 2007. 56
- [Namboodiri 10] Vinod Namboodiri and Lixin Gao. *Energy-aware tag anti-collision protocols for RFID systems*. Mobile Computing, IEEE Transactions on, Vol. 9, No. 1, pp. 44–59, 2010. 56, 73, 168, 169
- [Namboodiri 12] Vinod Namboodiri, Maheesha DeSilva, Kavindya Deegala and Suresh Ramamoorthy. *An extensive study of slotted Aloha-based RFID anti-collision protocols*. Computer Communications, Vol. 35, No. 16, pp. 1955–1966, 2012. 33, 77
- [Pan 09] Lei Pan and Hongyi Wu. *Smart trend-traversal: a low delay and energy tag arbitration protocol for large RFID systems*. In INFOCOM 2009, IEEE, pp. 2571–2575. IEEE, 2009. 43
- [Pan 11] Lei Pan and Hongyi Wu. *Smart trend-traversal protocol for RFID tag arbitration*. Wireless Communications, IEEE Transactions on, Vol. 10, No. 11, pp. 3565–3569, 2011. 43, 66
- [Park 06] Jongho Park, Min Young Chung and Tae-Jin Lee. *Identification of RFID tags in framed-slotted ALOHA with tag estimation and binary splitting*. In Communications and Electronics, 2006. ICCE'06. First International Conference on, pp. 368–372. IEEE, 2006. 58

BIBLIOGRAFÍA

- [Park 07] Jongho Park, Min Young Chung and Tae-Jin Lee. *Identification of RFID tags in framed-slotted ALOHA with robust estimation and binary selection*. Communications Letters, IEEE, Vol. 11, No. 5, pp. 452–454, 2007. 58
- [Peng 07] Qingsong Peng, Ming Zhang and Weimin Wu. *Variant enhanced dynamic frame slotted ALOHA algorithm for fast object identification in RFID system*. In Anti-counterfeiting, Security, Identification, 2007 IEEE International Workshop on, pp. 88–91. IEEE, 2007. 35
- [Pupunwiwat 09] Prapassara Pupunwiwat and Bela Stantic. *Unified Q-ary tree for RFID tag anti-collision resolution*. In Proceedings of the Twentieth Australasian Conference on Australasian Database-Volume 92, pp. 47–56. Australian Computer Society, Inc., 2009. 44
- [Pupunwiwat 10] Prapassara Pupunwiwat and Bela Stantic. *A RFID Explicit Tag Estimation Scheme for Dynamic Framed-Slot ALOHA Anti-Collision*. In Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on, pp. 1–4. IEEE, 2010. 32
- [Qian 10] Chen Qian, Yunhuai Liu, Hoilun Ngan and Lionel M Ni. *ASAP: scalable identification and counting for contactless RFID systems*. In Distributed Computing Systems (ICDCS), 2010 IEEE 30th International Conference on, pp. 52–61. IEEE, 2010. 58, 108
- [Qian 13] Chen Qian, Yunhuai Liu, R.H. Ngan and L.M. Ni. *ASAP: Scalable Collision Arbitration for Large RFID Systems*. Parallel and Distributed Systems, IEEE Transactions on, Vol. 24, No. 7, pp. 1277–1288, 2013. 58
- [Recknagel 11] Stefan Recknagel. *Low-Frequency RFID in a Nutshell*. Report technique, Texas Instruments, 2011. 69

- [Regulations 08] Radio Regulations. *International Telecommunication Union*. Radiocommunication Sector. ITU-R. Geneva, 2008. 9
- [Roberts 75] Lawrence G Roberts. *ALOHA packet system with and without slots and capture*. ACM SIGCOMM Computer Communication Review, Vol. 5, No. 2, pp. 28–42, 1975. 22, 28
- [Sarma 03] Sanjay E Sarma, Stephen A Weis and Daniel W Engels. *RFID systems and security and privacy implications*. In Cryptographic Hardware and Embedded Systems-CHES 2002, pp. 454–469. Springer, 2003. 35
- [Sayeed 09] Md Sakil Ibne Sayeed and Young Soo Kim. *A simple LMS algorithm based smart antenna to solve the reader collision problems in RFID system*. In Information and Multimedia Technology, 2009. ICIMT'09. International Conference on, pp. 426–430. IEEE, 2009. 23
- [Scharfeld 01] Tom Ahlkvist Scharfeld. *An analysis of the fundamental constraints on low cost passive radio-frequency identification system design*. PhD thesis, Massachusetts Institute of Technology, 2001. 10
- [Schoute 83] Frits Schoute. *Dynamic frame length ALOHA*. Communications, IEEE Transactions on, Vol. 31, No. 4, pp. 565–568, 1983. 31, 32, 36, 77
- [Semiconductors 02] Philips Semiconductors. *I-CODE1 System Design Guide*. Rapport technique, Philips Semiconductors, 2002. 29
- [Shih 06] Dong-Her Shih, Po-Ling Sun, David C Yen and Shi-Ming Huang. *Taxonomy and survey of RFID anti-collision protocols*. Computer communications, Vol. 29, No. 11, pp. 2150–2166, 2006. 13, 23, 25, 26, 38

