



UNIVERSIDAD DE LA REPÚBLICA
FACULTAD DE INGENIERÍA



Wi-Fi Controller

MEMORIA DE PROYECTO PRESENTADA A LA FACULTAD DE
INGENIERÍA DE LA UNIVERSIDAD DE LA REPÚBLICA POR

Guillermo Juliano, Camila Pertusso, Maximiliano
Rodríguez

EN CUMPLIMIENTO PARCIAL DE LOS REQUERIMIENTOS
PARA LA OBTENCIÓN DEL TÍTULO DE
INGENIERO ELECTRICISTA.

TUTOR

Gabriel Gómez..... Universidad de la República
Juan Pablo González..... Universidad de la República

TRIBUNAL

Germán Capdehourat Universidad de la República
Juan Pablo González..... Universidad de la República
Claudina Rattaro Universidad de la República

Montevideo
miércoles 27 julio, 2016

Wi-Fi Controller, Guillermo Juliano, Camila Pertusso, Maximiliano Rodríguez.

Esta tesis fue preparada en L^AT_EX usando la clase iietesis (v1.1).

Contiene un total de 127 páginas.

Compilada el miércoles 27 julio, 2016.

<http://iie.fing.edu.uy/>

Un buen jugador gana un partido; un buen equipo
gana campeonatos.

MICHAEL JORDAN

Esta página ha sido intencionalmente dejada en blanco.

Agradecimientos

A nuestras familiares y amigos que nos apoyaron a lo largo del proyecto...

Al personal de Plan Ceibal que colaboró en todo momento, Rossina, Federica y Haldó por su atención y tiempo...

Al personal de los Liceos N° 16 y 36 por permitirnos trabajar sin problemas.

A nuestros tutores por la ayuda y recomendaciones.

A nuestros compañeros de Ledefyl, que nos tuvieron paciencia durante todo este año.

A todos ellos muchas gracias!

Esta página ha sido intencionalmente dejada en blanco.

Resumen

WICO (WiFi Controller) surge de la necesidad del Plan Ceibal [1] de contar con una controladora centralizada capaz de comunicarse con los Access Points (APs) de la marca MikroTik [2], ya que los mismos carecen de una solución propietaria. El objetivo de la controladora es asignar canales y potencia a los APs de manera automática.

Se diseñó e implementó el software controlador que, en base a información tomada desde los APs, decide cuál es la mejor configuración para la red. Para ello se implementaron tres algoritmos distintos para la asignación de canales y dos para la asignación de potencia. Antes de realizar el desarrollo del software sobre el lenguaje de programación Python [3], se realizaron simulaciones utilizando MATLAB [4] para la validación de los algoritmos.

Luego de haberse obtenido la primera versión de la controladora se construyó una maqueta para hacer pruebas sobre los APs, utilizando un servidor de características similares a los que se encuentran en las instituciones beneficiarias de Plan Ceibal. Para las mismas se recabó información de dos centros educativos. Luego de depurar la solución, se llevó a cabo la puesta en funcionamiento de la controladora en una de estas instituciones para corroborar el correcto funcionamiento del sistema en un caso real.

El producto final es una controladora centralizada capaz de asignar canales y potencia de transmisión a los APs de forma automática y eficiente. Este servicio centralizado funciona en principio localmente en el servidor del centro educativo. Además si es ejecutado en forma periódica, brinda robustez frente a cambios en la topología de la red. El software diseñado es modular y está pensado para que se expanda a otros proveedores de hardware más allá de MikroTik.

Esta página ha sido intencionalmente dejada en blanco.

Tabla de contenidos

Agradecimientos	III
Resumen	V
1. Introducción	1
1.1. Descripción del proyecto	1
1.2. Objetivo y alcance del proyecto	2
2. Marco Teórico	5
2.1. Generalidades 802.11	5
2.1.1. Modelo básico	5
2.1.2. Medio compartido	6
2.1.3. Técnica de Acceso al medio	7
2.1.4. Canales en la banda 2.4GHz	8
2.2. Interferencia	8
2.2.1. Interferencia no Wi-Fi	8
2.2.2. Interferencia Co-Canal	8
2.2.3. Interferencia de canal adyacente	9
2.3. Optimización	9
2.3.1. Control de Potencia	9
2.3.2. Asignación de canales	10
2.4. Grafos	10
2.4.1. Introducción	10
2.4.2. Grafo	10
2.4.3. Grafo de Interferencia	11
2.5. Estado del Arte	11
2.6. Definiciones	14
2.6.1. Access Point Interno	14
2.6.2. Access Point Externo	14
2.6.3. Canales asignables	14
2.6.4. Access Point Vecino	14
2.6.5. Grado de saturación	14
2.6.6. Grado	14
2.6.7. Factor de interferencia para un canal	15
2.6.8. Factor de interferencia total de la red	17

Tabla de contenidos

3. Algoritmos de Asignación de Canales	19
3.1. Objetivo	19
3.2. Grafo de estado inicial	20
3.3. Flexibilidad de los algoritmos	21
3.3.1. Criterios de mejor canal (CMC)	21
3.3.2. Factor de canales adyacentes	21
3.3.3. Criterios de desempate	22
3.4. Algoritmos	23
3.4.1. Más Vecinos Asignados (MVA)	23
3.4.2. Menos Canales Libres (MCL)	29
3.4.3. Menos Comprometido Primero (MCP)	36
3.4.4. FI_{min} : Función auxiliar	43
4. Algoritmos de Potencia	45
4.1. Objetivo	45
4.2. Ejemplo	46
4.3. Algoritmo de control de potencia - Método Iterativo	47
4.3.1. Descripción del algoritmo	47
4.4. Algoritmo de control de potencia - Método Friis	48
4.4.1. Descripción del algoritmo	49
5. Implementacion	51
5.1. CentOS	51
5.2. MikroTik	52
5.2.1. MikroTik RB433	52
5.2.2. Interacción con la controladora	52
5.2.3. Comandos utilizados	53
5.3. Python	55
5.4. Pseudo código / Descripción del software	56
5.4.1. Archivo de configuración general - wico.conf	57
5.4.2. Archivo de configuración de APs - AP.csv	57
5.4.3. Lectura del archivo de configuración - LeerConfig.py	57
5.4.4. Creación de APs - CargarConfigcsv.py	57
5.4.5. Clase AccessPoint	57
5.4.6. Clase APExt	59
5.4.7. Definición de los algoritmos de asignación de canales - algoritmo_canales.py	59
5.4.8. Definición de los algoritmos de control de potencia - algoritmos_potencia.py	60
5.4.9. Cambio de configuración - configurar_AP.py	61
5.5. Posibilidad de expansión	61
6. Simulaciones y Puesta en Funcionamiento	63
6.1. Herramientas de Simulación	63
6.1.1. Representación gráfica de cobertura - graphAP.m	63
6.1.2. Espectro de frecuencias - plot_potencia.m	64

Tabla de contenidos

6.1.3. Simulacion.m	65
6.1.4. SimulacionEstadistica.m	65
6.1.5. Obtención de datos en archivo csv - escribir_archivo.py . . .	65
6.1.6. Simulación desde un archivo csv - wico_test.py	66
6.2. Simulaciones	67
6.2.1. Casos Particulares	71
6.3. Pruebas en Caso Real	87
6.3.1. Primera Visita o relevamiento de datos	87
6.3.2. Implementación en caso real	90
6.4. Comparación entre Algoritmos	92
7. Conclusiones	95
7.1. Conclusiones generales	95
7.2. Trabajos futuros	96
A. Manual de usuario	97
A.1. Procedimiento de instalación	97
A.2. Funcionalidades	97
A.2.1. Ejecución del programa principal	97
A.2.2. Módulo de Pruebas - wico_test.py	98
A.2.3. Configuración desde archivo csv - wico_configAP.py	99
B. Gestión de proyecto	101
Referencias	107
Índice de tablas	109
Índice de figuras	110
Contenido del CD	113

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 1

Introducción

Wi-Fi es una de las tecnologías de comunicación inalámbrica más usadas hoy en día. El requerimiento esencial para el usuario es que esté siempre disponible y con la mejor calidad posible. Para lograr esto se necesitaron muchos años de investigación y desarrollo hasta llegar a un estándar mundial. El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) define en su estándar 802.11 [5] especificaciones de control de acceso al medio (MAC) y capa física (PHY) para implementar en redes de área local inalámbricas (WLAN: Wireless Local Area Network).

Más allá de las ventajas que tienen las tecnologías inalámbricas como la movilidad y los bajos costos de instalación, existen problemas de desempeño por ser el aire un recurso compartido. Wi-Fi opera en bandas de frecuencias no licenciadas, es decir que cualquier persona puede transmitir en ellas sin la necesidad de tener un permiso para hacerlo. Esto implica mayor probabilidad de interferencia, por lo cual es importante un buen plan de optimización en el uso de los recursos del medio inalámbrico.

Este proyecto se centró en la optimización de recursos inalámbricos dentro del marco del Plan Ceibal. Ceibal es una institución gubernamental que se encarga de gestionar el Programa para la Conectividad Educativa de Informática Básica para el Aprendizaje en Línea. El objetivo de la misma es promover la inclusión digital para un mayor y mejor acceso a la educación digital. El Plan se basa en un completo sistema que busca garantizar el uso de recursos tecnológicos, la formación docente, elaboración de contenidos, además de la participación social. Para llevar esto a cabo se realizó el despliegue de una red inalámbrica en todos los centros educativos del país. [1]

1.1. Descripción del proyecto

El Plan Ceibal tiene como uno de sus objetivos brindar cobertura Wi-Fi a todos los centros educativos del país. En la etapa inicial del proyecto se instalaron

Capítulo 1. Introducción

Access Points (AP) de la marca MikroTik. Estos APs carecen de un controlador centralizado, llevando a que la asignación tanto de frecuencia (canal) a utilizarse, así como la potencia de transmisión y otros parámetros que afectan el desempeño de la red, se configuren de forma manual y generalmente sólo al momento de instalación de los equipos.

El hecho de que sean cientos de centros educativos configurados de forma manual, conlleva a que además de los altos costos operativos, las asignaciones se hagan de una manera poco rigurosa dejando por fuera factores importantes como interferencias de redes externas. Estas asignaciones además no tienen en cuenta que pueden existir cambios en la infraestructura del local, lo cual implica que la configuración inicial deje de ser la óptima.

El proyecto surgió de la necesidad de Ceibal de contar con un sistema centralizado que optimice la asignación de recursos de forma automática. Este se basa en información obtenida de los APs para decidir cuales son los valores óptimos para la configuración de canales y potencia de cada AP de la red.

1.2. Objetivo y alcance del proyecto

El objetivo de este proyecto es implementar un controlador que se comunique con los Access Points internos de la red Ceibal para obtener información de la misma. Luego con estos datos recabados decida cuál es la mejor asignación de canales y potencia de transmisión, y finalmente configure los Access Points con los valores que considere óptimos. Este proceso se debe realizar de forma periódica para garantizar el óptimo despliegue durante el mayor tiempo posible.

Para medir cuantitativamente los objetivos específicos del proyecto, se definió el *factor de interferencia*, este indicador tiene en cuenta la interferencia co-canal y de canales adyacentes. En conjunto con el cliente se acordó que 5dBm por encima de una configuración manual, era un margen que no afecta en forma significativa el desempeño de la red y es aceptable para un futuro despliegue de la controladora Wico dentro del Plan Ceibal.

El alcance del proyecto abarca el diseño y la implementación del correspondiente software controlador que se instalará en los servidores centrales existentes en cada centro educativo. El sistema deberá comunicarse con los Access Points del local para obtener la información necesaria y realizar los cambios en los parámetros correspondientes. El controlador deberá detectar y actuar ante casos de falla de alguno de los Access Point, reasignando la configuración óptima bajo las nuevas condiciones. En principio el software estará diseñado para interactuar con Access Points de la marca MikroTik, pero deberá estar implementado de forma modular para poder ampliarse a otras marcas de la forma más sencilla posible.

1.2. Objetivo y alcance del proyecto

El proyecto deberá constar del diseño de al menos dos algoritmos de elección de canal y potencia de transmisión, sin recurrir a la reubicación física de los Access Points. Se deberá incluir también la validación del producto mediante diferentes tipos de simulaciones y el despliegue en un caso real en un centro educativo.

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 2

Marco Teórico

Con el fin de comprender el problema de la asignación de recursos inalámbricos, se estudiaron las características y funcionamiento del protocolo IEEE 802.11 [5]. Además se investigó sobre la existencia de trabajos académicos sobre soluciones a problemas similares. En este capítulo se presenta un resumen de los puntos más relevantes para el entendimiento del proyecto, al final del mismo se definen conceptos que serán utilizados para explicar los algoritmos más adelante descriptos.

2.1. Generalidades 802.11

El estándar IEEE 802.11 define el uso de los dos niveles inferiores del modelo OSI (capa física y capa de enlace de datos), especificando sus normas de funcionamiento en una red de área local inalámbrica (WLAN). A continuación se detallan los conceptos más importantes para entender el funcionamiento básico del mismo.

2.1.1. Modelo básico

Wi-Fi es una de las más importantes y populares tecnologías inalámbricas. El estándar incluye la definición de varias familias como son 802.11a/b/g/n/ac. Entre las principales ventajas de este tipo de red están el bajo costo, la facilidad de despliegue y que funciona sobre bandas de frecuencias en las que cualquier dispositivo puede transmitir sin la necesidad de adquirir licencias.

En la mayoría de los casos, una red Wi-Fi se implementa en una infraestructura similar a la de la figura 2.1. En la misma se muestra cómo los dispositivos de los usuarios se conectan a la red por el medio inalámbrico a través de los Access Points.

Capítulo 2. Marco Teórico

Un Access Point es un equipo que funciona de vínculo entre los dispositivos inalámbricos y los servicios cableados. La mayoría de los APs cumplen con el estándar 802.11, lo que permite una compatibilidad con una gran variedad de equipos inalámbricos como teléfonos celulares, laptops, etc. En las redes domésticas generalmente se usa un sólo AP para conectar todos los dispositivos de la casa, en redes de mayor porte como las corporativas se pueden llegar a encontrar decenas de APs.

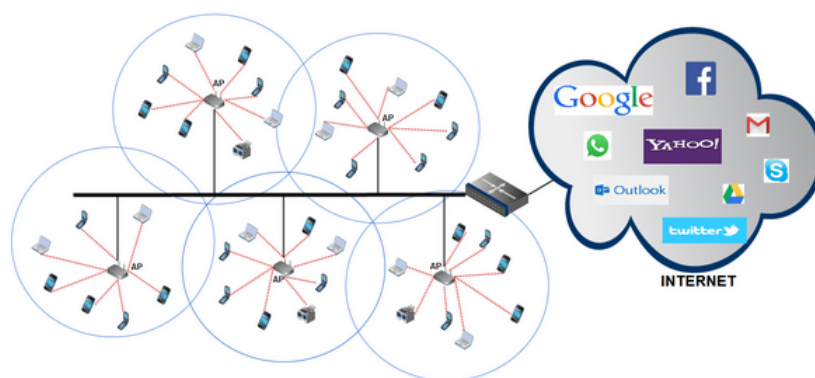


Figura 2.1: En rojo se representan los vínculos entre los APs y sus clientes

2.1.2. Medio compartido

Cuando se dice que una comunicación es inalámbrica, lo que se quiere decir es que el medio por el cual viaja la información es el aire. Para que la gran cantidad y diversidad de tecnologías que comparten el medio funcionen simultáneamente, los datos se envían a una frecuencia diferente. El receptor filtra todas las señales que llegan por el aire y bajo condiciones de SNR aceptables se queda sólo con las que son de su interés.

En la figura 2.2 se muestra el espectro de frecuencias y los diferentes tipos de información que se transmiten. En la figura 2.3 se muestra dónde se ubican en el espectro las tecnologías de transmisión de datos más comunes

En la actualidad dos bandas de frecuencias están disponibles para el estándar IEEE 802.11, la banda 2.4 GHz (también llamada ISM: Industrial, Scientific and Medical) y la 5 GHz. Una particularidad importante de estas bandas es que no son licenciadas. A diferencia de otras bandas como por ejemplo la de radio AM, donde una emisora debe pagar por la reserva del uso exclusivo de un espacio del espectro, en 2.4GHz o 5GHz cualquier dispositivo puede transmitir lo que quiera sin que se le exija pagar por ello. Esta libertad que existe, hace que al receptor le lleguen varias señales que se interfieren y en algunos casos hace que la demodulación de la señal sea imposible.

2.1. Generalidades 802.11

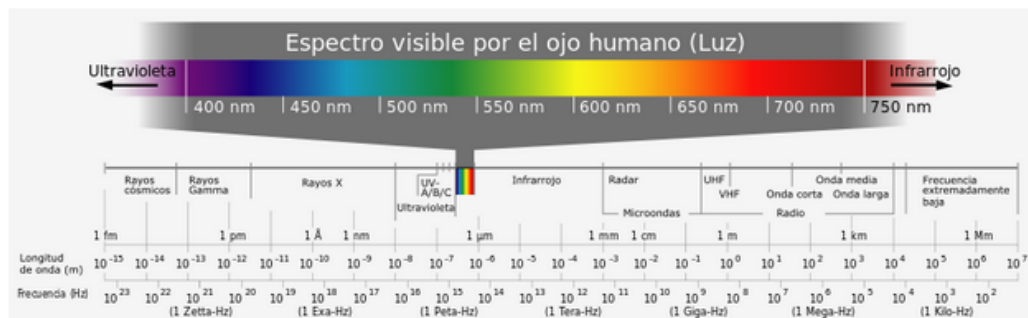


Figura 2.2: Uso del Espectro de frecuencias - Imagen obtenida de [6]

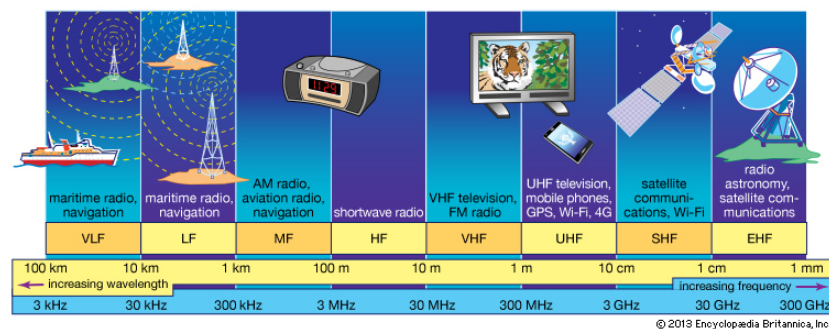


Figura 2.3: Ubicación espectral de las tecnologías de transmisión - Imagen obtenida de [7]

2.1.3. Técnica de Acceso al medio

Debido a las características del medio, compartido y no licenciado, es necesario un protocolo de acceso al medio que haga posible que múltiples estaciones lo utilicen en simultáneo sin problemas. Para esto el estándar define en la capa MAC el protocolo de contienda CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). Es una técnica en la cuál tanto los Access Points como clientes sensan el medio antes de transmitir. Cuando un dispositivo desea hacerlo y encuentra el medio libre, sensa durante un tiempo aleatorio (Backoff) en búsqueda de que ningún otro dispositivo comience a transmitir, si después de ese tiempo el medio continua libre comienza la transmisión. Cuando un dispositivo encuentra el medio ocupado, espera un tiempo aleatorio para volver a sensar el medio para intentar la transmisión.

El sensado en Wi-Fi se compone de dos funciones CCA (Clear Channel Assessment) y NAV (Network Allocation Vector). CCA es un sensado a nivel físico, donde el dispositivo mide la energía recibida en la frecuencia de trabajo. NAV es un sensado virtual de la portadora, donde el dispositivo extrae del encabezado de las tramas recibidas el tiempo reservado (Duration Field) por las mismas, durante ese tiempo el dispositivo asume que el canal está ocupado.

Con el fin de disminuir la probabilidad de las colisiones, existe un mecanismo

Capítulo 2. Marco Teórico

opcional de acceso al medio: RTS/CTS (Request-to-Send/Clear-to-Send) donde antes de transmitir un paquete el transmisor reserva el medio a través de un mensaje de RTS y el receptor responde con un mensaje de CTS si el medio está libre. [8] [9]

2.1.4. Canales en la banda 2.4GHz

El proyecto se enfoca en el estándar 802.11g sobre la banda de 2.4 GHz, que es el que tiene desplegado Ceibal al momento. Los equipos pueden elegir entre 11 canales separados 5MHz con un ancho de 22MHz como muestra la figura 2.4. Como es de esperar, los canales que no están a más de 25Mhz se solapan, esto implica que se generen interferencia entre ellos. Se identifican los canales 1, 6 y 11 como no solapados.

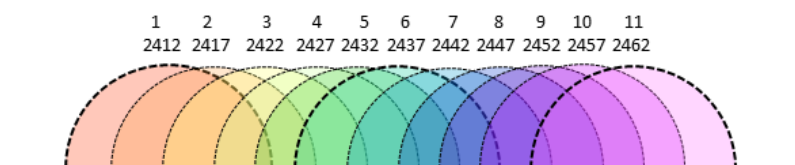


Figura 2.4: Canales en 2.4Ghz

2.2. Interferencia

Debido a la naturaleza del medio inalámbrico en que el protocolo IEEE 802.11 funciona, la interferencia en radiofrecuencia es un gran problema. Se estudiaron tres tipos de interferencia durante el desarrollo del proyecto.

2.2.1. Interferencia no Wi-Fi

En la banda ISM 2.4 GHz existen varias tecnologías no Wi-Fi como Bluetooth, periféricos de computadoras, teléfonos inalámbricos y emisiones de hornos microondas. Como no se pueden evitar y por lo general abarcan todos los canales, se desprecian en el cálculo de la interferencia.

2.2.2. Interferencia Co-Canal

El hecho de que dos Access Points trabajen en el mismo canal, puede causar que las transmisiones de ambos se interfieran. Gracias a que el estándar define el uso de CSMA, estas colisiones se evitan en su mayoría. El principal efecto de este tipo de interferencia es que los clientes de los dos APs tienen que compartir el

medio, o sea el tiempo disponible para transmitir, generando una degradación en la capacidad de la celda.

2.2.3. Interferencia de canal adyacente

Cuando dos Access Points usan canales que se solapan, causan interferencia entre ellos. Dadas las características del sensado del protocolo CSMA/CA, detectar que un AP está transmitiendo en un canal adyacente es más difícil que cuando transmiten en el mismo canal. Esto causa que las colisiones entre las transmisiones de los clientes de APs en canales adyacentes sean más frecuentes.

2.3. Optimización

Para intentar disminuir la interferencia y los efectos adversos que esta conlleva, es necesario un proceso de optimización y planeamiento antes y durante el despliegue de la red. Son varios los factores que se pueden alterar para conseguirlo, como ser la potencia de transmisión de los Access Points, el canal asignado o la ubicación de cada Access Point en la zona a cubrir.

2.3.1. Control de Potencia

El nivel de potencia de transmisión afecta la intensidad de señal, o sea afecta los nodos cercanos que pueden sensar las transmisiones de un Access Point. Los mecanismos de control de potencia son importantes para reducir la interferencia en el medio compartido y de este modo cooperar con la reutilización de canales. El nivel de potencia de transmisión afecta a la calidad de la señal recibida en el receptor y el área de cobertura, por ende la magnitud de la interferencia causada en los equipos cercanos. Son variadas las técnicas de optimización que se hallaron para la elección de potencias de transmisión. En general estas técnicas consisten en escoger una función objetivo y encontrar los valores de las variables que la maximizan o minimizan. Mientras algunas realizan el cálculo a partir de requerimientos de clientes a nivel de relación señal a ruido, otras lo hacen utilizando como función objetivo la interferencia en los APs o retrasos en la red. [10]

2.3.2. Asignación de canales

El protocolo 802.11g define varias frecuencias para transmitir, como se dijo anteriormente no es conveniente el uso de todos los canales en un mismo dominio de transmisión ya que puede haber solapamiento entre los mismos. Para optimizar el uso del espectro evitando solapamiento se utilizan los canales 1, 6 y 11 en los algoritmos de asignación. El objetivo es minimizar un indicador que represente los efectos adversos de la interferencia. Ejemplo de estos indicadores son: el uso efectivo de cada canal, la relación señal a ruido a nivel de usuario o la interferencia entre los Access Point. En general los problemas de asignación de canales se modelan como un problema de coloreado de grafos, siendo los vértices los AP y las aristas relaciones de interferencia. [11]

2.4. Grafos

2.4.1. Introducción

La teoría de grafos comienza en el siglo XVIII con un artículo [12] publicado por el matemático suizo Leonhard Euler. Los grafos sirven para analizar problemas de relaciones entre objetos. En el diseño de la solución al problema de asignación de frecuencias se optó por pensar el problema en términos de grafos coloreados. Un grafo se compone de un grupo de vértices y un grupo de aristas que conectan dichos vértices. El colorear un grafo es mapear cada vértice con un elemento de un conjunto finito de colores. Un coloreo ideal se da cuando no hay dos vértices conectados por una arista con el mismo color asignado. Hay casos en que el ideal no es posible porque el numero de colores disponible es finito.

2.4.2. Grafo

Un grafo se define como un par (V, E) , donde V es un conjunto de elementos denominados vértices y E es un subconjunto de pares no ordenados de vértices que reciben el nombre de aristas. Si $V = \{v_1, \dots, v_n\}$, los elementos de E se representan de la forma $\{v_i, v_j\}$, donde $i \neq j$. Dos vértices v_i y v_j se dicen adyacentes si $\{v_i, v_j\} \in E$.

La figura 2.5 proporciona un ejemplo de un grafo donde $V = \{a, b, c, d\}$ con $E = \{(a, b), (b, c), (a, c), (d, b)\}$.

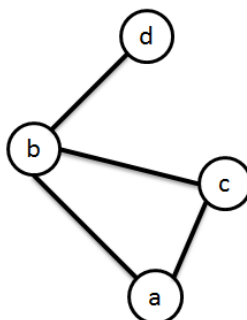


Figura 2.5: Ejemplo de un grafo

2.4.3. Grafo de Interferencia

Para la formulación del problema se considera a cada Access Point como un vértice del grafo. Las aristas se construyen a partir de la interferencia que se pueden causar dos APs si transmiten en el mismo canal, o sea dos APs son adyacentes si uno detecta la presencia del otro, en general si la potencia vista es mayor a -100 dBm.

Por otro lado el conjunto de colores serían los canales a asignar, en el estándar 802.11g existen tres canales que no se solapan entre sí: 1 (2412MHz), 6 (2437MHz) y 11 (2462MHz). Por lo tanto un algoritmo de asignación de frecuencias no es más que un modo de recorrer el grafo de interferencia construido a partir de la red.

2.5. Estado del Arte

Durante la primera etapa del proyecto se realizaron consultas de trabajos académicos para buscar procesos de optimización de redes Wi-Fi con el fin de entender cómo se encaran este tipo de problemas. Este estudio ayudó para obtener las ideas para diseñar nuevos algoritmos de optimización.

En primer lugar se encontraron dos clases de algoritmos, los que tienen la posibilidad de reubicar los APs y los que no [13]. En el caso de este proyecto, no existe la posibilidad de modificar la ubicación de los APs debido al alto costo operativo que esto implica, por lo que el estudio se concentró en los algoritmos que asignan canales o potencia a los APs dada una ubicación fija.

En [13] se definen dos tipos de infraestructuras Wi-Fi dependiendo de cómo están manejadas: *coordinada* y *descoordinada*. En las infraestructuras coordinadas se tiene el control sobre todos los APs, esto desemboca en una mejor asignación de recursos. Por otro lado, en las descoordinadas solo se tiene el control sobre un AP mientras que los canales y potencia de los APs cercanos están administrados por otras entidades. En este proyecto se tuvo en consideración una mezcla de los

Capítulo 2. Marco Teórico

dos tipos de infraestructura, ya que se tiene el control de todos los APs dentro de la red pero existen otros APs a los cuales no se tiene acceso (por ejemplo un AP de una casa vecina). Este enfoque mejora la asignación de los APs, ya que sólo teniendo en cuenta los APs internos a la red se pierde la información de los APs externos, un factor que afecta considerablemente el resultado de la asignación. Además trabajar con los APs como entes individuales, donde cada uno mira su entorno, no es eficiente porque no se tiene en cuenta que la configuración de los otros APs de la red también va a ser alterada. Una asignación coordinada tiene mejor desempeño.

