DESPLIEGUE DE WLANS EN EXTERIORES: DESARROLLO DE UNA HERRAMIENTA PARA LA TOMA DE MEDIDAS GEOREFERENCIADAS

Alexis Porro Pérez[*], Rafael Vidal Ferré[**]

[*] Ingeniero Técnico de Telecomunicaciones, especialidad en Telemática. UPC.
[**] Departamento de Ingeniería Telemática, Grupo de redes inalámbricas. UPC.
Contact mail: [*]alexis.porro@estudiant.upc.es, [**]rafael.vidal@entel.upc.es

1. INTRODUCCIÓN

El estándar IEEE 802.11b ha tenido una gran aceptación por parte de los usuarios y ha experimentado un gran *boom* comercial. Esta tecnología estuvo pensada en un principio para dar cobertura a interiores, aunque con el tiempo se ha extendido su uso a exteriores. Un par de ejemplos de este uso los podemos encontrar en la ciudad de Zamora con la empresa Afitel (http://www.afitel.com), o en Seattle (http://seattlewireless.net) pionera en este tipo de proyectos.

Este hecho provoca la necesidad de obtener medidas de potencia georeferenciadas en exteriores para poder estudiar el perfil de cobertura, y así poder determinar la ubicación de APs o problemas de interferencia entre los APs existentes. Con esta idea en mente se ha diseñado una aplicación gráfica que utiliza una tarjeta WLAN (compatible con el estándar IEEE 802.11b) y un dispositivo receptor de GPS para obtener la medida de potencia georeferenciada en cada punto. Mediante esta herramienta se puede guardar en ficheros toda la información referente a las redes inalámbricas, además de la información de posicionamiento. También ofrece la posibilidad de poder procesar estos datos posteriormente para ser interpretados por la aplicación MatLab.

El artículo empieza explicando como puede obtenerse la información necesaria para realizar medidas efectivas de una red 802.11. En segundo lugar se explica el diseño funcional de la aplicación, separándola en diversos módulos relacionados entre sí. La implementación de estos módulos es explicada a continuación, detallando el software utilizado así como el formato de los ficheros generados. Seguidamente, y a modo de aplicación práctica de la herramienta, se comentan los resultados obtenidos en su utilización para realizar medidas en los exteriores de la EPSC (Escuela Politécnica Superior de Castelldefels) para determinar la cobertura exterior que da la red WLAN instalada en su interior. Para terminar, en las conclusiones se comentan los problemas surgidos durante la implementación de la aplicación así como las líneas futuras.

2. OBTENCIÓN DE INFORMACIÓN DE REDES 802.11

Existen numerosas aplicaciones que permiten obtener información de redes IEEE 802.11, como por ejemplo Kismet, AirSnort, WiFiScanner o PrismStumbler para GNU/Linux, o NetStumbler para Win32. Todas ellas se basan en gran parte en la escucha de unas tramas de nivel 2 de gestión denominadas beacons, y en la medida de los niveles de la señal recibida mediante una tarjeta 802.11. Estas tramas son emitidas de manera periódica (el periodo de beacon) por parte de los Access Points (APs) a los nodos móviles (STAs) para anunciar su presencia. En esta apartado se comenta el formato de las tramas MAC 802.11 y en concreto de las tramas beacon para conocer toda la información que podemos obtener de ellas.

2.1 Trama MAC 802.11. Los Beacons

La trama MAC queda ilustrada en la figura 1, y se compone básicamente de los siguientes elementos: una MAC Header, el Frame Body de longitud variable y un FCS. El campo de control de trama se compone de los subcampos que aparecen en la figura 2.

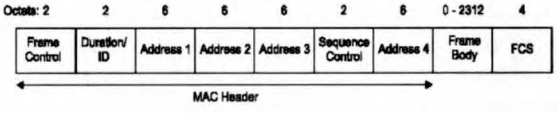

Fig. 1 *Formato de la trama*

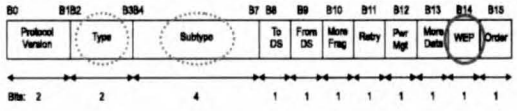

Fig. 2 *Campo Frame Control*

De estos subcampos, los que tienen especial interés son los que se explican a continuación:

〈 Subcampos **Type** y **Subtype**

El campo de Tipo tiene una longitud de 2 bits, y el de Subtipo de 4 bits. Ambos campos conjuntamente identifican la función de la trama. Existen 3 tipos de trama: las de control, datos y gestión. Dentro de éste último tipo tenemos las tramas de beacon (ver Fig. 3). Cada uno de los tipos de trama tiene diversos subtipos. La tabla 1 define la combinación válida de tipo y subtipo para la trama de beacon.