BIBLIOGRAFÍA

- [Shin 07] Jae-Dong Shin, Sang-Soo Yeo, Tai-Hoon Kim and Sung Kwon Kim. *Hybrid tag anti-collision algorithms in RFID systems*. In Computational Science–ICCS 2007, pp. 693–700. Springer, 2007. 58
- [Tang 07] Zhijun Tang and Yigang He. *Research of multi-access and anti-collision protocols in RFID systems*. In Anti-counterfeiting, Security, Identification, 2007 IEEE International Workshop on, pp. 377–380. IEEE, 2007. 25
- [Tran 07] Nhan Tran, Bomson Lee and Jong-Wook Lee. *Development of long-range UHF-band RFID tag chip using Schottky diodes in standard CMOS technology*. In Radio Frequency Integrated Circuits (RFIC) Symposium, 2007 IEEE, pp. 281–284. IEEE, 2007. 7
- [Violino 05] Bob Violino. *A Summary of RFID Standards*. RFID Journal, <http://www.rfidjournal.com/articles/view?1335/>, 2005. 60
- [Vogt 02a] Harald Vogt. *Efficient object identification with passive RFID tags*. In Pervasive Computing, pp. 98–113. Springer, 2002. 32, 34, 49, 56, 108
- [Vogt 02b] Harald Vogt. *Multiple object identification with passive RFID tags*. In Systems, Man and Cybernetics, 2002 IEEE International Conference on, volume 3, pp. 6–pp. IEEE, 2002. 32
- [Wang 06] Liang-Chin Wang and Hsin-Chin Liu. *A novel anti-collision algorithm for EPC Gen2 RFID systems*. In Wireless Communication Systems, 2006. ISWCS'06. 3rd International Symposium on, pp. 761–765. IEEE, 2006. 33
- [Wang 07a] Jianwei Wang, Dong Wang, Yuping Zhao and Timo Korhonen. *Fast anti-collision algorithms in rfid systems*. In Mobile Ubiquitous Computing, Systems, Services and Techno-

- logies, 2007. UBICOMM'07. International Conference on, pp. 75–80. IEEE, 2007. 24, 30
- [Wang 07b] Jianwei Wang, Yuping Zhao and Dong Wang. *A novel fast anti-collision algorithm for RFID systems*. In Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on, pp. 2044–2047. IEEE, 2007. 30
- [Wang 07c] Zhongxiang Wang, Dan Liu, Xiaofang Zhou, Xi Tan, Junyu Wang and Hao Min. *Anti-collision scheme analysis of RFID system*. Auto-ID Labs White Paper, 2007. 32
- [Wang 09] Weidong Wang, Yinghai Zhang, Yuan Sang and Shoufeng Wang. *Analysis of anti-collision algorithms in RFID system*. In Communications Technology and Applications, 2009. ICCTA'09. IEEE International Conference on, pp. 58–62. IEEE, 2009. 25
- [Want 04] Roy Want. *Enabling ubiquitous sensing with RFID*. Computer, Vol. 37, No. 4, pp. 84–86, 2004. 6
- [Want 06] Roy Want. *An introduction to RFID technology*. Pervasive Computing, IEEE, Vol. 5, No. 1, pp. 25–33, 2006. 9
- [Wasikon 08] Siti Mahfuzoh Wasikon and Zurinah Suradi. *A framework of tag anti-collision algorithm for fast identification in RFID system*. In Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on, pp. 1019–1022. IEEE, 2008. 35
- [Weiser 91] Mark Weiser. *The computer for the 21st century*. Scientific american, Vol. 265, No. 3, pp. 94–104, 1991. 4
- [Wieselthier 89] Jeffrey E Wieselthier, Anthony Ephremides and Larry A Michaels. *An exact analysis and performance evaluation*

