



Escola Politècnica Superior  
de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# TRABAJO DE FIN DE CARRERA

**TÍTULO:** Asignación de frecuencias en redes GSM

**AUTOR:** Carlos García Yuste  
Iñaki Villar Vinuesa

**DIRECTOR:** Javier Ozón Górriz

**FECHA:** 29 de junio de 2004



**Título:** Asignación de frecuencias en redes GSM

**Autor:** Carlos García Yuste  
Iñaki Villar Vinuesa

**Titulación:** Ingeniería Técnica de Telecomunicaciones

**Especialidad:** Sistemas de Telecomunicaciones

**Plan:** 2000

**Director:** Javier Ozón Górriz

**Departamento:** Ingeniería Telemática

**Vist i plau,**

**Director del TFC**

**Registre:**



**Título:** Asignación de frecuencias en redes GSM

**Autor:** Carlos García Yuste  
Iñaki Villar Vinuesa

**Director:** Javier Ozón Górriz

**Fecha:** 29 de junio de 2004

## Resumen

En el diseño de un sistema de comunicaciones celular, la zona de cobertura se divide en celdas, cada una de las cuales está cubierta por una estación base. Cada estación base puede incorporar más de un transmisor y cada transmisor trabaja a una cierta frecuencia en cada instante de tiempo. Además, una misma frecuencia se puede utilizar en diferentes celdas, permitiendo reducir el número total de frecuencias utilizadas (parámetro importante si se considera el elevado precio de la licencia de ocupación del espectro así como sus cuotas anuales).

No obstante, la reutilización de frecuencias en diferentes celdas está limitada por el nivel máximo de interferencias tolerable (7 u 8 dB en el caso de redes GSM). Debemos considerar además que para los servicios de datos esta limitación en la reutilización de frecuencias es un factor crítico si se desean conseguir velocidades de transmisión elevadas. De esta forma, el problema de la asignación de frecuencias consiste en encontrar una planificación que satisfaga unos ciertos requerimientos utilizando el mínimo número de frecuencias o utilizando eficientemente las frecuencias disponibles, teniendo en cuenta las restricciones o limitaciones de la red y tratando de minimizar el nivel de interferencia resultante. Todo esto con el propósito de optimizar la calidad de las comunicaciones de voz y aumentar el ancho de banda disponible para los enlaces de datos.

En este proyecto final de carrera se pretende definir un algoritmo de asignación de frecuencias (basado en algoritmos anteriores como *Hormigas* y *Simulated Annealing*) para redes que utilizan más de una frecuencia en cada transmisor (frequency hopping) con la finalidad de mejorar el comportamiento global del sistema.



**Título:** Asignación de frecuencias en redes GSM

**Autor:** Carlos García Yuste  
Iñaki Villar Vinuesa

**Director:** Javier Ozón Górriz

**Fecha:** 29 de junio de 2004

### **Abstract**

On the design of a cellular communication system, the coverage zone is divided into cells, each covered by a base station. Each station incorporates more than one transmitter and each transmitter works on a single frequency in every time slot. Moreover, the same frequency can be used in different cells, allowing to decrease the total number of used frequencies.

Nevertheless, the reuse of frequencies on different cells is limited by the level of interferences (7 or 8 dB on GSM). The problem of frequency assignment consists on finding a frequency design that satisfies some conditions using the minimal number of frequencies or using efficiently all available frequencies, considering restrictions and limitations of the network.

On this project we pretend to define an algorithm for frequency assignment using more than one frequency in each transmitter (frequency hopping) in order to improve the operation of the global system.



# INDICE

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO 1. ASIGNACIÓN DE FRECUENCIAS EN TELEFONÍA MOVIL.....</b>	<b>3</b>
<b>1.1 Introducción al sistema GSM.....</b>	<b>3</b>
1.1.1 Sistema radio.....	4
1.1.2 Reducción de nivel de interferencias.....	6
1.1.3 Frequency Hopping .....	7
1.1.4 Limitación del número de frecuencias.....	8
<b>1.2 Representación de una red GSM.....</b>	<b>9</b>
1.2.1 Especificaciones para la asignación de frecuencias.....	11
<b>CAPÍTULO 2. COLOREADO DE GRAFOS .....</b>	<b>15</b>
<b>2.1 Teoría de grafos. Definiciones.....</b>	<b>15</b>
2.1.1 Coloreado de grafos .....	18
2.1.2 Polinomios cromáticos.....	19
<b>CAPÍTULO 3. OPTIMIZACIÓN COMBINATORIA.....</b>	<b>23</b>
<b>3.1 Complejidad del coloreado de grafos.....</b>	<b>23</b>
<b>3.2 Problemas NP-completos .....</b>	<b>25</b>
<b>3.3 Simulated Annealing .....</b>	<b>25</b>
<b>3.4 Algoritmo Hormigas .....</b>	<b>28</b>
<b>3.5 Coloreado de grafos y generalización de Hormigas.....</b>	<b>29</b>
<b>CAPÍTULO 4. MULTICOLOREADO DE GRAFOS.....</b>	<b>33</b>
<b>4.1 Multicoloreado de grafos .....</b>	<b>33</b>
4.1.1 Cotas generales.....	34
4.1.2 Número multicromático de algunas familias de grafos.....	35
4.1.3 Grafos ajustables y grafos perfectos.....	39
<b>4.2 Reutilización de colores en el multicoloreado de grafos .....</b>	<b>40</b>
4.2.1 Construcción del subgrafo inducido .....	43
4.2.2 Algoritmo de reciclado .....	47
4.2.3 Algoritmo RCY paar el 3-coloreado.....	48

<b>CAPÍTULO 5. ASIGNACIÓN DE FRECUENCIAS EN REDES GSM .....</b>	<b>49</b>
<b>5.1 Asignación de frecuencias mediante coloreado de grafos .....</b>	<b>49</b>
<b>5.2 Implementación .....</b>	<b>50</b>
<b>5.3 Resultados y comparaciones .....</b>	<b>53</b>
5.3.1 Redes GSM reales .....	54
5.3.2 Redes generadas aleatoriamente .....	55
 <b>CAPÍTULO 6. CONCLUSIONES .....</b>	 <b>59</b>
<b>6.1 Líneas futuras.....</b>	<b>60</b>
 <b>BIBLIOGRAFÍA .....</b>	 <b>61</b>

## INTRODUCCIÓN

La telefonía móvil celular se ha convertido en uno de los campos más importantes de las telecomunicaciones, debido principalmente a la aparición del sistema GSM y al final del monopolio en este sector. De esta manera, por ejemplo, en número de usuarios de telefonía móvil del estado español ha pasado de 400 mil usuarios a finales de 1994 a 31 millones en el año 2003.

