

Capítulo 1

Cobertura en Redes WLAN

1.1

Descripción de las Redes WLAN

A continuación pasamos a describir qué son las redes WLAN y cuáles son sus usos.

Las redes de área local inalámbricas (WLANs) constituyen en la actualidad una solución tecnológica de gran interés en el sector de las comunicaciones inalámbricas de banda ancha. Estos sistemas se caracterizan por trabajar en bandas de frecuencia exentas de licencia de operación, lo cual dota a la tecnología de un gran potencial de mercado y le permite competir con otro tipo de tecnologías de acceso inalámbrico de última generación como UMTS y LMDS, pues éstas requieren de un importante desembolso económico previo por parte de los operadores del servicio. Ahora bien, ello también obliga al desarrollo de un marco regulatorio adecuado que permita un uso eficiente y de dominio público disponible.

Originalmente las redes WLAN fueron diseñadas para el ámbito empresarial. Sin embargo, en la actualidad han encontrado una gran variedad de escenarios de aplicación, tanto públicos como privados: entorno residencial y del hogar, grandes redes corporativas, PYMES, zonas industriales, campus universitarios, entornos hospitalarios, ciber-café, hoteles, aeropuertos, medios públicos de transporte, entornos rurales, etc. Incluso son ya varias las ciudades en donde se han instalado redes inalámbricas libres para acceso a Internet.

Básicamente, una red WLAN permite reemplazar por conexiones inalámbricas los cables que conectan a la red los PCs, portátiles u otro tipo de dispositivos, dotando a los usuarios de movilidad. Entre los componentes que permiten configurar una WLAN se pueden mencionar los siguientes: terminales de usuario o Clientes (dotados de una tarjeta interfaz de red que integra un transceptor de radiofrecuencia y una antena), puntos de acceso y controladores de puntos de acceso, que incorporan y autenticación de usuarios, etc.

El futuro de la tecnología WLAN pasa necesariamente por la resolución de cuestiones muy importantes sobre seguridad e interoperabilidad, en donde se centran

actualmente la mayor parte de los esfuerzos. Sin embargo, desde el punto de vista de los usuarios, también es importante reducir la actual confusión motivada por la gran variedad de estándares existentes.

1.2

Diseño Eficiente de Redes WLAN: Parámetros a tratar para garantizar su funcionamiento

Hemos tratado en el punto anterior las utilidades de las WLAN, ahora nos centraremos en los inconvenientes de dicha tecnología. Para ello vamos a detenernos en explicar más detalladamente los parámetros que la caracterizan, y centrarnos en cuál es el principal problemas de este tipo de redes y qué se puede hacer para resolver ese problema.

Como se ha comentado, las ventajas más relevantes en el uso de las WLANs descansan en su movilidad y en su flexibilidad. Sin embargo, el funcionamiento y sobre todo, la seguridad de las tecnologías WLAN todavía presentan deficiencias con respecto a las tecnologías LAN de cable. Con respecto a ello, y dado que la seguridad en las redes WLAN es una de las cuestiones más sensibles con respecto a las aplicaciones, los trabajos existentes sobre el tema se han centrado en estudiar el diseño de redes WLAN que garanticen su funcionamiento bajo ciertas condiciones de seguridad. Dicho diseño está estructurado en torno al concepto de AP (Punto de Acceso), entendiendo como tal un dispositivo que interconecta dispositivos de comunicación inalámbrica para formar una red inalámbrica, y el Problema del *Diseño de WLAN* consiste tanto en determinar la *localización* de los APs como la potencia y el canal de cada uno de ellos tal que garanticen el funcionamiento del sistema según los estándares de seguridad requeridos (ya que de esa manera cualquier persona con una terminal inalámbrica podría comunicarse con un punto de acceso privado si no se disponen de las medidas de seguridad adecuadas). La eficiencia del diseño requiere no sólo localizar los APs y configurarlos adecuadamente, sino también asimismo *minimizar* el número de APs para instalar en la red.