Tabla 1 *Combinación válida de tipo y subtipo para la trama de beacon*

Type value b3 b2	Type description	Subtype value b7 b6 b5 b4	Subtype description
...	...	...	...
00	**Management**	**1000**	**Beacon**
...	...	...	...

〈 Subcampo **WEP**

El campo WEP tiene una longitud de 1 bit y nos permite saber si se está enviando información cifrada utilizando el algoritmo WEP (valor «1») o no (valor «0»).

El formato de una trama de gestión es independiente del subtipo de trama, y está definido en la figura 3.

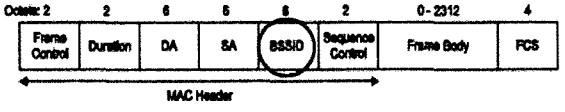

Fig. 3 Formato de la trama de gestión

El Frame Body de la trama de gestión de beacon contiene diversos campos fijos que son realmente importantes para la aplicación. Mediante el campo 4, **SSID**, se puede obtener el nombre de la red WLAN o ESSID (Extended Service Set Identifier), y mediante el campo 7, **DS Parameter Set**, se puede obtener el canal utilizado en la comunicación.
Otro de estos campos fijos es el *Capability Information*. Este campo 3 tiene una longitud de 2 bytes y queda ilustrado en la figura 4. A partir de los 2 primeros bits de este campo se puede conocer el modo en el que se está trabajando según la tabla 2.

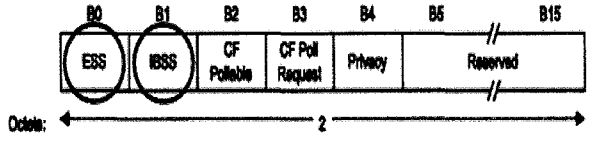

Fig. 4 Campo fijo Capability Information

Tabla 2*Modo de funcionamiento dependiendo de los bits ESS e IBSS*

Finalmente, y a modo de resumen, la tabla 3 indica los parámetros que se han obtenido a través de la tarjeta 802.11. Algunos de estos parámetros dependen del contenido de las tramas de beacon, y otros son completamente independientes.

Tabla 3 *Parámetros o campos obtenidos a partir de la tarjeta 802.11*

CAMPOS
WEP
BSSID
ESSID
CANAL
MODO
POTENCIA SEÑAL
POTENCIA RUIDO
SNR

3. ARQUITECTURA DE LA APLICACIÓN

La herramienta desarrollada ha sido pensada como un conjunto de diversos módulos que interaccionan entre ellos para un correcto funcionamiento de la aplicación. Estos módulos y sus interacciones son descritos a continuación.

3.1 Módulo de WLAN

Este módulo se encarga de la interacción entre la aplicación y la tarjeta WLAN a través de los drivers de ésta. De esta forma, permite obtener los valores de diversos parámetros de la tarjeta WLAN, así como establecer también el valor de algunos de estos parámetros.

Los parámetros que este módulo permite obtener son los que se listan a continuación:

- **Canal o Frecuencia** a la que está trabajando actualmente la tarjeta wireless.
- **Bitrate:** velocidad máxima a la que puede transmitir y recibir datos la STA.
- **Nickname:** indica el nombre (*nickname*) que tiene la STA para diferenciarla de las demás, aunque es meramente un simple accesorio. De hecho, este parámetro no se usa ya que los protocolos no trabajan con él.
- **Signal Level:** potencia de la señal que recibe la STA.
- **Noise Level:** potencia del ruido que recibe la STA.
- **SNR:** indica la relación señal a ruido de la señal recibida del AP
- **ESSID:** identificador de la red ESS en la que está trabajando la STA.
- **Dirección MAC del AP:** es el AP al que está asociada la STA en el momento actual.
- **WEP:** indica si la transmisión en la red ESS está cifrada (on) o no (off).
- **Mode:** modo en el que trabaja la STA o AP, que puede ser: Managed o Ad-Hoc

De ellos, este módulo sólo permite establecer el valor de **Canal o Frecuencia, ESSID, Dirección MAC del AP** y **Nickname**.