En el diseño de un sistema de comunicaciones celular, la zona de cobertura se divide en celdas, cada una de las cuales está cubierta por una estación base. Cada estación base puede incorporar más de un transmisor y cada transmisor trabaja a una cierta frecuencia. Además, una misma frecuencia se puede utilizar en diferentes celdas, permitiendo así reducir el número total de frecuencias utilizadas (parámetro importante si se considera el elevado precio de la licencia de ocupación del espectro). Sin embargo, la reutilización de frecuencias está limitada por el nivel máximo de interferencias tolerable. De esta manera, el problema de la asignación de frecuencias consiste en encontrar una asignación que satisfaga unos ciertos requerimientos de tráfico utilizando el mínimo número de frecuencias y considerando las restricciones por interferencias.

En este proyecto de final de carrera se propone un método de asignación de frecuencias en redes GSM capaz de realizar diseños de frecuencias múltiples (es decir asignando más de una frecuencia en cada transmisor) a partir de diseños simples.

La memoria está dividida en seis capítulos. En el primer capítulo se describen los aspectos básicos del sistema GSM, planteando el problema de la asignación de frecuencias.

En el segundo y el tercer capítulo se describe con más detalle el problema del coloreado de grafos –que es la representación en términos combinatorios del problema de asignación de frecuencias en redes GSM– y se presentan algunos métodos de optimización combinatoria para problemas NP-completos. También se describen los algoritmos Hormigas y Simulated Annealing y su funcionamiento básico.

En el capítulo cuarto se generaliza el problema de la asignación de frecuencias y se describe en términos de un coloreado múltiple de grafos en el que cada vértice puede recibir más de un color. Se incluye así una breve introducción del problema, definiéndose a continuación un conjunto de matrices ( $C$ ,  $C'$ ,  $D$  y  $D'$ ) asociadas a un grafo que permite reasignar colores a los distintos vértices de un grafo. La reasignación de colores se realiza a través del algoritmo de reciclado (RCY) que se describe al final de este capítulo.

En el quinto capítulo se resuelve el problema de la asignación de frecuencias en términos de coloreado de grafos y se describe el funcionamiento de los programas empleados en su resolución. Para finalizar realizamos pruebas con redes GSM reales y con redes aleatorias. Mediante unas tablas, analizamos los resultados obtenidos y comparamos el funcionamiento de los algoritmos en cada caso.

Finalmente, en el capítulo sexto se analizan los resultados obtenidos y se exponen una serie de conclusiones antes de señalar las posibles líneas de continuación del proyecto.

# Capítulo 1. Asignación de frecuencias en telefonía móvil

## Introducción

La licencia de explotación del espectro radioeléctrico es uno de los principales costes que debe soportar un operador de telefonía móvil. En estos casos, la reducción del número de frecuencias en el diseño de una red puede suponer un ahorro considerable. Igualmente, una vez contratada la licencia y determinado el número máximo de frecuencias disponibles, la capacidad de tráfico de la red puede ser incrementada mediante un mejor aprovechamiento del espectro, en el sentido de reutilizar frecuencias en distintas celdas de la red y poder transmitir simultáneamente desde distintos puntos con una misma portadora.

Una red de telefonía móvil celular se compone de una serie de celdas que pueden interferirse recíprocamente según distintos factores de proximidad, potencia de emisión o densidad de tráfico. La relación de interferencia entre las celdas permite modelar la red a través de un grafo, correspondiéndose los vértices con celdas (o las estaciones base que dominan cada celda) y designando las aristas la relación de interferencia entre celdas o vértices. De este modo el problema de asignación de frecuencias puede considerarse desde el punto de vista del coloreado de grafos y resolverse a partir de los métodos algorítmicos descritos en posteriores capítulos.

En el presente capítulo se describe el funcionamiento de una red celular GSM, el problema derivado de la asignación frecuencial y la definición de diferentes especificaciones (matrices y vectores) para la resolución de dicho problema. Finalmente se explica su representación en términos de la teoría de grafos.

## 1.1 Introducción al sistema GSM

El sistema GSM de telefonía móvil celular debe sus siglas al Group Spécial Mobile creado en 1980 con el objeto de definir y especificar un sistema de cobertura paneuropea (y naturaleza digital), capaz de dar servicio a un gran número de abonados y que pudiera integrarse fácilmente en las nuevas redes de telecomunicación RDSI. Este grupo inició sus trabajos en el año 1985 concluyendo la especificación del sistema en 1991.

El sistema GSM es un sistema:

1. *De telefonía móvil*: puesto que permite la comunicación entre usuarios móviles.
2. *Celular*: la zona de cobertura se divide en zonas llamadas celdas, a las cuales se asigna un cierto número de radiocanales o frecuencias. Estas

celdas cuentan con una infraestructura de estaciones base (a cada celda le corresponde una estación base) que transmiten con la potencia necesaria para alcanzar toda el área de cobertura. La división en celdas permite el reuso de frecuencias y el consiguiente aumento de abonados a los que la red puede dar servicio simultáneamente sin necesidad de aumentar el conjunto de frecuencias contratadas. Cada celda o estación base puede asimismo incluir uno o varios transmisores según el volumen de tráfico cursado. Así, una celda con tres transmisores podrá emitir con tres frecuencias simultáneamente y soportar tres veces más tráfico que una celda con un solo transmisor.

3. *Digital*: dado que supone la solución más económica, sencilla y eficaz para asumir la demanda del mercado e incorporar nuevas posibilidades de servicio, frente a la solución analógica empleada por otros sistemas de telefonía móvil.