Este problema, y su correspondiente modelización matemática, ha sido abordada por varios autores y formulada bajo diferentes modelos. En todos ellos destacan una serie de factores/parámetros a ser incorporados:

- * La localización de los puntos de demanda; de esa forma sabremos dónde es más ventajoso colocar los APs que utilizaremos.
- * El máximo número de estaciones por cada punto de demanda.
- * El conjunto de niveles de energía disponibles; para saber de qué potencia dispondremos para transmitir la señal.
- * El conjunto de canales (no superpuestos) disponible; con lo que tendremos un rango de frecuencias asignada (también llamada canal), para colocar nuestra señal.
- * El mínimo ancho de banda requerido por cada estación; ya que necesitamos un rango mínimo donde poner la señal sin que haya interferencias con otros canales.
- * La mínima SNR (señal a ruido); para saber qué potencia mínima debe llegar al receptor para la correcta captación de la señal.
- * El radio de atenuación entre los puntos demandados y las potenciales localizaciones de los APs, pues habrá un deterioro de la señal en el medio por el que se transmita.

Estos parámetros han de ser complementados con las hipótesis que se hagan sobre la organización y estructura de las WLAN (en función de la aplicación real de la que derive el modelo). Una de los modelos más amplios es el formulado por *Bahri et al.* (2005) que describimos a continuación. Básicamente, el modelo de estos autores formulan matemáticamente un problema de diseño de WLAN consistente en la localización de APs (así como de la potencia y el canal de cada uno), y el objetivo del problema es determinar *el mínimo número* de APs (así como la localización, potencia y canal de cada uno de ellos), de forma que se garantice el funcionamiento de la red WLAN con los estándares de seguridad y cobertura requeridos.

La formulación del modelo precisa de la siguiente notación:

Conjuntos: I denota a un conjunto de puntos de demanda (usuarios terminales), n_i el máximo número de estaciones en el punto de demanda $i \in I$, J denota al conjunto de potenciales localizaciones donde instalar los APs, P es el conjunto de niveles de potencia disponibles (en Watt), y C el conjunto de canales (no superpuestos) disponibles.

Variables de Decisión: $w_j, x_{ij}, y_j^p, z_j^p$ son todas ellas variables binarias tales que:

$w_j = 1$ si y sólo si un AP se instala en $j \in J$.

$x_{ij} = 1$ si y sólo si la demanda $i \in I$ está asociada con el AP instalado en $j \in J$.

$y_j^p = 1$ si y sólo si la potencia $p \in P$ se usa en el AP instalado en $j \in J$.

$z_j^p = 1$ si y sólo si el canal $c \in C$ se usa en el AP instalado en $j \in J$.

Constantes: $p_{\min}, t_{\min}, d_{\min}$, que representan:

$p_{\min}$: es la mínima potencia de señal (en Watt) desde un AP necesaria para ser asociado con dicho AP (potencia que puede estar especificada por el entorno, el emisor ó empresa proveedora, los obstáculos naturales, la orografía, etc.).

$t_{\min}$: el mínimo ancho de banda (en bps) requerido por cada estación.

$d_{\min}$: el mínimo SNR (signal-to-noise ratio) para reutilizar un canal.

Funciones: $\alpha(i, j)$ denota la razón de disminución (ó atenuación) desde el punto $i \in I$ a la localización $j \in J$, ($0 \leq \alpha(i, j) \leq 1$), $\beta(j, j')$ representa la razón de disminución entre $j \in J$ y $j' \in J$, y finalmente $\delta(n)$ es el mínimo ancho de banda (en bps) conteniendo n estaciones ($n > 0$, ya que $\delta(0)$ se fija a $\delta(1)$).

El modelo matemático establecido en *Bahri et al.* (2005) para el diseño óptimo de una red WLAN global con garantías de funcionamiento (WLNDP, WLAN Network Design Problem), está formulado como sigue:

WLNDP:

$$\min_{w, x, y, z} \sum_{j \in J} w_j \quad (1)$$

Sujeto a

Restricciones de asignación de punto de demanda:

$$\sum_{j \in J} x_{ij} = 1, \quad (i \in I) \quad (2)$$

$$x_{ij} \leq w_j \quad (i \in I, \quad j \in J), \quad (3)$$

$$(p_{\min} - \alpha(i, j) \sum_{p \in P} p y_j^p) x_{ij} \leq 0, \quad (i \in I, \quad j \in J) \quad (4)$$

$$\left(\alpha(i, j') \sum_{p \in P} p y_j^p - \alpha(i, j) \sum_{p \in P} p y_{j'}^p \right) x_{ij} \leq 0, \quad (i \in I, \quad j, j' \in J, \quad j < j') \quad (5)$$