El documento [14] muestra un estudio sobre las redes Wi-Fi en entornos urbanos. En primer lugar las define como caóticas, ya que la cantidad de APs es muy grande y además la configuración de los mismos no está hecha en forma inteligente. El estudio muestra que estadísticamente hay canales que son más usados que otros, por ejemplo el canal 6 es usado por el 41.15% de los APs, como se ve en la tabla 2.1. En este mismo documento se revela un estudio donde se muestra el efecto en el desempeño del tráfico cuando se asigna de forma óptima los canales o la potencia de los APs, y se compara con la situación en implementaciones caóticas mostrando una mejora significativa.

Canal	Porcentaje de uso
1	3.04
2	12.29
3	3.61
4	1.03
5	1.13
6	41.15
7	1.75
8	1.12
9	1.31
10	3.42
11	11.04

Tabla 2.1: Porcentaje de uso por canal

La técnica de diseñar el problema de asignación de frecuencias como un problema de coloreo de grafos es aplicada en la mayoría de los algoritmos estudiados. Son variadas las técnicas de recorrido de grafos, por ejemplo la citada en un trabajo realizado en Aachen University [15], en el que se utiliza el método de coloración DSATUR [16]. DSATUR define el grado de saturación de un vértice como la cantidad de colores utilizados por sus vecinos. Luego comienza por colorear el vértice

con mayor grado de saturación siempre coloreando con el menor color posible, es decir si el color 1 esta disponible se pinta con ese, si no se pasa al siguiente. La idea de este algoritmo es usar la menor cantidad de colores posibles. En este proyecto el objetivo no es pintar con la menor cantidad de colores sino mejorar la utilización del espectro de frecuencias, eligiendo siempre dentro de los tres canales no solapados.

En el artículo [13] se propone un algoritmo que utiliza el estándar 802.11F (IAPP - Inter AP Protocol) para que los Access Points intercambien información entre ellos. Cada AP envía la información acerca de la potencia con la que ve a otros APs. El algoritmo propone un protocolo similar a OSPF, todos los APs tienen la información del entorno de los otros nodos, crean un grafo de interferencia y mediante algún algoritmo de coloreo asignan su canal. Este enfoque es muy difícil de implementar en la realidad, ya que se necesitaría desarrollo en cada proveedor de hardware. Una solución que se encontró en este proyecto es centralizar la información en una controladora. El sistema consulta a todos los Access Points por la información de adyacencias, construye el grafo de interferencia, calcula la asignación óptima mediante diferentes técnicas de coloración y configura los APs.

Además de diseñar la forma en la que el grafo de interferencia es recorrido, se tuvo que decidir la manera de escoger el mejor canal una vez elegido el AP a asignar. El espectro de opciones es muy amplio, algunos trabajos se basan en encontrar el canal menos congestionado [13], otros en minimizar la interferencia entre los APs [17], minimizar la relación señal interferencia (SIR) a nivel de usuario [18] y otros intentan mejorar el ancho de banda de los usuarios [13]. Usar el canal menos congestionado en entornos donde los usuarios son muy dinámicos, implica que para tener siempre una asignación óptima, hay que recalcular los canales con mayor frecuencia. Esto conlleva también a una mala experiencia del usuario, ya que cada vez que se reasigna un canal, se pierde la asociación entre cliente y AP. Otro método que se descartó es el de obtener datos del usuario, ya que se entendió que no todos los APs cuentan con esta información. Dados los estudios vistos en [14] se concluyó que minimizando la interferencia entre los APs, se obtiene también una mejora del ancho de banda para los usuarios. Por esto se decidió utilizar la interferencia entre APs como la función objetivo a minimizar, teniendo en cuenta tanto la interferencia co-canal como la de canal adyacente.

A partir de [17], se obtuvo la idea de diseñar un indicador que tenga en cuenta la interferencia generada por los Access Points vecinos de un determinado AP y elija su canal de forma tal de minimizar dicho indicador.

2.6. Definiciones

En esta sección se definen los conceptos que más adelante serán utilizados para explicar el funcionamiento de los algoritmos desarrollados. Además se describen algunas consideraciones necesarias a tener en cuenta para comprender los mismos.

2.6.1. Access Point Interno

Son los Access Points que están dentro de la red que se va a optimizar. Son los que van a ser configurados por el controlador.

2.6.2. Access Point Externo

Son Access Points que no están dentro de la red a optimizar, pero que interfieren en la performance de la misma. Por ejemplo un router Wi-Fi de un vecino.

2.6.3. Canales asignables

Para evitar la asignación de canales que se solapan, los algoritmos solo asignan a APs internos los canales 1, 6, u 11. De esta forma se asegura que al menos entre los APs internos no hay interferencia co-canal. Los APs externos al no ser controlados por el algoritmo pueden tener cualquier canal entre 1 y 11.

2.6.4. Access Point Vecino

Un AP B es vecino del AP A, si la potencia de transmisión de B es suficiente para ser recibida por A. Generalmente la potencia recibida mínima con la que A ve a B es -100dBm.

2.6.5. Grado de saturación

Se define grado de saturación de un AP a la cantidad de canales asignables diferentes que tienen sus vecinos. Por ejemplo si un AP tiene cuatro vecinos con canales asignados 1, 1, 6 y 1 respectivamente el grado de saturación de ese AP es 2 ya que los canales 1 y 6 son usados por sus vecinos.

2.6.6. Grado

Se define grado de un AP como la cantidad de vecinos que este posee.

2.6.7. Factor de interferencia para un canal

El factor de interferencia es un indicador que se creó para cuantificar el uso efectivo de los canales. Este sirve para obtener cuál es el canal que se debe asignar a un AP. [17]

Interferencia co-canal

Es la suma de la potencia recibida de los APs que transmiten en el canal. La interferencia co-canal en el canal n (I_{cc}^n) se calcula como:

$$I_{cc}^n = \sum_k Pot_k \quad (2.1)$$

Con:

- k / AP_k transmite en el canal n
- Pot_k = Potencia recibida del AP_k

Interferencia de canal adyacente

Es la potencia recibida de los APs que trabajan en canales adyacentes, ponderadas según la distancia entre canales. La interferencia de los canales adyacentes al canal n se calcula como [17]:

$$I_{ady}^n = \sum_k \frac{I_{cc}^k}{|n - k|} \quad \text{con } |n - k| \leq 4 \quad (2.2)$$

Factor de interferencia FI_n

El factor de interferencia en el canal n se calcula como la suma de la interferencia co-canal del canal n y la de canal adyacente ajustada por la constante α . Esta constante hace que la influencia de la interferencia de canal adyacente sea mayor a la co-canal. Dadas las características del acceso al medio inalámbrico de Wi-Fi, el efecto de un AP transmitiendo en un canal adyacente es mucho más negativo que el de un AP transmitiendo en el mismo canal.

$$FI_n = I_{cc}^n + \alpha \times I_{ady}^n \quad (2.3)$$

Cálculo del FI_n : Ejemplo

En este ejemplo se muestra cómo se calcula el FI_6 en un entorno donde las potencias recibidas por APs vecinos son como se muestran en la figura 2.6. En la tabla 2.2 se detallan los valores de las potencias recibidas.

Capítulo 2. Marco Teórico

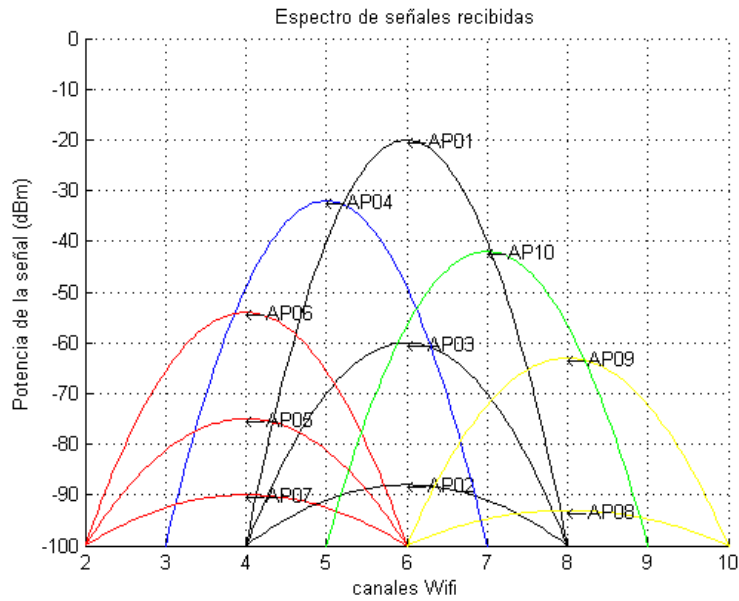


Figura 2.6: Ejemplo: Espectro de potencias recibidas

AP	Potencia Recibida (dBm)	Canal
AP01	-20	6
AP02	-88	6
AP03	-60	6
AP04	-32	5
AP05	-75	4
AP06	-54	4
AP07	-90	4
AP08	-93	8
AP09	-63	8
AP10	-42	7

Tabla 2.2: Potencias recibidas por los APs vecinos

Para el cálculo del FI_6 primero se calcula la interferencia co-canal en cada uno de los canales: ¹

$$I_{cc}^1 = -\infty$$

$$I_{cc}^2 = -\infty$$

¹Recordar que para sumar potencias que están expresadas en dB primero hay que pasarlas a Watts $P_{dBm} = 10 * \log(P_w)$

$$\begin{aligned}
I_{cc}^3 &= -\infty \\
I_{cc}^4 &= Pot_{AP05} + Pot_{AP06} + Pot_{AP07} \approx -20 \\
I_{cc}^5 &= Pot_{AP04} = -32 \\
I_{cc}^6 &= Pot_{AP01} + Pot_{AP01} + Pot_{AP01} \approx -54 \\
I_{cc}^7 &= Pot_{AP010} = -42 \\
I_{cc}^8 &= Pot_{AP09} + Pot_{AP08} \approx -63 \\
I_{cc}^9 &= -\infty \\
I_{cc}^{10} &= -\infty \\
I_{cc}^{11} &= -\infty
\end{aligned}$$

Luego se calcula el I_{ady}^6 , para ello sumamos los I_{cc}^k / $|6 - k| \leq 4$.

$$I_{ady}^6 = \sum_{k=2}^5 I_{cc}^k + \sum_{i=7}^{10} I_{cc}^i = -19,7dBm$$

Finalmente el Factor de interferencia en el canal 6 es:

$$FI_6 = I_{cc}^6 + \alpha \times I_{ady}^6 = -54dBm + 1000 \times -19,7dB \approx 11dBm^2$$

2.6.8. Factor de interferencia total de la red

Se define el Factor de interferencia total de la red FI_{tot} como la suma de los factores de interferencia recibidos por cada AP en el canal que están configurados. Este indicador muestra una imagen global de cómo está la red en cuestión de interferencia. En el proyecto se usó el FI_{tot} para comparar el desempeño de los algoritmos.

²Para esta suma también se deben pasar las potencias a watts (antes de multiplicar por α), en el ejemplo se muestra todo en dBm por simplicidad.

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 3

Algoritmos de Asignación de Canales

Como se comentó en capítulos anteriores WiFi transmite en la banda de 2.4GHz y 5GHz. Al ser éstas frecuencias no licenciadas, puede haber transmisiones de diferentes fuentes dentro de todo el espectro. Cuando dos equipos Wi-Fi están configurados para transmitir en la misma frecuencia, estos deben compartir el tiempo en el que transmiten, lo que causa que la capacidad se reduzca a la mitad. Peor es el caso en que dos equipos transmiten en canales adyacentes, como están escuchando en canales diferentes no pueden darse cuenta cuando el otro está transmitiendo y la probabilidad de que transmitan al mismo tiempo es muy alta, lo que se traduce en alta probabilidad de colisiones, o sea errores de transmisión.

3.1. Objetivo

Para evitar los efectos de la interferencia en la red, se diseñaron algoritmos de asignación de canales que tienen como objetivo intentar evitar que a Access Point vecinos se le asigne el mismo canal o canales adyacentes. Como los algoritmos solo pueden actuar sobre los APs internos y muchas veces la cantidad de APs vecinos es grande, la asignación sin interferencia co-canal o de canales adyacentes es un ideal que no siempre se puede lograr.

En el capítulo 2 se definió el factor de interferencia (FI_{tot}) con el fin de tener un indicador que sirva para comparar dos asignaciones. Una buena asignación minimiza el factor de interferencia, en el capítulo 6 se ve mediante simulaciones que no siempre es única la asignación con factor de interferencia mínima. Es por esto que además de intentar minimizar el factor de interferencia los algoritmos tienen en cuenta los siguientes detalles que hacen que la asignación sea óptima:

- 1- Minimizar el Factor de Interferencia
- 2- Hacer el mayor uso posible de los canales. Si dos asignaciones cumplen con el punto 1 se tomará como mejor la que utilice más canales diferentes
- 3- Utilizar para la asignación los canales 1(2412 MHz), 6 (2437 MHz) u 11 (2462 MHz)

3.2. Grafo de estado inicial

Antes de empezar a decidir cuál es la mejor asignación se tienen que obtener datos de los APs, en particular lo que interesa saber es qué APs son vecinos y con qué potencia se ve cada uno. En general este dato se puede obtener fácilmente.

Una vez obtenidos todos los datos de vecindades de los APs, se obtiene un grafo inicial donde los vértices del grafo son los APs y las aristas indican si hay vecindad entre ellos. En la figura 3.1 se muestra un ejemplo de un grafo inicial donde se diferencian los APs internos de los externos.

Los algoritmos de asignación de canal van recorriendo el grafo asignando canales a los APs, cuando a un AP se le asigna un canal se le llama Access Point Asignado. Como se puede observar en la figura 3.1 los APs externos ya tienen un canal asignado, por lo que se asumen como Access Points Asignados durante la aplicación del algoritmo.

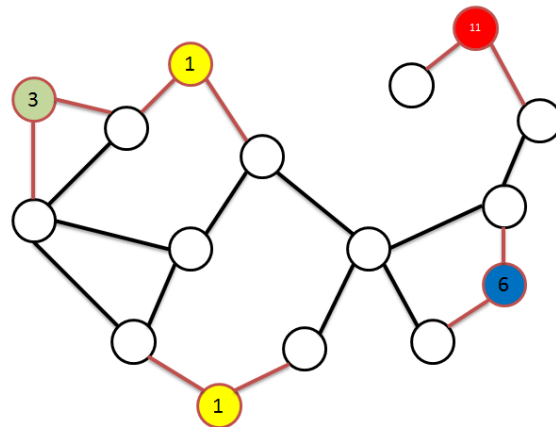


Figura 3.1: Grafo inicial - En negro y rojo se identifican los APs internos y externos respectivamente

3.3. Flexibilidad de los algoritmos

Para hacer el sistema lo más flexible se introducen variables de configuración para que el usuario las pueda ajustar a medida de su problema.

3.3.1. Criterios de mejor canal (CMC)

Para elegir el mejor canal para un AP se diseñaron dos alternativas. La primera es buscar el canal en el que el AP tenga el menor FI_n , y la segunda es buscar el canal donde el AP tenga menor cantidad de vecinos. Estas alternativas fueron diseñadas para poder comparar la performance y los resultados de los algoritmos usando diferentes funciones objetivo

- **Potencia:** En este modo, para la elección del mejor canal se tiene en cuenta la potencia recibida por APs vecinos en cada canal. El mejor canal (n) se calcula como:

$$n \quad / \quad \min\{FI_n = I_{cc}^n + \alpha \times I_{ady}^n\} \quad (3.1)$$

I_{cc}^n : Suma de la potencia recibida de los APs que transmiten en el canal n.

I_{ady}^n : Potencia recibida por APs que trabajan en canales adyacentes al n, ponderadas según la distancia entre canales.

α : Factor de canales adyacentes

- **Cantidad de vecinos:** En este modo, lo que se tiene en cuenta a la hora elegir el mejor canal, es la cantidad de vecinos que tiene cada AP si transmitiese en ese canal.

$$n \quad / \quad \min\{CV_n + \beta \times CV_{ady}^n\} \quad (3.2)$$

CV_n : Cantidad de APs en el canal n

CV_{ady}^n : Cantidad de APs en canales adyacentes al canal n, ponderadas según la distancia entre canales.

β : Factor de canales adyacentes

3.3.2. Factor de canales adyacentes

Recordando la observación que se mencionó en capítulos anteriores sobre el efecto que tiene la interferencia de canales adyacentes, se buscó darle un peso a ésta a la hora de elegir el mejor canal para un AP. En consecuencia, se introdujeron las variables α y β . Las mismas se tendrán en cuenta cuando se calcula el mejor canal por el criterio de potencia o cantidad de vecinos respectivamente. A mayor o

Capítulo 3. Algoritmos de Asignación de Canales

menor valor en cualquiera de ellas, el usuario podrá darle más o menos importancia a la interferencia de canales adyacentes.

Los valores de α y β se definen según la preferencia del administrador de la red. Para la puesta en funcionamiento del proyecto se eligieron valores que evitan completamente la interferencia de canal adyacente. Se llegó mediante simulaciones que con valores de $\alpha = 1000$ y $\beta = 100$ se logra ese objetivo.

3.3.3. Criterios de desempate

Al momento de decidir cuál es el mejor canal para un AP, son muchos los casos donde hay más de un canal que cumple las condiciones para minimizar el FI_{tot} . Con esto en mente, se implementaron distintos criterios de desempate. Esta alternativa resulta en una mejor utilización del espectro, en el capítulo Simulaciones [6] se muestran ejemplos donde se aprecian los efectos de la misma.

- **Normal:** Utilizando esta opción el software elegirá el primer canal de la lista de canales disponibles que cumplan con las condiciones. Por ejemplo si los canales a elegir son el 1 y 6, se elige el 1.
- **Random:** Utilizando esta opción el software elegirá de manera aleatoria entre las opciones que tenga para elegir el canal.
- **Round-robin:** Utilizando esta opción el software elegirá de manera secuencial entre las opciones que tenga para elegir el canal. Por ejemplo si se puede elegir entre los canales 1 y 6, y el último canal que se eligió fue el 1, entonces el software elegirá el 6.

3.4. Algoritmos

Durante el desarrollo del proyecto se diseñaron tres algoritmos de asignación de canales. Una vez conformado el grafo de interferencia asociado a la red, los algoritmos lo recorren de forma inteligente asignando los canales a los APs. La diferencia entre los algoritmos diseñados es la forma en que el grafo es recorrido. Como herramienta para comparar los algoritmos, se desarrollo una función auxiliar que se describe al final del capítulo.

3.4.1. Más Vecinos Asignados (MVA)

Este algoritmo se basa en encontrar el AP que tenga la mayor cantidad de vecinos asignados y asignar el mejor canal para este AP y todos sus vecinos. Cuando un AP tiene muchos vecinos asignados, la libertad de elección de canal se encuentra muy limitada. Dada la popularidad de Wi-Fi, la mayoría de las redes se ve muy cargada por APs externos, la idea de diseño de este algoritmo se basa en la suposición de que el grafo inicial tiene más APs asignados (externos) que sin asignar.

Descripción del Algoritmo

El funcionamiento del algoritmo se describe en los siguientes pasos:

- 1- Setear la potencia al máximo en todos los APs y construir el grafo inicial en base a las potencias recibidas en cada AP.
- 2- Calcular el grado de cada AP
- 3- Calcular la cantidad de vecinos asignados de cada AP
- 4- Calcular la potencia recibida en cada AP desde todos sus vecinos
- 5- Buscar el AP (AP_A) con más vecinos asignados. En caso de haber empate en la comparación elegir el que tenga mayor grado. Si sigue habiendo empate elegir el AP con menor potencia recibida de vecinos.
- 6- Asignar el mejor canal según el CMC (Criterio de Mejor Canal).
- 7- Buscar el (AP_B) entre los vecinos sin asignar del AP_A con el mismo criterio que el paso 5
- 8- Asignar a AP_B el mejor canal según el CMC.
- 9- Si AP_A tiene vecinos sin asignar volver al paso 7
- 10- Si hay APs sin asignar volver al paso 5
- 11- Terminar

Capítulo 3. Algoritmos de Asignación de Canales

Ejemplo

En el siguiente ejemplo se muestra el funcionamiento del algoritmo. Se comienza con el grafo de adyacencias, en el cual:

- Cada vértice representa un AP.
- Los colores y números de los vértices representan el canal asignado.
- Un AP interno se representa con contorno negro.
- Un AP externo se representa con contorno rojo.
- Las aristas unen dos APs si estos son vecinos.
- En azul se representan las potencias recibidas en cada AP por su vecino.

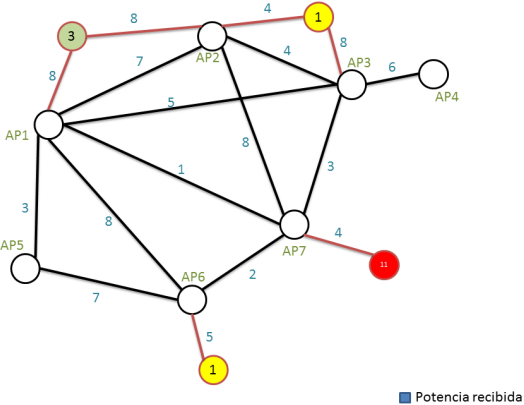
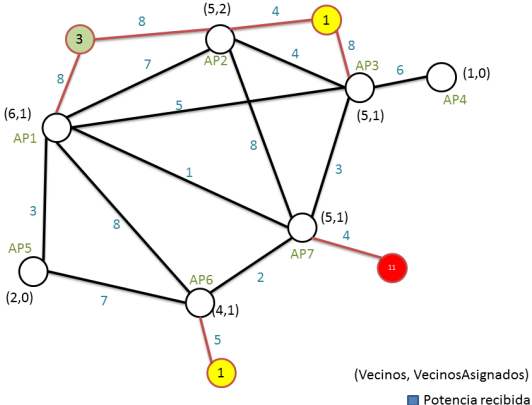
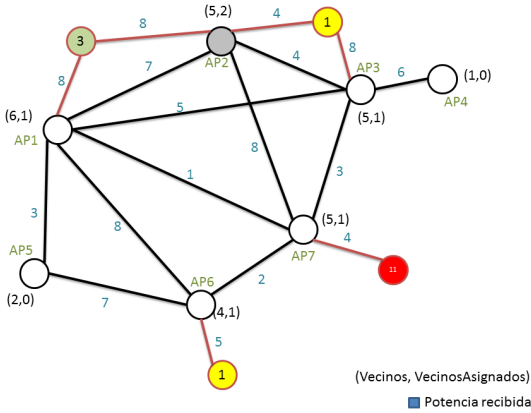
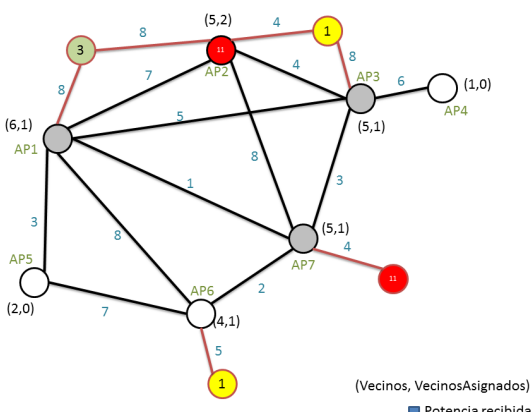
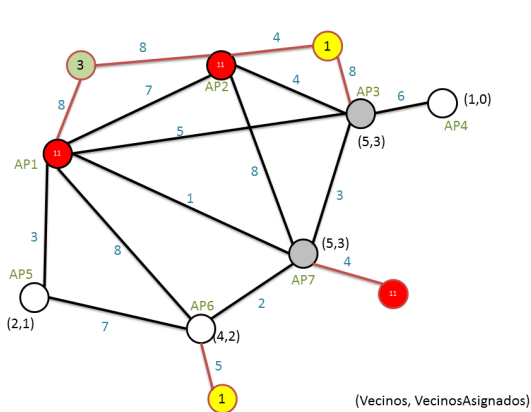
Se construye el grafo inicial de la red	
Se calcula para todos los APs la cantidad de vecinos y la cantidad de vecinos asignados, representados entre paréntesis en la figura.	

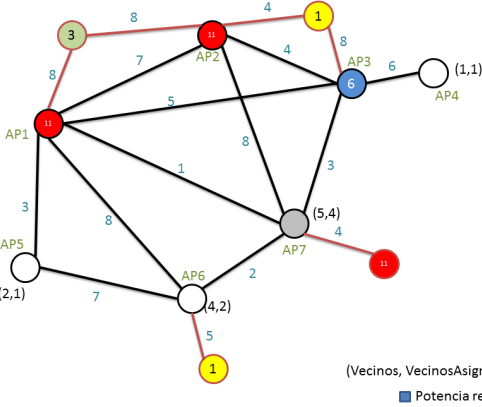
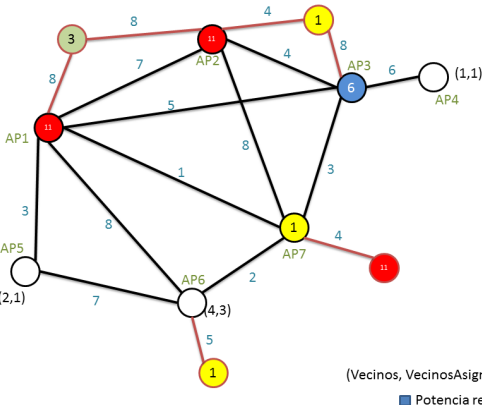
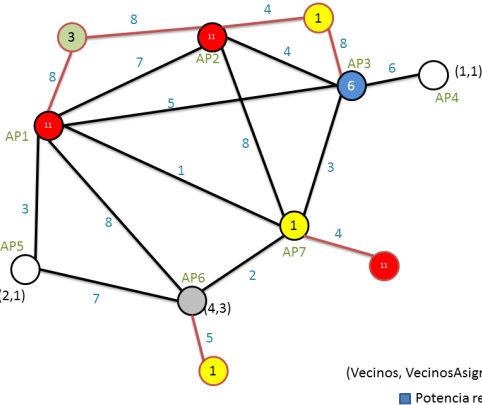
Tabla 3.1: Ejemplo para el algoritmo MVA

3.4. Algoritmos

Se comienza eligiendo el AP con más vecinos asignados. Al comienzo los únicos APs asignados son los externos, por lo que se empieza a recorrer el grafo por el AP_2	 (Vecinos, VecinosAsignados) ■ Potencia recibida
Se calcula el mejor canal para el AP seleccionado en el paso anterior. En este ejemplo el mejor canal es el 11 ya que los APs vecinos están transmitiendo en los canales 1 y 3. A continuación se evalúan solo los APs vecinos del AP_2 (sombreados en gris) para buscar aquel con más vecinos asignados.	 (Vecinos, VecinosAsignados) ■ Potencia recibida
Se selecciona el AP_1 ya que como criterio de desempate frente a igual cantidad de vecinos asignados, se opta por aquel con más vecinos. Al AP_1 se le asigna el mejor canal, en este caso el 11. Esto es así porque es preferible asignar dos APs con el mismo canal frente a dos APs con canales adyacentes (el canal 3 se solapa con el 1 y el 6).	 (Vecinos, VecinosAsignados) ■ Potencia recibida

Ejemplo para el algoritmo MVA

Capítulo 3. Algoritmos de Asignación de Canales

Se selecciona el AP_3 ya que frente al empate de vecinos asignados y vecinos, el algoritmo se queda con el de mayor potencia total recibida. A este AP se le calcula el mejor canal, resultando ser el 6	 (Vecinos, VecinosAsignados) ■ Potencia recibida
Se selecciona el AP_7 y se le asigna el canal 1.	 (Vecinos, VecinosAsignados) ■ Potencia recibida
Se busca entre los APs que quedan por recorrer al AP con más vecinos asignados. En este ejemplo se selecciona el AP_6	 (Vecinos, VecinosAsignados) ■ Potencia recibida

Ejemplo para el algoritmo MVA

3.4. Algoritmos

The diagram illustrates the channel assignment process for Access Points (APs) in a network. The network topology is shown with 7 APs (AP1 to AP7) and their assigned channels. The process is shown in three steps:

- Initial Assignment:** Channel 6 is assigned to AP6. The legend indicates that the number in parentheses next to the AP label represents (Vecinos, VecinosAsignados) and the color of the AP label represents the Potencia recibida (Received Power).
- Calculation of Best Channel:** The best channel for AP5 is calculated, resulting in channel 1. The legend indicates that the number in parentheses next to the AP label represents (Vecinos, VecinosAsignados) and the color of the AP label represents the Potencia recibida (Received Power).
- Search for AP without assigned channels:** The search identifies AP4 as the AP without assigned channels. The legend indicates that the number in parentheses next to the AP label represents (Vecinos, VecinosAsignados) and the color of the AP label represents the Potencia recibida (Received Power).