Por otra parte, este módulo tiene dos modos de funcionamiento:

- **Normal:** en este modo se puede obtener la información de los parámetros anteriormente citados sobre el funcionamiento actual de la tarjeta. También está implementada una función para poder cambiar los parámetros de la tarjeta.
- **De exploración:** en este modo se obtienen los parámetros que caracterizan a un AP. Para esto, se van explorando todos los canales del espectro y se va obteniendo la información que contienen las tramas de gestión de beacon.

3.2 Módulo de GPS

Para este módulo es necesario un dispositivo receptor de

GPS, ya que la aplicación necesita obtener la información de posicionamiento: latitud, longitud y altitud de cada punto. Para tal fin se utiliza una aplicación externa a la propia como interfaz para obtener los datos del dispositivo receptor de GPS.

3.3 Módulo de captura de datos en ficheros

Este módulo permite capturar en ficheros toda la información que obtienen los 2 módulos anteriores (la información referente a la tarjeta WLAN y la referente al dispositivo de GPS), para un posterior estudio de la información obtenida, o un post-procesado de los datos (como se explica en el siguiente módulo). El usuario especifica el nombre del fichero donde se guardan los datos.

3.4 Módulo de tratamiento de capturas

Una vez se tengan los datos en un fichero, se ha pensado en hacer un tratamiento a posteriori de estos. Existen herramientas muy potentes para hacerlo, como el MatLab o sus clones de libre distribución como el Octave o SciLab. Lo que se pretende es aprovechar la potencia de cálculo de estas herramientas transformando los datos a un formato compatible con estas aplicaciones.

Este módulo no interactúa con ninguno de los demás, ya que realiza un post-procesado de los datos capturados por el módulo anterior. Este módulo trabaja con 2 ficheros: el de captura de datos generado por el módulo anterior, y el que será compatible con MatLab, que contendrá la información del fichero anterior de tal forma que la aplicación MatLab sea capaz de interpretarla. Así, a partir de este fichero MatLab, se podrá generar un gráfico en 3D sobre las medidas de potencia georeferenciadas obtenidas.

3.5 Módulo de interfaz gráfica (GUI)

Este módulo concierne toda la interacción con el usuario final a través de los elementos visuales. Está compuesto por elementos típicos de aplicaciones con interfaces gráficas: ventanas, barras de menús, botones, listas deslizables, imágenes en ventanas (como en los mensajes de error) y demás. Con este módulo se consigue la unificación total de la aplicación, ya que a través de la interfaz gráfica se pueden controlar todos los demás módulos.

La figura 5 muestra las relaciones existentes entre los diversos módulos que componen la aplicación. En la siguiente sección se entrará en más detalle en la implementación de esta aplicación.

4. IMPLEMENTACIÓN DE LA APLICACIÓN

Esta aplicación se ha desarrollado completamente bajo una plataforma GNU/Linux. Para la implementación de esta aplicación sobre un PC ha sido necesario que éste incorporase un adaptador de ISA a PCMCIA (para la comunicación con la tarjeta WLAN) y un puerto serie (para la comunicación con el dispositivo GPS). La potencia del procesador y la cantidad de memoria RAM no son significativas, siempre y cuando el PC sea capaz de ejecutar Red Hat Linux 9.0, la distribución elegida, que incorpora el kernel 2.4.20, con ligereza.