Restricciones de unicidad de nivel de potencia AP:

$$\sum_{p \in P} y_j^p = w_j \quad (j \in N) \quad (6)$$

Restricciones de unicidad de valor de canal AP:

$$\sum_{c \in C} z_j^c = w_j \quad (j \in N) \quad (7)$$

Restricciones de mínimo ancho de banda:

$$t_{\min} - \delta \left(\sum_{i \in I} n_i x_{ij} \right) \leq 0 \quad (j \in J) \quad (8)$$

Restricciones de reutilización de cana :

$$\left(d_{\min} \sum_{j' \in J, j' \neq j} \left(\beta(j, j') z_{j'}^c \sum_{p \in P} p y_{j'}^p \right) - p_{\min} \right) z_j^c \leq 0, \quad (j \in J, \quad c \in C) \quad (9)$$

Restricciones de variables enteras:

$$\mathbf{w} \in B^{|J|}, \quad \mathbf{x} \in B^{|I||J|}, \quad \mathbf{y} \in B^{|I||P|}, \quad \mathbf{z} \in B^{|I||C|} \quad (10)$$

Nótese que la función objetivo (1) representa el número de APs instalados en la red, y las restantes restricciones (de (2) a (10)) garantizan los requerimientos sobre las variables así como sobre la asignación, potencia, canal, ancho de banda, y reutilización de la red.

En el mismo trabajo, *Bahri et al.* (2005) establecen que el anterior problema WLNDP es NP-duro. En términos algorítmicos, un problema NP-duro (intrínsecamente difícil, o intratable) es aquél para el cual no se conoce un algoritmo de resolución en tiempo polinomial para cualquier ejemplar de entrada del problema. Dado que el carácter del problema respecto a su computabilidad condiciona las estrategias y algoritmos de resolución, y que los problemas de cobertura son frecuentemente NP-duros (como se verá a lo largo de esta memoria), dedicamos unas líneas a describir brevemente algunas nociones básicas sobre complejidad computacional.

1.3

Problemas tratables e intratables

Uno de los principales problemas que aborda la teoría de la complejidad computacional es establecer la dificultad computacional de los problemas, o en otras palabras, si es posible resolverlos mediante algoritmos eficientes. La eficiencia de un algoritmo está relacionada con los recursos requeridos (espacio y tiempo) para producir la solución. Básicamente, *el tiempo* se mide como el número de *operaciones elementales* realizadas (comparaciones, operaciones y asignaciones) mientras que *el espacio* afecta a la cantidad de memoria usada. Dado que la cantidad de memoria está en función del tamaño del problema y éste repercute en el tiempo, el análisis de la eficiencia de los algoritmos y su complejidad se centra, fundamentalmente, en el estudio del *tiempo de ejecución*. El tiempo de ejecución viene descrito mediante una función $f(n)$, que expresa el número de operaciones elementales requeridas por el algoritmo para resolver un problema de longitud n (tamaño del problema). Asociado al tiempo de ejecución de un algoritmo está su “Orden” (notación “ O ”), que expresa el *crecimiento asintótico* del tiempo de ejecución en el peor caso al aumentar el tamaño n del problema. Así, diremos que un algoritmo es de complejidad polinomial si su peor caso de tiempo ejecución es de orden $O(n^k)$, donde n es el tamaño de la entrada, y k una constante. Esta permite introducir una clasificación de los problemas en “tratables”, ó resolubles mediante un algoritmo de orden polinomial, e “intratables”, o no resolubles en tiempo polinomial. Y además existen problemas cuyo estatus es desconocido, esto es, no se ha descubierto un algoritmo de tiempo polinomial para ellos ni se conoce si su complejidad es superior a la polinomial, aunque se cree que estos problemas también son intratables (no polinomiales).

