Evaluación de un mecanismo MAC p-persistente para redes inalámbricas de sensores con tráfico por eventos

J. Vales Alonso, E. Egea López, J. L. Sieiro Lomba, Mª V. Bueno Delgado, J. García Haro
Departamento de Tecnologías de la Información y las Comunicaciones
Universidad Politécnica de Cartagena, ETSI de Telecomunicación
Plaza del Hospital, 1. Campus Muralla del Mar. 30202 - Cartagena (Spain)
E-mail: {javier.vales, esteban.egea, joseluis.sieiro, mvictoria.bueno, joang.haro}@upct.es

***Resumen*—En las redes de sensores inalámbricas con tráfico basado en eventos los nodos solamente transmiten información si el nivel alcanzado por las magnitudes físicas bajo monitorización ha alcanzado un punto crítico que obliga a lanzar una señal de alarma. En estas redes, el tráfico exhibe una alta correlación espacial, ya que es muy probable que nodos vecinos detecten y traten de notificar los mismos eventos físicos. Por lo tanto, la probabilidad de que colisione una transmisión aumenta, ya que los nodos vecinos entran simultáneamente a la contienda, y consecuentemente podría aumentar el retardo de notificación del evento y el gasto energético asociado. Este efecto es el contrario al deseable en redes con tráfico por eventos. En este trabajo proponemos el uso de un mecanismo p-persistente en el nivel de acceso al medio. El objetivo es reducir las colisiones, ahorrar energía y reducir la latencia de notificación. En este trabajo calculamos el p óptimo para un despliegue de red verosimil y describimos una implementación experimental de nuestro mecanismo. Tanto los cálculos teóricos, como los resultados prácticos, muestran una importante mejora, especialmente en términos energéticos, incluso en comparación con mecanismos ideales (sin colisiones).**

I. Introducción

Las redes de sensores inalámbricas son redes *ad-hoc* destinadas a la monitorización de diversos ambientes, y están basadas en dispositivos sensores de bajo coste y pequeño rango de transmisión (típicamente inferior a 100 metros). Los datos son transmitidos para su procesado a nodos especiales, denominados sumideros (*sinks*), haciendo uso de encaminamiento multisalto entre los propios nodos sensores. En las redes de sensores se preveen fundamentalmente dos tipos de tráfico [1]:

1. Tráfico de *datos periódico*, donde todos los nodos transfieren periódicamente las muestras obtenidas a los sumideros.
2. Tráfico *basado en eventos*, en este caso sólo se enviarán mensajes a los sumideros cuando se verifiquen ciertas condiciones sobre las magnitudes bajo control, por ejemplo: la temperatura es mayor que 50° centigrados, la presión es inferior a 100 kilo-pascal, etc.

En este trabajo nos centraremos exclusivamente en el último caso, de tráfico por eventos. Denominaremos "evento" al proceso físico que dispara la transmisión del mensaje, y "notificación" al propio mensaje enviado por los nodos para señalar dicho evento. Habitualmente las redes con tráfico basado en eventos se asocian con aplicaciones sensibles al retardo, y demandan retardos de notificación lo más pequeños posibles.

Por otro lado, la duración de las baterías sigue siendo un condicionante fundamental para el diseño de las redes de sensores, y en muchos casos, en el funcionamiento de los protocolos. En el nivel de acceso al medio (MAC), los sensores deben activar sus radios sólo cuando sea estrisctamente necesario, para evitar gastos enegéticos innecesarios. De hecho, escuchar el medio en los periodos inactivos se considera la mayor fuente de ineficiencia energética, y las estrategias para combatirla actúan durmiendo los dispositivos en dichos periodos. Para ello, es necesario que todos los nodos de la red trabajen en un ciclo coordinado de actividad y sueño (ver figura 1), lo que conduce a unos periodos de actividad ranurados en el tiempo. Existen diversas propuestas de algoritmos de contienda adaptados a este tipo de funcionamiento, como por ejemplo [2], [3], [4]. En este caso, el intercambio de datos sólo puede tener lugar durante la parte activa del periodo. Dentro de ella, los nodos que deseen transmitir seleccionan aleatoriamente (habitualmente según una variable aleatoria uniforme) el momento en que comenzarán su transmisión. Los posibles momentos de comienzo de transmisión son un conjunto discreto, por lo que utilizaremos la terminología de microranura para referirnos a cada uno de los $k = 1, \ldots, K$ periodos de comienzo de la transmisión. Un nodo comenzará la transmisión de sus datos si antes de su microranura elegida (k) ningún otro nodo ha ocupado el medio. Evidentemente, las colisiones ocurrirán solamente si la primera microranura de transmisión es escogida por dos o más nodos.