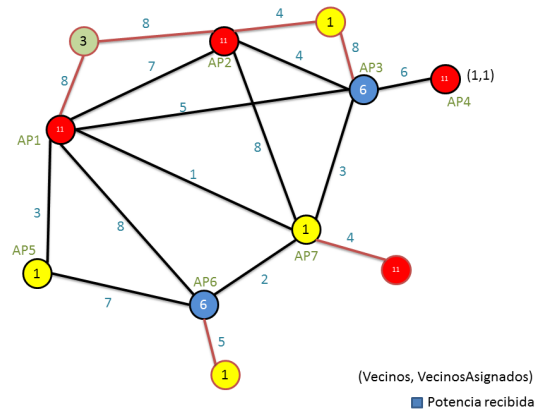
Ejemplo para el algoritmo MVA

Capítulo 3. Algoritmos de Asignación de Canales

A este AP se le asigna el mejor canal, siendo el mismo el canal 11. Para esta asignación se asume el canal antes asignado fue el 1 y se supone como criterio de desempate round-robin.

$$FI_{tot} = \sum FI_{AP}$$

$$FI_{tot} = 7 + 7 + 0 + 0 + 0 + 0 + 0 = 14$$



Ejemplo para el algoritmo MVA

3.4.2. Menos Canales Libres (MCL)

Para el diseño de este algoritmo se tuvo en cuenta la técnica DSATUR de coloración de grafos [16]. Para el recorrido del grafo se tiene en cuenta el *grado de saturación*, el *grado* y la potencia total recibida por cada AP. El grado de saturación indica la cantidad de colores que los vecinos de un AP están usando. La idea del recorrido por grado de saturación es pasar primero por los APs que tienen menos opciones de colores. El grado y la potencia recibida, se utilizan en caso que haya empate en grado de saturación entre dos o más APs. Dado que los Access Points externos pueden estar en cualquier canal del 1 al 11, para el cálculo del grado de saturación se aproximan al canal asignable más cercano¹.

Descripción del Algoritmo

El funcionamiento del algoritmo se describe en los siguientes pasos:

- 1- Setear la potencia al máximo en todos los APs y construir el grafo inicial en base a las potencias recibidas en cada AP.
- 2- Calcular el grado de cada AP.
- 3- Calcular la potencia recibida total en cada AP.
- 4- Calcular el grado de saturación de cada AP.
- 5- Buscar el AP con mayor grado de saturación. En caso de haber empate en la comparación elegir el que tenga mayor grado. Si sigue habiendo empate, elegir el que tenga mayor potencia recibida
- 6- Asignar el mejor canal según el CMC.
- 7- Si quedan APs por asignar ir al paso 4
- 8- Terminar

¹Canales asignables: 1, 6, y 11

Capítulo 3. Algoritmos de Asignación de Canales

Ejemplo

En el siguiente ejemplo se muestra el funcionamiento del algoritmo. Se comienza con el grafo de adyacencias, en el cual:

- Cada vértice representa un AP.
- Los colores y números de los vértices representan el canal asignado.
- Un AP interno se representa con contorno negro.
- Un AP externo se representa con contorno rojo.
- Las aristas unen dos APs si estos son vecinos.
- En azul se representan las potencias recibidas en cada AP por su vecino.

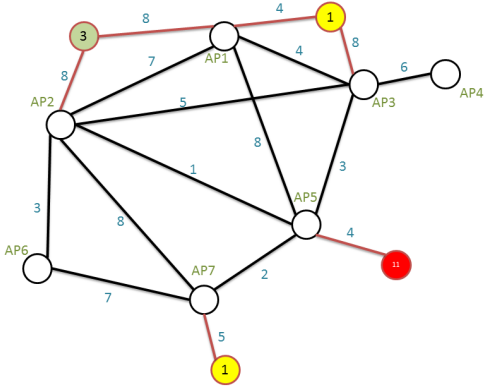
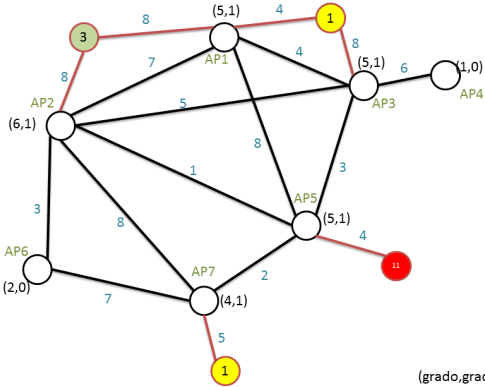
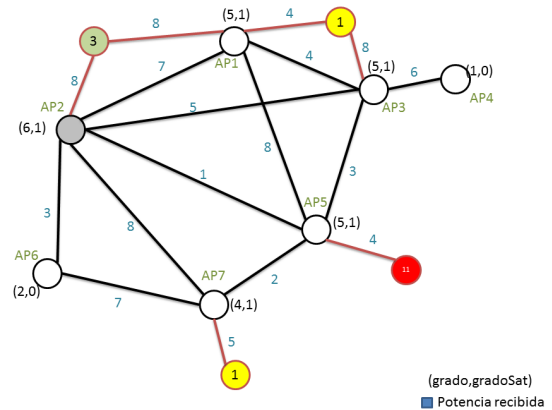
Como primer paso se aproximan los canales de los APs externos al canal más cercano entre las opciones 1, 6 y 11. El AP externo que se encuentra en el canal 3 se aproxima al 1.	
Se calcula el grado, potencia recibida y grado de saturación de cada AP.	

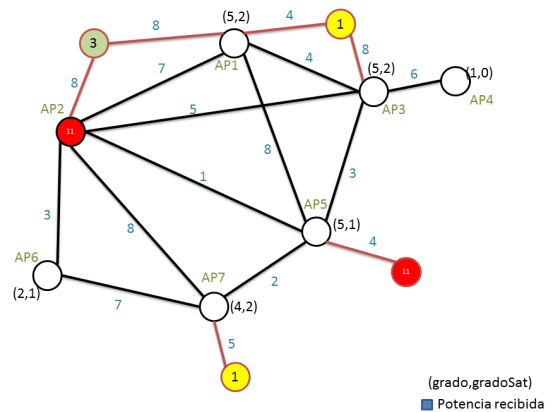
Tabla 3.2: Ejemplo para el algoritmo MCL

3.4. Algoritmos

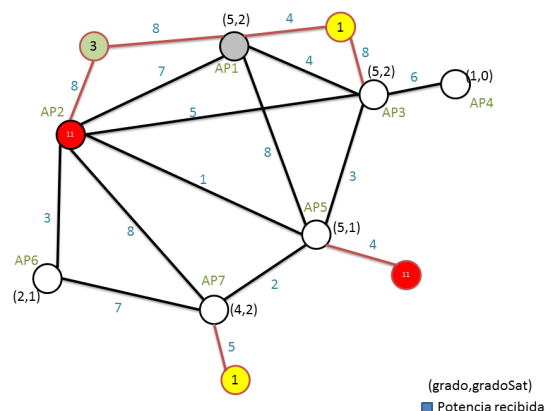
Para elegir el primer AP a asignar se evalúa el grado de saturación. Dado que varios APs tienen este valor igual a 1 (AP_1 , AP_2 , AP_3 , AP_5 y AP_7) se desempata con el grado, por lo que se elige el AP_2 con grado 6 para comenzar.



Se elige el mejor canal para el AP_2 . Los canales 1 y 6 se encuentran afectados por el AP externo por lo que se elige el 11 y se actualizan los grados de saturación de sus vecinos.



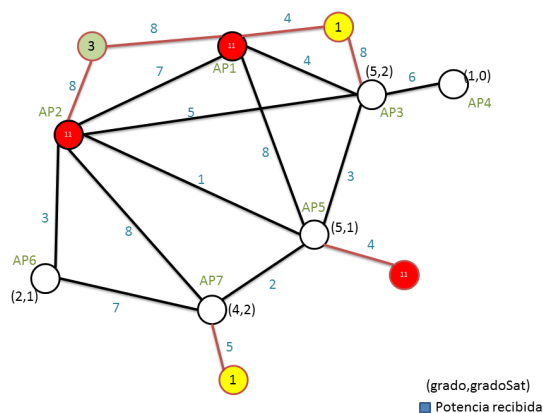
Para elegir el próximo AP a asignar también se tiene un empate en el grado de saturación de los APs 1, 3 y 7. También se tiene un empate en el grado entre los APs 1 y 3, se elige el AP_1 por tener mayor potencia total recibida.



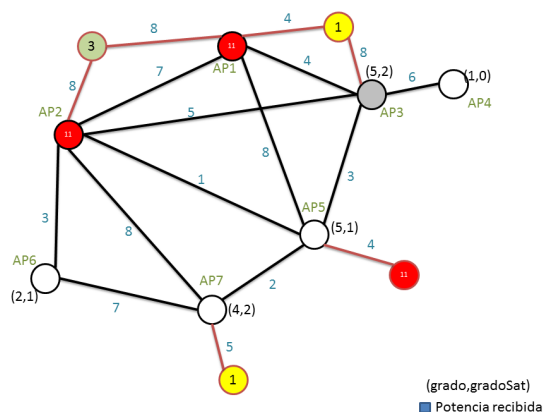
Ejemplo para el algoritmo MCL

Capítulo 3. Algoritmos de Asignación de Canales

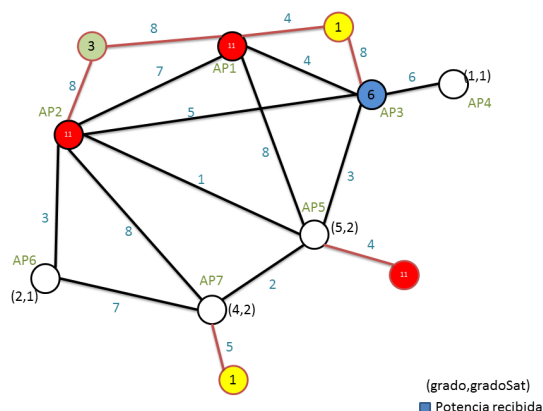
Se asigna canal al AP_1 . Dado que los canales 1 y 6 están ocupados por sus vecinos se elige el canal 11. Luego se actualiza el grado de saturación de sus vecinos.



AP_3 y AP_7 tienen grado de saturación igual a 2, se elige AP_3 ya que tiene mayor grado (5).



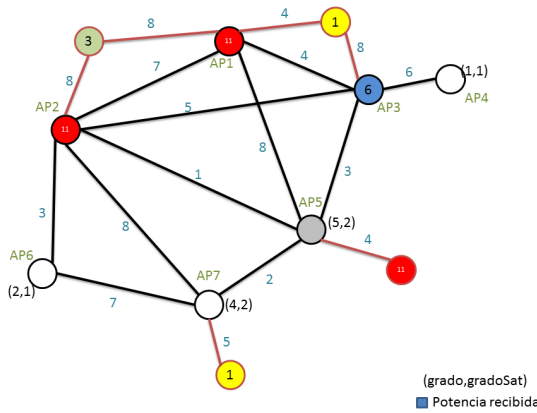
Se asigna el canal 6 al AP_3 y se actualiza el grado de saturación de sus vecinos.



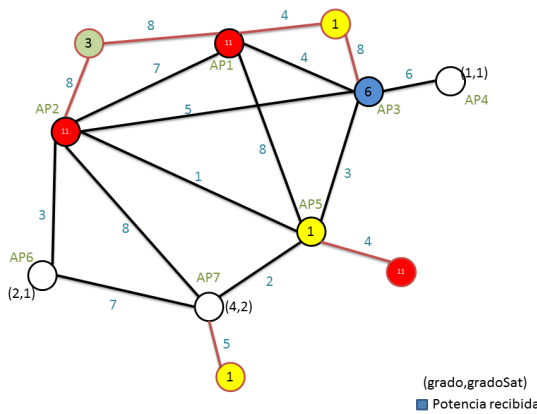
Ejemplo para el algoritmo MCL

3.4. Algoritmos

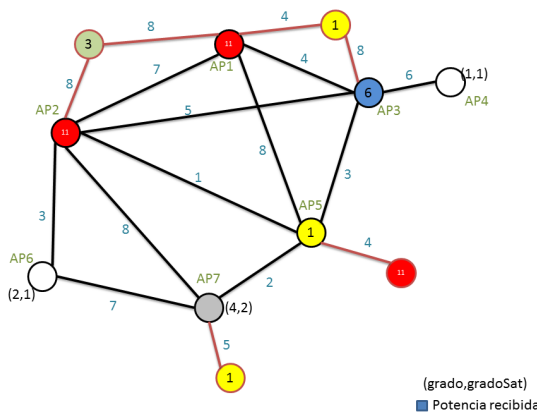
AP_5 y AP_7 tienen grado de saturación igual a 2, se elige AP_5 ya que tiene mayor grado (5).



Se asigna el canal 1 al AP_5 y se actualiza el grado de saturación de sus vecinos.



Se elige AP_7 por ser el AP con mayor grado de saturación (2)

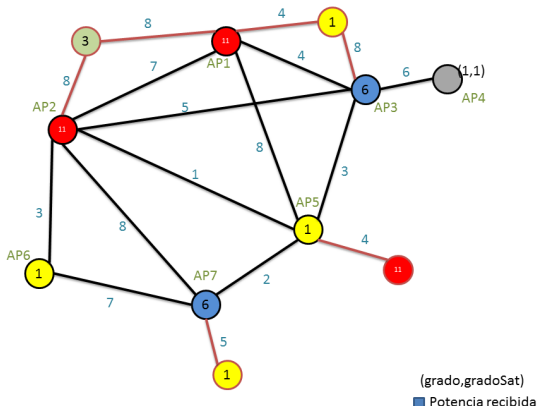
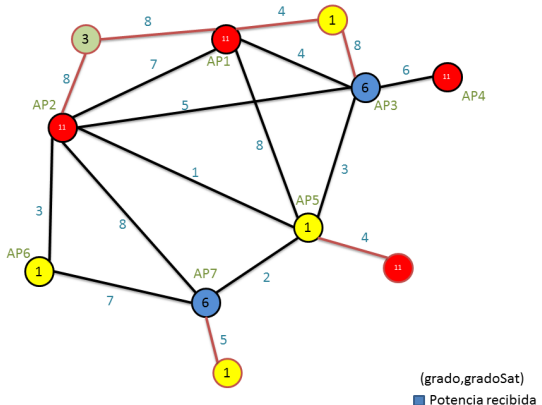


Ejemplo para el algoritmo MCL

Capítulo 3. Algoritmos de Asignación de Canales

Se asigna el canal 6 al AP_7 y se actualiza el grado de saturación de sus vecinos.	
Se elige AP_6 por ser el AP con mayor grado de saturación (2).	
Se asigna el canal 1 al AP_6 y se actualiza el grado de saturación de sus vecinos.	

Ejemplo para el algoritmo MCL

Se termina por el AP_4 ya que es el único que queda por recorrer.	
Se le asigna al AP_4 el canal 11. Para esta asignación se asume el canal antes asignado fue el 1 y se supone como criterio de desempate round-robin. $FI_{tot} = \sum FI_{AP}$ $FI_{tot} = 7 + 7 + 0 + 0 + 0 + 0 + 0 + 0 = 14$	

Ejemplo para el algoritmo MCL

Capítulo 3. Algoritmos de Asignación de Canales

3.4.3. Menos Comprometido Primero (MCP)

Este algoritmo se diseñó para comparar los algoritmos antes descritos con uno que tenga un recorrido más simple. El recorrido se basa en calcular el mejor canal de cada AP según el CMC, y asignar el AP que tenga el mayor Factor de interferencia en ese canal. El AP que tiene el mayor factor de interferencia en el mejor canal disponible, se considera que es el que está más comprometido para la elección del canal.

Descripción del Algoritmo

El funcionamiento del algoritmo se describe en los siguientes pasos:

- 1- Setear la potencia al máximo en todos los APs y construir el grafo inicial en base a las potencias recibidas en cada AP.
- 2- Calcular el grado de cada AP
- 3- Calcular el mejor canal de cada AP sin asignar según el CMC
- 4- Calcular el factor de interferencia en todos los AP en los canales calculados en 3.
- 5- Buscar el AP con el mayor factor de interferencia calculado en 4
- 6- Asignar el canal según el CMC
- 7- Si quedan APs por asignar ir a 3
- 8- Terminar

Ejemplo

En el siguiente ejemplo se muestra el funcionamiento del algoritmo. Se comienza con el grafo de adyacencias, en el cual:

- Cada vértice representa un AP.
- Los colores y números de los vértices representan el canal asignado.
- Un AP interno se representa con contorno negro.
- Un AP externo se representa con contorno rojo.
- Las aristas unen dos APs si estos son vecinos.
- En azul se representan las potencias recibidas en cada AP por su vecino.

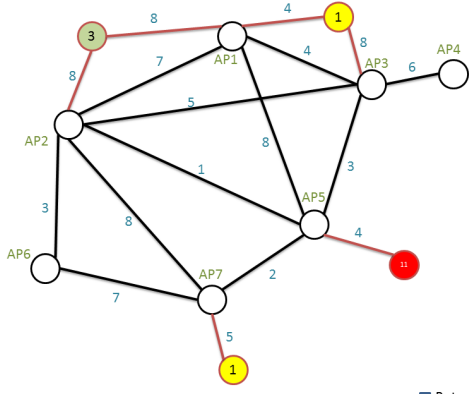
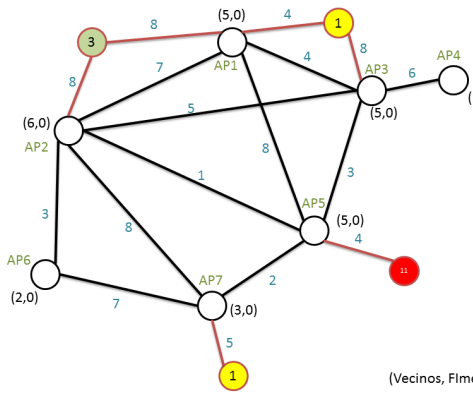
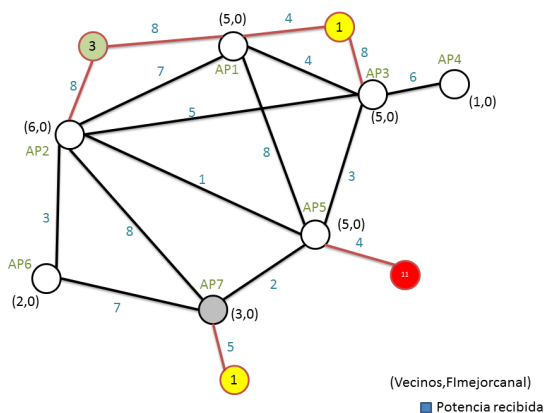
Se construye el grafo inicial de la red.	
Se calcula para cada AP cual es el mejor canal disponible y el $FI_{mejorcanal}$, representados entre paréntesis en la figura.	

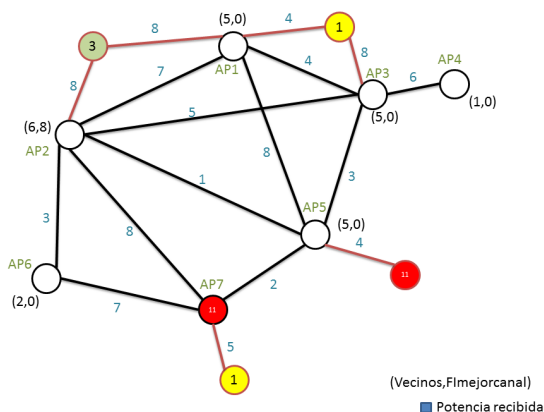
Tabla 3.3: Ejemplo para el algoritmo MCP

Capítulo 3. Algoritmos de Asignación de Canales

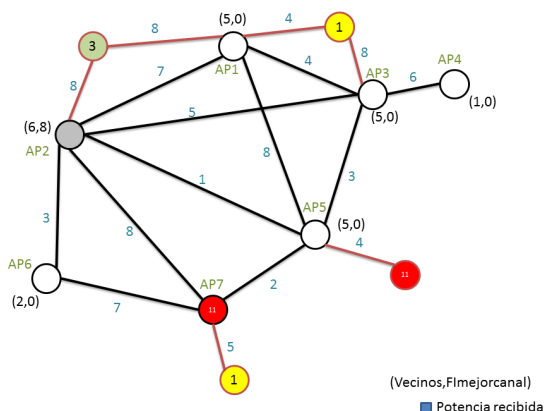
Se elige el AP que tenga el mayor $FI_{mejorcanal}$. Se elige AP_7 por ser el último de la lista.



Se asigna el mejor canal a AP_7 . Tanto 6 como 11 tienen potencia recibida 0, dependiendo del método de desempate elegido se decide cuál de los dos canales asignar. En este ejemplo se elige el canal 11. Se recalcula el en todos los AP.

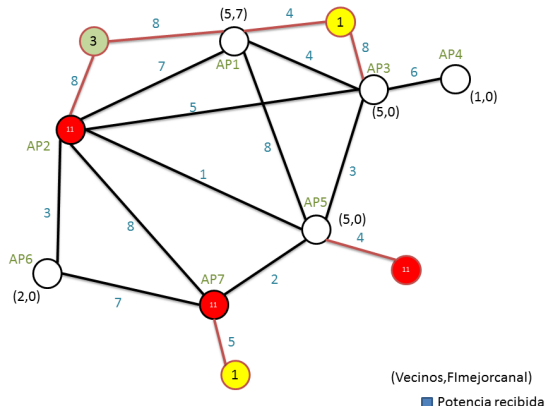
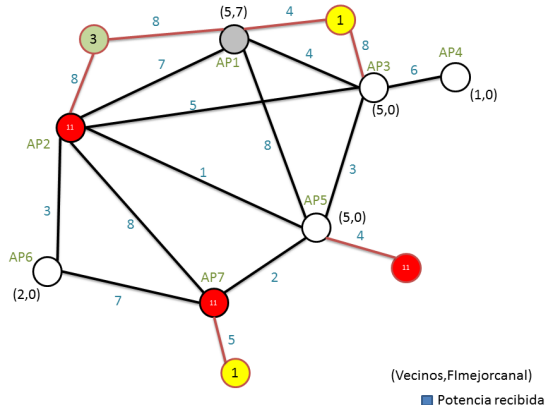
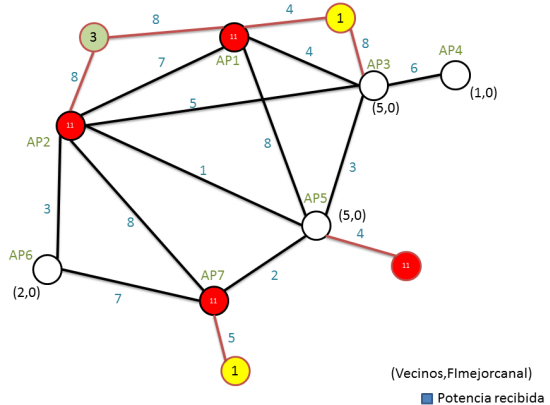


Se elige AP_2 porque tiene el mayor $FI_{mejorcanal}$. El mejor canal para AP_2 es el 11 ya que hay un AP externo en el canal 3 que solapa tanto al 1 como al 6. El FI en el canal 11 es la suma de las potencias recibidas en ese canal, en el ejemplo $FI_{11}^{AP_2} = 8$.



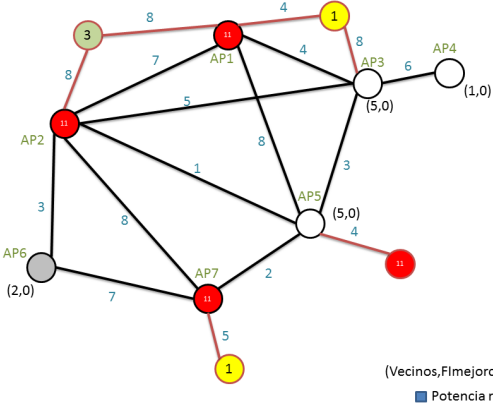
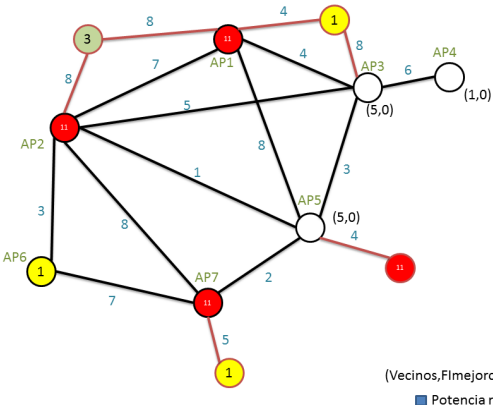
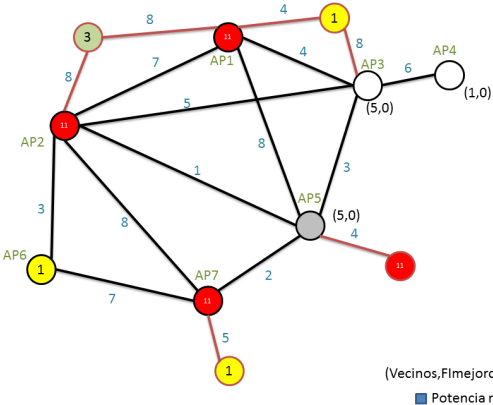
Ejemplo para el algoritmo MCP

3.4. Algoritmos

Se asigna el canal 11 a AP_2 y se actualiza el $FI_{mejorcanal}$ en todos los APs no asignados	 (Vecinos, $FI_{mejorcanal}$) ■ Potencia recibida
Se elige AP_1 ya que tiene el mayor $FI_{mejorcanal}$ (7)	 (Vecinos, $FI_{mejorcanal}$) ■ Potencia recibida
Se asigna el canal 11 a AP_1. Se actualiza el $FI_{mejorcanal}$ en todos los APs no asignados	 (Vecinos, $FI_{mejorcanal}$) ■ Potencia recibida

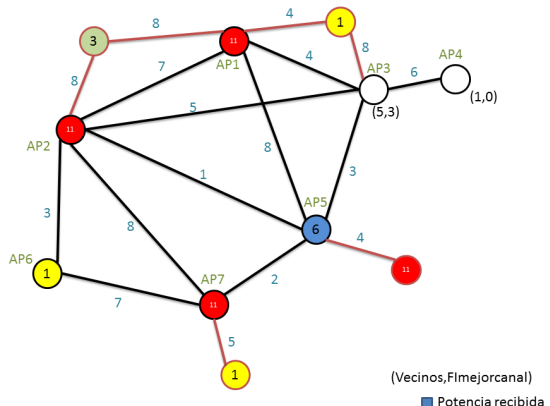
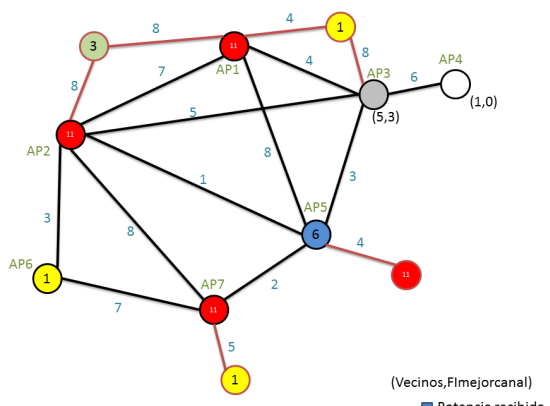
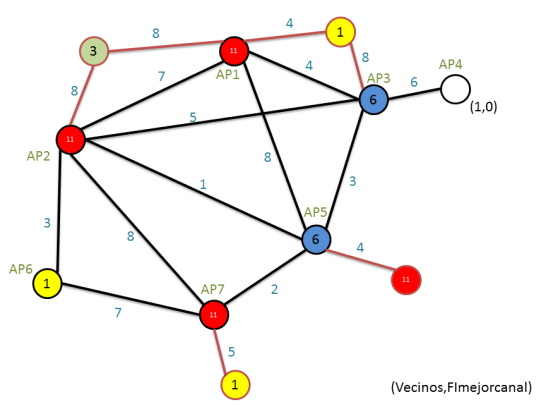
Ejemplo para el algoritmo MCP

Capítulo 3. Algoritmos de Asignación de Canales

Todos los APs sin asignar tienen $FI = 0$. Se elige AP_6 por ser el último en la lista	 (Vecinos, $FI_{mejorcanal}$) ■ Potencia recibida
Dependiendo del método de desempate se asigna el canal a AP_2, asumiendo round-robin se elige el canal 1. Se actualiza el $FI_{mejorcanal}$ en todos los APs no asignados.	 (Vecinos, $FI_{mejorcanal}$) ■ Potencia recibida
Todos los APs sin asignar tienen $FI = 0$. Se elige AP_5 por ser el último en la lista.	 (Vecinos, $FI_{mejorcanal}$) ■ Potencia recibida

Ejemplo para el algoritmo MCP

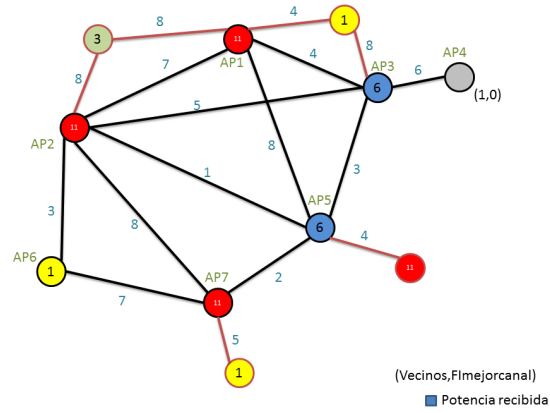
3.4. Algoritmos

Dependiendo del método de desempate se asigna el canal a AP_2, asumiendo round-robin se elige el canal 6. Se actualiza el $FI_{mejorcanal}$ en todos los APs no asignados.	
Se elige AP_3 ya que tiene el mayor $FI_{mejorcanal}$.	
Se asigna el canal 6 a AP_3. Se actualiza el $FI_{mejorcanal}$ en todos los APs no asignados.	