Para no extendernos demasiado en la fundamentación teórica sobre las clases de complejidad (un análisis riguroso basado en las máquinas de Turing puede encontrarse en el clásico texto de *Garey and Johnson*, 1979), diremos que el problema de la complejidad algorítmica se reduce al estudio de los Problemas de Decisión, de forma que se considera un algoritmo como un mecanismo que permite

resolver un problema de decisión concreto, esto es, obtener una respuesta “SI” o “NO” al problema. En base a ello se puede establecer la siguiente clasificación:

- La clase de complejidad P es el conjunto de todos los problemas de decisión que pueden ser resueltos en tiempo polinomial.
- La clase de complejidad NP es el conjunto de todos los problemas de decisión para los cuales una *solución* (que se postula o conjetura) puede ser *verificada* como respuesta-SI en tiempo polinomial.

Nótese que el hecho de que el problema sea NP , no quiere decir que la solución correspondiente sea fácil de obtener, sino que, dada ésta última, puede verificarse como respuesta afirmativa en tiempo polinomial. Claramente, $P \subseteq NP$. Sin embargo la igualdad $P = NP$ no ha podido ser demostrada todavía. Dicho de otro modo, todavía no se ha encontrado un algoritmo que **resuelva en tiempo polinomial un problema en NP** (pues de conseguirse, se habría demostrado la anterior igualdad).

La teoría de la NP -completitud se basa en el concepto de transformación polinomial. Así, la clase **NP -completo** recoge todos los problemas en NP que son equivalentes mediante transformaciones polinomiales, esto es, aplicando algoritmos que en tiempo polinomial transforman un problema en otro. De esta forma, se puede decir que un problema de decisión D_1 es NP -completo si pertenece a la clase NP y si otro problema D_2 que también pertenece a NP se puede transformar polinomialmente a D_1 . De hecho, la palabra “completitud” refleja que la solución de un problema de decisión NP contiene, de alguna forma, la solución a todos los problemas de decisión de la clase NP . A partir del *Teorema de Cook* (1971) que estableció que el problema de satisfactibilidad (SAT) es NP -completo, se ha podido establecer la NP -completitud de un gran número de problemas mediante la reducción polinomial de los mismos al problema SAT. De todo lo anterior, un posible “esquema” de los problemas NP sería como sigue:

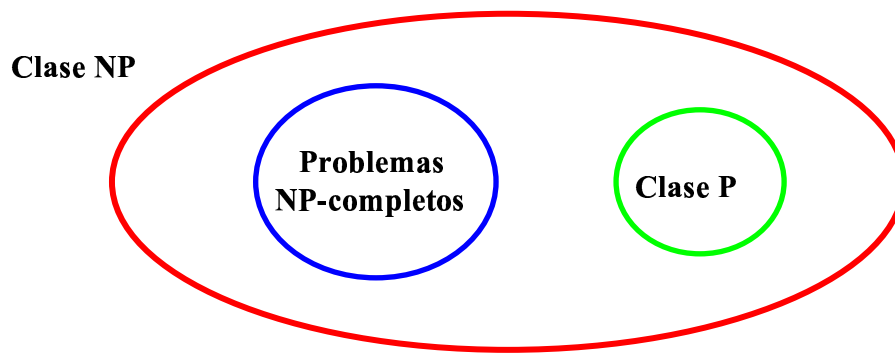


Figura 1.1: Esquema de los problemas NP

Todo Problema de Optimización es formulable en términos de Problema de Decisión, sin más que establecer la correspondiente pregunta (cuya respuesta será SI o NO) en función de una cota. Por ejemplo, el problema del viajante de comercio (optimización): “*Dado un conjunto de ciudades, encontrar una ruta de longitud mínima que pase por todas las ciudades, sin repetir ninguna, y vuelva a la ciudad de partida*”, se puede transformar en la siguiente versión decisional: “*Dada una longitud positiva $k > 0$, ¿existe un camino de las características requeridas cuya longitud sea inferior a k ?*”. Claramente, si se encuentra solución a un problema de decisión, entonces se encuentra también solución al problema de optimización correspondiente.

De esta forma, si para un problema de optimización se establece el carácter NP-completo de su correspondiente versión de decisión, dicho problema de optimización será tan difícil (tan duro) como cualquier problema en NP, y entonces será intratable. De hecho, el término **NP-duro (NP-hard)** se usa en literatura para describir los problemas de optimización cuya versión de decisión está en la clase NP-completo, reflejando con ello la intratabilidad del problema.