Si el tráfico muestra un patrón independiente entre los distintos nodos de la red (una hipótesis clásica en la literatura), el funcionamiento basado en ranuras temporales reduce notablemente el consumo y proporciona un método de acceso al medio sumamente flexible. Sin embargo, en aquellas redes donde el tráfico está orientado a eventos, esta hipótesis no se verifica. En este caso las transmisiones tendrán una alta correlación espacial, es decir, será muy probable que nodos cercanos capturen el mismo evento físico y que, consecuente-

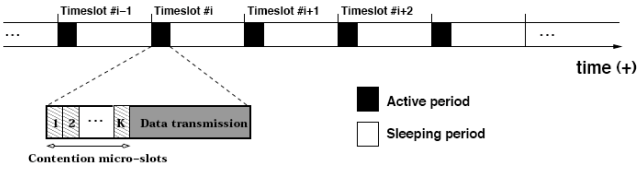

Figura 1. Los nodos se coordinan en periodos de actividad y sueño, lo que da lugar a un funcionamiento organizado en ranuras temporales

mente, intenten notificarlo simultánemente. Además, puesto que todas las transmisiones se agrupan en la parte activa de la trama, la probabilidad de colisión aumenta, y en consecuencia también lo hace el retardo de notificación, justo al reves de lo deseable.

En este artículo proponemos una modificación de los protocolos de contienda basados en operación ranurada en el tiempo para minimizar el retardo medio de notificación y reducir el consumo energético. La modificación está basada en la introducción de un mecanismo p-persistente en el nivel de acceso al medio. Esto es, en cada ranura temporal, los nodos que desean transmitir se activan (con probabilidad p), o continuan durmiendo (con probabilidad $1-p$). En este trabajo calculamos la persistencia p óptima en el sentido de minimizar el retardo de recepción de la primera notificación del evento. En realidad, el efecto conseguido mediante esta modificación es doble, ya que al seguir durmiendo los nodos consumen mucha menos energía que con los mecanismos convencionales. Por ejemplo, si un evento es captado por 15 nodos sensores, con el algoritmo clásico en el mejor caso (con una transmisión exitosa al primer intento) los 15 nodos consumen la energía correspondiente a un periodo activo. En el caso p-persistente, en media sólo $15 \times p$ de los nodos se despetarían y consumirían energía. Para un valor bajo de p, las ganancias energéticas pueden resultar considerables. De hecho, como mostraremos más adelante, el ahorro es mayor cuando el número de nodos que desean transmitir simultáneamente crece. En este trabajo se proporciona un método práctico para el cálculo del valor óptimo de p, así como diferentes resultados analíticos y experimentales que demuestran las ventajas del algoritmo propuesto frente a la aproximación clásica (1-persistente).

El resto de este trabajo se organiza como sigue: La sección II destaca los trabajos relacionados. La sección III está dedicada al cálculo de la persistencia óptima en función del número de nodos presentes y otros parámetros de configuración de la red. La siguiente sección, IV, presenta los resultados de ahorro energético y en terminos de latencia calculados a partir de los resultados teóricos. La sección V presenta los experimentos prácticos realizados para contrastar los estudios teóricos de las secciones anteriores. En los experimentos se emplearon nodos sensores de la familia Mica2 corriendo el protocolo S-MAC. Por último, la sección VI resume los principales resultados alcanzados en este trabajo.

Nota: En las secciones siguientes se empleará la siguiente notación:

- Las probabilidades se denotarán como Pr[Evento].
- Las variables aleatorias (va) se notarán como $\boldsymbol{x}$.
- Los valores medios se notarán como $\overline{\boldsymbol{x}}$.

II. Trabajos relacionados

La gran cantidad de protocolos MAC para redes de sensores que han sido propuestos en los últimos años refleja el interés que este campo ha adquirido. La mayoría de los desarrollos están enfocados al problema energético, que a menudo supedita otras cuestiones consideredas de menor importancia, como por ejemplo la minimización de la latencia de entrega de los paquetes.