En el año 1993 había 36 redes GSM en 22 países. Aunque el sistema GSM es un sistema típicamente europeo, existen redes GSM operativas en alrededor de 80 países de todo el mundo. Con motivo de este desarrollo, se ha asignado un nuevo término al acrónimo GSM: Global System for Mobile Communications. El sistema GSM ofrece las siguientes características:

1. Utilización eficiente del medio de transmisión.
2. Buena calidad de la señal de voz (incluso en condiciones de recepción débil de señal y alta interferencia) a pesar de que el uso del codificador de voz puede provocar ciertas variaciones en el timbre de voz y un retardo hasta cierto punto apreciable.
3. Posibilidad de comunicación entre distintos países sin la necesidad de cambiar de terminal, con el único requisito de acuerdos pertinentes entre los diferentes operadores.
4. Conversaciones seguras por la encriptación de la información en el canal radio.

### **1.1.1 Sistema radio**

El subsistema radio, que constituye el enlace físico entre las estaciones móviles y la red fija, está formado por una serie de canales. El sistema GSM emplea una modulación multiplexada en el tiempo y en frecuencia. De esta forma, una celda puede transmitir diversas frecuencias (llamadas portadoras) simultáneamente. Cada frecuencia portadora puede soportar además distintos canales (en general dedicados al tráfico de llamadas) gracias a la multiplexación en tiempo, dividiéndose cada trama en 8 slots (o segmentos de tramas), correspondientes con otros tantos canales lógicos (generalmente 1 de control y 7 de tráfico por cada portadora, o bien 8 de tráfico si otra de las

frecuencias portadoras de la misma celda ya contiene un canal de control). El estándar GSM distingue los canales de usuario, que cursan y gestionan las llamadas, de los canales de control propiamente dichos, según la jerarquía siguiente:

- Canales de usuario
  - Canal de tráfico
  - Canal de control (asociados durante la comunicación)
    - Lento: Control de potencia, medidas, avance temporal
    - Rápido: Información de traspaso
  
- Canales de control
  - Difusión (Base → Móviles)
  - Acceso (Móviles → Base)
  - Dedicados

Como se ha dicho, los canales de usuario ocupan los slots no destinados a los canales de control (apareciendo éstos como máximo 1 vez por trama), y por tanto quedando 7 u 8 slots de una trama destinados al tráfico y al control de tráfico según el siguiente esquema:

1. Los canales de tráfico soportan la transmisión de la información de voz y datos en forma de ráfagas.
  
2. Los canales de control asociados aparecen una vez se ha establecido la comunicación y son propios del usuario. Se distinguen dos tipos de canales de control asociado:
  - Lentos: transporta información de control de potencia (en bucle cerrado, es decir, la estación base informa al móvil si ha de aumentar o disminuir la potencia transmitida), medidas de potencia (del móvil a la estación base para que ésta decida el control de potencia) e información de avance temporal.
  
  - Rápidos: transmiten la información de traspaso cuando el usuario se traslada de una celda a otra, que debe transferirse al móvil antes de que se interrumpa la comunicación. Por este motivo la transmisión se lleva a cabo sobre un canal de tráfico, pudiendo producirse una ligera degradación de la transmisión.

Por otra parte, los canales de control son comunes a todos los usuarios de una celda siendo utilizados en los pasos previos al establecimiento de la llamada. Estos canales, que han de estar presentes en todas las celdas de la red, ocupan normalmente un solo slot de una de las portadoras de la celda, respondiendo a la siguiente jerarquía:

1. Los canales de difusión se transmiten en el enlace descendente (de la estación base al terminal móvil) y proporcionan distintos tipos de información: configuración del sistema, frecuencia portadora de la estación base (permitiendo así sintonizar el móvil) y frecuencias con canal de control, sincronización con la trama de la estación base, etcétera.
2. Los canales de acceso son canales ascendentes o de subida (es decir, van del móvil a la estación base) que, al ser comunes a todos los usuarios, requieren la presencia de un mecanismo de acceso al medio. Entre otras funciones controlan las peticiones de llamadas de los usuarios.
3. Finalmente, se encuentran los canales de control dedicados, destinados a transmitir información de control entre la red y el móvil con funciones específicas, como el intercambio de datos de usuario entre la estación base y el móvil antes de realizarse la comunicación.

### 1.1.2 Reducción del nivel de interferencias

Como todo sistema de telefonía celular, GSM permite el reuso de frecuencias en diferentes estaciones base con el objeto de aumentar la capacidad de tráfico de la red. Esta reutilización comporta, sin embargo, la posible aparición de interferencias entre celdas. De este modo, junto al diseño de una planificación frecuencial adecuada para determinar cuál debe ser la distancia de reuso entre celdas y la utilización de una modulación robusta, el estándar GSM incorpora tres mecanismos para compensar los efectos de las interferencias, siendo decisión de la operadora incluirlos o no en su servicio:

- Control de potencia: permite modificar dentro de un cierto rango la potencia transmitida en ambas direcciones, de móvil a base y viceversa. En GSM se aplica un control de potencia, independiente en ambas direcciones, en función de los niveles de recepción medidos por la estación base y el terminal móvil.
- Transmisión discontinua: detecta la existencia de silencios en la comunicación e inhibe la transmisión radio durante este tiempo (aproximadamente el 50% del tiempo de una llamada nos encontramos a la escucha) para no interferir en celdas contiguas.
- Frequency hopping o conmutación de frecuencia (FH): varía la frecuencia portadora de transmisión, de una trama a otra y a lo largo de una misma comunicación.

El control de potencia y la conmutación de frecuencia además de reducir las interferencias, solucionan algunos problemas relacionados con la utilización del canal radio como medio de transmisión. Así, el primero de ellos permite compensar la atenuación debida a los obstáculos o atenuación lenta, mientras

que el segundo contribuye a atenuar los efectos de la propagación multicamino (que induce desvanecimientos rápidos o de "Rayleigh").

### 1.1.3 Frequency Hopping

La técnica del Frequency Hopping (también llamada conmutación de frecuencia) consiste en la transmisión de las sucesivas tramas a frecuencias diferentes según una secuencia definida para cada una de las celdas. En este caso, un canal no se corresponde con un time-slot y una única frecuencia portadora, sino con un time-slot y una secuencia de salto. Tras la transmisión de una ráfaga el transmisor conmuta a una frecuencia distinta dentro del grupo de frecuencias de hopping correspondientes, asignadas según criterios de minimización del número de interferencias. Como es lógico, la capacidad de tráfico de una celda viene determinada por el número de frecuencias que la celda es capaz de manejar en un instante de tiempo determinado, y por tanto no depende del número de frecuencias de hopping sino del número de transmisores. De este modo, el número de transmisores de una celda determinará el máximo número de frecuencias que la estación base podrá emitir simultáneamente, con independencia del número de frecuencias con las que pueda trabajar.