Ejemplo para el algoritmo MCP

Capítulo 3. Algoritmos de Asignación de Canales

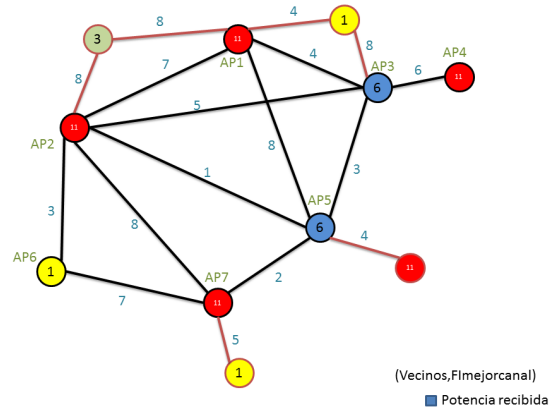
Se elige AP_4 ya que es el último AP a asignar.



Dependiendo del método de desempate se asigna el canal a AP_4 , asumiendo round-robin se elige el canal 11.

$$FI_{tot} = \sum FI_{AP}$$

$$FI_{tot} = 7 + 15 + 3 + 0 + 3 + 0 + 8 = 36$$



Ejemplo para el algoritmo MCP

3.4.4. FI_{min} : Función auxiliar

Esta función se desarrolló con el objetivo de comparar los resultados de los algoritmos descritos anteriormente contra la asignación con mínimo FI . La función se basa en evaluar el FI_{tot} obtenido en todas las combinaciones posibles de canales en los APs de la red. La cantidad de combinaciones posibles de asignaciones en un escenario con n APs se calcula como $CR_n^3 = \frac{(n+3-1)!}{(3-1)!n!}$.

Si bien esta herramienta asegura la asignación con factor de interferencia mínimo, el problema que enfrenta es el tiempo de ejecución. Se obtuvo una aproximación de los tiempos requeridos para escenarios con distintas cantidades de APs, en la tabla 3.4 se muestran los resultados. Se observó que para redes con más de 15 APs los tiempos de ejecución no eran tolerables.

Cantidad de APs	Combinaciones posibles	Tiempo de ejecución
64	3×10^{30}	7×10^{19} años
30	2×10^{14}	4500 años
20	3×10^9	1 mes
17	129 millones	1 día
16	43 millones	8 horas
15	14 millones	3 horas
10	59049	40 segundos

Tabla 3.4: Tiempos de ejecución del algoritmo FI_{min} , medidos con un laptop con procesador i7 y 12Gb de RAM

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 4

Algoritmos de Potencia

El control de la potencia de transmisión en redes Wi-Fi es utilizado en general para optimizar el consumo energético y reducir la interferencia en el medio compartido. Los algoritmos de control de potencia son ejecutados luego de una asignación óptima de canales. Hay casos de redes pequeñas donde es posible deshacerse totalmente de la interferencia co-canal, pero no siempre es así en redes de alta densidad como lo son las de Plan Ceibal. Cuando luego de una asignación de canales hay APs que siendo vecinos tienen el mismo canal asignado, es necesario un ajuste de potencia para eliminar la interferencia co-canal.

4.1. Objetivo

El control de potencia tiene como objetivo eliminar la interferencia co-canal entre los APs de la red. Si hay APs que tienen asignado el mismo canal y son vecinos, el algoritmo debe reducir la potencia de transmisión de ambos para que la potencia recibida por su vecino sea menor que la del umbral de detección¹. Reducir la potencia de transmisión trae aparejado una disminución en el área de cobertura, sin embargo si antes de aplicar el ajuste de potencia se hizo una buena asignación de canales, es de esperar que haya otros APs cercanos cubriendo el área.

Los Access Points tienen un rango acotado de potencia de transmisión, en los APs MikroTik este rango va desde los 0dBm hasta los 30dBm. Si los APs están a una distancia tal que transmitiendo a la mínima potencia siguen recibiendo la transmisión de su vecino, no va a ser posible eliminar la interferencia co-canal. Para estos casos se decidió dejar transmitiendo a los APs a su máxima potencia ya que se cree que si no se va a eliminar la interferencia entonces no amerita disminuir el radio de cobertura.

Se diseñaron dos algoritmos de control de potencia. Estos algoritmos se diferencian principalmente en el proceso que llevan a cabo elegir la potencia de

¹En la mayoría de los APs el umbral de detección es -100dBm, si la potencia recibida es menor a este umbral el AP deja de ver al vecino

Capítulo 4. Algoritmos de Potencia

transmisión. Mientras que uno realiza un procedimiento iterativo el otro lo hace a partir de cálculos teóricos.

4.2. Ejemplo

En este ejemplo se detalla cómo actúa un algoritmo de control de potencia.

- Un AP se representa con una cruz en la figura
- Los círculos representan el área de cobertura de cada AP
- El color de los círculos representa el canal asignado al AP
 - Amarillo: Canal 1
 - Azul: Canal 6
 - Rojo: Canal 11

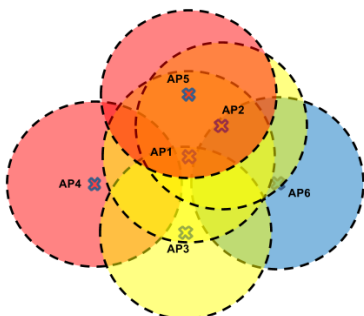
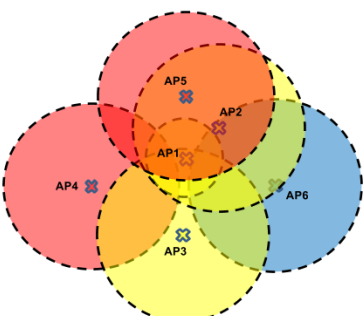
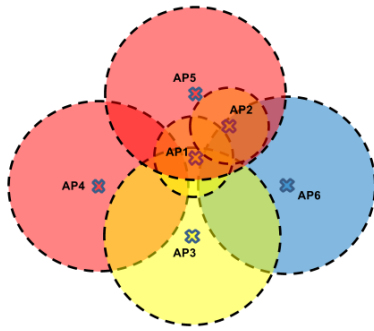
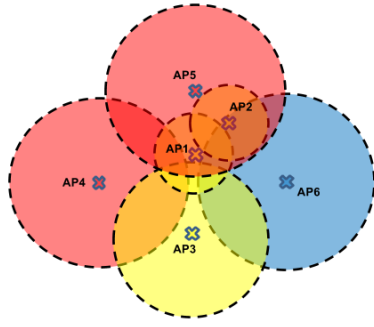
Este es el estado inicial de la red. En este caso se encuentran los APs 1, 2 y 3 transmitiendo todos en el canal 1 e interfiriendo uno con otro, por lo que es necesario aplicar control de potencia.		Tabla de Potencias recibidas (dBm) <table><tr><th></th><th>AP_1</th><th>AP_2</th><th>AP_3</th></tr><tr><th>AP_1</th><td>x</td><td>-40</td><td>-85</td></tr><tr><th>AP_2</th><td>-40</td><td>x</td><td>$-\infty$</td></tr><tr><th>AP_3</th><td>-85</td><td>$-\infty$</td><td>x</td></tr></table> $FI_{tot} = -37dBm$		AP_1	AP_2	AP_3	AP_1	x	-40	-85	AP_2	-40	x	$-\infty$	AP_3	-85	$-\infty$	x
	AP_1	AP_2	AP_3															
AP_1	x	-40	-85															
AP_2	-40	x	$-\infty$															
AP_3	-85	$-\infty$	x															
Control de Potencia sobre AP1: Al AP_1 lo detectan el AP_2 y el AP_3 . El AP_2 es quien lo ve con más potencia por lo que se disminuye la potencia de transmisión del AP_1 (P_{tx}^1) hasta que AP_2 no lo ve.		Tabla de Potencias recibidas (dBm) <table><tr><th></th><th>AP_1</th><th>AP_2</th><th>AP_3</th></tr><tr><th>AP_1</th><td>x</td><td>-40</td><td>-85</td></tr><tr><th>AP_2</th><td>$-\infty$</td><td>x</td><td>$-\infty$</td></tr><tr><th>AP_3</th><td>$-\infty$</td><td>$-\infty$</td><td>x</td></tr></table> $FI_{tot} = -40dBm$		AP_1	AP_2	AP_3	AP_1	x	-40	-85	AP_2	$-\infty$	x	$-\infty$	AP_3	$-\infty$	$-\infty$	x
	AP_1	AP_2	AP_3															
AP_1	x	-40	-85															
AP_2	$-\infty$	x	$-\infty$															
AP_3	$-\infty$	$-\infty$	x															

Tabla 4.1: Ejemplo Control de Potencia

4.3. Algoritmo de control de potencia - Método Iterativo

Control de Potencia sobre AP_2: Al AP_2 lo detecta únicamente el AP_1, por lo que se disminuye P_{tx}^2 hasta que AP_1 no lo detecte.		Tabla de Potencias recibidas (dBm) <table><tr><th></th><th>AP_1</th><th>AP_2</th><th>AP_3</th></tr><tr><th>AP_1</th><td>x</td><td>$-\infty$</td><td>-85</td></tr><tr><th>AP_2</th><td>$-\infty$</td><td>x</td><td>$-\infty$</td></tr><tr><th>AP_3</th><td>$-\infty$</td><td>$-\infty$</td><td>x</td></tr></table> $FI_{tot} = -85dBm$		AP_1	AP_2	AP_3	AP_1	x	$-\infty$	-85	AP_2	$-\infty$	x	$-\infty$	AP_3	$-\infty$	$-\infty$	x
	AP_1	AP_2	AP_3															
AP_1	x	$-\infty$	-85															
AP_2	$-\infty$	x	$-\infty$															
AP_3	$-\infty$	$-\infty$	x															
Control de Potencia sobre AP_3: Al AP_3 sólo lo ve el AP_1, se disminuye P_{tx}^3 hasta que no interfiera con AP_1.		Tabla de Potencias recibidas <table><tr><th></th><th>AP_1</th><th>AP_2</th><th>AP_3</th></tr><tr><th>AP_1</th><td>x</td><td>$-\infty$</td><td>$-\infty$</td></tr><tr><th>AP_2</th><td>$-\infty$</td><td>x</td><td>$-\infty$</td></tr><tr><th>AP_3</th><td>$-\infty$</td><td>$-\infty$</td><td>x</td></tr></table> $FI_{tot} = 0$		AP_1	AP_2	AP_3	AP_1	x	$-\infty$	$-\infty$	AP_2	$-\infty$	x	$-\infty$	AP_3	$-\infty$	$-\infty$	x
	AP_1	AP_2	AP_3															
AP_1	x	$-\infty$	$-\infty$															
AP_2	$-\infty$	x	$-\infty$															
AP_3	$-\infty$	$-\infty$	x															

Ejemplo Control de Potencia

4.3. Algoritmo de control de potencia - Método Iterativo

El objetivo de un algoritmo de control de potencia es eliminar la interferencia co-canal entre APs, para eso se disminuye la potencia de transmisión de los APs hasta que se dejen de interferir. Este algoritmo se basa en un método recursivo para lograrlo, la idea es ir bajando de a poco la potencia de transmisión de un AP y sensar como lo ven sus vecinos. Este proceso se repite hasta que ningún vecino transmitiendo en el mismo canal detecte su presencia.

El algoritmo recorre todos los APs con interferencia co-canal. Para cada AP busca el vecino que lo ve con mayor intensidad de señal, reduce la potencia de transmisión en pasos de 5dBm hasta que el AP vecino deja de recibir la señal. Un AP MikroTik deja de ver a un vecino cuando la potencia recibida es menor a -100dBm.

4.3.1. Descripción del algoritmo

Suponiendo que el rango de la potencia de transmisión del AP es $P_{tx} = [P_{tx_{min}}, P_{tx_{max}}]$, se asume al comienzo del algoritmo $P_{tx} = P_{tx_{max}}$.

Se realiza el siguiente procedimiento para cada uno de los APs (AP_i):

Capítulo 4. Algoritmos de Potencia

- 1- AP_i transmite con potencia P_{tx}^i . Se define AP_{cc} como el conjunto de APs $AP_{1..n}$ que son vecinos de AP_i y transmiten en mismo canal que él. Si este conjunto de APs no es vacío continuar.
- 2- Se busca AP_m dentro de AP_{cc} que ve a AP_i con mayor potencia. Se llama a esta potencia P_{mi} .
- 3- Si $P_{tx}^i - 5dBm > P_{tx_{min}}$ se disminuye P_{tx}^i en 5 dBm, en caso contrario ir a 5
- 4- Se mide P_{mi} . Si $P_{mi} > -100dBm$ (no son vecinos) volver a 3, sino ir a 6.
- 5- Se fija $P_{tx} = P_{tx_{max}}$.
- 6- Terminar

4.4. Algoritmo de control de potencia - Método Friis

Para eliminar la interferencia co-canal es necesario disminuir la potencia de transmisión de los APs. Este algoritmo se basa en la ecuación de transmisión de Friis [19], el objetivo es calcular teóricamente cuál es la potencia de transmisión requerida para que la potencia recibida por un vecino sea menor a cierto umbral. Para los APs MikroTik el umbral de detección de APs cercanos es -100dBm.

La ecuación de transmisión de Friis establece:

$$P_{rx} = G_{tx}G_{rx}KP_{tx} \quad (4.1)$$

Donde:

- P_{rx} : Potencia recibida por el AP vecino
- P_{tx} : Potencia de transmisión del AP
- G_{tx} : Ganancia de la antena del AP
- G_{rx} : Ganancia de la antena del AP vecino
- K : Constante que depende de la distancia y la longitud de onda

El objetivo es que la potencia recibida (P_{rx}) por los APs vecinos sea menor a -100dBm entonces:

$$\left. \begin{array}{l} P_{rx}^i = G_{tx}G_{rx}KP_{tx}^i \\ P_{rx}^f = G_{tx}G_{rx}KP_{tx}^f \end{array} \right\} P_{tx}^f = \frac{P_{tx}^i P_{rx}^f}{P_{rx}^i} \quad (4.2)$$

Si se convierte todo a dBm:

$$P_{tx}^f = P_{tx}^i + P_{rx}^f - P_{rx}^i \quad (4.3)$$

Un AP MikroTik no detecta a un vecino si la potencia recibida es menor a -100dBm. Ajustar la potencia de transmisión para que la potencia recibida en el AP

4.4. Algoritmo de control de potencia - Método Friis

vecino sea exactamente -100dBm, no es eficaz debido a que el medio es muy sensible a interferencias. El AP receptor puede detectar la señal a -100dBm en un instante y luego detectarla a un valor cercano. Si se ajusta a -100dBm seguramente los APs se interfieran en algunos momentos, es por esto que se decidió definir una variable de ajuste γ para asegurar que los APs no sean vecinos a pesar de la inestabilidad del medio debida a las características propias de la propagación indoor. Mediante simulaciones se definió el valor de γ en -10dBm.

$$P_{tx}^f = P_{tx}^i - 100dBm - P_{rx}^i + \gamma \quad (4.4)$$

4.4.1. Descripción del algoritmo

Suponiendo que el rango de la potencia de transmisión del AP es $P_{tx} = [P_{tx_{min}}, P_{tx_{max}}]$, se asume al comienzo del algoritmo $P_{tx} = P_{tx_{max}}$.

Se realiza el siguiente procedimiento para cada uno de los APs (AP_i):

- 1- AP_i transmite con potencia P_{tx}^i . Se define AP_{cc} como el conjunto de APs $AP_{1..n}$ que son vecinos de AP_i y transmiten en mismo canal que él. Si este conjunto de APs no es vacío continuar.
- 2- Se busca AP_m dentro de AP_{cc} que ve a AP_i con mayor potencia.
- 3- Se calcula la potencia de transmisión de AP_i según la fórmula 4.4 P_{friis} .
- 4- Si el resultado anterior es menor a $P_{tx_{min}}$, no se modifica la potencia en caso contrario $P_{tx} = P_{friis}$.
- 5- Terminar

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 5

Implementacion

Las redes Ceibal que necesitan un controlador para la optimización de canales y potencia, cuentan con un servidor centralizado y varios Access Points de la marca MikroTik. Los servidores tienen instalado una versión de CentOS ajustada a las necesidades de Ceibal. Cuando Ceibal propuso el proyecto estableció que el software de la controladora debía ejecutarse sobre estos servidores.

Para la fase de implementación del proyecto Ceibal brindó cinco Access Points y un servidor para poder tener un laboratorio de similares condiciones a las de un centro educativo.

Se decidió desarrollar el software en Python, un lenguaje de programación de simple aprendizaje. El producto final fue un sistema que se conecta a los APs y obtiene los datos para correr los algoritmos, realiza los cálculos de optimización y realiza los cambios de configuración en los APs.

A pesar de que toda la implementación se basó en Access Points de la marca MikroTik, el desarrollo del software tuvo en cuenta la posibilidad de ser expandido a otras marcas mediante el desarrollo de pequeños módulos adaptadores.

5.1. CentOS

El proyecto CentOS (Community ENTERprise Operating System) es software libre basado en esfuerzo comunitario con la meta de entregar un ecosistema robusto de código abierto. Como ventajas hacia los usuarios, tiene una plataforma administrable que encaja en una gran variedad de despliegues. Para las comunidades de software libre, se ofrece una base predecible, así como también una fuente grande de recursos para construir, probar, desplegar y mantener sus códigos. [20]

5.2. MikroTik

Para el proyecto se utilizaron APs de la marca MikroTik provistos por Plan Ceibal. Se estudiaron los comandos necesarios para obtener la información relevante para los algoritmos y también los relativos a la configuración de los equipos.

5.2.1. MikroTik RB433

MikroTik es una empresa letona fundada en 1996 para desarrollar routers y sistemas inalámbricos. Mikrotik provee hardware y software para conectividad hacia internet en la mayoría de los países del mundo. MikroTik RouterOS es el sistema operativo de los equipos MikroTik utilizados. RouterOS es un sistema operativo basado en el kernel de Linux v2.6, tiene entre otras, funcionalidades de routing, firewall y QoS. Para las pruebas se utilizaron MikroTik RB433 como el que se muestra en la figura 5.1. [21] [2]

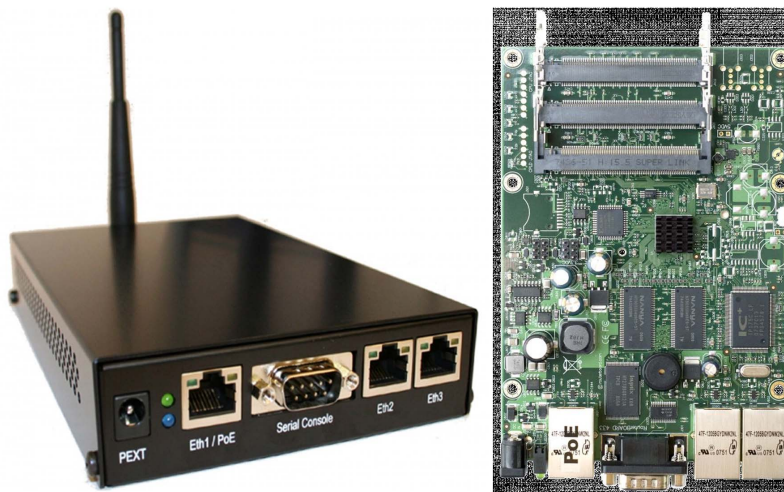


Figura 5.1: Mikrotik RouterBoard RB433 [22]

5.2.2. Interacción con la controladora

Los Access Points Mikrotik tienen varias formas de conectarse para hacer consultas o cambios de configuración:

- Telnet
- SSH
- Cable serial
- SNMP

- Web (Solo algunos modelos)
- Winbox
- API (Application programmable interface)

Dentro de todas estas posibilidades se decidió utilizar API. Es un servicio que corre en el puerto TCP:8728 diseñado para la interacción con sistemas de control. En la web de la comunidad de MikroTik [23] hay publicado un desarrollo en Python para la interacción con los APs utilizando API. Este desarrollo permite conectarse a los equipos remotamente y ejecutar comandos de lectura y escritura muy similares a los comandos del CLI (Command Line Interface).

5.2.3. Comandos utilizados

En esta sección se describen los comandos utilizados por la controladora, para la recolección de datos y cambios de configuración.

a- Obtención de datos:

Con el siguiente comando se obtienen los datos de SSID, MAC-Address, canal (frequency) y potencia (tx-power):

```
[admin@MikroTik] > interface wireless print advanced
Flags: X - disabled, R - running
0 name="wlan1" mtu=1500 mac-address=00:0C:42:63:DB:C1 arp=enabled disable-running-check=no
  interface-type=Atheros AR5413 radio-name="000C4263DBC1" mode=ap-bridge ssid="WICO-AP9" area=""
  frequency-mode>manual-txpower country=no country-set antenna-gain=0 frequency=2412
  band=2.4ghz-b/g scan-list=default rate-set=default supported-rates-b=1Mbps,2Mbps,5.5Mbps,11Mbps
  supported-rates-a/g=6Mbps,9Mbps,12Mbps,18Mbps,24Mbps,36Mbps,48Mbps,54Mbps basic-rates-b=1Mbps
  basic-rates-a/g=6Mbps max-station-count=2007 ack-timeout=dynamic tx-power=18
  tx-power-mode=all-rates-fixed noise-floor-threshold=default periodic-calibration=default
  periodic-calibration-interval=60 burst-time=disabled dfs-mode=none antenna-mode=ant-a
  wds-mode=disabled wds-default-bridge=none wds-default-cost=100 wds-cost-range=50-150
  wds-ignore-ssid=no update-stats-interval=disabled default-authentication=yes
  default-forwarding=yes default-ap-tx-limit=0 default-client-tx-limit=0
  proprietary-extensions=post-2.9.25 wmm-support=disabled hide-ssid=no security-profile=default
  disconnect-timeout=3s on-fail-retry-time=100ms preamble-mode=both compression=no
  allow-sharedkey=no station-bridge-clone-mac=00:00:00:00:00:00 hw-retries=4 frame-lifetime=0
  adaptive-noise-immunity=none hw-fragmentation-threshold=disabled hw-protection-mode=none
  hw-protection-threshold=0 frequency-offset=0
```

Figura 5.2: interface wireless print advanced

b- Configuración de frecuencia:

El siguiente comando es utilizado para configurar la frecuencia o canal del AP:

```
[admin@MikroTik] >
[admin@MikroTik] > interface wireless set wlan1 frequency=2412
[admin@MikroTik] >
```

Figura 5.3: interface wireless set wlan1 frequency=2412

Capítulo 5. Implementacion

c- Configuración de Potencia de transmisión:

El siguiente comando es utilizado para configurar la potencia de transmisión del AP:

```
[admin@MikroTik] >  
[admin@MikroTik] > interface wireless set wlan1 tx-power=18  
[admin@MikroTik] >
```

Figura 5.4: interface wireless set wlan1 tx-power=18

c- Búsqueda de APs vecinos:

Mediante el siguiente comando se hace la búsqueda de vecinos del AP:

```
[admin@MikroTik] > interface wireless scan wlan1 duration=12  
Flags: A - active, B - bss, P - privacy, R - routers-network, N - nstreme  
ADDRESS SSID BAND FREQ SIG NF SNR RADIO-NAME  
ABP 84:74:2A:02:6B:15 Consultorio Odontologico 2.4ghz-g 2412 -72 -99 27  
ABP 34:4B:50:2E:C2:65 Sezena 2.4ghz-g 2412 -69 -99 30  
ABP 34:4B:50:2E:1B:90 mape 2.4ghz-g 2412 -78 -99 21  
ABP 38:46:08:50:7D:00 Pertusso 2.4ghz-g 2412 -26 -99 73  
ABP 14:60:80:3C:9E:07 holandes 2.4ghz-g 2427 -89 -99 10  
ABP 34:4B:50:2E:C0:B2 paddy 2.4ghz-g 2432 -52 -99 47  
ABP 00:25:9C:2F:4A:A5 Joao 2.4ghz-g 2437 -69 -99 30  
ABP 14:60:80:42:22:E6 CLARICEROSA6 2.4ghz-g 2437 -70 -99 29  
ABP 94:0C:6D:EF:4B:B2 Flia Bouret Wifi 2.4ghz-g 2437 -73 -99 26  
ABP 84:74:2A:01:D3:AD TP-LINK 2.4ghz-g 2437 -62 -99 37  
ABP 14:CC:20:D3:0D:A5 Flia Bouret Wifi 2.4ghz-g 2437 -93 -99 6  
ABP 84:74:2A:05:14:E1 Carrau 2.4ghz-g 2462 -80 -99 19  
ABP 90:F6:52:56:F5:3E TIMON 2.4ghz-g 2462 -54 -99 45  
ABP 54:22:F8:A1:D0:FF DIEGO 2.4ghz-g 2462 -84 -99 15  
ABP DC:02:8E:A5:8A:4D dgamboa 2.4ghz-g 2462 -88 -99 11  
AB C8:D7:19:B3:17:8A laube602-invitado 2.4ghz-g 2462 -86 -99 13  
ABP 34:4B:50:29:AB:42 GABRIEL 2.4ghz-g 2412 -83 -99 16  
ABP 84:74:2A:01:47:71 esperanza 2.4ghz-g 2412 -86 -99 13  
ABP 54:22:F8:A1:A9:E5 carlos 2.4ghz-g 2437 -82 -99 17  
ABP 68:7F:74:78:6E:F1 videofan 2.4ghz-g 2437 -84 -99 15  
ABP C8:D7:19:B3:17:88 laube602 2.4ghz-g 2462 -86 -99 13  
ABP 14:60:80:3E:CF:5B Antelhcba2 2.4ghz-g 2442 -76 -99 23  
ABP 14:60:80:3E:93:AB Antel66HFY 2.4ghz-g 2457 -87 -100 13  
ABP 8C:E0:81:7D:8C:3A magdalena 2.4ghz-g 2462 -87 -99 12  
ABP E4:71:85:20:4B:60 tuculandia4 2.4ghz-g 2442 -87 -99 12  
ABP B0:48:7A:AF:D0:B4 ojodigital 2.4ghz-g 2412 -92 -99 7  
ABP 64:68:0C:C3:E9:AD Nightroad2 2.4ghz-g 2412 -93 -99 6  
ABP 84:74:2A:01:2D:D6 Lolo 2.4ghz-g 2437 -87 -99 12
```

Figura 5.5: interface wireless scan wlan1 duration=12

El comando hace que el AP realice un escaneo de todas las señales recibidas de Access Points que transmiten en los diferentes canales de Wi-Fi. La sentencia "duration = 12" es necesaria para indicar el tiempo de espera del escaneo

5.3. Python

Python es un lenguaje de programación orientada a objetos muy sencillo con una sintaxis muy simple y fácil de aprender. Es un software de código abierto, lo que permite que sea utilizado libremente incluso en despliegues comerciales. Existen muchas librerías diseñadas por terceros que facilita la implementación de códigos nuevos. En 2008 se publicó la versión 3 de Python que agrega nuevas funcionalidades al lenguaje, sin embargo la mayoría de los usuarios aún no han migrado y siguen utilizando la versión Python 2.