Los protocolos MAC para redes de sensores se ubican en las categorías tradicionales de protocolos de contienda y protocolos de acceso por división en el tiempo. Los protocolos de contienda clásicos son simples, escalables y extremadamente flexibles, su mayor problema (al considerar su aplicación a redes de sensores) es que para recibir datos es necesario estar continuamente a la escucha (ya que no se conoce de antemano cuando se transmitirán), por tanto, el eficiencia energética se ve severamente comprometida. Por ello, las propuestas para WSN de protocolos basados en acceso multiple con detección de portadora (CSMA) se diseñan siempre incluyendo mecanismos que minimicen los tiempos de escucha vacia. Como se indicó en el apartado anterior, la solución más habitual es hacer que los nodos apaguen sus radios durante los instantes de inactividad. El protocolo S-MAC [2] fue el primero que propuso organizar los nodos en ciclos periódicos de actividad y sueño (*listen/sleep*). Este mecanismo hace que los nodos se activen durante un corto periodo de actividad (tiempo de escucha) y los duerme (es decir, apaga la etapa de radiotransmisión) durante el resto del tiempo (tiempo de sueño). Al ratio entre la duración de los tiempos de escucha y de sueño se le denomina ciclo de trabajo (*duty cycle*). Evidentemente, si se decremente éste, la duración de las baterías se incrementará proporcionalmente, a expensas de un menor caudal y mayor retardo. Los nodos coordinan el sueño con sus vecinos mediante la transmisión periódica de paquetes de sincronización *broadcast* (SYNC *packets*). Así, la parte de actividad del ciclo se divide en dos: una parte de sincronización y otra para el intercambio de datos. Durante el intercambio de datos, los nodos usan una variante del CSMA/CA de IEEE802.11 [5]: la ventana de contienda es de tamaño fijo. Además, los paquetes RTS/CTS evitan el problema de los nodos ocultos. Con estas características el protocolo S-MAC reduce notablemente el desperdicio energético, pero el uso de periodos constantes de sueño puede incrementar la latencia bajo tasas de tráfico variables.

El problema de minimización de la latencia bajo tráfico por eventos ha sido también tratado en [6], donde los autores proponen el uso de un CSMA no persistente en el que la microranura se seleccina con una distribución de probabilidad especial no uniforme (la denotaremos como $\boldsymbol{f}$, es decir Pr[“Comenzar la transmisión en la microranura j”] $= f(j)$)

que maximiza la probabilidad de éxito en la transmisión cuando N estaciones compiten. Por tanto, la distribución ($\boldsymbol{f}$) minimiza la latencia de la primera notificación. Sin embargo, calcular esta distribución óptima requiere conocer el número real de contendientes (N), lo que habitualmente no es factible. Para evitar este problema, los autores proporcionan una distribución aproximada, llamada Sift [7].

Nuestro mecanismo es sub-óptimo en comparación con la distribución propuesta en [6], pero tiene como ventaja que no necesita activar todos los nodos durante la fase de actividad, por lo que se ahorra energía en comparación con el primero.

III. Análisis para el cálculo de la persistencia óptima

En esta sección calculamos el valor óptimo para el mecanismo p-persistente descrito en la sección I. El criterio de optimalidad es la minimización del retardo de la primera notificación del evento (nótese que si N nodos capturan un evento todos intentarán notificarlo, es decir, se intentarán enviar N notificaciones independientes). El tiempo de notificación se calculará como el tiempo desde que el evento físico se produce hasta que el mensaje de notificación llega al nodo sumidero.

Antes de proceder con el análisis, es necesario asumir un modelo de red adecuado, ya que el valor óptimo de p dependerá del número esperado de nodos que capturen un evento. Para nuestro modelo supondremos que el número de nodos distribuidos se escogerá de tal modo que la probabilidad de perder un evento (porque no hay ningún nodo cercano que lo capture) es inferior a un umbral determinado. Este umbral deberá escogerse como parámetro de configuración de la red, y aplicaciones de seguridad o críticas demandarán umbrales extremadamente bajos. A continuación se detallan las asunciones que realizaremos para nuestro modelo.

III-A. Modelo de red

- Asumiremos que la red está formada por n nodos, distribuidos aleatoriamente en un área de tamaño A. Por tanto el número de nodos que capturarán cada evento será una variable aleatoria, que denotaremos como $\boldsymbol{N} \leq n$. Consideraremos que la distribución de los nodos es uniforme en el área objetivo. Para calcular $\boldsymbol{N}$, asumiremos que los eventos serán capturados por un nodo si se producen a una distancia de él inferior a r (es decir, la zona de captura es un circulo de radio r y centrado en el lugar donde se produce el evento), y que el radio de comunicación es R. El tamaño del área donde se produce el evento es entonces $a = \pi r^2 \ll A$. Suponiendo que $2r < R$, todos los nodos que capturen el evento competirán para transmitirlo (ya que se solapan sus zonas de cobertura radio) hacia el sumidero. Podemos calcular entonces la función de masa de probabilidad de $\boldsymbol{N}$, que vendrá dada por la ecuación (1).