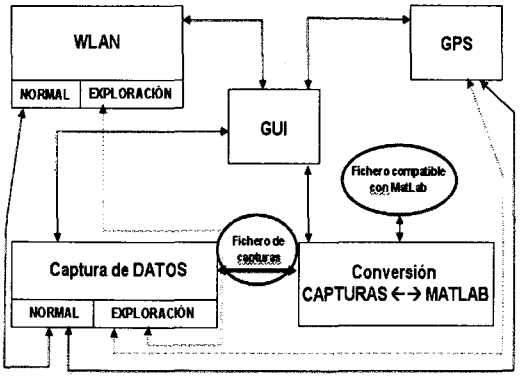

Fig. 5 Diagrama de bloques sobre las relaciones entre módulos

4.1 Módulo de WLAN

Para la realización de este primer módulo se ha utilizado la programación en lenguaje C junto con llamadas al sistema (a través del kernel), haciendo uso de las Wireless Extensions y las Wireless Tools (ambas desarrolladas por Jean Tourrilhes). Se han utilizado algunas de las funciones de las Wireless Tools, otras se han adaptado a las necesidades de la aplicación, y también se han creado nuevas funciones. Básicamente para tal fin se ha utilizado, en primer lugar, un socket (canal de comunicación) entre el kernel y el módulo, mediante el que se ha interaccionado con el driver de la tarjeta WLAN. En segundo lugar, y para establecer o conseguir los valores de los parámetros de la WLAN a través del socket anteriormente creado, se ha utilizado la función de C
ioctl (int sock, int request, ...) A esta función, y dependiendo si queremos establecer u obtener información de la tarjeta WLAN (get/set), se le pasará una estructura de tipo *wireless_info*, que se define en el fichero *wireless.h*. Este fichero contiene la API de las Wireless Extensions, y normalmente se encuentra bajo el directorio /usr/include/linux/.

Los *drivers* PCMCIA de las tarjetas WLAN utilizados en este proyecto son los que se incluyen en la versión 3.2.4 del paquete *pcmcia-cs*. Dependiendo del *driver* y la tarjeta utilizada se han podido obtener y establecer unos u otros parámetros:

- **Tarjeta Cisco Aironet 350 Series y driver airo_cs:** permite explorar el espectro en busca de APs mediante los *beacons* que le llegan. Sin embargo no puede obtener la potencia de la señal de ruido, y por tanto

tampoco la SNR, aunque puede obtener la potencia de la señal.

- **Tarjeta Lucent WaveLAN Silver y driver orinoco_cs:** permite establecer todos los parámetros que permite esta aplicación excepto el AP al que asociarse (esta operación no está soportada por su driver). Por otro lado permite obtener la SNR, ya que también puede obtener la potencia de ruido. Sin embargo, el mayor inconveniente que tiene es que no permite explorar el espectro, tan sólo puede obtener la información del AP al que está asociada en un momento determinado.

4.2 Módulo de GPS

El dispositivo receptor de GPS utilizado es un ***GARMIN GPS 76***, con diferentes formatos de salida de datos, de los cuales, el que necesitamos es el NMEA 0183, ya que es el único que tiene en común con la aplicación que controla al dispositivo receptor de GPS. Esta aplicación es el demonio de GPS ***gpsd***, cuyo autor es Remco Treffkorn (http://www.pygps.org/gpsd/downloads). ***Gpsd*** es un demonio o servidor que obtiene la información de un dispositivo receptor de GPS a través de una interfaz de puerto serie, y espera conexiones de clientes en un puerto determinado (por defecto es el 2947). Esta aplicación espera obtener datos de salida del receptor de GPS en formato NMEA 0183 o en formato binario de Rockwell.

Para interactuar con esta aplicación, se ha hecho una pequeña función que utiliza la comunicación mediante sockets, aunque a diferencia del módulo de WLAN esta vez no se comunica con el kernel, sino con un servidor en un host, que en este caso es la propia máquina que tiene conectado el receptor de GPS.

4.3 Módulo de captura de datos en ficheros

Para este módulo se han utilizado funciones típicas de la gestión de ficheros, como son ***open***, ***close***, ***read*** o ***write***. En este módulo tan sólo se trabaja con un único fichero en el que se almacenan los datos que se van capturando, y que se obtienen a través de los 2 módulos anteriores.

Para este módulo, se han almacenado los datos pertenecientes a la WLAN junto con los datos que nos ofrece el dispositivo GPS. El formato general del fichero es el siguiente:

- 3 bytes de cabecera
- datos variables de captura del módulo de WLAN y/o módulo de GPS

Dependiendo de si la tarjeta tiene la capacidad de detectar el ruido o no, hay 2 tipos de formatos de ficheros y datos.