De este modo la calidad del enlace puede cambiar de una ráfaga a otra, esto es, una ráfaga con tasa de error elevada puede ir seguida con otra baja, puesto que:

- Los desvanecimientos rápidos son distintos a frecuencias diferentes, debido a que la longitud de los caminos medida en longitudes de onda depende de la frecuencia.
- El nivel de interferencia es distinto para frecuencias diferentes ya que la señal interferente puede ocupar una zona determinada del espectro.

El hecho de que la tasa de error pueda variar en dos tramas sucesivas, junto con la utilización de un código corrector de errores y un sistema de entrelazado de la información, hace posible la reconstrucción de los datos originales incluso en situaciones de interferencias. Sin hopping, en general, esto no ocurre dado que todas las ráfagas o son buenas o son malas.

De otro lado, el hopping no se aplica en ningún caso sobre la frecuencia portadora del canal de control, que se utiliza como referencia de la potencia de las celdas vecinas en los traspasos de celda (cuando el usuario se traslada de un punto a otro) para decidir el instante en que debe realizarse el traspaso.

En GSM el frequency hopping o salto de frecuencia puede implementarse mediante dos técnicas distintas conocidas con el nombre de:

1. Hopping en banda base: a cada transmisor se le asigna una única frecuencia y cada ráfaga de una llamada es encaminada, por un

procesador de trama, hacia el transmisor correspondiente de acuerdo con la secuencia de hopping establecida.

2. Hopping sintetizado: en contraposición al anterior, todas las ráfagas de una comunicación son enviadas por el mismo transmisor, que tiene la capacidad de sintetizar las distintas frecuencias de transmisión, según la secuencia de hopping asignada.

Para entrar en modo de operación frequency hopping la red informa al móvil (a través del canal de control de cada celda) del conjunto de frecuencias asignadas y del algoritmo de salto.

#### 1.1.4 Limitación del número de frecuencias

A pesar de que un operador suele contratar un número fijo de frecuencias es posible encontrar diseños frecuenciales GSM en rangos de distinta amplitud, en el sentido no sólo de que el número de frecuencias hopping empleadas en cada celda sea distinto de un diseño a otro, sino de que el propio número de frecuencias disponibles por la red pueda variar significativamente. En primer lugar, la asignación frecuencial se realiza normalmente por separado, repartiendo de un lado las frecuencias para los canales de tráfico (que incorporan en general técnicas de frequency hopping) y de otro las frecuencias asignadas a los canales de control (que, como se ha dicho, no realizan la conmutación de frecuencia y requieren normalmente una única frecuencia portadora por celda). De otro lado la planificación frecuencial puede llevarse a cabo de forma segmentada dividiendo el área de cobertura (una región, un área metropolitana o una ciudad de dimensiones variables) en distintas coronas e incluyendo pequeñas celdas de apoyo llamadas micro y picoceldas (figura 1.1)

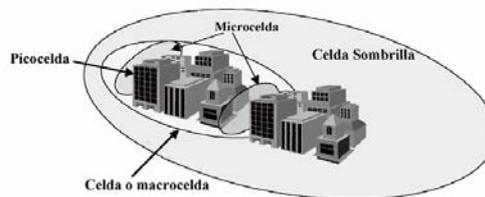


Figura 1.1: Estructura jerárquica de una red de acceso GSM.

Es posible encontrar una planificación frecuencial en la cual la zona correspondiente al centro de una ciudad requiera un número de frecuencias menor que la periferia, debido a la existencia de un número determinado de microceldas, es decir pequeñas celdas (que disponen de un grupo específico de frecuencias reservadas) cuyo área de cobertura es mucho menor que el área de una celda (figura 1.1). Las microceldas están provistas de antenas directivas dispuestas a escasa altura del suelo y suelen cubrir calles o avenidas, requiriendo un número menor de transmisores que las celdas “normales”, aunque es usual que trabajen en saturación. El tráfico que no

pueden cursar las microceldas es transferido a las celdas de nivel superior. Asimismo, el reuso de frecuencias entre microceldas no reviste grandes complicaciones gracias a la disposición de las antenas, cuya escasa altura permite a los mismos edificios atenuar las posibles interferencias. De esta forma, la zona de la periferia puede requerir un mayor número de frecuencias hopping debido a la escasa presencia de microceldas fuera del centro comercial de la ciudad (aun así no es descartable la ubicación de algunas microceldas en la periferia como sucede, en el caso de Barcelona, en algunos tramos de la Avenida Diagonal).

Finalmente cabe la posibilidad de que una red GSM incorpore picoceldas, esto es, pequeñas estaciones capaces de ofrecer cobertura en interiores (como por ejemplo estaciones de metro o centros comerciales) y cuyas frecuencias no producen interferencias con el exterior (excepto en las zonas limítrofes, como pueden ser las escaleras de bajada al metro), así como de incorporar celdas sombrilla, que pueden cubrir un determinado número de celdas de una jerarquía inferior, y cursar el tráfico desbordado. Esta jerarquización (más o menos pronunciada) de la red de acceso a un sistema GSM (así como la presencia de las frecuencias de control) se traduce finalmente, como se ha dicho, en diseños que incorporan en cada caso distintos números de frecuencias.

## **1.2 Representación de una red GSM**

Un sistema de telefonía celular, como se ha visto, divide la zona de cobertura en distintas áreas denominadas celdas. Cada celda está dominada por una estación base provista de uno o varios transmisores (o transmisores-receptores, también llamados transceptores), encargados de enlazar los terminales móviles con la red telefónica. Esta división de la zona de cobertura, que permite de un lado reducir la potencia de emisión y de otro reutilizar frecuencias en celdas diferentes (con el objeto de aumentar el número de canales disponibles) provoca sin embargo la aparición de interferencias entre celdas.

De este modo, es preciso antes de definir la asignación frecuencial realizar un estudio riguroso de las interferencias electromagnéticas inducidas por las diferentes celdas de la red. Dada la multiplicidad de factores que deben tenerse en cuenta este análisis no puede ser únicamente teórico, debiendo ir acompañado de un trabajo de campo que permita ajustar los resultados previstos por la teoría.