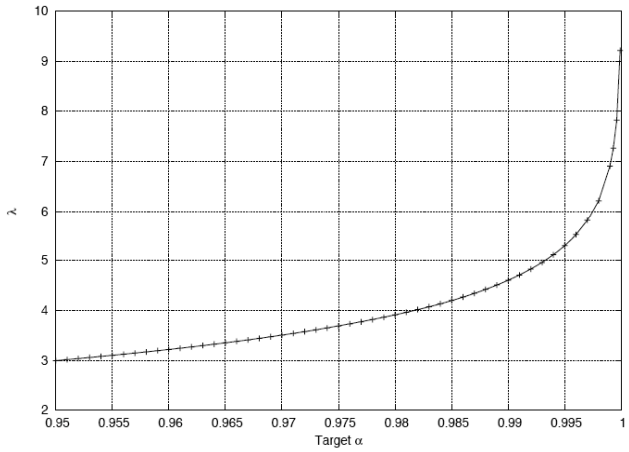

Figura 2. λ versus target α

$$\begin{aligned} \Pr[\boldsymbol{N} = i] &= \binom{n}{i}\left(\frac{a}{A}\right)^i\left(1-\frac{a}{A}\right)^{n-i} \approx \\ &\approx \frac{(n\frac{a}{A})^i e^{-n\frac{a}{A}}}{i!} \\ &\Rightarrow \boldsymbol{N} \approx \boldsymbol{Poisson}\{n\frac{a}{A}\} \end{aligned} \quad (1)$$

En la ecuación anterior hemos empleado la aproximación de la distribución binomial como distribución poissoniana. Esta aproximación es correcta si $\frac{a}{A} \ll n\frac{a}{A} \ll n$, condición que se cumple claramente en una red de sensores ya que n será un número muy grande y el ratio $\frac{a}{A}$ muy pequeño. A partir de la nomenclatura característica de una distribución de Poisson podemos definir $\lambda = n\frac{a}{A}$. En nuestro modelo λ debe ser interpretado como el *número medio de sensores que reciben un evento*.
Además, es sencillo establecer una relación entre λ y la probabilidad de cobertura de un evento (que denotaremos como α), esto es, la probabilidad de que al menos un nodo capture el evento,

$$\begin{aligned} 1-\alpha = \Pr[\boldsymbol{N}=0] \Rightarrow \alpha = 1-e^{-\lambda} \Rightarrow \\ \Rightarrow \begin{cases} \lambda = -ln(1-\alpha) \\ n = -\frac{A}{a}ln(1-\alpha) \end{cases} \end{aligned} \quad (2)$$

A modo de ejemplo, si se considera un despliegue en un área A de un quilómetro cuadrado, el radio de captura de un evento es de 20 metros y se desea una probabilidad de pérdida de eventos de $1-\alpha$=0.001, obtendríamos $lambda$=6.9, y $n \approx 5500$ nodos. Con los datos de este ejemplo se puede verificar fácilmente que estamos en las condiciones de la aproximación poissoniana, ya que $\frac{a}{A} = 0{,}00126 \ll n\frac{a}{A} = 6{,}9 \ll n = 5500$.
Así, para un nivel de cobertura de eventos objetivo podemos calcular el número de sensores que habrá que desplegar, y la λ asociada a dicha configuración de red.

La figura 2 muestra el valor de λ *versus* el nivel de cobertura elegido desde el 95 % hasta casi un 100 %.

- Como segunda asunción sobre el modelo de red, supondremos que el tiempo de notificación es mucho menor que el tiempo entre-eventos. Es decir, un evento nuevo se transmitirá en una red "vacia" (sin otro tráfico de aplicación). De este modo, en nuestro estudio no tendremos que considerar el efecto de eventos físicos simultáneos. Esta asunción es coherente con el hecho de que las posibles alarmas a transmitir suceden de modo esporádico (en caso contrario, posiblemente una red con transmisión periódica de datos ofrecería mejores prestaciones).
- Supondremos también que existe un mecanismo de apropiación que permite que un nodo puente (un nodo en la ruta desde el nodo que captura el evento hasta el nodo sumidero) transmita los datos fuera de contienda. Añadir este mecanismo es lógico ya que el retardo se minimiza si un nodo puente gana siempre el canal. Dicho sistema de apropiación puede ser implementado fácilmente por medio de un retardo variable antes de la etapa de detección de portadora, como por ejemplo el uso que se hace de los retardos SIFS y DIFS en IEEE 802.11 [5]. En conclusión, minimizar el retardo de la primera transmisión es equivalente a minimizar el retardo extremo a extremo (nodo a sumidero) de la primera notificación.

I-B. Retardo medio de notificación

Al comienzo de cada periodo de actividad, todos los nos que capturen un evento (N) tratarán de transmitir la otificación pertinente. Sea K el número de microranuras e transmisión, y denotemos $\boldsymbol{q}$ a la variable aleatoria que elecciona la ranura de contienda inicial. Supondremos que la icroranura de transmisión se selecciona uniformemente, es ecir, $\boldsymbol{q} = \boldsymbol{Uniform}\{1, K\}$. Denotermos $q_k = \Pr[\boldsymbol{q} = k] =$ para todo $k \in [1, \ldots, K]$.