La estrategia para abordar los problemas de optimización NP-duros pasa por aplicar algoritmos *heurísticos* que, en un tiempo computacional razonable, producen buenas soluciones (soluciones subóptimas o aproximadas a la óptima). Existe una amplia colección de procedimientos heurísticos para abordar este tipo de problemas, uno de los más eficaces se basa en la **Búsqueda Tabú**. Este algoritmo es aplicado en *Bahri et al.*

(2005) para resolver el anterior problema WLNDP. A continuación haremos una explicación más detallada del mismo:

1.4

Algoritmo de Búsqueda Tabú

Entre los distintos métodos y técnicas heurísticas de resolución de problemas combinatorios surge, en un intento de dotar de "inteligencia" a los algoritmos de búsqueda local, el algoritmo de búsqueda Tabú .

La búsqueda Tabú, a diferencia de otros algoritmos basados en técnicas aleatorias de búsqueda de soluciones cercanas, se caracteriza porque utiliza una estrategia basada en el uso de *estructuras de memoria* para escapar de los óptimos locales, en los que se puede caer al "moverse" de una solución a otra por el espacio de soluciones.

Este algoritmo se dota, por tanto, de una "memoria" donde se almacenan los últimos movimientos realizados, y que puede ser utilizada para "recordar" aquellos movimientos que hacen caer de nuevo en soluciones ya exploradas. Esta "memoria" serviría para impedir la evolución hacia esas soluciones.

Ejemplo Ilustrativo:

A modo de ejemplo muy simple, si se parte de una solución actual x_0 , que resulta ser un óptimo local, y se procede a buscar la mejor solución en un "entorno" de x_0 , se obtendrá una solución x_1 que pasará a ser la nueva solución actual, aunque tenga mayor coste, (movimiento $x_0 Y x_1$). Al continuar con x_1 buscando en el espacio de soluciones dentro de su entorno, puede que sea x_0 la mejor de todas las soluciones (movimiento $x_1 Y x_0$). Es obvio que se vuelve a la misma solución, entrando en un ciclo que impide salir de esa región y explorar otras diferentes. Para evitar caer en óptimos locales, este algoritmo considera ciertos movimientos "tabú", de forma que no puedan ser aceptados durante un cierto tiempo o un cierto número de iteraciones. En el caso anterior, podría considerarse tabú el

movimiento $x_1 Y x_0$ de tal forma que no se aceptaría nuevamente x_0 como solución actual, sino otra solución, evitando así caer en un ciclo.

Algoritmo

A partir de esta idea nace el concepto de Lista Tabú (LT). Cada vez que un movimiento es realizado, se introduce su inverso en una lista circular (la primera en entrar es la primera en salir, estrategia FIFO, *First In, First Out*) de tamaño L de forma que los elementos de dicha lista están penalizados durante un cierto tiempo. Un movimiento, por tanto, no será aceptado siempre y cuando permanezca en la lista tabú. Esta lista constituye la estructura de memoria del algoritmo.

Como se verá más adelante, se contempla la posibilidad de establecer un cierto criterio mediante el cual un movimiento tabú pueda ser aceptado, se trata del nivel de aspiración.

Una forma simple de representar el algoritmo es:

```
LT <-- {0} {lista tabú vacía inicialmente}
x* <-- x0 {solución inicial}
F* <-- F(x0) {valor óptimo inicial}
+Q MIENTRAS ((n_iter_total - n_iter_cambio) < N_MAX_ITER) HACER
* {si nº de iteraciones sin mejorar es menor que un máximo}
* n_iter_total <-- n_iter_total+1 {nº total de iteraciones}
* xn <-- x* {solución actual}
* F' <-- 4 {coste mínimo de solución cercana}
* +Q PARA Iteracion=1 HASTA Card(V(xn)) HACER {para todas las cercanas}
* * x <-- V(xn) {se genera solución cercana}
* * +Q SI ( F(x)<F' Y ( [xn->x]óLT O ([xn->x]0LT Y F(x)<Fa) ) ) ENTONCES
* * * {si es la de menor coste y no es tabu o siendo tabu es aceptada}
* * * x' <-- x {mejor solución cercana}
* * * F' <-- F(x) {coste mínimo cercano}
* * .Q FINSI
* .Q FINPARA
* xn+1 <-- x' {mejor solución cercana}
* LT <-- LT + [xn+1->xn] - [1º_si_llena] {incorpora como tabú}
* +Q SI F'< F* ENTONCES {si la mejor solución cercana es mínima}
* * x* <-- x' {nueva solución óptima}
* * F* <-- F' {nuevo valor óptimo}
* .Q FINSI
```