- Si la tarjeta puede obtener la potencia de la señal de ruido, entonces podemos conseguir la SNR, y en este caso guardamos la potencia de la señal en dBm, la potencia del ruido en dBm y la SNR en dB. Por tanto, la cabecera se define como ***ALL*** indicando que se ha guardado todo.
- Si por el contrario la tarjeta no puede obtener la potencia de la señal de ruido, no se puede obtener la SNR, por tanto no guardamos ni la potencia del ruido ni la SNR. En este caso la cabecera se define como ***SIG*** indicando que sólo se ha guardado la potencia de la señal (*SIGnal level*) en dBm.

- En el caso que la tarjeta pueda obtener la potencia del ruido (***ALL***), el formato del contenido del fichero es el siguiente:

ESSID BSSID CANAL
SEÑAL **RUIDO** **SNR** MODO
ENCRIPTACIÓN LATITUD
LONGITUD ALTITUD

Las siguientes líneas son un ejemplo del formato del fichero de capturas que contiene toda la información posible de la WLAN y además contiene la posición:

Essid1 00:11:22:33:44:55 1 -62.00
-90.00 28.00 Managed on
41.275232 1.987380

4.4 Módulo de tratamiento de capturas

En este módulo, al igual que el anterior, se utilizan funciones de gestión de ficheros, así como funciones específicas para cadenas de caracteres o *strings* (incluidas en string.h), como ***strstr***, ***strrstr***, ***index***, ***rindex***, etc.
En este módulo, y a diferencia del anterior, se trabaja con 2 ficheros al mismo tiempo:

- El de **capturas**, que se abre en modo sólo-lectura
- El **compatible con MATLAB**, que se abre en modo sólo-escritura

En este módulo se genera un fichero compatible con MATLAB en el que se crea una matriz con las coordenadas de los puntos (latitud y longitud) para poder obtener una gráfica en 3D, en la que el eje z representa el nivel de señal recibido en dBm o la SNR en dB, dependiendo de las capacidades de la tarjeta utilizada y sus drivers respectivos. En la figura 9 se puede apreciar con más detalle un ejemplo de esta gráfica en 3D.

4.5 Módulo de interfaz gráfica (GUI)

En este módulo, realizado en lenguaje C como todos los demás en esta aplicación, se han utilizado las librerías gráficas de GTK+ en su versión 1.2. Con estas librerías «libres» se ha podido diseñar la parte gráfica de la aplicación, es decir, las ventanas, botones, listas, entradas de texto y otros. Para la implementación de este módulo, se ha utilizado una herramienta de desarrollo gráfico llamada GLADE en su versión 0.6.4. Esta herramienta lo que hace es automatizar y acelerar la construcción de interfaces gráficas utilizando las librerías gráficas GTK+.

En las figuras 6 y 7 se puede observar el aspecto de la

aplicación en pleno funcionamiento tanto en el modo de funcionamiento normal como en el de exploración.

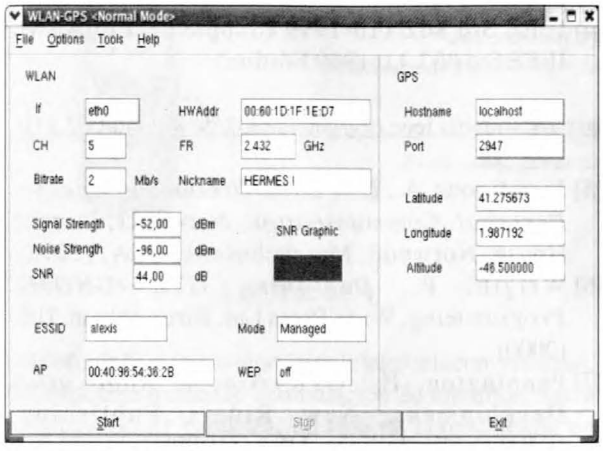

Fig. 6 *Ejemplo del modo de funcionamiento normal*

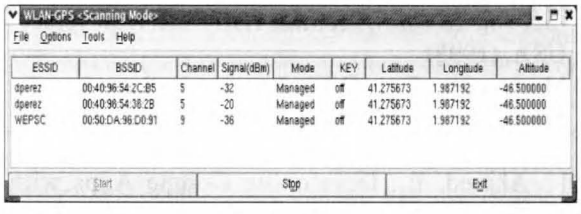

Fig. 7 *Ejemplo del modo de funcionamiento de exploración*

5. PRUEBAS

Una vez finalizada la implementación de la aplicación, y habiendo comprobado que funciona correctamente, se ha pasado a la realización de capturas de datos en el exterior de la EPSC a modo de ejemplo práctico de las capacidades de la herramienta. Para capturar los datos se ha ejecutado la aplicación en un portátil con la tarjeta WLAN Cisco Aironet 350 Series (con el *driver* airo_cs), que es la que permite explorar el espectro, aunque no permite obtener la potencia de la señal de ruido, y el dispositivo receptor de GPS ***GARMIN GPS 76***.