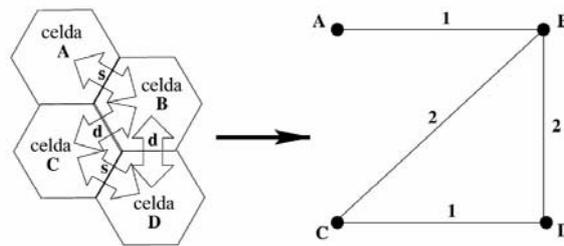


Figura 1.2: Modelado de una red GSM empleando una frecuencia portadora por celda. Las interferencias cocanal se indican con una *s* y las interferencias de canal adyacente con una *d*.

Una red celular puede ser representada mediante un grafo simple con pesos en las aristas, donde los vértices se corresponden con las frecuencias hopping que deben asignarse a cada celda y las aristas unen vértices correspondientes a celdas interferentes (figuras 1.2 y 1.3). El peso de una arista representa la distancia que han de guardar las frecuencias asignadas a cada uno de los vértices con el fin de no sufrir interferencias. El problema así planteado puede traducirse finalmente en un problema de coloreado de grafos, donde las frecuencias se representan con colores o números enteros, de manera que la separación entre colores de vértices adyacentes (definido como el valor absoluto de su diferencia) ha de ser como mínimo el peso de la arista que los une.

El nivel de interferencias entre celdas viene representado por la matriz de restricciones, cuyos elementos expresan las distancias que deben guardar las frecuencias de las distintas celdas de la red. Aunque pueden considerarse otras posibilidades, el valor de las restricciones es en general un número entero siendo los valores más habituales 1 y 2. Un valor en la matriz de restricciones igual a 1 requiere una separación de por lo menos una unidad entre las frecuencias de las celdas correspondientes, con el objeto de blindar las interferencias producidas por reflexiones, radiación de los lóbulos traseros de un transmisor, cambios rápidos de celda, etc. De otro lado, una restricción igual a 2, presente cuando existe un solapamiento entre celdas, requiere una separación frecuencial de por lo menos dos unidades. La figura 1.2 muestra un ejemplo de modelado de una red GSM.

En una red GSM existe normalmente más de un transmisor por celda siendo el número de transmisores menor o igual que el número de frecuencias hopping. Por otro lado, la asignación frecuencial (en la que tenemos tantos vértices por celda como frecuencias hopping) indicará cuáles son las frecuencias que utilizarán los transmisores, siendo el orden de utilización de éstas distinto en cada uno (siguiendo las secuencias de salto definidas por las tarjetas hardware de la estación base). De este modo cada celda se desdoblará en un número de vértices igual al número de frecuencias de hopping asignadas. Dos vértices serán adyacentes si las celdas de las que proceden interfieren, correspondiéndose el peso de las aristas con el valor respectivo de la matriz de restricciones (figura 1.3). En el caso en que no se emplee la técnica del

frequency hopping el número de frecuencias coincidirá en cada celda con el número de transmisores dado que en esta situación cada uno de los transmisores emite todo el tiempo en una única frecuencia portadora.

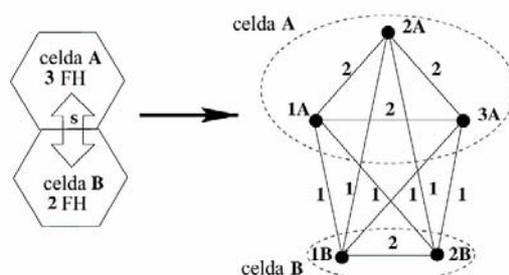


Figura 1.3.: Modelado de una red GSM con más de una frecuencia hopping por celda. El número de frecuencias hopping por celda es 2 y 3 respectivamente.

Finalmente, los vértices correspondientes a una misma celda deberán guardar una distancia de dos unidades. En primer lugar, y dado que los distintos transmisores de una celda deben conmutar las distintas frecuencias disponibles, no tiene sentido repetir una frecuencia en una misma celda. Podría darse además el caso de que dos transmisores emplearan simultáneamente la misma frecuencia (en la misma celda), interfiriéndose mutuamente. De otro lado, y debido al control dinámico de potencia, dos terminales móviles emplazados en una misma celda pueden transmitir simultáneamente a potencias distintas (en función de su distancia a la estación base) pudiendo de este modo los lóbulos laterales del móvil de mayor potencia interferir el lóbulo principal del móvil de potencia menor. Debe evitarse así el uso en una misma celda de frecuencias a distancia uno, siendo de este modo el peso de las aristas correspondientes igual o mayor que 2.

### 1.2.1 Especificaciones para la asignación de frecuencias

El problema de la asignación de frecuencias (con o sin hopping) puede caracterizarse, según se ha descrito en el apartado anterior, mediante una matriz de restricciones, un vector que indique el número de transmisores por celda, un vector de requerimientos de frecuencias hopping, un vector de índice de ocupación de la celda y un plan de frecuencias fijas, alguna de cuyas informaciones puede no emplearse en asignaciones sencillas. Así, mientras la matriz de restricciones y el vector de requerimientos hopping (o el vector de transmisores en caso de no conmutar en frecuencia) modelan la red en términos de teoría de grafos, el vector de transmisores y el vector de ocupación permiten (en el caso de conmutación de frecuencias) calcular la función de coste en el transcurso de la asignación. El plan de frecuencias fijas indica las celdas que no intervienen en la asignación dado que contienen frecuencias que han sido fijadas previamente.

## Matriz de restricciones

La matriz de restricciones o de compatibilidad es una matriz cuadrada de  $n$  filas y  $n$  columnas (siendo  $n$  el número de celdas de la red) cuyos elementos recogen la separación frecuencial que ha de existir entre la celda fila y la celda columna correspondientes. Un 0 significa que las dos celdas no interfieren y que, por tanto, pueden utilizar la misma frecuencia. Un valor de 1 indica que existen interferencias cocanal y que por lo tanto las frecuencias entre las celdas han de situarse, cuando menos, a una distancia igual a 1. Finalmente un valor igual a 2 indica la presencia de interferencia de canal adyacente (los lóbulos laterales interfieren la señal principal) de tal modo que la separación frecuencial ha de ser como mínimo de dos unidades. Existe la posibilidad de incorporar restricciones no enteras (valores reales entre 0 y 2) que indicaran los distintos niveles de interferencias entre celdas de la red. Asimismo, los algoritmos deben mantener una separación mayor o igual a 2 entre las frecuencias de una misma celda.