Como se mencionó en 5.2.2 en el sitio web de MikroTik hay disponible una interfaz para la conectividad con los APs desarrollada en Python. Esta interfaz es de fácil implementación y la documentación es lo suficientemente accesible como para implementarlo sin mayores inconvenientes. Si a esto le sumamos la recomendación de los tutores, más la sencillez del lenguaje fue fácil decidir que Python era el lenguaje adecuado para la implementación. Se decidió además implementar el código de la controladora en Python 2 ya que los servidores CentOS instalados en las distintas instituciones de Ceibal solo tienen instalada esta versión.

5.4. Pseudo código / Descripción del software

En esta sección se describen los diferentes módulos de software implementados, en la figura 5.6 se muestra el diagrama de flujo del sistema.

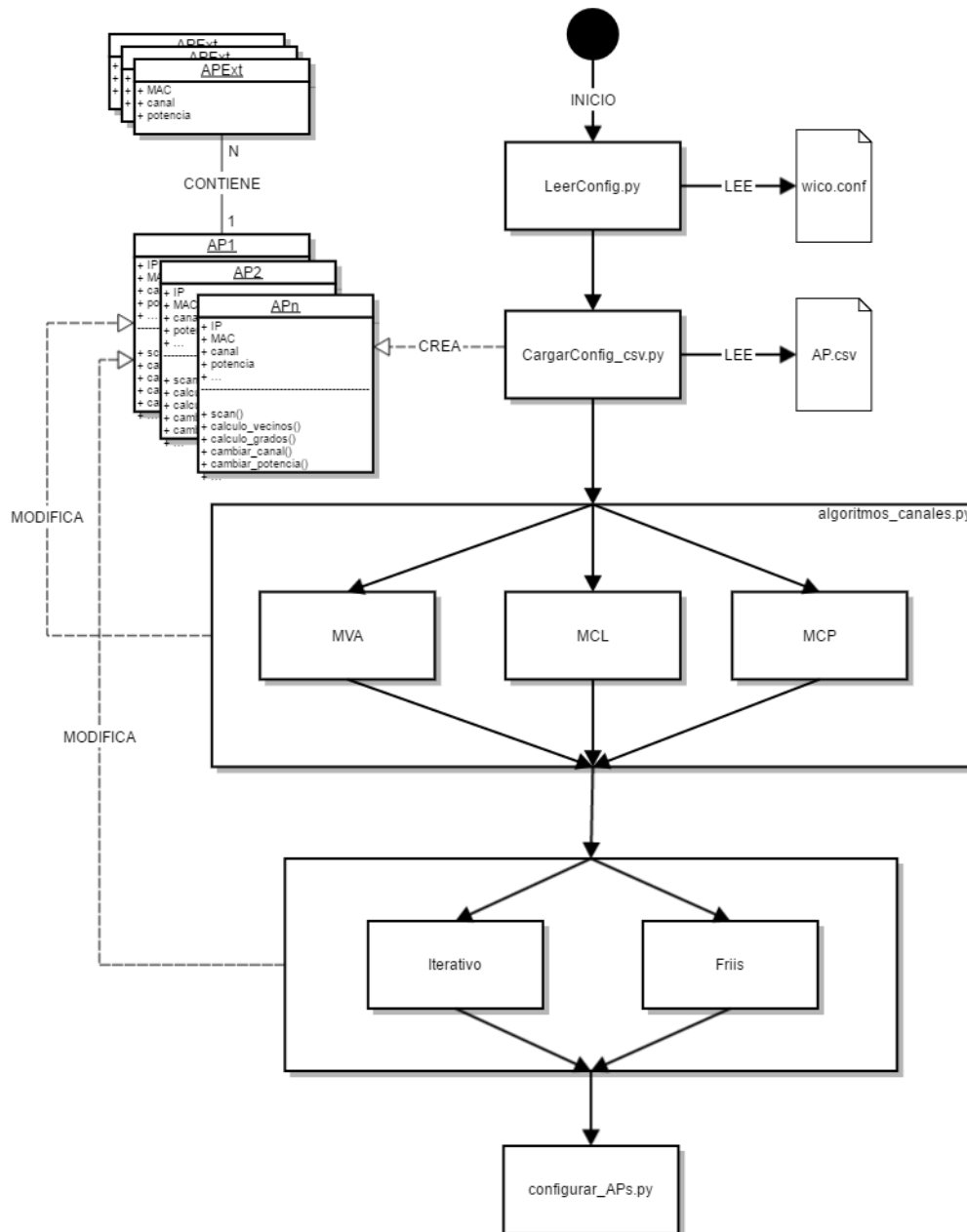


Figura 5.6: Diagrama de flujo del software implementado

5.4.1. Archivo de configuración general - wico.conf

Es el archivo de configuración del sistema que contiene las variables editables por el usuario:

- **Directorio AP CSV:** Variable que especifica donde está ubicado el archivo de configuración de los Access Point. Por defecto está ubicado en la misma carpeta que wico.py.
- **Algoritmo Asignación Canales:** Algoritmo de asignación de canales elegido (MVA, MCL, MCP o none para ninguno) [3]
- **Algoritmo Asignación Potencia:** Algoritmo de control de potencia elegido (Friis, Iterativo o none para ninguno) [4]
- **CMC:** Criterio de mejor canal (potencia o cantidad) [3.3.1]
- **Alpha:** Factor de canales adyacente utilizado con el CMC potencia. [3.3.2]
- **Beta:** Factor de canales adyacente utilizado con el CMC cantidad [3.3.2]
- **Desempate:** Criterio utilizado para decidir que canal elegir en caso de empate. [3.3.3]

5.4.2. Archivo de configuración de APs - AP.csv

Es el archivo donde se listan los Access Point de la red que se va a optimizar. El formato del archivo es *.csv* (variables separadas por coma) donde las variables son:

Modelo de AP, IP, Usuario, Contraseña

5.4.3. Lectura del archivo de configuración - LeerConfig.py

Este módulo se encarga de leer el archivo de configuración y cargar las variables al sistema.

5.4.4. Creación de APs - CargarConfigcsv.py

Este módulo lee el archivo AP.csv y crea las instancias de los objetos AP que se van a usar en los algoritmos.

5.4.5. Clase AccessPoint

Conjunto de atributos y métodos definidos para caracterizar cada nodo de la red interna. Como se explica en la sección 5.5 se definieron varias clases AccessPoint por cada modelo o marca diferente de AP.

Capítulo 5. Implementacion

Atributos

- IP: Dirección IP de la interfaz física por la que se comunica el sistema al AP.
- user: usuario que se usa para conectarse al equipo
- password: contraseña del usuario user
- MAC: Dirección MAC de la interfaz inalámbrica
- canal: Canal asignado al AP
- potencia: Potencia asignada al AP
- vecinosInt: Lista de instancias de la clase AP, que representan los APs que son vecinos internos de este objeto.
- vecinosExt: Lista de instancias de la clase APExt, que representan los APs que son vecinos externos de este objeto.
- grado: Cantidad total de vecinos, coincide con la suma del tamaño de las listas vecinosInt y vecinosExt
- gradoSat: Grado de saturación del AP.
- asignado: Indica con un 1 si el AP está asignado para un algoritmo de optimización de canales o con un 0 si no.
- menorSat: Atributo usado por el algoritmo MCP, para indicar el FI en el mejor canal.

Métodos

- calculo_grado(): Calcula el grado del AP.
- calculo_grado_sat(): Calcula el grado de saturación.
- calculo_vecinos(listaaps): Este método se conecta al AP y obtiene la lista de vecinos mediante un escaneo de canales. Si la MAC de los vecinos obtenidos está dentro de la lista *listaaps* entonces lo agrega a la lista *vecinosInt* si no crea un objeto *APExt* y lo agrega a la lista *vecinosExt*. *listaaps* es una lista con todas las instancias de los APs de la red interna.
- calculo_potencia_rx(): Calcula la potencia recibida total por el AP.
- buscar_mac(): Se conecta al AP y obtiene la dirección MAC de la interfaz inalámbrica del AP
- buscar_pot(): Se conecta al AP y obtiene la potencia de transmisión configurada en el AP.
- buscar_canal(): Se conecta al AP y la frecuencia configurada en el AP.

5.4. Pseudo código / Descripción del software

- `cambiar_canal(nuevocanal)`: Se conecta al AP y configura el canal en el que transmite a *nuevocanal*
- `cambiar_potencia(nuevapotencia)`: Se conecta al AP y configura la potencia de transmisión a *nuevapotencia*
- `potencia_chan(chan)`: Calcula la interferencia co-canal en el canal *chan*
- `potencia_chan1(alpha, beta)`: Calcula el FI en el canal 1.
- `potencia_chan6(alpha, beta)`: Calcula el FI en el canal 6.
- `potencia_chan11(alpha, beta)`: Calcula el FI en el canal 11.
- `potencia_mejor_canal(alfa, beta, desempate, ultimo_canal)`: Calcula el mejor canal según el FI y el criterio de desempate.
- `cantidad_chan(chan)`: Calcula la cantidad de APs vecinos en el canal *chan*
- `cantidad_chan1(alpha, beta)`: Calcula el FI en el canal 1 según la cantidad de APs vecinos en los canales
- `cantidad_chan6(alpha, beta)`: Calcula el FI en el canal 6 según la cantidad de APs vecinos en los canales
- `cantidad_chan11(alpha, beta)`: Calcula el FI en el canal 11 según la cantidad de APs vecinos en los canales
- `cantidad_mejor_canal(alfa, beta, desempate, ultimo_canal)`: Calcula el mejor canal según el FI y el criterio de desempate, cuando el FI fue calculado usando cantidad de APs vecinos.

5.4.6. Clase APExt

Esta clase representa los Access Points vecinos de un determinado AP que no pertenecen a la red interna. Contiene los siguientes atributos:

- `MAC`: Dirección MAC sensada por el AP.
- `canal`: Canal o frecuencia en la que transmite el Access Point externo.
- `potencia`: Potencia vista por el AP interno transmitida por APExt .

5.4.7. Definición de los algoritmos de asignación de canales - algoritmo_canales.py

Este archivo contiene los algoritmos de optimización de uso de canales implementados:

- `MVA(listaaps, criterio_mejor_canal, alfa, beta, desempate, ultimo_canal)`

Capítulo 5. Implementacion

- `MCL(listaaps, criterio_mejor_canal, alfa, beta, desempate, ultimo_canal)`
- `MCP(listaaps, criterio_mejor_canal, alfa, beta, desempate, ultimo_canal)`

Cada algoritmo tiene como salida una lista con los objetos APs con los respectivos nuevos canales asignados como muestra la figura 5.7

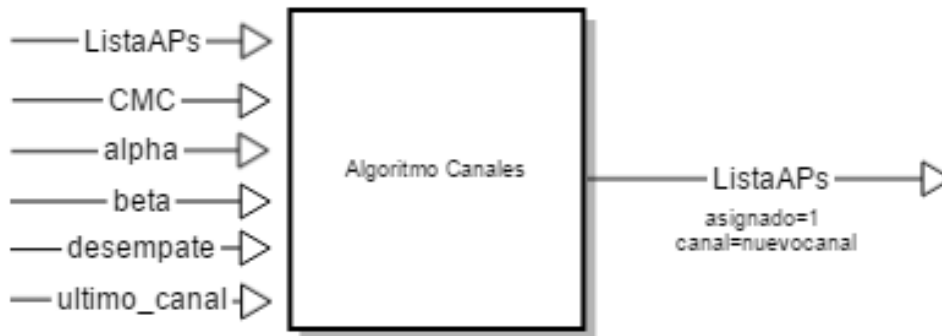


Figura 5.7: Diagrama de entrada y salida de los archivos de asignación de canales

5.4.8. Definición de los algoritmos de control de potencia - `algoritmos_potencia.py`

Este archivo contiene los algoritmos de control de potencia implementados

- `friis(lista_aps)`
- `iterativo(lista_aps)`

Cada algoritmo tiene como salida una lista con los objetos APs con las nuevas potencias como muestra la figura 5.8

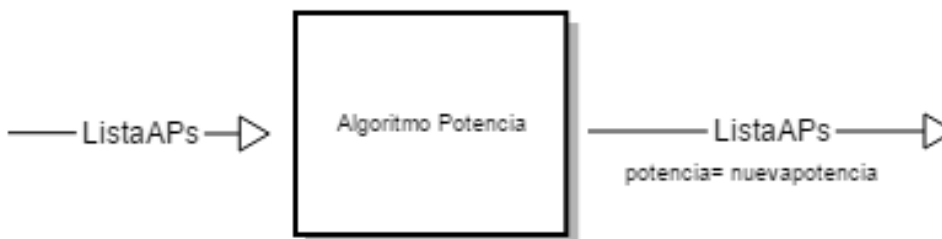


Figura 5.8: Diagrama de entrada y salida de los algoritmos de potencia

5.4.9. Cambio de configuración - configurar_APs.py

Este es el módulo encargado realizar los cambios de configuración de canales calculados previamente. Hace uso de los métodos definidos en la clase `AccessPoint` 5.4.5 para conectarse al AP y modificar el canal del mismo.

5.5. Posibilidad de expansión

El sistema se diseñó de forma modular con el fin de poder extender la funcionalidad del mismo a redes con APs de otros fabricantes. Un sistema de este tipo consiste en módulos simples que interactúan entre sí para facilitar la construcción de un software de mayor porte. Esto lo hace más flexible para la adaptación a nuevos casos de uso, como pueden ser nuevas funcionalidades en la controladora, cambios en versiones de software del AP o incluso cambios de marca de los mismos.

Dado que las redes de Ceibal pueden tener APs de otras marcas más allá de MikroTik, y aún varias versiones de software de la misma, es posible adaptar el programa creando una nueva clase de Access Point. Esta clase debe tener los mismos atributos y métodos que la clase `AccessPoint` 5.4.5. Esta nueva clase tendrá variantes como la forma de conectarse a los equipos, los comandos ejecutados y las salidas de los mismos. El formato de nombre del archivo de la clase debe ser `AccessPoint_Modelo`, donde *Modelo* es la variable para diferenciar las diferentes marcas o versiones de software. Esto se define a la hora de cargar los APs en el archivo `AP.csv` 5.4.2 mediante el primer valor, este debe ser igual al nuevo modelo creado.

Asimismo el software permite agregar nuevos algoritmos, para ello simplemente se debe agregar el código de los mismos en los archivos `algoritmoscanales.py` o `algoritmospotencia.py` y para usarlos simplemente referenciarlos en el archivo de configuración `wico.conf`.

Estas características hacen que Wico pueda ser modificado fácilmente para ser desplegado en nuevos escenarios más complejos.

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 6

Simulaciones y Puesta en Funcionamiento

Para validar el funcionamiento del código desarrollado se hicieron diferentes pruebas y simulaciones. En primer lugar se realizaron funciones en MATLAB [4] como una etapa inicial de implementación con el fin de corroborar y comparar el funcionamiento de los algoritmos. Este análisis se hizo también para casos particulares que se consideraron importantes para la validación.

Una vez obtenida una versión depurada de los algoritmos, se realizaron pruebas en un entorno de laboratorio similar a una red de un centro educativo. Finalmente se procedió a llevar el sistema a un liceo donde se comprobó el funcionamiento real del mismo.

6.1. Herramientas de Simulación

Para poder evaluar el funcionamiento de los algoritmos se crearon herramientas de software en MATLAB y Python. En esta sección se describen las principales funciones desarrolladas.

6.1.1. Representación gráfica de cobertura - graphAP.m

Esta función fue creada para obtener un gráfico representativo de un mapa de cobertura de una red de APs.

Capítulo 6. Simulaciones y Puesta en Funcionamiento

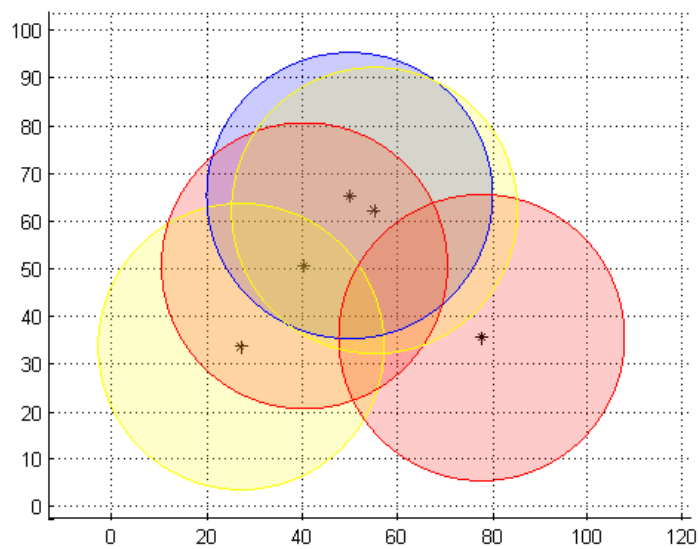


Figura 6.1: Mapa de cobertura creado por la función graphAP

6.1.2. Espectro de frecuencias - plot_potencia.m

Esta función permite visualizar el espectro de un ambiente simulado.

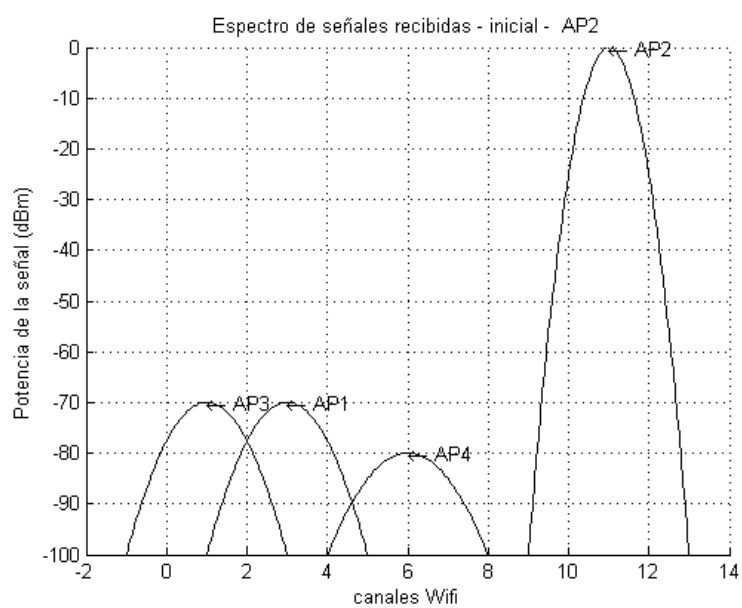


Figura 6.2: Espectro de frecuencias simulado en MATLAB

6.1. Herramientas de Simulación

6.1.3. Simulacion.m

Este programa fue utilizado para realizar las primeras pruebas de verificación de los algoritmos, el mismo simula una red donde los Access Point tienen ubicación, canales y potencia aleatorios. El programa ejecuta los algoritmos de optimización y muestra los gráficos comparativos. En la figura 6.3 se muestra los resultados de una simulación con cinco APs internos y 2 externos.

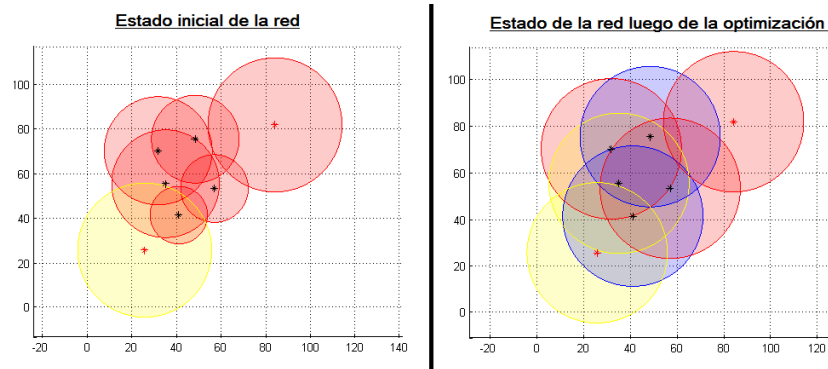


Figura 6.3: Resultado de la aplicación de los algoritmos de optimización

6.1.4. SimulacionEstadistica.m

Este programa de MATLAB fue desarrollado para analizar cómo se mejora el factor de interferencia simulando varios escenarios aleatorios. En la figura 6.4 se muestra una simulación de 100 casos diferentes, donde cada caso corresponde a una distribución aleatoria de la ubicación de los APs, en la misma se observa cómo mejora el FI cuando se aplican los algoritmos de optimización.

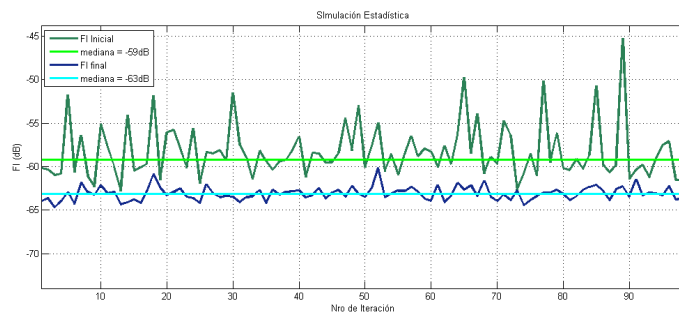


Figura 6.4: Factor de interferencia para 100 escenarios aleatorios.

6.1.5. Obtención de datos en archivo csv - escribir_archivo.py

Este programa de Python permite obtener varios archivos de texto con la información de los vecinos de cada AP. A partir de estos archivos y con las herramientas desarrolladas en MATLAB se obtiene una representación del estado general de la red. Esta herramienta tiene como salida los siguientes archivos:

Capítulo 6. Simulaciones y Puesta en Funcionamiento

- canalesint_scan.csv: Información de los canales de los AP internos.
- canalesext_scan.csv: Información de los canales de los AP externos.
- potint_scan.csv: Matriz con la información de con qué potencia ven los APs a los APs internos
- potext_scan.csv: Matriz con la información de con qué potencia ven los APs a los APs externos

6.1.6. Simulación desde un archivo csv - wico_test.py

Esta función permite simular la operación del sistema a partir de datos ingresados manualmente. Para ser ejecutado necesita que en la carpeta donde está el programa existan los siguientes archivos:

- canalesint_scan.csv: En este archivo se representa en una columna los canales de APs internos
- canalesext_scan.csv: En este archivo se representan las direcciones MAC y los canales de los AP externos en dos columnas
- potint_scan.csv: Matriz con la información de con qué potencia ven los APs a los APs internos
- potext_scan.csv: Matriz con la información de con qué potencia ven los APs a los APs externos

En la figura 6.5 se muestra un ejemplo de estos archivos.

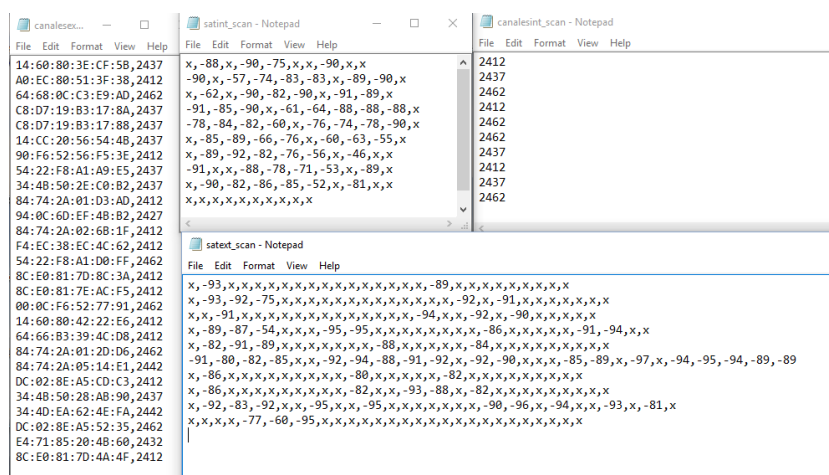


Figura 6.5: Archivos necesarios para la ejecución de maindesdearchivo.py

6.2. Simulaciones

El primer paso para la implementación de los algoritmos fue escribir el código de los mismos en MATLAB. Con la ayuda de los gráficos de cobertura se pudo depurar los algoritmos y verificar su buen funcionamiento. En la figuras 6.6 a 6.10 se muestra una simulación con 25 APs realizada en MATLAB. En la 6.6 se muestra el estado inicial de la red que se obtuvo de forma aleatoria. En las figuras 6.7 y 6.8 se puede observar la asignación de canales realizada por el algoritmo MVA y la posterior optimización de potencia de transmisión. Lo mismo se muestra en las figuras 6.9 y 6.10 para el algoritmo MCL.