.Q FINMIENTRAS

Imprimir_Optimo(x^*) {se llega a la mejor}

Elementos básicos:

Los elementos básicos de la búsqueda tabú se describen a continuación:

- **Selección de una solución inicial x_0 .** Un factor muy importante a tener en cuenta es la posible influencia que tenga comenzar la búsqueda tabú con una solución inicial más o menos buena. Esta solución dependerá del algoritmo específico que la genera. Con una solución inicial buena, de bajo coste, generada de forma algorítmica, se puede pensar que es posible evolucionar, a corto plazo, hacia soluciones mejores, aunque podría suponer un gran perjuicio computacional si realmente evoluciona la búsqueda hacia regiones de soluciones más desfavorables. Es, por tanto, necesario evaluar la conveniencia de considerar un método algorítmico o no. En cualquier caso, siempre será posible generar una solución de forma aleatoria.
- **Elección del entorno $V(x_a)$.** Para evolucionar hacia otras soluciones, el algoritmo de búsqueda tabú selecciona éstas en un entorno de x_a . Hay que definir, por tanto, el concepto de solución cercana de x_a para proceder a seleccionar una nueva solución dentro de dicho entorno. Lo natural sería la selección completa ($V(x_a) - \{Lista\ Tabú\}$), evaluando cada una de las soluciones y quedándose con la mejor que no sea tabú. Según se defina el entorno, es decir, el conjunto de soluciones a las que se puede acceder desde x_a , así será su tamaño. Para realizar una búsqueda completa, es deseable que el tamaño no sea elevado. Si se considera un entorno de tamaño grande, con objeto de reducir el tiempo de computación, se puede realizar la búsqueda en un subconjunto tomado aleatoriamente, o bien realizar la búsqueda hasta que se mejora el coste de la solución actual.
- **Elección del tamaño de la lista tabú (L).** Varios autores toman el valor 7 como "número mágico" sin explicación lógica. Más recientemente, se toman valores dependientes del tamaño del problema. En cualquier caso, constituye un parámetro importante cuya influencia habría que analizar y del cual dependerá la evolución del algoritmo en gran medida.

- **Elección de los atributos para almacenar en la lista tabú.** Almacenar la descripción completa de las últimas soluciones exploradas y comprobar si cada movimiento se encuentra en la lista puede ocupar mucho tiempo. Como alternativa se puede almacenar un atributo representativo del movimiento (p.e., la diferencia de costes). Los atributos que se consideren, así como la forma de almacenarlos dependerán, en cierta medida, del problema a resolver.
- **Nivel de Aspiración.** Si todos los movimientos de la lista tabú se prohíben, se evita entrar en ciclos, pero se pueden perder movimientos que acerquen a mejores soluciones. En definitiva, el nivel de aspiración supone un criterio para aceptar soluciones que están incluidas en la lista tabú. Una posibilidad podría ser definir una función de aspiración F_a para cada coste, de forma que un movimiento tabú podría ser elegido si la nueva solución tiene un coste menor que $F_a(F(x_a))$, siendo x_a la solución actual. También se puede considerar como tal nivel el coste de la mejor solución alcanzada hasta el momento $F(x^*)$. Podría ocurrir que un movimiento tabú diese lugar a una solución cuyo coste fuese menor que dicho nivel. En tal caso, parece no tener sentido rechazarla. Este caso podría ser posible según sean los atributos considerados para caracterizar a un movimiento tabú.
- **Criterio de finalización.** Se puede establecer un número máximo de iteraciones, o un número máximo de pasos sin mejorar el coste.