La probabilidad de éxito en la contienda (gana un nodo) es nción del número de competidores (N). Evidentemente si $= 1$ la probabilidad es 1, para $N \geq 2$ está dada por (ver 5]):

$$\begin{aligned}\pi(N) &= N \sum_{s=1}^{K-1} q_s (1 - \sum_{r=1}^{s} q_r)^{(N-1)} = \\ &= \frac{N}{K} \sum_{s=1}^{K-1} (1 - \frac{s}{K})^{(N-1)}\end{aligned} \tag{3}$$

Ahora, consideremos la introducción del mecanismo persis- nte, en este caso, la probabilidad de éxito vendrá dada por expresión (4).

$$\pi(N, p) = \sum_{c=1}^{N} \pi(c) \binom{N}{c} p^c (1-p)^{N-c} \tag{4}$$

Dado un número N de nodos contendientes, podemos definir variable aleatoria $\boldsymbol{T}_N$ como el "número de ranuras tempo- les hasta que uno de los N nodos gane el canal y transmita". Su función de masa de probabilidad viene dada por la ecuación (5).

$$\Pr[\boldsymbol{T}_N{=}j] = \pi(N,p)(1 - \pi(N,p))^{j-1} \tag{5}$$

para cada $j \geq 0$.

En las últimas expresiones hemos considerado el parámetro N como una constante. Sin embargo, como discutimos anteriormente, es en realidad una variable aleatoria $\boldsymbol{N}$. De las ecuaciones (1) y (5) podemos calcular el número medio de ranuras temporales hasta que un nodo consiga ganar la contienda ($\overline{\boldsymbol{T}}$), para un número aleatorio de contendientes $\boldsymbol{N}$, que es función del nivel de persistencia p escogido. A saber,

$$\boldsymbol{T(p)} = \sum_{i=1}^{n} \overline{\boldsymbol{T_i}} \Pr[\text{N=i}] \tag{6}$$

Debemos remarcar que la expresión anterior proporciona el número medio de ranuras necesarias para transmitir la *primera* notificación.

III-C. Cálculo del p óptimo

Del análisis anterior, podemos expresar nuestro criterio de minimización como,

$$\begin{aligned} p &= \arg\min_p \{\overline{\boldsymbol{T(p)}}\} = \\ &= \arg\min_p \left\{ \sum_{i=1}^{n} \overline{\boldsymbol{T_i}} \Pr[\text{N=i}]\} \right\} = \\ &= \arg\min_p \left\{ \sum_{i=1}^{n} \left(\sum_{j=1}^{\infty} j \Pr[\boldsymbol{T}_i = j] \right) \Pr[\text{N=i}] \right\} \end{aligned}$$

Simplificando,

$$\begin{aligned}\overline{\boldsymbol{T(p)}} &= \frac{1}{p} \sum_{i=1}^{n} \left(\frac{1}{\sum_{c=1}^{i} \pi(c) \binom{i}{c} p^{c-1} (1-p)^{i-c}} \frac{\lambda^i e^{\lambda}}{i!} \right) \\ &= \frac{f(p)}{p}\end{aligned} \tag{7}$$

Derivando la expresión anterior e igualandola a cero para calcular el mínimo, obtenemos,

$$\frac{d\overline{\boldsymbol{T(p)}}}{dp} = \frac{d(f(p)/p)}{dp} = 0 \Rightarrow p = \frac{f(p)}{f'(p)} = g(p) \tag{8}$$

De la ecuación (8) el valor óptimo de p puede ser calculado utilizando el teorema del punto fijo de Banach [8]. Este teorema establece que para una función contractiva $g(x)$ de un subconjunto cerrado F⊆E en un espacio F, siendo E y F espacios de Banach, existe $x \in$ F tal que $g(x) = x$. En nuestro problema la función contractiva es $g(p) = \frac{f(p)}{f'(p)}$. Además, el teorema proporciona un método constructivo para calcular el punto fijo: se establece un p_0 inicial y se calcula $p_{i+1} = g(p_i) = \frac{f(p_i)}{f'(p_i)}$. La sucesión p_i converge al p óptimo. Además, ya que el teorema garantiza que el punto fijo es único, es el óptimo buscado.