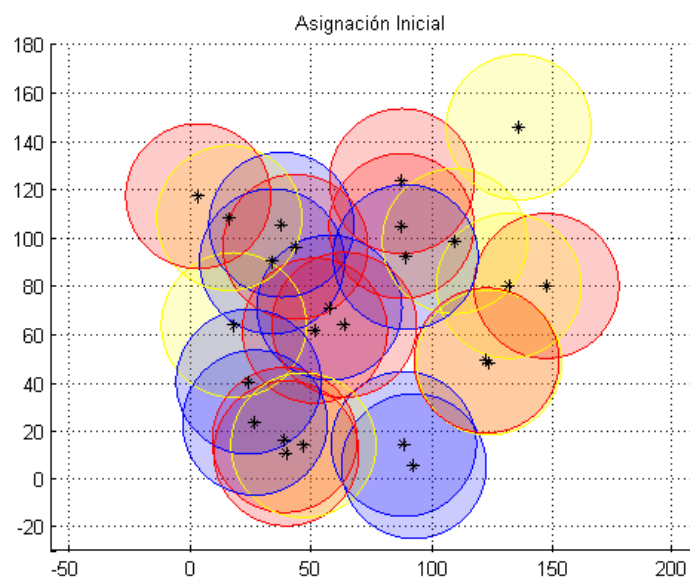


Figura 6.6: Simulación 25 APs: Mapa de cobertura inicial

Capítulo 6. Simulaciones y Puesta en Funcionamiento

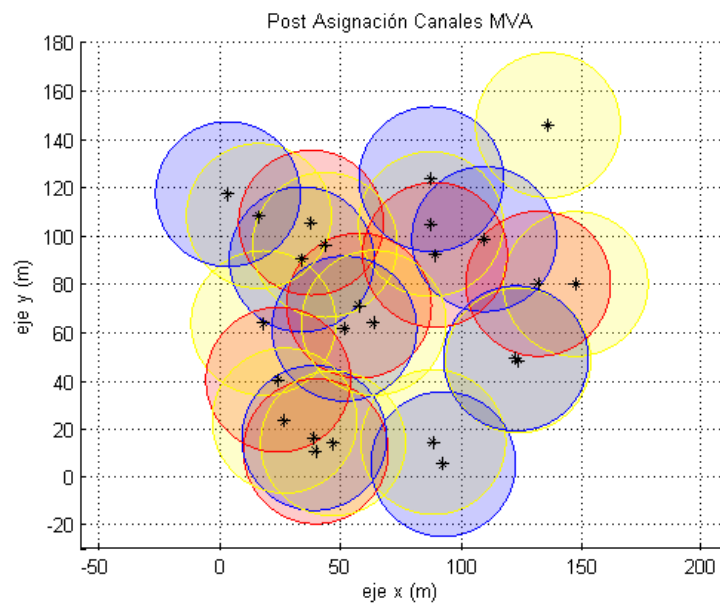


Figura 6.7: Simulación 25 APs: Mapa de cobertura post MVA

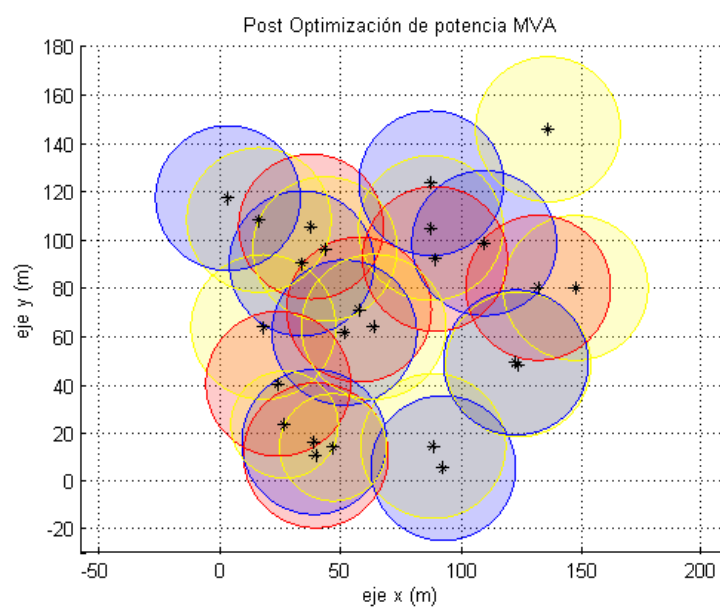


Figura 6.8: Simulación 25 APs: Mapa de cobertura post MVA y potencia

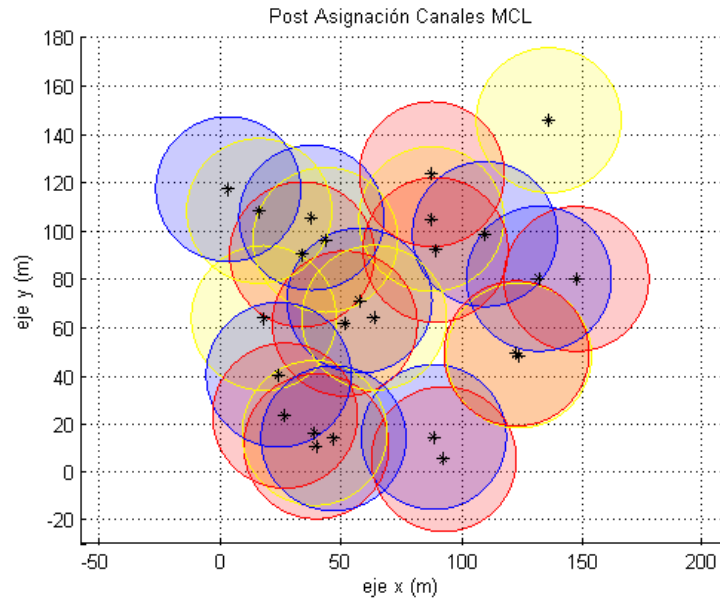


Figura 6.9: Simulación 25 APs: Mapa de cobertura post MCL

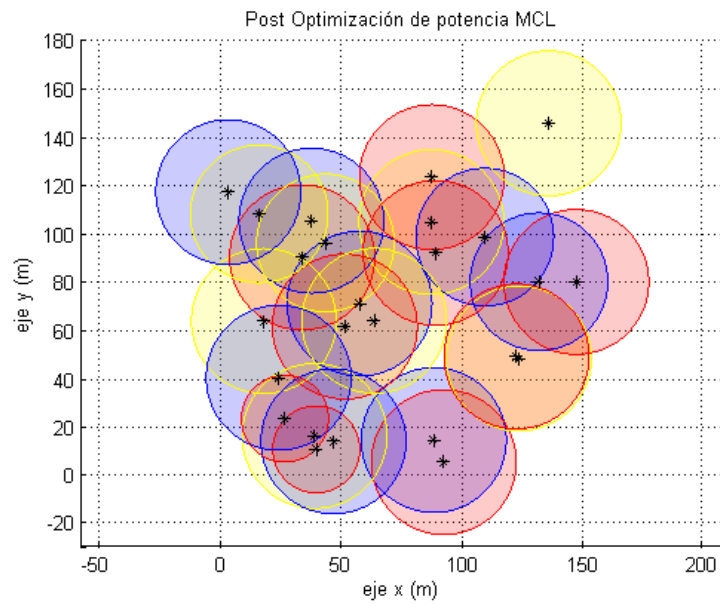


Figura 6.10: Simulación 25 APs: Mapa de cobertura post MCL y potencia

Adicionalmente se realizó un análisis estadístico de los algoritmos mediante un programa de MATLAB que simula cien escenarios aleatorios de veinticinco APs. Como se ve en la figura 6.11 la aplicación de los algoritmos de asignación de canales disminuye en promedio unos 5dBm el factor de interferencia total de la red.

Capítulo 6. Simulaciones y Puesta en Funcionamiento

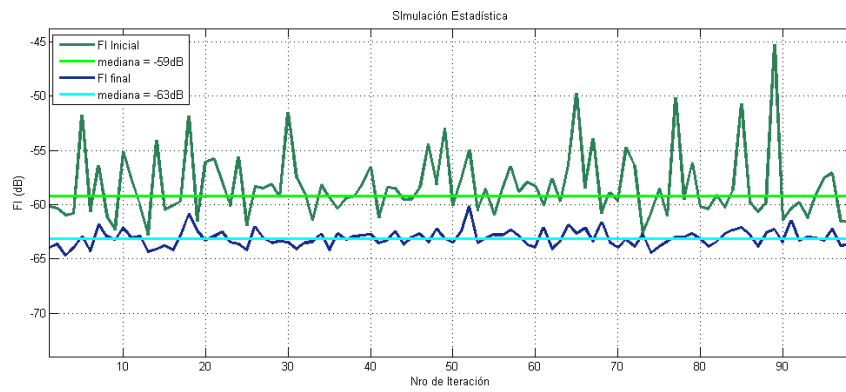


Figura 6.11: Simulación Estadística

6.2.1. Casos Particulares

Una vez comprobado el funcionamiento de los algoritmos se procedió a implementar el código en Python. Se utilizaron como pseudocódigo los trabajos realizados en MATLAB. Además de la implementación del sistema central se elaboraron módulos de prueba para analizar el funcionamiento de los algoritmos en casos particulares.

Se analizaron seis casos diferentes que se caracterizan por ubicaciones particulares de los APs. El objetivo de este análisis fue comparar el funcionamiento de los algoritmos en situaciones que son sencillas de resolver manualmente por un técnico, en las cuales se quería observar el resultado de la asignación hecha por el programa.

Caso 1

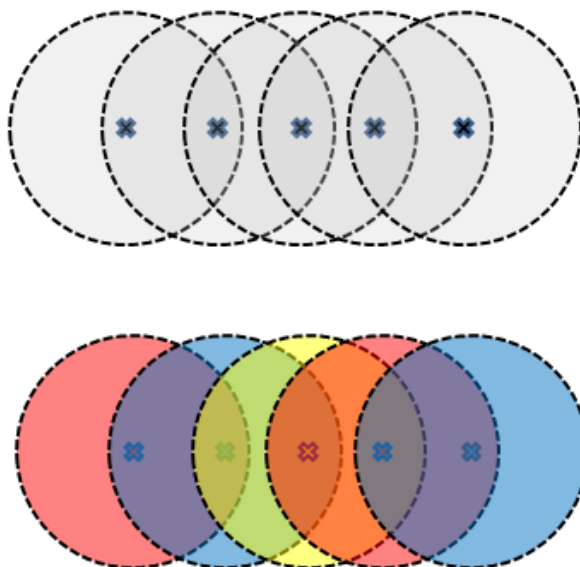


Figura 6.12: Caso Particular 1



Figura 6.13: Caso Particular 1: Grafo Asociado

En este caso se tienen cinco APs distribuidos en fila como se muestra en la figura 6.12. El grafo asociado a la red se muestra en la figura 6.13. El objetivo de

Capítulo 6. Simulaciones y Puesta en Funcionamiento

la asignación de canales es lograr un factor de interferencia total de la red igual a cero. Son varias las combinaciones de canales que logran este objetivo, pero la asignación óptima se da cuando además se utilizan los canales del espectro de forma lo más distribuida posible. Por ejemplo, para este caso se podría lograr una asignación con factor interferencia igual a cero utilizando solo dos canales. En las figuras 6.14 y 6.15 se muestran dos asignaciones con $FI = 0$ utilizando dos y tres canales respectivamente. Para una mejor utilización del espectro siempre es mejor utilizar la mayor cantidad de canales posibles. En las figuras 6.16 y 6.17 podemos comparar el uso del espectro para ambos casos.

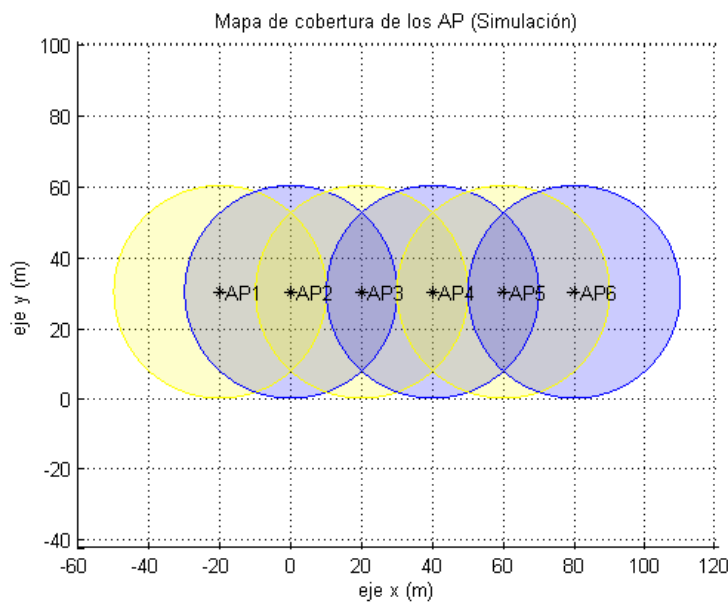


Figura 6.14: Caso Particular 1 : Asignación con 2 canales

6.2. Simulaciones

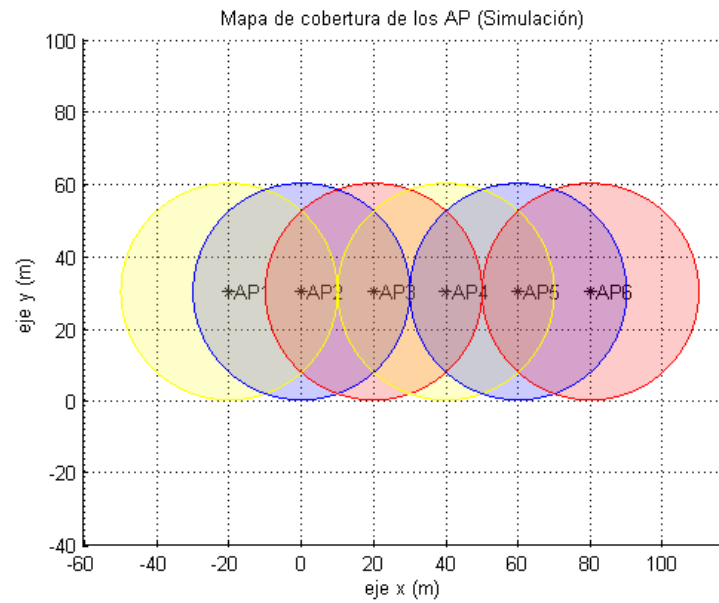


Figura 6.15: Caso Particular 1: Asignación con 3 canales

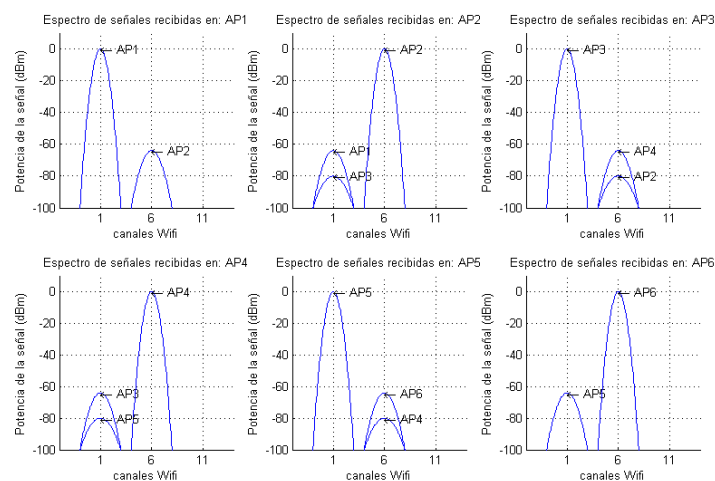


Figura 6.16: Caso Particular 1: Espectro 2 canales utilizados

Capítulo 6. Simulaciones y Puesta en Funcionamiento

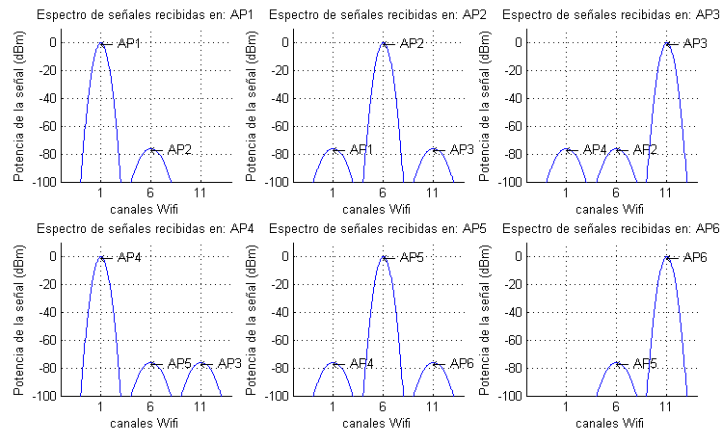


Figura 6.17: Caso Particular 1: Espectro 3 canales utilizados

El análisis se hizo para todos los algoritmos y con todas las variaciones de CMC(3.3.1) y criterios de desempate (3.3.3) posibles. En todos los casos se logró $FI = 0$, pero se observó que utilizando el criterio de desempate normal la asignación quedaba con dos canales, mientras que para los demás casos se logró asignación óptima.

Caso 2

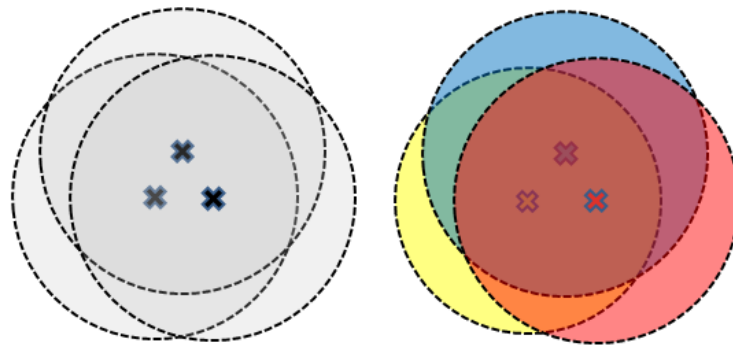


Figura 6.18: Caso Particular 2

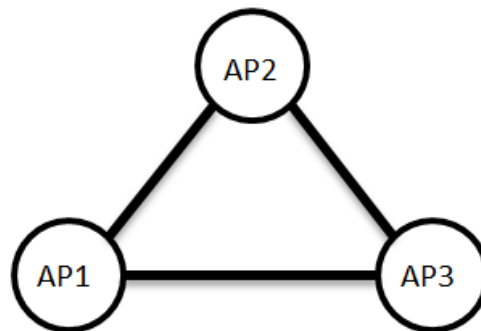


Figura 6.19: Caso Particular 2: Grafo Asociado

Este es un escenario simple donde hay tres APs que se ven entre sí como se muestra en la figura 6.18. La asignación óptima es en la que todos los APs quedan asignados con canales diferentes. Se comprobó que con todas las combinaciones de variables posibles, todos los algoritmos lograban el objetivo.

Capítulo 6. Simulaciones y Puesta en Funcionamiento

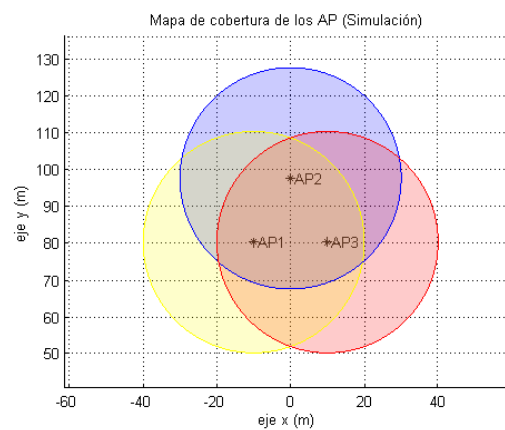


Figura 6.20: Caso Particular 2: Asignación óptima

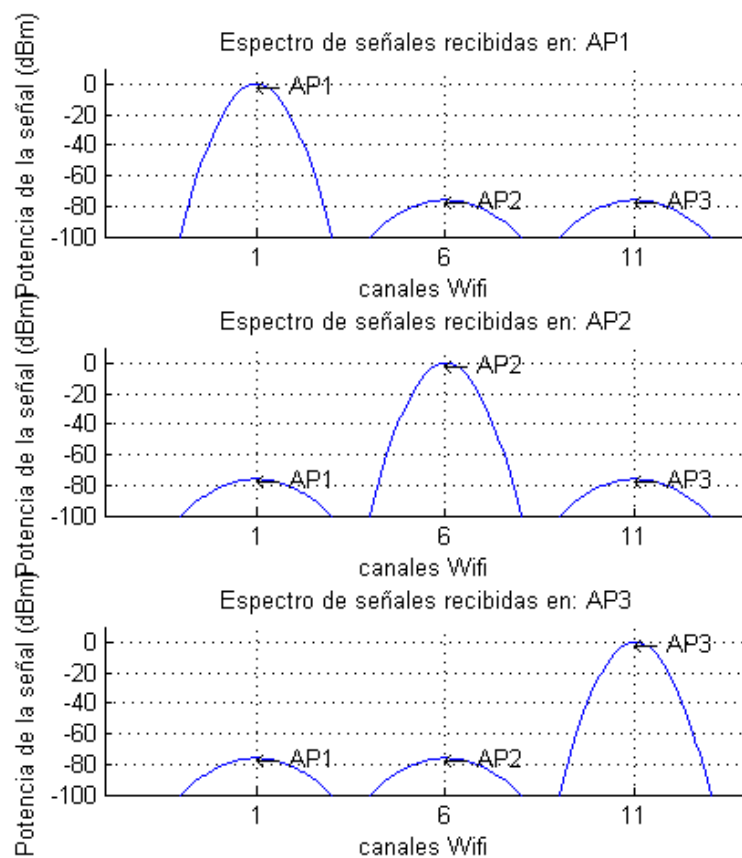


Figura 6.21: Caso Particular 2: Utilización del espectro

Caso 3

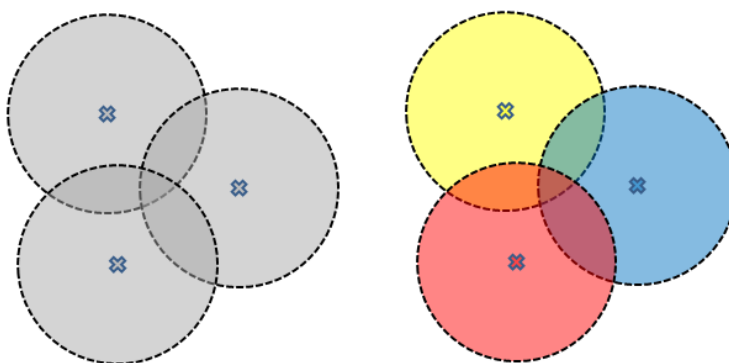


Figura 6.22: Caso Particular 3

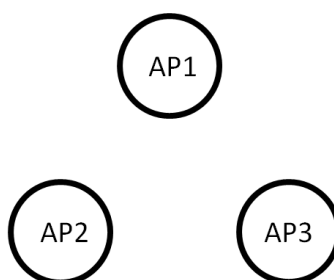


Figura 6.23: Caso Particular 3: Grafo Asociado

En este caso la red consiste en tres APs que no se ven entre sí. En este escenario con cualquier asignación se obtiene factor de interferencia igual a cero, pero al igual que se explicó en el caso 1 la mejor asignación se logra obteniendo el mejor uso del espectro, es decir asignando canales diferentes a cada AP.

Para lograr el objetivo de asignación óptima los algoritmos deben ser capaces de llevar un registro de los canales ya utilizados al momento de asignar un nuevo canal. Esto es así porque el AP que se está recorriendo no tiene información de la asignación de los APs que no son sus vecinos. Roundrobin es el único criterio de desempate que lleva un registro de los canales ya utilizados, mediante simulaciones se logró mostrar que solo utilizando este criterio se lograba la asignación óptima con todos los algoritmos.

Capítulo 6. Simulaciones y Puesta en Funcionamiento

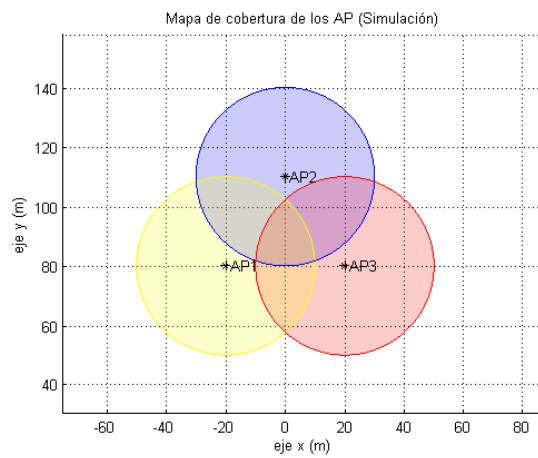


Figura 6.24: Caso Particular 3: Asignación óptima

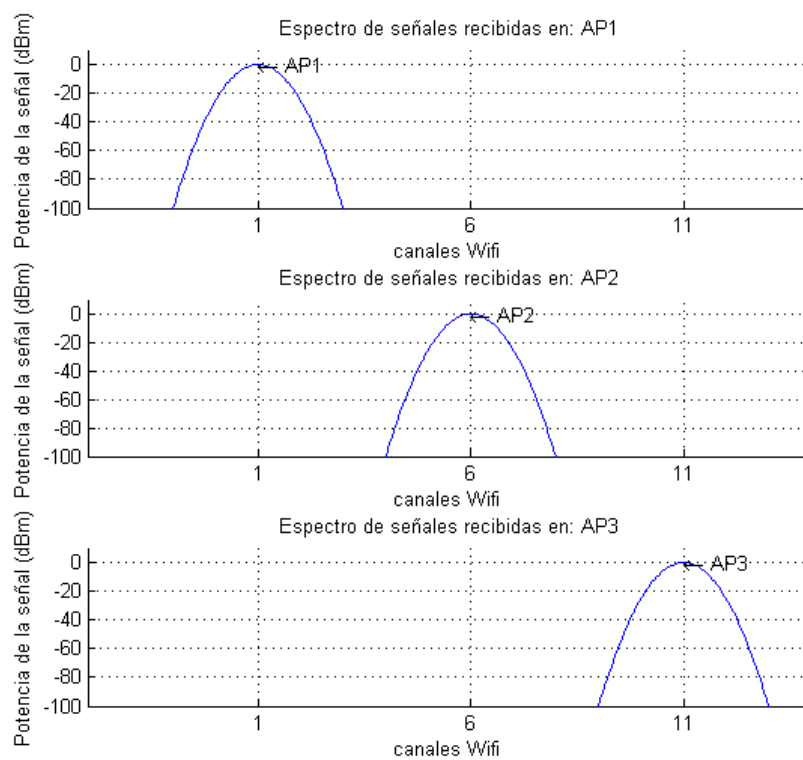


Figura 6.25: Caso Particular 3: Utilización del espectro

Caso 4

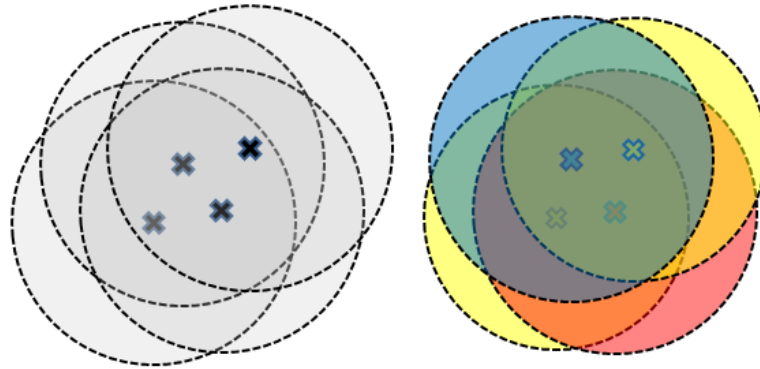


Figura 6.26: Caso Particular 4

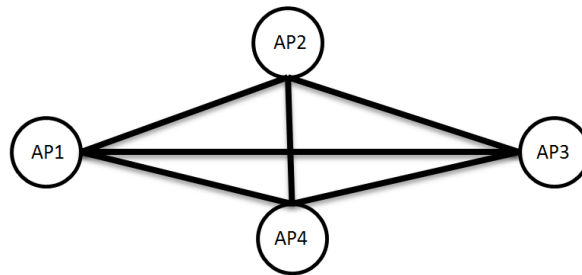


Figura 6.27: Caso Particular 4: Grafo Asociado

Este es uno de los casos más interesantes para analizar, en el mismo hay cuatro APs que se ven todos entre sí. Como se puede deducir la asignación con factor de interferencia igual a cero no es posible, porque solo se cuenta con tres canales para la asignación. En otras palabras, en todas las asignaciones posibles siempre hay al menos dos APs a los que se le asigna el mismo canal.