La calidad del servicio depende en gran medida de la precisión de la matriz de restricciones dado que las frecuencias asignadas estarán relacionadas con los valores de los elementos de la matriz. Por esta razón, la matriz de restricciones se ha de construir con extremo detalle, de manera que refleje las características de la red de la forma más precisa posible y siguiendo criterios conservadores en la elección de los valores.

El criterio utilizado para obtener la matriz de restricciones varía según las características del sistema, como pueden ser el control dinámico de potencia, la transmisión discontinua y el frequency hopping. La misma red puede caracterizarse con diferentes matrices dependiendo de si se representa el subsistema de tráfico, que radia a potencia media cuando hay conversación, o el subsistema de control, que radia con máxima potencia de forma continua con el objeto de facilitar la sintonización de las frecuencias portadoras.

$$M = \begin{pmatrix} 2 & 0 & 0 & 1 & 2 & 1 & 0 & 2 & 1 & 1 \\ 0 & 2 & 1 & 0 & 2 & 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 & 1 & 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 & 1 & 2 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 2 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix} \quad R = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 2 \\ 1 \\ 1 \\ 1 \\ 2 \\ 1 \\ 2 \end{pmatrix}$$

Figura 1.4: Ejemplo de matriz de restricciones y vector de requerimientos para una red de 10 celdas. El vector de requerimientos indica el número de frecuencias hopping por celda o, en el caso en que no se conmute en frecuencia, el número de transmisores por celda.

### **Vector de requerimientos de hopping**

Indica el número de frecuencias (en el caso en que se emplee el frequency hopping) que deben asignarse a cada celda. Este número, como se ha visto, determina el número de vértices asociados a cada celda. El número de frecuencias depende de varios factores y ha de ser siempre mayor o igual al número de transmisores de la celda (ya que cada transmisor emitirá en cada instante a una frecuencia distinta del resto de transmisores de la misma celda). La elección del número de frecuencias hopping puede ser la misma para todas las celdas de la red o variable en función del número de transmisores y del tráfico de la celda.

### **Vector de transmisores**

Indica el número de transmisores asignados a cada celda, es decir, el número de frecuencias simultáneas radiadas (cifra que no tiene por qué coincidir con el número de frecuencias de hopping). El número de transmisores se escoge en función del tráfico que ha de cursar la celda que, a su vez, depende del índice de población, del número total de abonados, del tráfico en la hora punta y del grado del servicio ofrecido por el operador de red.

### **Vector de ocupación**

Indica el porcentaje de ocupación de cada celda. Las restricciones incumplidas entre frecuencias pertenecientes a celdas más ocupadas (porcentajes mayores) tienen una mayor influencia en la interferencia global del sistema.

### **Plan de frecuencias fijas**

Contiene un conjunto de frecuencias predeterminado que no puede ser modificado en el proceso de asignación. Esta restricción suele darse en el caso de ampliación de una red previamente planificada, la cual no puede modificar las frecuencias ya existentes en algunas celdas específicas.



## Capítulo 2. Coloreado de grafos

### Introducción

En el presente capítulo se ofrecen algunas definiciones de la teoría de grafos así como del problema clásico del coloreado de grafos. Se definen también los polinomios cromáticos.

### 2.1 Teoría de grafos. Definiciones

Un grafo simple  $G$  es un par  $(V(G), E(G))$  tal que  $V(G)$  es un conjunto finito de elementos denominados vértices y  $E(G)$  un conjunto finito de pares no ordenados de vértices. Cada uno de los elementos de  $E(G)$  se denomina arista. Tanto los vértices como las aristas de un grafo pueden incluir etiquetas definidas a partir de aplicaciones  $\Phi : V(G) \rightarrow \mathbf{Z}$  y  $\Phi' : E(G) \rightarrow \mathbf{Z}$ . El orden  $n$  de un grafo  $G=(V,E)$  es el número de vértices o cardinal de  $V(G)$ . Asimismo se define el tamaño  $E$  de un grafo  $G=(V,E)$  como el cardinal de  $E(G)$  o el número de aristas de  $G$ .

Según la definición, un grafo simple no puede tener aristas repetidas, esto es, pares de vértices unidos por más de una arista, ni tampoco lazos (aristas que enlazan un mismo vértice). En lo sucesivo cada vez que nos refiramos a un grafo  $G$  se entenderá que  $G$  es un grafo simple.

Se dice que dos vértices  $u$  y  $v$  son adyacentes cuando los une una arista. En este caso se dice que los vértices  $u$  y  $v$  inciden sobre la arista  $uv$  o bien que la arista  $uv$  incide sobre los vértices  $u$  y  $v$ . Dos aristas son adyacentes cuando tienen un vértice en común. El grado  $\delta(v)$  de un vértice  $v$  es el número de aristas que inciden sobre  $v$ .

Dado un grafo simple de orden  $n$  el número máximo de aristas que puede contener es igual a  $n(n-1)/2$ . Se define así la densidad  $\rho(G)$  de un grafo  $G$  de orden  $n$  como el cociente entre el número de aristas del grafo  $G$  y el máximo número de aristas que puede contener un grafo de orden  $n$ . De este modo

$$\rho(G) = \frac{E}{n(n-1)/2} = \frac{2E}{n(n-1)}$$

Un clique  $C \subset V(G)$  de un grafo  $G$  es un conjunto de vértices de  $V(G)$  tal que todo par de vértices de  $C$  son adyacentes entre sí. De otro lado se denomina conjunto independiente a un conjunto  $S \subset V(G)$  de vértices tales que no son adyacentes dos a dos.

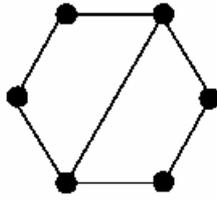


Figura 2.1: Ejemplo de grafo conexo

Una secuencia de aristas es una sucesión de aristas consecutivas  $V_0V_1, V_1V_2, \dots, V_{m-1}V_m$ . De este modo la secuencia dibuja un camino continuo sobre el grafo. Una secuencia en la que no se repiten aristas se denomina cola y si tampoco se repite ningún vértice, trayecto. Un trayecto cerrado tal que el primer y el último vértice coinciden se denomina circuito.