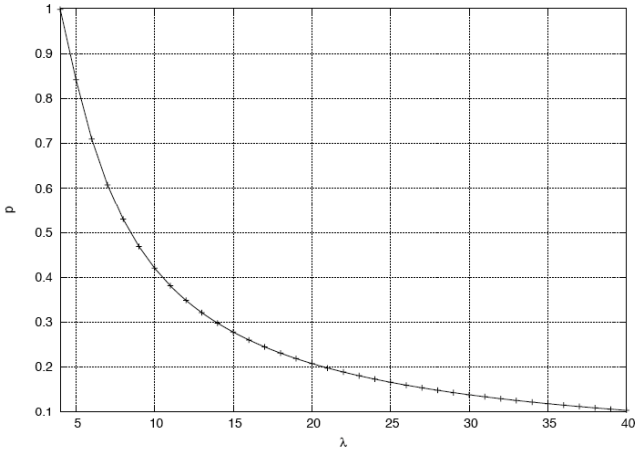

Figura 3. p óptimo versus λ

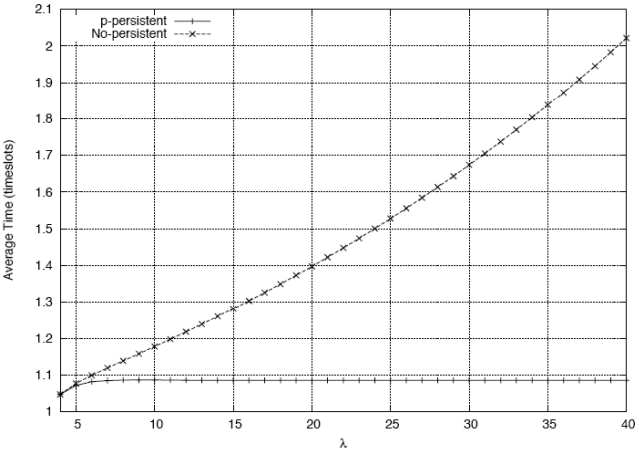

Figura 4. $\overline{\boldsymbol{T(p)}}$ versus λ para el valor de p óptimo

IV. Resultados

La figura 3 muestra el p óptimo obtenido mediante el teorema del punto fijo descrito en el apartado anterior. Para valores de λ menores que 3.5 se obtiene $p > 1$, que no tienen una correspondencia física en el mecanismo propuesto. En estos casos, debemos seleccionar $p = 1$ para obtener las mejores prestaciones. Es decir, que el mecanismo se comporta como un algoritmo 1-persistente. Sin embargo, como comentabamos anteriormente en cualquier despliegue práctico el número de nodos a utilizar debe ser seleccionado de tal modo que la probabilidad de perder eventos sea suficientemente baja (típicamente deberá ser menor al 1 %). Observando la figura 2, vemos que estos casos tendrán asociada una $\lambda > 4$. Para estos valores de λ la persistencia óptima será $p < 1$, y el uso del mecanismo propuesto cobra sentido.

En la figura 4 se representa el número medio de periodos de activad (número de ranuras temporales) necesarios para la notificación del primer evento para el caso p-persistente y el 1-persistente (i.e., el mecanismo de acceso convencional).

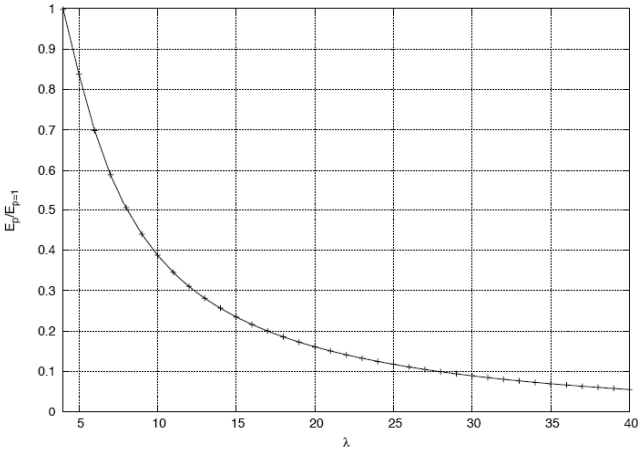

Figura 5. Consumo energético relativo del mecanismo p-persistente comparado con el mecanismo 1-persistente, en función de λ, para el valor de p óptimo

Como se aprecia, la utilización de la persistencia controla y ajusta la contienda, manteniendo el retardo de notificación casi constante. Por el contrario, con el esquema 1-persistente, el retardo se incrementa a medida que aumenta el número de contendientes (mayor λ). De hecho, el incremento puede ser mayor, dependiendo del tamaño de la ventana de contienda (número de microranuras, K) Los resultados mostrados han sido calculados para $K = 32$.