Capítulo 6. Simulaciones y Puesta en Funcionamiento

La asignación óptima se da cuando los APs que repiten el canal están lo más alejados posible (se ven con menor potencia). En la figuras 6.28 se muestra el mapa de cobertura de esta asignación

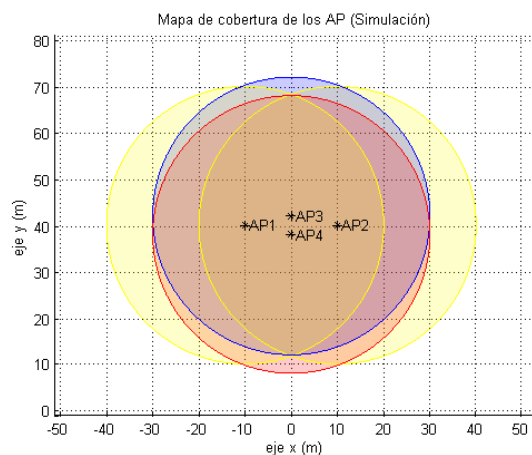


Figura 6.28: Caso Particular 4: Asignación óptima

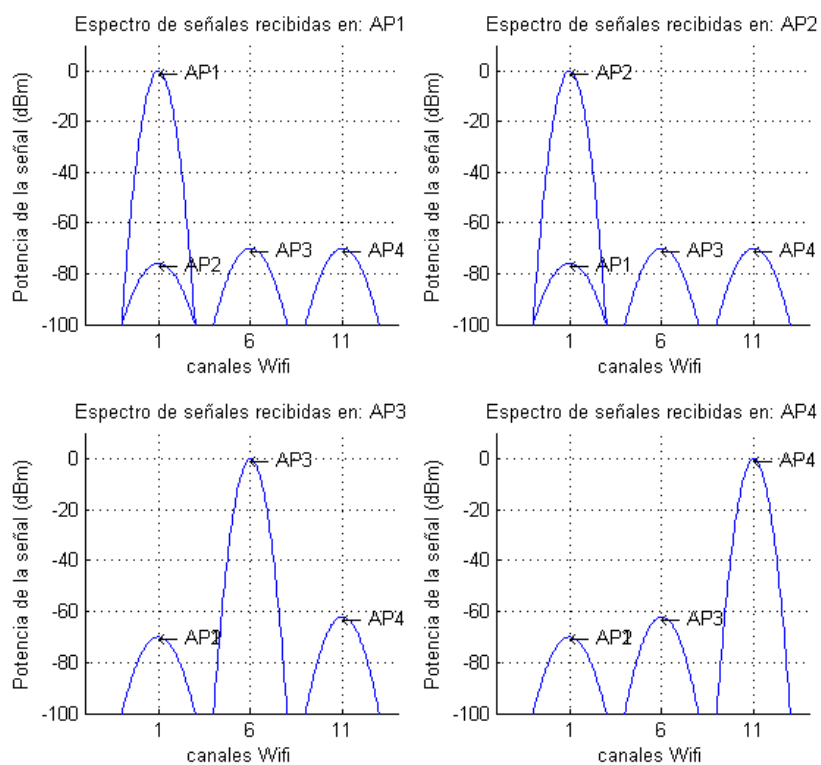


Figura 6.29: Caso Particular 4: Utilización del espectro

6.2. Simulaciones

Si luego de la asignación de canales se utiliza control de potencia, se observa que asignando canales de forma óptima se logra factor de interferencia igual a cero permitiendo que cada AP cubra el mayor radio de cobertura posible. En las figuras 6.30 y 6.31 se muestran dos asignaciones con control de potencia. En la primera, luego de una asignación óptima de canales. Y en la segunda, el caso en el que los APs que repiten el canal son los que se encuentran más cercanos.

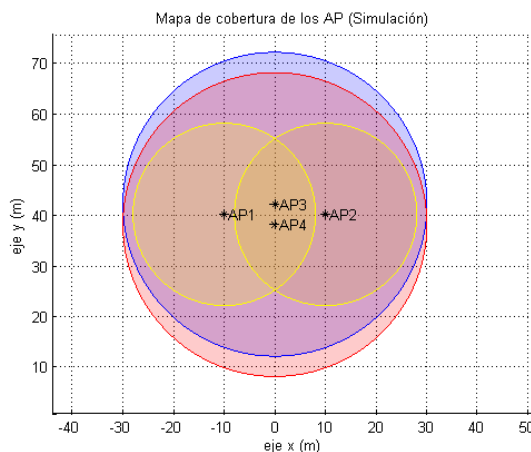


Figura 6.30: Caso Particular 4: Asignación óptima con control de potencia

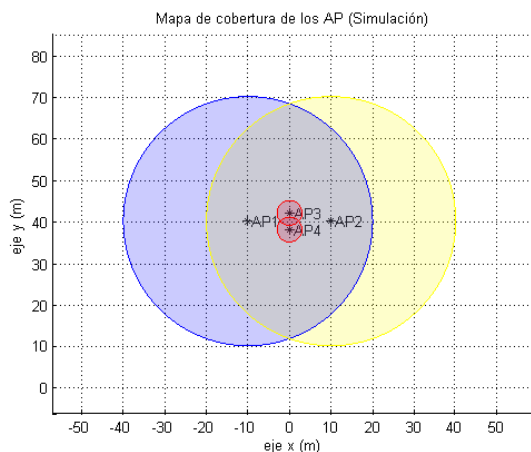


Figura 6.31: Caso Particular 4: Peor caso de asignación con control de potencia

Cuando se realizó el análisis del desempeño de los algoritmos para este caso, se comprobó que tanto con MVA como con MCL se logra asignación óptima. MCP es un algoritmo muy sencillo, que se creó con la finalidad de comparar los resultados que se obtienen cuando el grafo es recorrido de forma inteligente. Este caso presenta la particularidad que todos los APs tienen el mismo grado, eso hace que el recorrido de MCP dependa de factores como el orden en que se listan los APs. Es por esto que MCP no logra el mismo resultado.

Capítulo 6. Simulaciones y Puesta en Funcionamiento

Caso 5

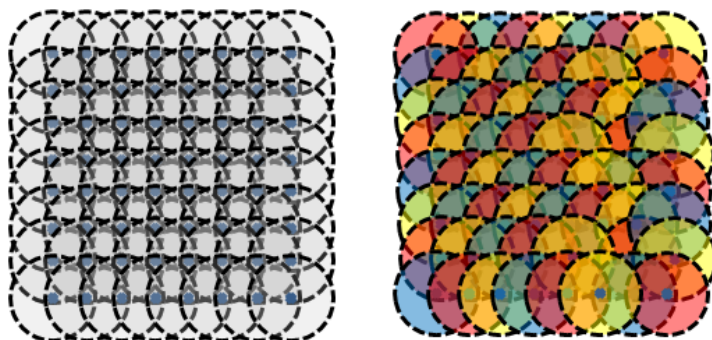


Figura 6.32: Caso Particular 5

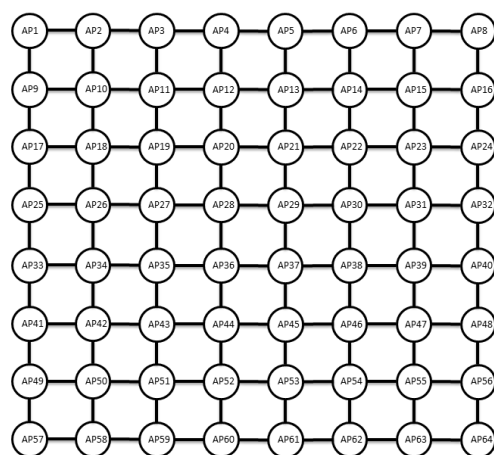


Figura 6.33: Caso Particular 5: Grafo Asociado

El caso 5 se diseñó para comprobar el funcionamiento de los algoritmos en un escenario con una gran densidad de APs. La particularidad de este caso es que a pesar de la gran cantidad de nodos, es posible obtener una asignación con factor de interferencia igual a cero. En la figura 6.34 se observa una asignación óptima y en la figura 6.35 se muestra el espectro resultante para algunos APs. Se comprobó mediante estas simulaciones que todos los algoritmos lograban obtener la mejor asignación.

6.2. Simulaciones

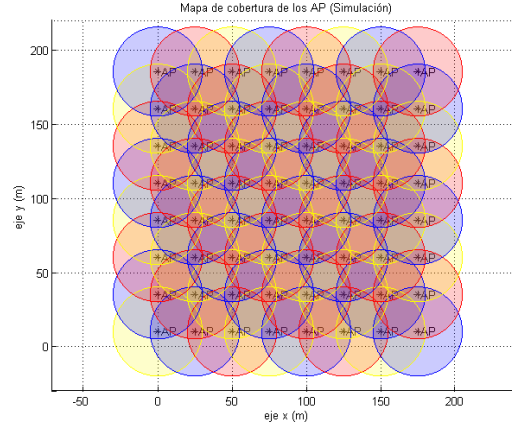


Figura 6.34: Caso Particular 5: Mapa de cobertura

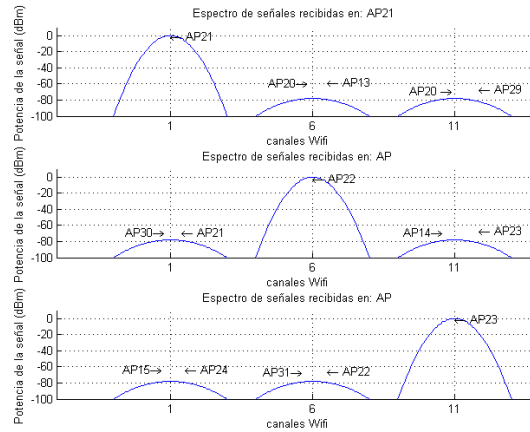


Figura 6.35: Caso Particular 5: Utilización del espectro

Cuando se implementó el código del sistema, se creó un algoritmo [?] que busca entre las asignaciones posibles, la que resulta con menor factor de interferencia. Este algoritmo nos sirvió para saber qué tan buena era la asignación obtenida con los algoritmos MVA, MCL y MCP. La cantidad de combinaciones posibles de asignaciones en un escenario con n APs se calcula como $CR_n^3 = \frac{(n+3-1)!}{(3-1)!n!}$. En un escenario como este, donde se tienen 64 APs, la cantidad de combinaciones posibles es aproximadamente 3×10^{30} . Esta cantidad de iteraciones es imposible de hacer con una computadora ordinaria, los algoritmos desarrollados llegan a la solución mínima en aproximadamente 26ms. Esto demuestra la fortaleza de la asignación por recorrido del grafo asociado.

Caso 6

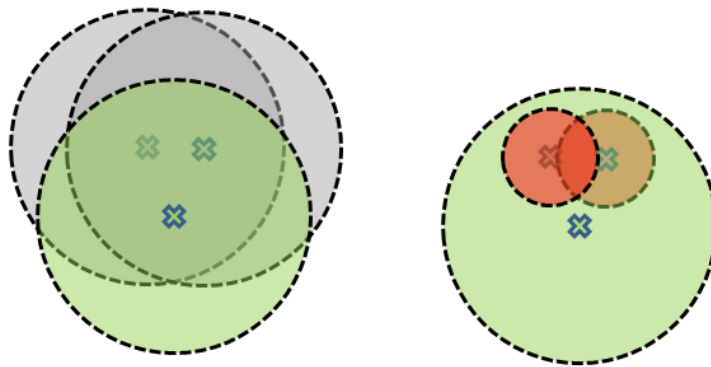


Figura 6.36: Caso Particular 6

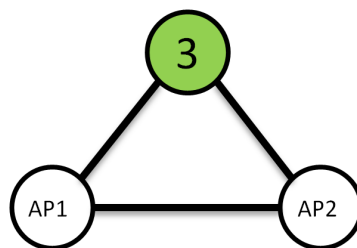


Figura 6.37: Caso Particular 6: Grafo Asociado

Este escenario se analizó para comprobar el funcionamiento de los algoritmos cuando aparece un AP externo en un canal diferente a los asignables (1, 6 u 11). Como se muestra en la 6.36 en este caso hay dos APs internos y uno externo con el canal 3 asignado. En la 6.37 se muestra el grafo asociado.

6.2. Simulaciones

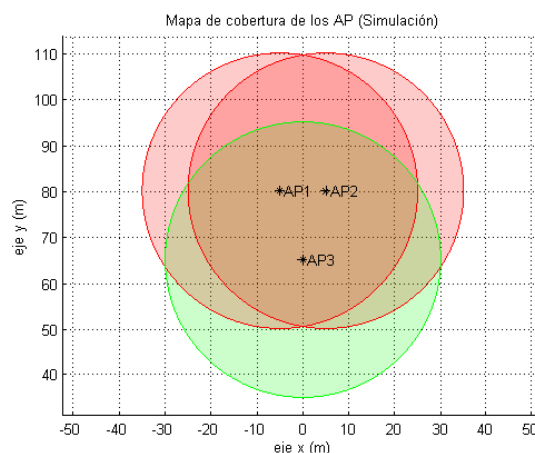


Figura 6.38: Caso Particular 6: Mapa de cobertura

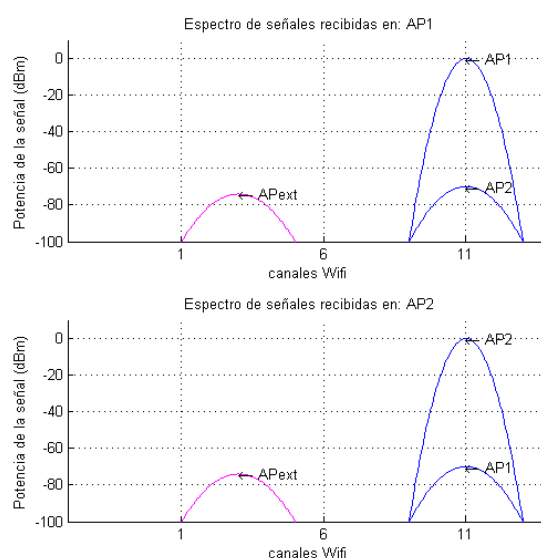


Figura 6.39: Caso Particular 6: Utilización del espectro

La asignación óptima para este caso es evitar que los APs queden asignados con canales adyacentes al del AP externo. Es decir a ambos APs se le debe asignar el canal 11. Esto es así porque es preferible la interferencia co-canal a la de canal adyacente. Mediante las simulaciones realizadas se demostró que todos los algoritmos, con todas sus variaciones, lograban el objetivo planteado.

Este es un caso particular donde se muestra además que la asignación de potencia no siempre es favorable. Como se muestra en la figura 6.40 al aplicar el control de potencia, el área de cobertura de los APs internos se ve muy reducida. En un caso como este, es preferible tener interferencia co-canal, a no tener una buena cobertura.

Capítulo 6. Simulaciones y Puesta en Funcionamiento

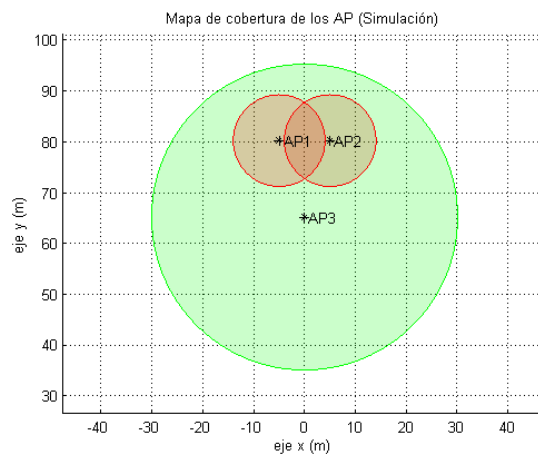


Figura 6.40: Caso Particular 6: Control de potencia contraproducente

6.3. Pruebas en Caso Real

Luego de ya haber culminado las etapas de implementación en Python, se probó el sistema en un caso real. Las redes a optimizar son puramente conformadas por Access Points de la marca MikroTik, por lo que las pruebas se llevaron a cabo en un centro que cumplía esta característica.

El trabajo de campo consistió en dos etapas. La primera fue de relevamiento de estado de la red de dos liceos, para realizar simulaciones sobre escenarios reales y conocer cómo se comportan los algoritmos desarrollados. La segunda etapa fue una prueba del sistema como producto final.

6.3.1. Primera Visita o relevamiento de datos

Para el relevamiento inicial Ceibal propuso visitar dos centros educativos: el Liceo 16 “Desembarco de la Agraciada” que tenía instalada una red de seis Access Points Mikrotik, y el Liceo 36 “Batlle y Ordoñez” cuya red Ceibal consistía de once Access Points también MikroTik. Ambos liceos cuentan con un servidor central CentOS.

El proceso de relevamiento consistió en obtener, mediante el comando `interface wireless scan`, los datos de vecindades de todos los APs. De cada AP se obtuvo el SSID, dirección MAC, canal en el que transmite y potencia recibida de cada vecino.

A partir de los datos recolectados se obtuvieron los grafos asociados a ambas redes. En las figuras 6.41 y 6.42 se muestran los mismos.

Como se puede observar, en los casos reales los grafos son mucho más complejos que los analizados en casos particulares. La gran cantidad de Access Points externos hace que el recorrido del grafo poco influya en los resultados obtenidos. Mediante la herramienta desarrollada en Python *wico.test.py*, se hicieron simulaciones de asignación de canales para los distintos algoritmos. En las tablas 6.1 y 6.2 se muestran los FI obtenidos. En la figura 6.43 se muestra una representación del espectro obtenido en uno de los liceos.

Algoritmo	Factor de interferencia
FI inicial	-56.22dBm
MVA	-54.02dBm
MCL	-54.02Bm
MCP	-56.48dBm
FI mínimo	-56.29dBm

Tabla 6.1: FI obtenidos en simulaciones para los datos relevados del liceo 16

Capítulo 6. Simulaciones y Puesta en Funcionamiento

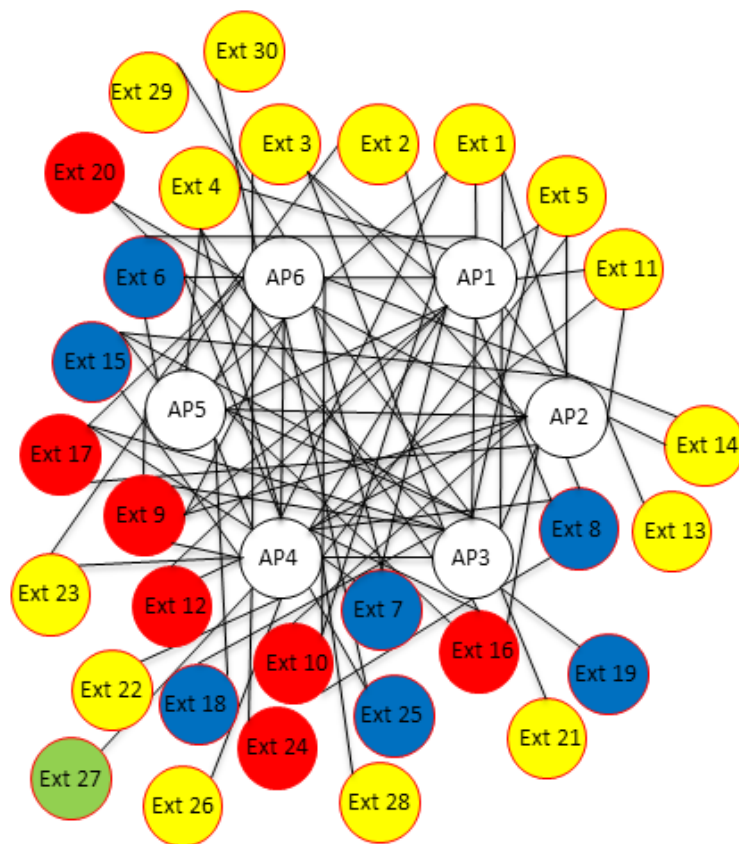


Figura 6.41: Grafo asociado a la red relevada del liceo 16

Algoritmo	Factor de interferencia
FI inicial	-47.43dBm
MVA	-60.28dBm
MCL	-60.10dBm
MCP	-60.28dBm
FI mínimo	-60.28dBm

Tabla 6.2: FI obtenidos en simulaciones para los datos relevados del liceo 36

6.3. Pruebas en Caso Real

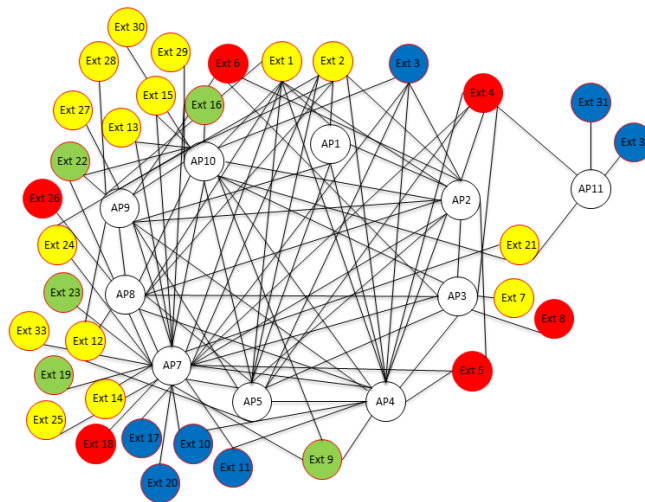


Figura 6.42: Grafo asociado a la red relevada del liceo 36

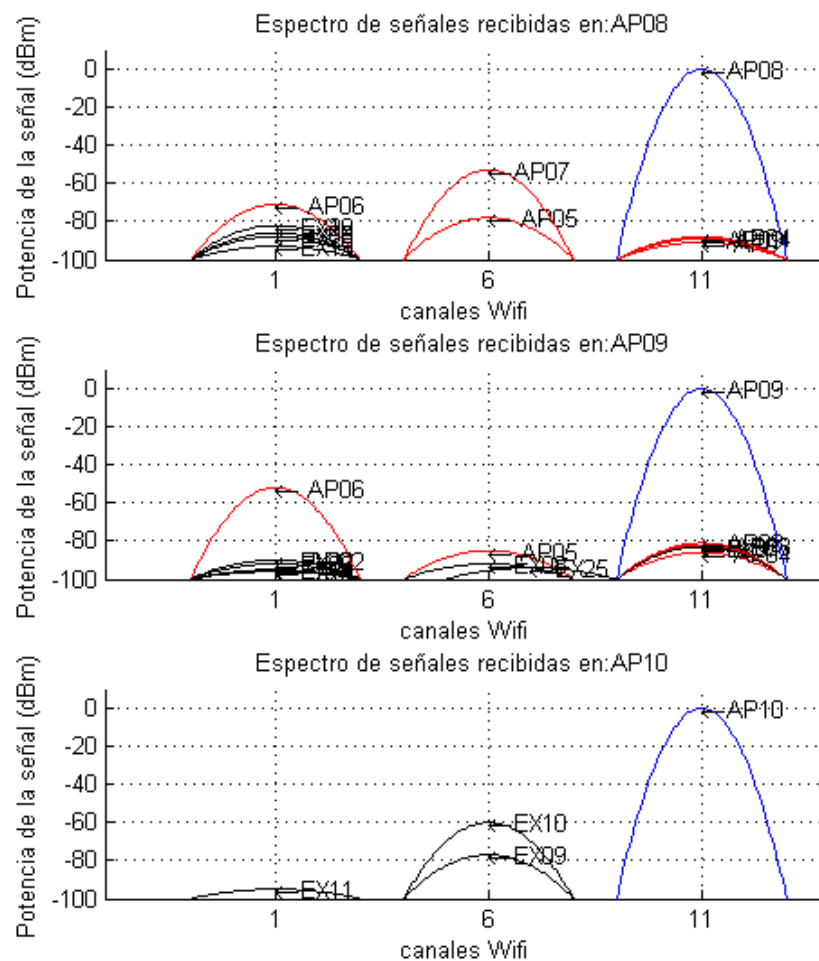


Figura 6.43: Utilización del espectro resultante de la aplicación de algoritmos en el liceo 36

6.3.2. Implementación en caso real

Se analizaron las redes de los dos liceos para decidir en cuál se iba a implementar el sistema. El liceo 36 presentaba una topología más compleja, donde no todos los APs se veían entre sí, incluso uno estaba aislado del resto. Además, a partir de las simulaciones realizadas se concluyó que en el liceo 16 el margen de mejora del FI era muy bajo. Como el objetivo de la visita era analizar el desempeño del sistema, se decidió hacerla en el liceo 36.

Luego de instalado y probado el sistema en un servidor proporcionado por Ceibal, se procedió con la visita al liceo. La primer ejecución del programa falló. Luego de horas de depuración, se descubrió que el problema era debido a que las versiones de software de los APs eran diferentes. El sistema había sido desarrollado para la versión 4.11 de RouterOS, algunas versiones instaladas en los APs del liceo tenían diferencias en los comandos utilizados para configurar y ver el estado de los APs. Para solucionar este inconveniente se debió desarrollar un nuevo módulo de software para el sistema, en realidad lo que se realizó fue crear una clase nueva para los APs que contaban con versiones que se comportaban diferente a la 4.11. Además se modificó el archivo de configuración (AP.csv). A partir de ese momento en el mismo debe especificarse la versión de software instalada en cada uno de los APs. La solución al problema fue sencilla ya que el programa se había desarrollado en forma modular, de la misma forma que se adaptó el sistema a diferentes versiones de software, se podría ajustar el mismo a otras marcas de Access Point.

Ya con estos inconvenientes solucionados, se realizó una última visita al liceo para la implementación del software. La misma fue exitosa ya que se logró ejecutar Wico sin errores. Se probó con todas las combinaciones posibles de algoritmos y variables y se compararon los resultados. Como conclusión inmediata se pudo observar que el FI_{total} de la red bajó. En la tabla 6.3 y el gráfico 6.44 se muestran los resultados obtenidos.

6.3. Pruebas en Caso Real

Algoritmo	CMC	Desempate	FI_i	FI_f	FI_{min}	$FI_i - FI_f$	$FI_f - FI_{min}$
MVA	potencia	normal	-49	-62	-70	13	8
MVA	potencia	random	-50	-62	-69	12	8
MVA	potencia	round-robin	-51	-58	-70	7	12
MVA	cantidad	normal	-47	-54	-69	7	15
MVA	cantidad	random	-50	-50	-70	0	20
MVA	cantidad	round-robin	-45	-53	-61	8	9
MCL	potencia	normal	-45	-58	-70	12	13
MCL	potencia	random	-48	-58	-70	10	12
MCL	potencia	round-robin	-49	-63	-69	14	7
MCL	cantidad	normal	-46	-52	-69	6	16
MCL	cantidad	random	-49	-54	-70	6	15
MCL	cantidad	round-robin	-50	-53	-70	4	17
MCP	potencia	normal	-47	-66	-70	19	4
MCP	potencia	random	-46	-63	-70	17	7
MCP	potencia	round-robin	-32	-65	-70	33	4
MCP	cantidad	normal	-35	-53	-70	18	17
MCP	cantidad	random	-40	-55	-70	15	15
MCP	cantidad	round-robin	-42	-55	-69	13	13

Tabla 6.3: FI del liceo para todos los algoritmos. -Valores en dBm-

Capítulo 6. Simulaciones y Puesta en Funcionamiento

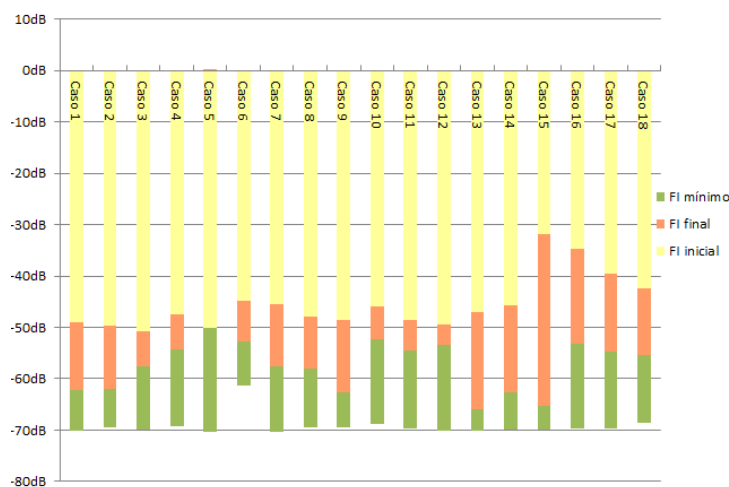


Figura 6.44: Resultados obtenidos en la prueba en el liceo 36

Como se puede observar el sistema pudo mejorar en todos los casos la asignación de canales. En promedio se obtuvo una mejora de 12dBm respecto a la asignación hecha por el técnico de Ceibal. Además se observa que con el criterio de mejor canal *potencia*, el resultado obtenido es muy cercano a la asignación mínima. En promedio unos 8dBm por debajo de la misma.

En todos los casos se utilizó el algoritmo de control de potencia que calcula la potencia a transmitir según la ecuación de Friis de propagación en el espacio libre [4.4].

6.4. Comparación entre Algoritmos

En las secciones anteriores se vio el comportamiento de los diferentes algoritmos con sus variaciones en distintos escenarios. En base a estos análisis, se comparó el desempeño de los mismos.