Se ha recorrido todo el edificio en 4 pasadas, una por cada pared, haciendo zig-zag para conseguir el mayor alcance posible. El itinerario seguido en las capturas se muestra en la figura 8.

Una vez explorado todo el exterior de la escuela, se ha realizado un post-procesado de los datos mediante la herramienta (para obtener ficheros MatLab), que nos ha permitido conseguir diferentes gráficas en 3D de la cobertura WLAN en el exterior de la Escuela, como la que se muestra en la figura 9. En esta figura se puede observar la cobertura que existe en el exterior de la Escuela para la red ESS «WEPSC», representando sólo la información del canal 1, es decir, filtrando desde la aplicación por ESSID

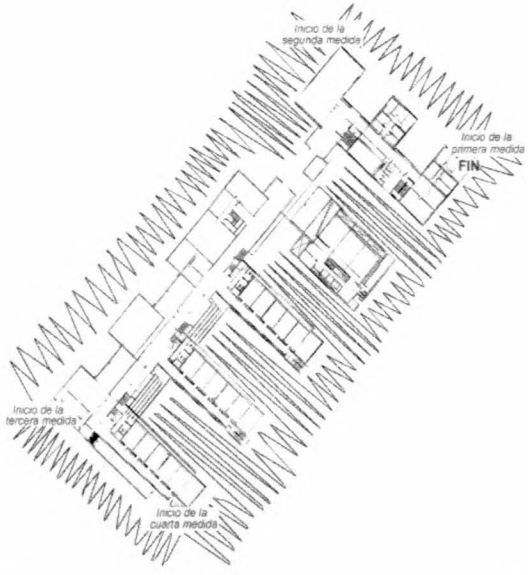

Fig. 8 *Mapa que muestra el itinerario seguido para las medidas en el exterior*

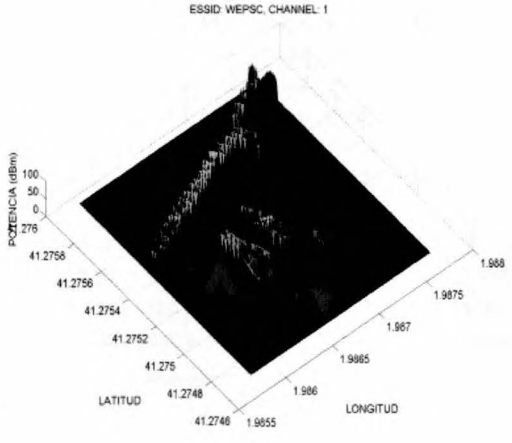

Fig. 9 *Gráfica en 3D sobre la cobertura de la red «WEPSC», canal 1*

y canal **1**. Se puede apreciar el perfil del edificio de la EPSC.

6. CONCLUSIONES

Con el desarrollo de la aplicación se han conseguido diversos objetivos. El primero, la implementación de una herramienta (con página oficial en http://asterx.upc.es/~alexis/wlan-gps.html) para la toma de medidas WLAN georeferenciadas, mediante la cual poder guardar la información de redes IEEE 802.11b y la información de posicionamiento en ficheros. El segundo, poder convertir los ficheros de capturas de datos en ficheros MatLab, aprovechando así la potencia y flexibilidad que ofrece la aplicación MatLab para la generación de gráficos en 3D, así como cualquier otra operación matemática con los datos.

Además se han realizado diversas medidas en el exterior de la EPSC a modo de ejemplo de aplicación práctica de la herramienta. Así se puede comprobar que la aplicación es funcional y útil. Para validar la aplicación, se han comparado las medidas obtenidas con esta aplicación con las obtenidas con aplicaciones del fabricante de las tarjetas WLAN, y con los datos que se muestran en el display del receptor de GPS. Sin embargo han habido limitaciones de implementación, ya que no todas las tarjetas que se han utilizado ofrecen las mismas funcionalidades, ni los *drivers* respectivos son lo suficientemente buenos para ofrecer toda la información necesaria.