Por último, el ahorro energético puede ser calculado aproximadamente como sigue: supongamos que un nodo consume en su parte activa una energía G. Si no se usa la persistencia, la energía consumida para la transmisión de una notificación es $E_{p=1} \approx \overline{\boldsymbol{T(1)}}\lambda G$ (recordemos que λ representa el número medio de nodos que captan un evento). En el caso del mecanismo p-persistente la energía consumida es $E_p \approx p\overline{\boldsymbol{T(p)}}\lambda G$. Nótese que el tiempo medio $\overline{\boldsymbol{T(p)}}$ es diferente en cada caso. Así, el beneficio es doble ya que la energía es reducida porque hay nodos que no se despiertan y porque se necesitan menos ranuras para transmitir la notificación. Como se muestra en la figura 5, utilizando el mecanismo p-persistente la energía consumida puede ser tan sólo un 5 % de la necesaria con la aproximación convencional 1-persistente para redes de muy alta densidad. Para escenarios más realistas, como el proporcionado en la sección III-A, con $\lambda = 6{,}9$, los ahorros están en torno al 50 %.

V. Pruebas experimentales

Adicionalmente, para verificar su viabilidad práctica, hemos implementado el mecanismo p-persistente sobre un protocolo MAC de funcionamiento ranurado: el S-MAC [2]. Para ello utilizando como nodos sensores equipos de la familia Mica2. Nuestro objetivo era fundamentalmente demostrar la factibilidad práctica de la implementación del mecanismo p-persistente en hardware real empleado en redes de sensores. Evidentemente, un despliegue con un modelo de red como el indicado en la sección III-A es inviable en condiciones

de laboratorio (debido a la gran superficie y número de sensores necesarios. Por ello, en estas pruebas empleamos un modelo simplificado cuyo objetivo era estudiar la viabilidad del mecanismo como procedimiento potencial de ahorro energético.

Como se indicó en II, S-MAC es un protocolo diseñado especialmente para redes de sensores y que coordina a los nodos en un ciclo de actividad y sueño. S-MAC usa una ventana de contienda de tamaño fijo con $K = 32$ microranuras seleccionadas uniformemente, y que permite integrar fácilmente la modificación p-persistente. Los nodos Mica2 se usan ampliamente como solución hardware para WSN, son desarrollados por Crossbow Inc. y están basados en una etapa de radiocomunicación Chipcon CC1000 que trabaja en la banda de 868/916 MHz. Los nodos Mica2 funcionan bajo el sistema operativo TinyOS [9] y para su programación se emplea como lenguaje de programación el nesc [10].

En nuestros experimentos (ver figura 6) teníamos:

- Un nodos (*coordinador*) que actuaba como sumidero y que además lanzaba los eventos. Para lanzar eventos el coordinador incorporaba una placa de sensores MTS300, que entre otras capacidades, cuenta con un detector y emisor de tonos (ambos a 4 KHz).
- Hasta 8 nodos sensores, que incorporaban también placas MTS300.

En los experimentos:

1. El coordinador emitía un tono a 4 KHz durante medio segundo.
2. Todos los nodos lo detectaban vía el hardware detector de tonos.
3. Todos los nodos entraban en una fase de contienda para transmitir al coordinador la notificación usando el mecanismo p-persistente.
4. Cuando la primera notificación era recibida correctamente por el coordinador establecía un *flag* en la siguiente trama de sincronismo (en nuestra modificación todos los nodos permanecen activos durante la parte de recepción de sincronismos). para evitar notificaciones posteriores.

El experimento anterior se realizó para un número diferente de nodos (n=2,4,8) y de niveles de persistencia (p=0.2,0.5,0.8). La figura 6 muestra el montaje con $n = 8$. Debemos señalar que en nuestros experimentos *todos los nodos capturaban los eventos*, y por tanto el tráfico no puede ser modelado como un patrón poissoniano, y por tanto las fórmulas obtenidas en las secciones anteriores no pueden ser empleadas. Para cada configuración obteniamos 30 muestras del retardo de notificación (medido en número de ranuras). El valor medio obtenido se muestra en la tabla I. Además, mostramos en esta tabla el ahorro esperado de energía para cada configuración, en comparación con un mecanismo de acceso perfecto (es decir, suponiendo que no hay colisiones y las notificaciones se consiguen transmitir siempre al primer intento). En este caso, el ahorro relativo por nodo puede ser calculado como $1 - pT$, siendo T el retardo medio de notificación (en número de ranuras) y p el nivel de persistencia. Dicho valor de ahorro relativo es el indicado en los resultados de la tabla I. Como se aprecia, se ahorra una considerable cantidad de energía, confirmando los resultados teóricos presentados en la sección III.