En la primer versión de los algoritmos desarrollados la decisión del mejor canal para un APs se calculaba según la potencia recibida en los canales asignables. Cuando dos o más canales estaban en iguales condiciones, el algoritmo simplemente elegía siempre el primero de la lista. En los casos particulares vistos en la sección anterior, se vio que este tipo de asignación no es óptima ya que no se hace un buen uso del espectro. Para solucionar este problema se crearon dos métodos de desempate, uno que elige aleatoriamente entre los canales disponibles (*random* [3.3.3]) y otro que lleva un registro del último canal asignado y en caso de tener que desempatar elige el siguiente al último (*round-robin* [3.3.3]). En el análisis del caso 3 [6.2.1], donde los APs no se ven entre sí y la elección de canal para cada AP se tiene que hacer independientemente del canal asignado a los otros, se observó que sólo utilizando el criterio de desempate *round-robin* se lograba una asignación

6.4. Comparación entre Algoritmos

con buen uso del espectro.

Para analizar el desempeño de los algoritmos desarrollados[3] (MVA y MCL), se creó otro algoritmo (MCP) con menor inteligencia a la hora de recorrer el grafo. A la hora de estudiar el funcionamiento de los algoritmos en un caso real se demostró que el resultado de FI total no dependía del algoritmo de recorrido de grafo elegido, aunque se pudo demostrar que eligiendo un buen criterio de elección del mejor canal se obtienen mejores resultados. Con el análisis del caso 4, se demostró que el algoritmo con menor inteligencia de recorrido de grafo, no resuelve de manera óptima la asignación de redes con topologías complejas.

El objetivo de la asignación de canales es disminuir el FI , por esta razón se creó un algoritmo[3.4.4] que calcula el FI en todas las asignaciones posibles y se queda con la de FI mínimo. Si bien este algoritmo resuelve el problema para todos los casos planteados, la asignación no siempre resuelve el problema con el mejor uso del espectro. Además se comprobó que para redes con más de 15 APs, el tiempo requerido para el procesamiento no era tolerable. Los algoritmos desarrollados (MCP, MVA y MCL) tienen tiempos de ejecución menores a los 100ms ¹.

En los análisis de los casos particulares se vio que no siempre es posible la asignación con $FI = 0$, para estos casos se diseñaron dos algoritmos de control de potencia para disminuir el radio de cobertura de los APs vecinos que comparten el canal asignado. Se observó que en algunos casos, el resultado del control de potencia resulta en una disminución total de la cobertura, pero si la distribución geográfica de los APs y la asignación de canales es buena el efecto del control de potencia es favorable.

Los dos algoritmos de potencia desarrollados cumplen los objetivos. Se observó además que los algoritmos de potencia no siempre pueden asignar la potencia óptima debido a que los APs trabajan con un rango de potencia de transmisión limitado.

¹Medidos en un Notebook con procesador core i7 y 12Gb de RAM

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 7

Conclusiones

7.1. Conclusiones generales

A lo largo del proyecto se logró comprender el protocolo que utilizan las redes WiFi, su funcionamiento y sus flaquezas. Se recurrió a trabajos académicos para investigar sobre métodos de asignación de canales y control de potencia. Gracias a este estudio se logró diseñar un indicador que permite medir la calidad de la asignación de canales y potencia en una red WiFi.

Para lograr el objetivo del proyecto, se diseñaron tres algoritmos de asignación de canales y potencia que optimizan el uso del espectro. Estos algoritmos tienen tiempos de ejecución que se adaptan a las necesidades de una red en servicio, y durante la aplicación de los mismos no se afecta considerablemente el desempeño de la red.

El sistema realizado se desarrolló sobre una herramienta de software libre, y los requerimientos para su implementación son mínimos. Esto permite el despliegue de la aplicación en grandes redes con costos muy bajos. Además el código desarrollado permite ser adaptado a cualquier marca de Access Point.

El objetivo principal del proyecto se cumplió ampliamente, el sistema automático permitió mejorar la asignación con respecto a la realizada por técnicos especializados. Al comienzo del proyecto se planteó como objetivo no empeorar el FI en más de 5dBm, no solo se cumplió con esto, sino que se logró diseñar algoritmos que mejoran el mismo en más de 5dBm.

A pesar que todos los algoritmos desarrollados lograron el objetivo en una topología real, se concluyó a partir de simulaciones de casos particulares que un buen recorrido del grafo asociado a la red, mejora la calidad de la asignación. Es por esto que se concluye que la mejor configuración del sistema es utilizar el algoritmo MCL con método de desempate round-robin, y asignación del mejor canal según la potencia recibida.

7.2. Trabajos futuros

El proyecto queda abierto a ampliaciones del sistema. El código desarrollado de forma modular permite la simple adaptación del mismo a otras marcas de Access Points diferentes a MikroTik. Para esta ampliación lo que se necesita es desarrollar una clase de Python para los APs nuevos que contengan los mismos atributos y métodos que la clase `AccessPoint.py`. El nombre de la clase creada deberá tener el formato `AccessPoint.NOMBRE.py`, y luego especificar *NOMBRE* en el modelo del AP dentro del archivo de configuración de APs (`AP.csv`).

Además si se quisiese utilizar otro algoritmo de recorrido de grafo, basta con adjuntar el código del mismo dentro del archivo `algoritmos_canales.py` y referenciarlo en el archivo de configuración (`wico.conf`). Esto mismo se puede lograr tanto como para nuevos métodos de desempate, criterios de mejor canal, e incluso algoritmos de control de potencia.

Durante el desarrollo del proyecto se creo una función de que calculaba el FI_{min} comparando entre todas las asignaciones posibles. Se menciono que los tiempos requeridos, cuando la cantidad de APs era muy grande, eran inaceptables. Esta función se podría pensar como un nuevo algoritmo a utilizar cuando la cantidad de APs es baja.

Otra posibilidad de algoritmo de asignación que se manejó durante el proyecto, pero estuvo fuera del alcance del mismo, fue la de obtener datos estadísticos de la utilización de los APs por los usuarios, con esta información decidir qué APs son más importantes para la red y asignar el mejor canal recorriendo el grafo según importancia.

En la implementación del proyecto se asumió el valor mínimo de potencia recibida para la detección de un AP vecino en 100dBm. Se podría hacer más flexible esto agregando una variable configurable dentro del archivo `wico.conf`.

Otro desarrollo que puede ser útil para el sistema, puede ser el control del roaming de los usuarios. Es decir, agregarle la inteligencia a los APs para que la movilidad de los usuarios entre APs sea transparente.

También se podría realizar un sistema más centralizado, donde los algoritmos corran en algún equipo en un Data Center en vez de en una maquina en cada sitio.

Apéndice A

Manual de usuario

A.1. Procedimiento de instalación

El único requerimiento del sistema donde se va a instalar Wico, es que tenga Python 2 instalado. La instalación del sistema es muy simple, únicamente se debe descomprimir el archivo `wico.zip` en el servidor. Específicamente los pasos a seguir serían:

- Transferir el archivo `wico.zip` por ftp al servidor central del centro educativo.
- Ubicarse en la carpeta donde se copió el archivo.
- Descomprimir el archivo mediante el comando: `unzip wico.zip`.

A.2. Funcionalidades

Dentro de las funcionalidades se destaca el programa principal, `wico.py`, y dos herramientas auxiliares para realizar pruebas y simulaciones: `wico_test.py` y `wico_configAP.py`

A.2.1. Ejecución del programa principal

Para la ejecución del programa principal `wico.py` es necesario editar previamente el archivo `wico.conf` donde es posible elegir las opciones de las variantes de optimización. Las variables a definir son:

- `DIRECTORIO_AP_CSV`. Por defecto tiene valor `AP.csv`
- `ALGORITMO_ASIGNACION_CANALES` = `[mva, mcl, mcp, none]`. Por defecto es igual a `mcl`.
- `ALGORITMO_ASIGNACION_POTENCIA` = `[iterativo, friis, none]`. Por defecto es igual a `friis`.

Apéndice A. Manual de usuario

- CMC = [potencia, cantidad]
- ALFA = 10000 por defecto
- BETA = 100 por defecto.
- DESEMPATE = [normal, roundrobin, random]. Por defecto es igual a roundrobin

Es importante evitar dejar líneas en blanco en el archivo dado que puede dar errores en el parseo del mismo.

Por otro lado, se le deben indicar al programa en el archivo AP.csv detalles de los Access Points internos a la red a optimizar. El formato de este archivo es una fila por AP, donde cada fila debe ser:

`Modelo_AP, IP, usuario, password`

Siendo `Modelo_AP` el nombre de la versión de software del AP. Al momento las únicas opciones son MKv4 o MKv5.

El programa se ejecuta mediante el siguiente comando: `python wico.py`. Se recomienda ejecutar el programa periódicamente. Para ello modificar el `crontab` del usuario acordemente. Cabe destacar que cuando el programa consulta el estado de la red en cada APs (Recolección de vecinos), el vínculo entre los APs y sus clientes se interrumpe. Es por esto que se recomienda ejecutar el mismo en horarios de baja actividad de la red.

A.2.2. Módulo de Pruebas - `wico_test.py`

Con este programa se pueden hacer pruebas y simulaciones mediante las cuales se pueden ver resultados de ejecución de los algoritmos sin necesidad de interactuar con los Access Points.

- `canalesint_test.csv` - es una columna con los canales de todos los APs internos de la red.
- `potint_test.csv` - matriz nxn donde cada fila indica con qué potencia un AP ve a sus vecinos internos. Se debe marcar con la letra x cuando dos APs no son vecinos.
- `canalesext_test.csv` - es una matriz de 2 columnas y m filas. En la primer columna se indica la dirección MAC de los APs externos y en la segunda su canal.
- `potext_test.csv` = matriz de m columnas y n filas donde cada fila indica con que potencia un AP interno ve sus vecinos externos. Se debe marcar con la letra x cuando dos APs no son vecinos.

A.2. Funcionalidades

El programa se ejecuta mediante el siguiente comando: `python wico_test.py`. El módulo despliega en pantalla la solución final y su FI total.

A.2.3. Configuración desde archivo csv - `wico_configAP.py`

Este programa implementa la configuración de la red a partir del archivo `nuevaconfig.csv`. El formato del archivo es:

- IP, canal, potencia

El programa se ejecuta mediante el siguiente comando: `wico_configAP.py`

Esta página ha sido intencionalmente dejada en blanco.

Apéndice B

Gestión de proyecto

Al comienzo del proyecto se realizó la planificación de las tareas necesarias para el cumplimiento del mismo. En la tabla TablaB.1 se puede ver el detalle de la planificación original de horas y en la figura B.1 y el diagrama de Gantt correspondiente.

Como en todo proyecto el transcurso del mismo no se ajustó exactamente a la planificación inicial, ésta debió ser levemente modificada durante el mismo. Particularmente una de las mayores modificaciones que se tuvieron que hacer fue aumentar el tiempo requerido para la primer tarea ya que se consideró clave para la finalización del proyecto. Además se modificó el orden de las tareas dándole un mayor énfasis a la creación de los pseudocódigos ya que los mismos nos permitirían pasar a la versión final con mayor facilidad. Otras tareas que debieron ser reprogramadas fueron las visitas a las instituciones, aplazándolas hacia un período donde no se interfiriera con el normal funcionamiento de las instituciones. A pesar de todos estos cambios, el tiempo de finalización del proyecto no se vio afectado gracias a los buffers, y a que ciertas tareas tomaron menos tiempo de lo previsto. En la tabla(B.2) se muestra el detalle de la replanificación.

Apéndice B. Gestión de proyecto

Tarea	Prec.	Descripción	Recursos	Tiempo
1. Elegir el o los algoritmos de optimización que se adecuan a nuestro problema	-	-	-	150
1.1 Estudiar especificaciones estándar 802.11.	-	Lectura para estudiar los conceptos básicos	3	10
1.2 Estudiar optimización en redes WLAN, modo infrastructure BSS.	-	Enfocarnos en métodos actuales para la solución al problema.	3	10
1.3 Análisis de algoritmos de asignación de canales dentro de la banda 2.4 GHz.	1.1 1.2	Enfocarnos en métodos actuales para la solución al problema.	3	80
1.4 Análisis de algoritmos de control de potencia.	1.1 1.2	-	3	80
1.5 Estudiar viabilidad de los algoritmos en nuestro caso de aplicación: MikroTik.	1.3 1.4	Traducir lo que existe a los APs Mikrotik	3	10
1.6 Definir el o los algoritmos a usar por el sistema.	1.4	Elegir al menos dos algoritmos a comparar para ver cuál implementar	3	30
1.7 Documentación macro tarea 1.	1.6	-	2	10
2. Decidir en qué forma se va a dar la comunicación controlador - AP.	-	-	-	38
2.1 Analizar opción conectarse vía telnet / ssh a cada AP y ejecutar comandos de consulta y configuración.	-	Estudiar opciones para el diálogo entre los APs y la controladora	2	4
2.2 Analizar opción de realizar consulta y configuración vía SNMP.	-	Estudiar la opción de usar SNMP para la solución.	2	4
2.3 Familiarizarnos con los equipos MikroTik particularmente	-	Estudiar detalladamente las opciones que tienen los Mikrotik	3	18
2.4 Documentación macro tarea 2.	2.1 2.2 2.3	-	2	10

Tabla B.1: Detalle de las tareas

Tarea	Prec.	Descripción	Recursos	Tiempo
3. Implementar software de algoritmos.		Diseñar algoritmo de software para la solución general		180
3.1 Diseño	1.7		2	30
3.2 Pseudocódigo	3.1		2	30
3.3 Desarrollo del código	3.2		2	80
3.4 Testeo	3.3		3	10
4. Implementar software del driver MikroTik.		Aplicar un módulo de solución que sea específico para APs Mikrotik		160
4.1 Diseño	2.3		2	30
4.2 Pseudocódigo	4.1		2	30
4.3 Desarrollo del código	4.2		2	80
4.4 Testeo	4.3		3	10
5. Integración de las tareas 3 y 4.		Unir la solución específica para MikroTik al algoritmo central		112
5.1 Desarrollo del código	3.4 4.4		2	30
5.2 Pruebas y simulaciones	5.1		3	40
5.3 Documentación tarea 5	5.2		2	10
6. Validar el funcionamiento en un caso real		Se buscará comparar la solución diseñada con una manual		32
6.1 Elegir la red Ceibal en centro educativo donde se implementará el controlador.		Elegir el centro según disponibilidad, dificultad de la solución, conectividad con el servidor	1	60
6.2 Solicitar ajuste manual por un técnico Ceibal	6.1	Se pedirá un ajuste manual a cargo de Ceibal	1	60
6.3 Instalación y puesta en funcionamiento del software en servidor central.	5.2	Se pondrá en funcionamiento el controlador en un servidor de Ceibal	3	10
6.4 Documentación macro tarea 6.	6.3		2	10

Detalle de las tareas

Apéndice B. Gestión de proyecto

Tarea	Prec.	Descripción	Recursos	Tiempo
7. Verificar que se optimiza el desempeño de la red.		Luego de las pruebas se pasará a estudiar los resultados		270
7.1 Medir resultados ajuste manual.	6.2	Se estudiará los detalles de la solución manual	2	10
7.2 Medir resultados luego de optimización con controlador.	7.1	Se verá cuál fue el resultado luego de los cambios realizados por la controladora	2	10
7.3 Comparar resultados.	7.2		3	20
7.4 Documentación macro tarea 6.	7.3		2	10
8. Documentación final	7.4		3	80

Detalle de las tareas

Nombre	Fecha inicio	Fecha fin	Nuevo Inicio	Nuevo fin
T1.Elegir algoritmo optimización	20/2/15	11/6/15	20/2/15	23/7/15
T2. Decidir en qué forma se va a dar la comunicación controlador - AP	27/4/15	21/5/15	18/5/15	12/6/15
T3. Implementar software algoritmos	12/6/15	15/10/15	16/4/15	19/8/15
T4.Implementar software de driver AP MikroTik	22/5/15	10/9/15	15/6/15	2/10/15
T5. Integración de tareas 3 y 4	16/10/15	1/1/16	18/9/15	16/11/15
T6. Validar funcionamiento caso real	1/6/15	22/01/16	12/10/15	1/1/16
T7. Verificar optimización	24/8/15	4/3/16	8/12/15	12/2/16
T8. Documentación final	7/3/16	29/4/16	23/2/16	18/4/16

Tabla B.2: Detalle de los cambios en las fechas

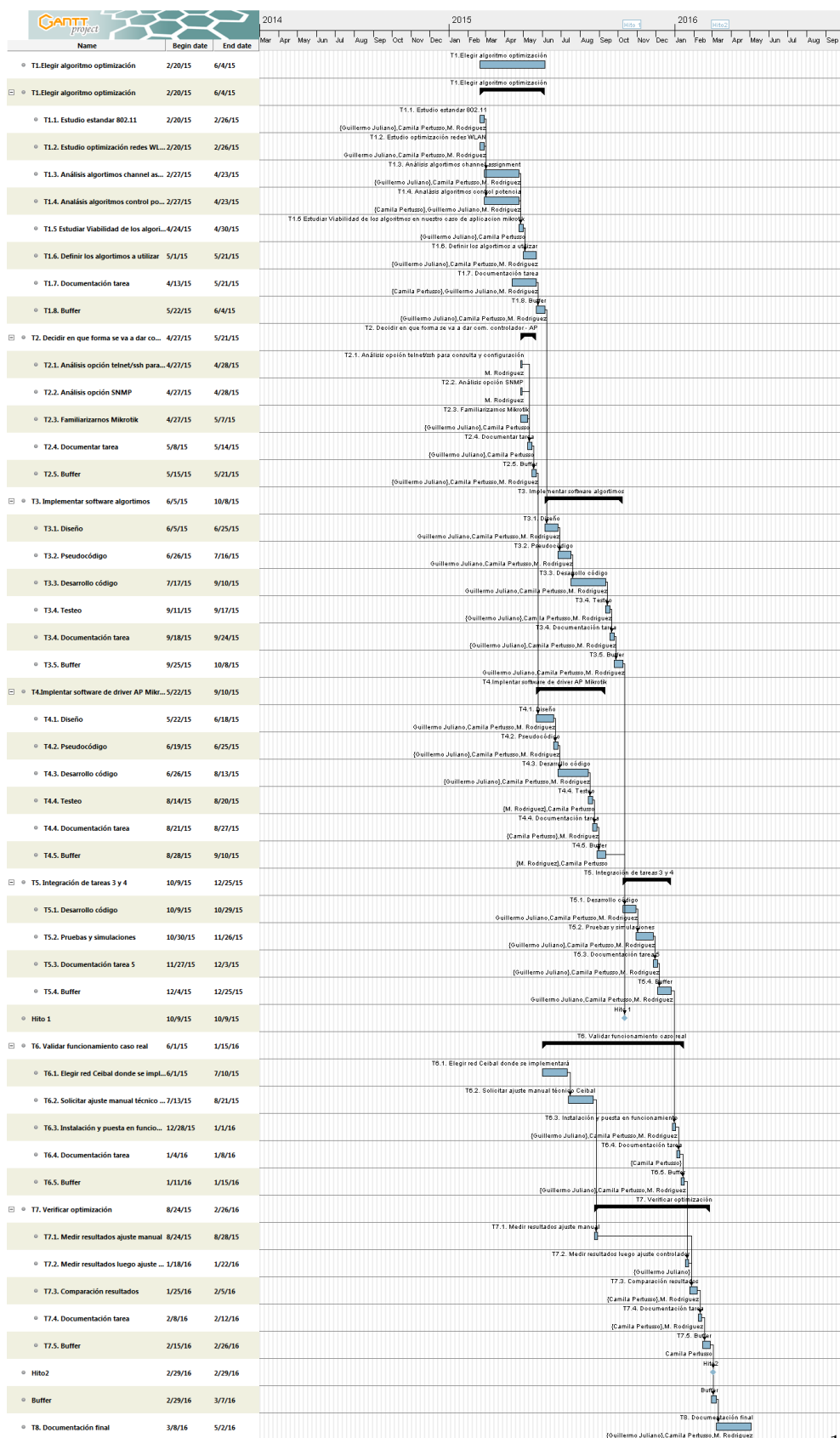


Figura B.1: Cronograma Inicial

Esta página ha sido intencionalmente dejada en blanco.

Referencias

- [1] Plan Ceibal. <http://www.ceibal.edu.uy/>.
- [2] Mikrotik. <http://www.mikrotik.com/aboutus>.
- [3] Sitio web de Python. <https://www.python.org/about/>.
- [4] Sitio web de MATLAB. <http://www.mathworks.com/products/matlab/>.
- [5] 802.11-2012 - IEEE Standard for Information technology– Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. 2012.
- [6] Wikipedia: Espectro electromagnetico . https://es.wikipedia.org/wiki/Espectro_electromagn%C3%A9tico#/media/File:Electromagnetic_spectrum-es.svg.
- [7] Enciclopedia Britanica - Espectro de radiofrecuencia . <http://www.britannica.com/science/radio-frequency-spectrum>.
- [8] David D.Coleman and David A. Westcott. *CWNA: Certified Wireless Network Administrator Official Study Guide*. Sybex inc, 2012.
- [9] Understundig WIFI carrier Sense. <http://www.revolutionwifi.net/revolutionwifi/2011/03/understanding-wi-fi-carrier-sense.html>.
- [10] G. J. Foschini and Z. Miljanic. A Simple Distributed Autonomous Power Control Algorithm and its Convergence. *IEEE Transactions on Vehicular Technology*, 42:641–646, 1993.
- [11] Micheal Driberg and Fu-Chun Zheng. Centralized Channel assignment for IEEE 802.11 WLANs: Utilization MinMax-Sum. *Wireless Personal Multimedia Communications (WPMC), 2012 15th International Symposium on*, pages 633–637, 2012.
- [12] Leonhard Euler. Solutio Problematis ad Geometriam Situs Pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, 7:128–140, 1741.
- [13] Surachai Chiochan, Ekram Hossain, and Jeffrey Diamond. Channel Assignment Schemes for Infrastructure-Based 802.11 WLANs: A Survey. *IEEE Community Surveys Tuts.*, 12:124–136, 2010.

Referencias

- [14] Aditya Akella, Glenn Judd, Srinivasan Seshan, and Peter Steenkiste. Self-Management in Chaotic Wireless Deployments. *Electrical and Computer Engineering, 2008. CCECE 2008. Canadian Conference on*, pages 1169 – 1174, May 2008.
- [15] J. Riihijarvi, M. Petrova, and P. Mahonen. *Frequency Allocation for WLANs Using Graph Colouring Techniques*. PhD thesis, Aachen University Department of Wireless Networks Kackertstrasse 9 52072 Aachen Germany, 2005.
- [16] D. Brelaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22:251–256, 1979.
- [17] Robert Akl and Anurag Arepally. *Dynamic Channel Assignment in IEEE 802.11 Networks*. PhD thesis, Proc. IEEE International Conference on Portable Information Devices, 2007.
- [18] Mohamad Haidar, Rabindra Ghimire, Hussain Al-Rizzo, Robert Akl, and Yupo Chan. CHANNEL ASSIGNMENT IN AN IEEE 802.11 WLAN BASED ON SIGNAL - TOINTERFERENCE RATIO. *IEEE Community Surveys Tuts.*, 12:124–136, 2010.
- [19] Ecuación de propagación de Friis. https://en.wikipedia.org/wiki/Friis_transmission_equation.
- [20] CentOS. <https://www.centos.org>.
- [21] RouterOS. http://www.mikrotik.com/download/pdf/what_is_routeros.pdf.
- [22] RouterBoard 433. <http://routerboard.com/RB433>.
- [23] Wiki mikrotik - Manual API. <http://wiki.mikrotik.com/>.

Índice de tablas

2.1. Porcentaje de uso por canal	12
2.2. Potencias recibidas por los APs vecinos	16
3.1. Ejemplo para el algoritmo MVA	24
3.2. Ejemplo para el algoritmo MCL	30
3.3. Ejemplo para el algoritmo MCP	37
3.4. Tiempos de ejecución del algoritmo FI_{min} , medidos con un laptop con procesador i7 y 12Gb de RAM	43
4.1. Ejemplo Control de Potencia	46
6.1. FI obtenidos en simulaciones para los datos relevados del liceo 16 .	87
6.2. FI obtenidos en simulaciones para los datos relevados del liceo 36 .	88
6.3. FI del liceo para todos los algoritmos. -Valores en dBm-	91
B.1. Detalle de las tareas	102
B.2. Detalle de los cambios en las fechas	104

Esta página ha sido intencionalmente dejada en blanco.

Índice de figuras

2.1. En rojo se representan los vínculos entre los APs y sus clientes . . .	6
2.2. Uso del Espectro de frecuencias - Imagen obtenida de [6]	7
2.3. Ubicación espectral de las tecnologías de transmisión - Imagen obtenida de [7]	7
2.4. Canales en 2.4Ghz	8
2.5. Ejemplo de un grafo	11
2.6. Ejemplo: Espectro de potencias recibidas	16
3.1. Grafo inicial - En negro y rojo se identifican los APs internos y externos respectivamente	20
5.1. Mikrotik RouterBoard RB433 [22]	52
5.2. interface wireless print advanced	53
5.3. interface wireless set wlan1 frequency=2412	53
5.4. interface wireless set wlan1 tx-power=18	54
5.5. interface wireless scan wlan1 duration=12	54
5.6. Diagrama de flujo del software implementado	56
5.7. Diagrama de entrada y salida de los archivos de asignación de canales	60
5.8. Diagrama de entrada y salida de los algoritmos de potencia	60
6.1. Mapa de cobertura creado por la función graphAP	64
6.2. Espectro de frecuencias simulado en MATLAB	64
6.3. Resultado de la aplicación de los algoritmos de optimización	65
6.4. Factor de interferencia para 100 escenarios aleatorios.	65
6.5. Archivos necesarios para la ejecución de maindesdearchivo.py . . .	66
6.6. Simulación 25 APs: Mapa de cobertura inicial	67
6.7. Simulación 25 APs: Mapa de cobertura post MVA	68
6.8. Simulación 25 APs: Mapa de cobertura post MVA y potencia . . .	68
6.9. Simulación 25 APs: Mapa de cobertura post MCL	69
6.10. Simulación 25 APs: Mapa de cobertura post MCL y potencia . . .	69
6.11. Simulación Estadística	70
6.12. Caso Particular 1	71
6.13. Caso Particular 1: Grafo Asociado	71
6.14. Caso Particular 1 : Asignación con 2 canales	72
6.15. Caso Particular 1: Asignación con 3 canales	73
6.16. Caso Particular 1: Espectro 2 canales utilizados	73

Índice de figuras

6.17. Caso Particular 1: Espectro 3 canales utilizados	74
6.18. Caso Particular 2	75
6.19. Caso Particular 2: Grafo Asociado	75
6.20. Caso Particular 2: Asignación óptima	76
6.21. Caso Particular 2: Utilización del espectro	76
6.22. Caso Particular 3	77
6.23. Caso Particular 3: Grafo Asociado	77
6.24. Caso Particular 3: Asignación óptima	78
6.25. Caso Particular 3: Utilización del espectro	78
6.26. Caso Particular 4	79
6.27. Caso Particular 4: Grafo Asociado	79
6.28. Caso Particular 4: Asignación óptima	80
6.29. Caso Particular 4: Utilización del espectro	80
6.30. Caso Particular 4: Asignación óptima con control de potencia . . .	81
6.31. Caso Particular 4: Peor caso de asignación con control de potencia	81
6.32. Caso Particular 5	82
6.33. Caso Particular 5: Grafo Asociado	82
6.34. Caso Particular 5: Mapa de cobertura	83
6.35. Caso Particular 5: Utilización del espectro	83
6.36. Caso Particular 6	84
6.37. Caso Particular 6: Grafo Asociado	84
6.38. Caso Particular 6: Mapa de cobertura	85
6.39. Caso Particular 6: Utilización del espectro	85
6.40. Caso Particular 6: Control de potencia contraproducente	86
6.41. Grafo asociado a la red relevada del liceo 16	88
6.42. Grafo asociado a la red relevada del liceo 36	89
6.43. Utilización del espectro resultante de la aplicación de algoritmos en el liceo 36	89
6.44. Resultados obtenidos en la prueba en el liceo 36	92
B.1. Cronograma Inicial	105

contenido del cd

Esta es la última página.
Compilado el miércoles 27 julio, 2016.
<http://iie.fing.edu.uy/>