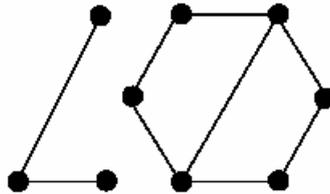


Figura 2.2: Ejemplo de grafo no conexo

Se dice que un grafo  $G$  es conexo cuando es posible trazar al menos un trayecto entre dos vértices cualesquiera de  $V(G)$ . Cada uno de los conjuntos conexos de vértices en que puede descomponerse un grafo no conexo se denomina componente conexa del grafo. La distancia  $d(u,v)$  entre dos vértices  $u$  y  $v$  se define como el cardinal mínimo de todos los trayectos entre  $u$  y  $v$ , es decir, el número mínimo de aristas que hay que recorrer para llegar de un vértice a otro. Cuando un grafo no es conexo y dos vértices pertenecen a componentes distintas se dice que su distancia es infinita. Se denomina diámetro de un grafo  $G$  a la distancia máxima entre dos vértices cualesquiera de  $G$ .

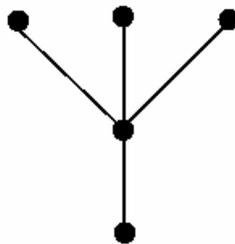


Figura 2.3: Ejemplo de árbol.

Un grafo conexo sin ningún circuito es un árbol. Un grafo completo  $K_n$  es un grafo simple tal que todos sus pares de vértices están unidos por una arista. De

este modo el tamaño de  $K_n$  es igual a  $n(n-1)/2$ . De otro lado, un grafo cuyos vértices tienen todos el mismo grado es un grafo regular. Concretamente se dice que  $G$  es un grafo regular de grado  $r$  si todos sus vértices tienen grado  $r$ .

Un grafo  $G$  es bipartito si su conjunto  $V(G)$  de vértices admite una partición en dos subconjuntos  $V_1$  y  $V_2$  tales que  $V_1 \cup V_2 = V(G)$  y  $V_1 \cap V_2 = \emptyset$  y tales que cada una de las aristas de  $E(G)$  une un vértice del subconjunto  $V_1$  con un vértice de  $V_2$ . De este modo ningún par de vértices de  $V_1$  ni de  $V_2$  son adyacentes entre sí. En el caso en que cada uno de los vértices de  $V_1$  sea adyacente a cada uno de los vértices de  $V_2$  diremos que el grafo es un grafo bipartito completo  $K_{n,m}$  siendo  $n$  y  $m$  los cardinales respectivos de  $V_1$  y  $V_2$ .



Figura 2.4: Ejemplo de grafo bipartito.

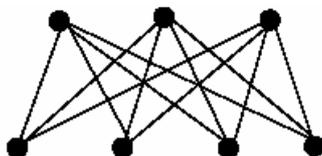


Figura 2.5: Ejemplo de grafo bipartito completo  $K_{4,3}$ .

Dados dos grafos cualesquiera  $G_1 = (V_1, E_1)$  y  $G_2 = (V_2, E_2)$ ,  $G_2$  es subgrafo de  $G_1$  si  $V_2 \subseteq V_1$  y  $E_2 \subseteq E_1$ . Si  $V_2 = V_1$  entonces  $G_2$  es subgrafo generador de  $G_1$ . En caso de que  $G_2$  conserve las aristas de  $G_1$ , es decir, en el caso de que todo par de vértices del grafo  $G_2$  sean adyacentes si y sólo si son adyacentes en  $G_1$ , diremos que  $G_2$  es un subgrafo inducido de  $G_1$ .

Un grafo  $G = (V, E)$  es plano si puede dibujarse sobre el plano sin que ninguna de sus aristas se corte. De otro lado un grafo  $G = (V, E)$  es planar si existe algún grafo isomorfo a  $G$  y plano.

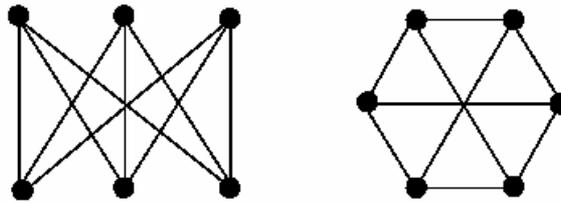


Figura 2.6: Ejemplo de grafos isomorfos.

### 2.1.1 Coloreado de grafos

Un grafo  $G$  es  $k$ -coloreable o  $k$ -partito si existe una aplicación  $\Phi: V(G) \rightarrow \{1, \dots, k\}$  tal que  $\Phi(u) \neq \Phi(v)$  para todo par de vértices  $u$  y  $v$  adyacentes (es decir, tal que dos vértices adyacentes tengan siempre imágenes distintas). La aplicación  $\Phi$  suele denominarse coloreado de los vértices del grafo  $G$  y cada uno de los enteros asignados a cada vértice color. El número cromático  $X(G)$  de un grafo  $G$  es el mínimo entero  $k$  tal que  $G$  es  $k$ -coloreable.

Puede asimismo definirse el coloreado de las aristas como una función  $\Phi': E(G) \rightarrow \{1, \dots, k\}$  tal que dos aristas adyacentes tienen imágenes distintas (i.e. colores distintos asignados). En este caso se dice que el grafo  $G$  es  $k$ -arista coloreable. Finalmente, el índice cromático  $X'(G)$  de un grafo  $G$  es el mínimo entero  $k$  tal que  $G$  es  $k$ -arista coloreable.

El problema del arista coloreado de un grafo puede transformarse en un problema de vértice coloreado a través del grafo línea. Dado un grafo  $G = (V, E)$  se define el grafo línea  $\Gamma(G)$  de  $G$  como un grafo tal que sus vértices se corresponden con las aristas de  $G$  y tal que dos vértices de  $\Gamma(G)$  son adyacentes si y sólo si las aristas correspondientes de  $G$  son adyacentes entre sí.

El número cromático de un grafo completo  $K_n$  es  $n$ . De otro lado, el número cromático de un grafo bipartito es por definición igual a 2. Puede demostrarse que un grafo es bipartito si y sólo si no contiene ningún circuito con un número impar de aristas.