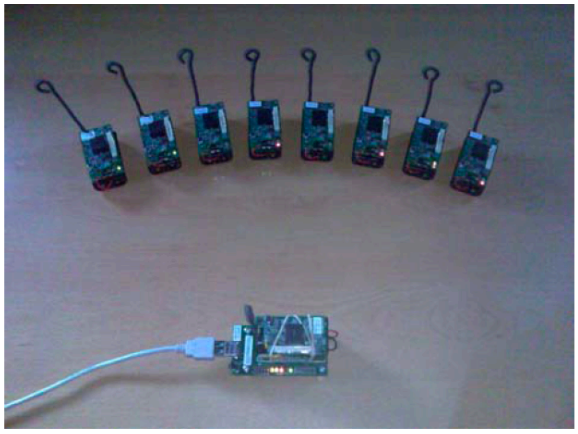

Figura 6. El montaje experimental contiene un nodo sumidero (abajo) y hasta ocho nodos que reciben eventos acústicos (arriba)

	p = 0.2		p = 0.5		p = 0.8	
	Delay	Saving	Delay	Saving	Delay	Saving
N = 2	2.67	46.6 %	1.32	34 %	1.06	15.2 %
N = 4	1.63	67.4 %	1.14	43 %	1	20 %
N = 8	1.63	67.4 %	1.05	47.5 %	1	20 %

Cuadro I
RETARDO DE NOTIFICACIÓN (EN NÚMERO DE RANURAS) Y AHORRO ENERGÉTICO PREVISTO

VI. CONCLUSIONES

En este trabajo hemos propuesto una modificación para los protocolos de contienda de nivel MAC en redes de sensores. En esta propuesta se incluye un mecanismo de persistencia que ha sido analizado como solución para reducir el tiempo de notificación de eventos, y para reducir el consumo energético. La aproximación p-persistente puede reducir significativamente el consumo energético en el caso de trabajar en redes densas actuando de modo doble: por un lado reduce el número de colisiones, ya que entran menos nodos en la contienda, y por otro lado evita que los nodos tengan que permanecer despiertos durante todo su ciclo activo. La persistencia óptima (p) se ha expresado como función de la densidad de nodos presentes en la red, y se ha presentado un método basdo en el teorema del punto fijo para espacios normados que permite calcularlo de modo eficiente. Asimismo, se ha implementado en sensores de la familia Mica2 el mecanismo propuesto, usando como protocolo de base el S-MAC, los resultados experimentales han mostrado ahorros signiticativos que respaldan la viabilidad del mecanismo propuesto.

Como trabajo futuro pretendemos extender nuestro análisis para considerar otros modelos de red y modos de operación,

por ejemplo, considerando una distribución de los nodos no uniforme.

VII. Agradecimientos

Esta investigación ha sido financiada por los proyectos DEP2006-56158-C03-03/EQUI del Ministerio de Educación y Ciencia y TEC2007-67966-01/TCM (CON-PARTE-1) del Ministerio de Industria, Turismo y Comercio. Asimismo, se ha desarrollado en el contexto del *Programa de Ayudas a Grupos de Excelencia de la Región de Murcia*, de la Fundación Séneca, Agencia de Ciencia y Tecnología de la Región de Murcia (Plan Regional de Ciencia y Tecnología 2007/2010).

Referencias

[1] Akyildiz, I., Kasimoglu, I. (2004). "Wireless sensor and actor networks: research challenges". *Elsevier Ad Hoc Networks*, vol. 2, issue 4, pp. 351–367, Octubre 2004.

[2] Ye, W., Heidemann, J., Estrin, D. (2004). "Medium Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks". *IEEE/ACM Transactions on Networking*, vol. 12, issue 3, pp. 493–506, Junio 2004.

[3] Dam, T., Langendoen, K. (2003). "An adaptive energy–efficient MAC protocol for wireless sensor networks". *Proc. ACM Int. Conf. on Embedded Networked Sensor Systems*, pp. 171–180, 2003.

[4] C.C. Enz et al. WiseNet: An Ultralow-Power Wireless Sensor Network Solution IEEE Comp., vol. 37, n0 8, Aug. 2004

[5] IEEE 802.11, 1999 Edition (ISO/IEC 8802-11:1999). Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

[6] Y.C. Tay, K. Jamieson and H. Balakrishnan, Collision-Minimizing CSMA and Its Applications to Wireless Sensor Networks IEEE Journal on Selected Areas in Communications, Vol. 22, No. 6, August 2004

[7] K. Jamieson, H. Balakrishnan and Y.C. Tay Sift: A MAC Protocol for Event-Driven Wireless Sensor Networks IEEE Journal on Selected Areas in Communications, Vol. 22, No. 6, August 2004

[8] L. Debnath and M. Mikusinski, P. Introduction to Hilbert Spaces with Applications San Diego, CA: Academic Press, 1990.

[9] TinyOS: Open-source operating system for wireless embedded sensor networks. [Online]. Disponible en: http://www.tinyos.net

[10] Gay, D., Levis, P., von Behren, R., Welsh, M., Brewer, E., Culler, D. (2003). "The nesC language: a holistic approach to networked embedded systems". *Proc. ACM SIGPLAN 2003 Conf. on Programming Language Design and Implementation*, pp. 1–11, 2003.