BIBLIOGRAFÍA

- of framed ALOHA with capture.* Communications, IEEE Transactions on, Vol. 37, No. 2, pp. 125–137, 1989. 31
- [Woodland 52] Norman Joseph Woodland and Bernard Silver. *Classifying apparatus and method*, October 7 1952. US Patent 2,612,994. 2
- [Wu 10] Haifeng Wu and Yu Zeng. *Bayesian tag estimate and optimal frame length for anti-collision aloha RFID system.* Automation Science and Engineering, IEEE Transactions on, Vol. 7, No. 4, pp. 963–969, 2010. 32, 108
- [Wu 13] Haifeng Wu, Yu Zeng, Jihua Feng and Yu Gu. *Binary Tree Slotted ALOHA for Passive RFID Tag Anticollision.* IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 1, pp. 19–31, 2013. 58, 108
- [Yan 08] Xinqing Yan, Zhou ping Yin and Xiong You-lun. *A comparative study on the performance of the RFID tag collision resolution protocols.* In Future Generation Communication and Networking, 2008. FGCN'08. Second International Conference on, volume 1, pp. 469–472. IEEE, 2008. 166
- [Yan 09] Xinqing Yan and Guiliang Zhu. *An enhanced query tree protocol for RFID tag collision resolution with progressive population estimation.* In Mobile Adhoc and Sensor Systems, 2009. MASS'09. IEEE 6th International Conference on, pp. 935–940. IEEE, 2009. 43
- [Yan 10] Xinqing Yan and Xuemei Liu. *Evaluating the Energy Consumption of RFID Tag Collision Resolution Protocols.* In Mobile Ad-hoc and Sensor Networks (MSN), 2010 Sixth International Conference on, pp. 215–221, 2010. 37, 166

- [Yan 11a] Xinqing Yan and Yan Xu. *Analytic Properties of the Binary Tree Based Multiple Access Protocol with Application to RFID Tag Collision Resolution*. In Mobile Ad-hoc and Sensor Networks (MSN), 2011 Seventh International Conference on, pp. 376–382. IEEE, 2011. 37
- [Yan 11b] Xinqing Yan and Guiliang Zhu. *EBQP: Enhanced Binary Query Protocol for RFID tag collision resolution with progressive population estimation*. International Journal of Ad Hoc and Ubiquitous Computing, Vol. 8, No. 1, pp. 78–84, 2011. 43
- [Yang 10] Lei Yang, Jinsong Han, Yong Qi, Cheng Wang, Yunhao Liu, Ying Cheng and Xiao Zhong. *Revisiting tag collision problem in RFID systems*. In Parallel Processing (ICPP), 2010 39th International Conference on, pp. 178–187. IEEE, 2010. 12, 37
- [Yang 11] Ching-Nung Yang and Jyun-Yan He. *An effective 16-bit random number aided query tree algorithm for RFID tag anti-collision*. Communications Letters, IEEE, Vol. 15, No. 5, pp. 539–541, 2011. 71
- [Yeh 07] Ming-Kuei Yeh and Jehn-Ruey Jiang. *Adaptive k-way splitting and pre-signaling for RFID tag anti-collision*. In Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE, pp. 40–45. IEEE, 2007. 39
- [Yeh 11] Ming-Kuei Yeh and Jehn-Ruey Jiang. *Parallel Splitting for RFID tag anti-collision*. International Journal of Ad Hoc and Ubiquitous Computing, Vol. 8, No. 4, pp. 249–260, 2011. 39
- [Yeo 10] Woon-Young Yeo and Gyung-Ho Hwang. *Efficient anti-collision algorithm using variable length ID in RFID sys-*

BIBLIOGRAFÍA

- tems*. IEICE Electronics Express, Vol. 7, No. 23, pp. 1735–1740, 2010. 45
- [Yu 08] Jiexiao Yu, Kaihua Liu and Ge Yan. *A novel RFID anti-collision algorithm based on sdma*. In Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference on, pp. 1–4. IEEE, 2008. 23
- [Zheng 11] Yuanqing Zheng, Mo Li and Chen Qian. *PET: Probabilistic estimating tree for large-scale RFID estimation*. In Distributed Computing Systems (ICDCS), 2011 31st International Conference on, pp. 37–46. IEEE, 2011. 108
- [Zhou 03] Feng Zhou, Dawei Jin, Chenling Huang and Min Hao. *Optimize the power consumption of passive electronic tags for anti-collision schemes*. In ASIC, 2003. Proceedings. 5th International Conference on, volume 2, pp. 1213–1217. IEEE, 2003. 47, 68
- [Zhou 04] Feng Zhou, Chunhong Chen, Dawei Jin, Chenling Huang and Hao Min. *Evaluating and optimizing power consumption of anti-collision protocols for applications in RFID systems*. In Proceedings of the 2004 international symposium on Low power electronics and design, pp. 357–362. ACM, 2004. 68
- [Zhu 11] Lei Zhu and T-SP Yum. *A critical survey and analysis of RFID anti-collision mechanisms*. Communications Magazine, IEEE, Vol. 49, No. 5, pp. 214–221, 2011. 25, 32, 37