1.5

Tratamiento Heurístico del Problema del Diseño

WLAN

El anterior procedimiento heurístico se ha descrito al objeto de resolver el problema **W NDP** que estábamos tratando, esto es, el problema de diseñar eficientemente una red WLAN, proponiendo un modelo de optimización para seleccionar tanto la localización como la potencia de canal de los APs (Puntos de Acceso). Como ya se ha señalado, el carácter NP-duro del modelo matemático resultante motiva el uso de procedimientos heurísticos de resolución, para obtener soluciones subóptimas en tiempo de computación razonable. Usualmente, el procedimiento heurístico seleccionado (el algoritmo Tabú, en

este caso) no se aplica de forma aislada, sino que se ejecuta una fase previa para, mediante un heurístico general adaptado al problema, obtener una solución factible inicial. Dicha solución es posteriormente mejorada mediante el heurístico de búsqueda que se haya seleccionado (al Tabú, en nuestro problema).

En el caso del problema WNDP, el tratamiento hecho por *Bahri et al.* (2005) trabaja de la forma descrita, esto es:

- Fase Previa: En la cual se aplicó un heurístico voraz (Greedy) para obtener una solución inicial factible. El voraz más usual es el que parte de una solución vacía y, en cada iteración, sucesivamente, dicha solución se va incrementando con el candidato que optimice el decremento (o incremento) de la función objetivo (decremento en WNDP, dado que es un objetivo de minimización).
- Algoritmo Tabú: Donde se aplicó el procedimiento de Búsqueda Tabú descrito para mejorar dicha solución inicial.

El algoritmo se aplicó a un ejemplo numérico, consistente en el diseño de una red WLAN en edificios de hasta 10 plantas, cada una de 5000 m² de superficie, y donde se consideraron 400 localizaciones potenciales para instalar los APs (esto es, $|J| = 400$). Se introdujeron una serie de hipótesis sobre los parámetros en función de las características físicas del entorno (por ejemplo, sobre los valores de los α y β) y, en cada instancia del problema, $|I|$ puntos de demanda y $|J|$ puntos correspondientes a las potenciales ubicaciones fueron localizados en el edificio siguiendo una distribución uniforme. Los resultados computacionales mostraron que el procedimiento heurístico basado en la búsqueda Tabú produjo relativamente buenas soluciones con bastante rapidez. También se constató la influencia que la arquitectura del entorno tiene en el diseño de la red. Así por ejemplo se compararon los resultados obtenidos para un edificio de 10 plantas (torre de oficinas) y para un edificio de una planta (un aeropuerto) obteniéndose que en la mayoría de los casos se obtienen mejores soluciones para el edificio de una planta. Esto se debe al hecho de que la propagación de la señal es mejor en un edificio de una planta que en uno de 10 plantas.

1.6

Otros Modelos para el Diseño de Redes WLAN

En este apartado comentamos brevemente algunas variantes al anterior modelo. El problema WNDP que se acaba de ver aborda la minimización del número de APs (junto con su localización y potencia) para que la red preste servicio (bajo unos requerimientos establecidos) a un conjunto de puntos de demanda, y la formulación matemática del problema es un problema de optimización que sigue los patrones de un problema de optimización combinatoria clásico: El problema del conjunto cobertura de mínimo cardinal, donde la función objetivo es lineal. Como se ha visto, el carácter NP-duro del problema requiere el uso de heurísticos para obtener soluciones sub-óptimas.

Sin embargo el anterior modelo no considera la posible superposición de áreas cubiertas por distintos puntos de acceso, lo que puede provocar soluciones que limiten la capacidad del sistema, o que den lugar a zonas cubiertas con distinta intensidad. Es por ello que en el diseño de redes WLAN se han estudiado variantes al anterior modelo. Entre ellas, citamos los trabajos de *Amaldi et al.* (2004 y 2005), que establecen formulaciones para el problema del diseño de WLAN que tienen en cuenta la superposición de la cobertura entre APs y su impacto en la capacidad de la red. Para ello, utilizan funciones objetivo hiperbólicas y cuadráticas en la formulación matemática del problema (en lugar de la anterior función objetivo lineal). Los problemas resultantes siguen siendo NP-duros, lo que implica el tratamiento heurístico de los problemas y la posterior discusión de los resultados.