La herramienta tiene otras aplicaciones prácticas potenciales además de la que se ha querido conseguir en primer lugar. Por ejemplo se podría utilizar para estudiar de manera aproximada el diagrama de radiación de antenas no comerciales y/o experimentales, o también para detectar APs o ESSIDs no registrados en una zona dónde no deberían estar. A pesar que existen diversas aplicaciones con funcionalidades similares a la presente, ésta se diferencia de ellas por las diversas funcionalidades extra que incorpora, como el establecimiento de diversos parámetros de la tarjeta WLAN, y la obtención de ficheros preparados para ser procesados por MatLab.

Como líneas futuras queda la representación de las medidas realizadas en cada punto sobre un mapa real y a escala de cualquier zona. Para esta tarea existen diversas aplicaciones, como por ejemplo la aplicación GPSMap (http://gpsmap.sourceforge.net/). También se podría utilizar los valores de altitud en cada punto para realizar una gráfica teniendo en cuenta este parámetro, que es muy importante en zonas donde la orografía no es plana. Además, la aplicación MatLab proporciona un gran abanico de posibilidades para la creación de scripts especializados en, por ejemplo, la búsqueda de puntos con interferencias significativas o con potencias por debajo de un umbral mínimo.

7. AGRADECIMIENTOS

Esté trabajo ha sido financiado por el proyecto TIC2003-01748.

8. BIBLIOGRAFÍA

[1] Porro, A. y Vidal, R., *Despliegue de WLANs en exteriores: Desarrollo de una herramienta para la toma de medidas georeferenciadas*. Trabajo Fin de Carrera , EPSC, Julio 2003.

[2] Prasad, N. and Prasad A., WLAN *Systems and Wireless IP for Next Generation Communications*, Artech House, Norwood, Massachusetts, USA, (2002)

[3] ANSI/IEEE Std 802.11, 1999 Edition
http://standards.ieee.org/getieee802/download/802.11-1999.pdf

[4] IEEE Std 802.11b-1999 (Supplement to ANSI/IEEE Std 802.11, 1999 Edition)

http://standards.ieee.org/getieee802/download/802.11b-1999.pdf

[5] Jamalipour, A., *Low Earth Orbital Satellites for Personal Communication Networks*, Artech House, Norwood, Massachusetts, USA, (1998)

[6] Wrigth, P., *Beginning GTK+/GNOME Programming*, Wrox Press Ltd, Birmingham, UK, (2000)

[7] Pennington, H., *GTK+/Gnome Application Development*, New Riders Publishing, Indianapolis, Indiana, USA, (1999)

[8] Stevens, W. R., *UNIX Network Programming Volume 1, Networking APIs: Sockets and XTI (Second Edition)*, Prentice Hall PTR, New Jersey, USA, (1998)

[9] Moritsugu, S. y DTR Bussiness Systems, *La biblia de UNIX*, Anaya Multimedia, Madrid, (1999)

[10] Ahmed, E., Developing Gnome Apps with Glade, (2002)
http://writelinux.com/glade/index.php

[11] Gale, T. and Main, I., *GTK v1.2 Tutorial*, página oficial de GTK, (2000)
http://www.gtk.org/tutorial1.2/

AUTORES

Alexis Porro, Ingeniero Técnico de Telecomuni-caciones en la especialidad de telemática por la EPSC (UPC) desde el año 2003. Actualmente está cursando el primer curso del segundo ciclo de Ingeniería Superior de Telecomunicaciones en la EPSC y trabaja como becario en el Grupo de redes inalámbricas.

Rafael Vidal, Ingeniero de Telecomunicaciones por la ETSETB (UPC) y profesor del Departamento de Ingeniería Telemática desde el año 2000, con docencia en la EPSC (UPC). Forma parte del grupo de investigación de redes inalámbricas desde el año 1998. Su ámbito de trabajo es el soporte a la movilidad en redes IP. Ha participado en diferentes proyectos de financiación pública y privada. Actualmente trabaja en los proyectos RUBI (Red Ubicua Basada en IP, TIC2003-01748) e I2Cat.