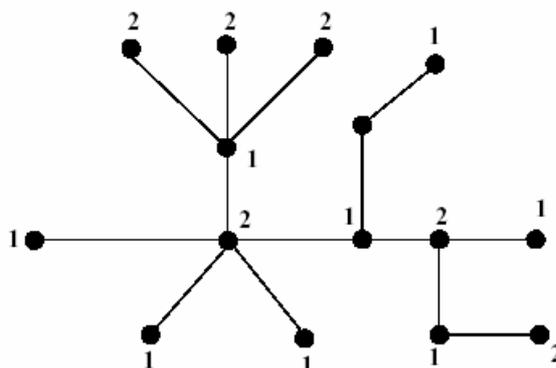


Figura 2.7: Ejemplo 2-coloreado de un grafo bipartito

Dado un grafo cualquiera  $G$  su número cromático es siempre menor o igual al número de vértices. De otro lado, un grafo que contiene un clique de orden  $r$  (es decir un subgrafo completo de orden  $r$ ) tiene un número cromático mayor o igual a  $r$ . Estas cotas pueden ajustarse en función del grado de los vértices del grafo.

**Teorema 2.1** *Un grafo  $G = (V, E)$  de grado máximo  $\Delta$  es  $(\Delta + 1)$ -coloreable.*

**Demostración.** Por construcción del grafo. Supóngase que tenemos una lista de  $\Delta + 1$  colores y que el grafo se forma añadiendo los vértices de forma sucesiva y las aristas correspondientes en cada caso. En este caso cada vez que se añada un vértice  $v$  podrá asignársele un color de la lista de  $\Delta + 1$  colores que sea diferente de los colores de, a lo sumo, los  $\Delta$  vértices adyacentes a  $v$ . De este modo en el momento en que se añada el último vértice se habrá obtenido además un  $\Delta + 1$ -coloreado del grafo. ■

Esta cota puede mejorarse para el caso en que  $G$  sea un grafo simple conexo y no completo de grado máximo  $\Delta \geq 3$ . En este caso  $G$  es  $\Delta$ -coloreable. Estas cotas no son sin embargo restrictivas en general. En el caso del grafo bipartito completo  $K_{n,m}$  el grado máximo del grafo  $\Delta = \max\{n, m\}$  puede ser arbitrariamente grande resultando siempre un número cromático igual a 2. De otro lado puede demostrarse que todo grafo planar es 5-coloreable.

### 2.1.2 Polinomios cromáticos

Sea  $G$  un grafo simple y sea  $P_G(k)$  el número de coloreados distintos que admite  $G$  con un número de colores igual a  $k$ . Dado que como se demuestra a continuación  $P_G(k)$  es una función polinómica que depende del número  $k$  de colores,  $P_G(k)$  se denomina polinomio cromático de  $G$ .

Así, por ejemplo, para el grafo de la figura 2.8 el polinomio cromático es  $P_G(k) = k(k - 1)^2$  dado que el vértice central puede ser coloreado con cualquiera

de los  $k$  colores disponibles y los extremos con cualquier otro color distinto del vértice central. Este resultado puede generalizarse para cualquier árbol siendo el polinomio cromático de un árbol  $A_n$  de orden  $n$  el polinomio  $P_{A_n}(k) = k(k-1)^{n-1}$ .



Figura 2.8: Grafo de polinomio cromático  $k(k-1)^2$

Del mismo modo para el grafo completo de orden 3 se tiene  $P_{K_3}(k) = k(k-1)(k-2)$  y en general en  $P_{K_n}(k) = k(k-1)\cdots(k-(n-1))$  el caso en que  $k \geq n$ .

Para todo entero  $k$  menor que el número cromático de un grafo  $X(G)$  se tiene  $P_G(k) = 0$ . De otro lado si  $k \geq X(G)$  entonces  $P_G(k) > 0$ . El siguiente teorema y corolario demuestran además que dado un grafo simple  $G$  de orden  $n$  la función  $P_G$  es un polinomio de grado  $n$ .

**Teorema 2.2** Sean  $v$  y  $w$  dos vértices no adyacentes de un grafo  $G$ . Sea  $G_1$  el grafo obtenido de añadir a  $G$  una arista entre  $v$  y  $w$  y sea  $G_2$  el grafo que resulta de identificar los dos vértices. En este caso se cumple:

$$P_G(k) = P_{G_1}(k) + P_{G_2}(k)$$

**Demostración.** Dado un  $k$ -coloreado cualquiera de  $G$  los vértices  $v$  y  $w$  pueden tener el mismo color o bien colores distintos. El número de coloreados en los cuales  $v$  y  $w$  tienen colores distintos es igual a  $P_{G_1}$ . De otro lado el número de coloreados en los cuales  $v$  y  $w$  comparten el mismo color es igual a  $P_{G_2}$ . De este modo se cumple que

$$P_G(k) = P_{G_1}(k) + P_{G_2}(k). \quad \blacksquare$$

Esta propiedad queda ilustrada en el ejemplo de la figura 2.9. Calculando el polinomio cromático de cada uno de los grafos puede verse que efectivamente se cumple la igualdad del teorema 2.2 de modo que:

$$k(k-1)(k-2)^2 = k(k-1)(k-2)(k-3) + k(k-1)(k-2)$$

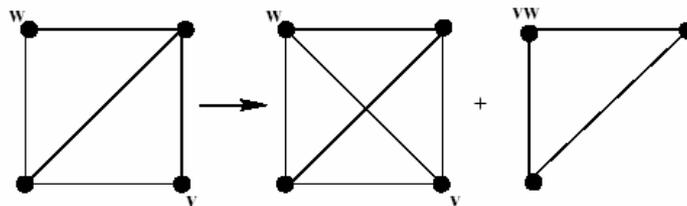


Figura 2.9: Descomposición del grafo  $G$  en  $G_1$  y  $G_2$ .

**Corolario 2.3** *La función cromática de un grafo simple de orden  $n$  es un polinomio mónico de grado  $n$ .*

**Demostración.** Se itera el procedimiento de descomposición del teorema 2.2 tomando en cada caso un par de vértices no adyacentes. De este modo el proceso terminará cuando cada uno de los pares de vértices de los grafos resultantes sean adyacentes, es decir, cuando todos los grafos resultantes sean completos. De otro lado, según el teorema 2.2 el polinomio cromático  $P_G(k)$  es igual a la suma de los polinomios  $P_{K_i}(k)$  de los grafos completos generados. Como estos valores, según se ha visto, son funciones polinómicas  $P_G(k)$  será también un polinomio. De otro lado, como en ningún paso se han introducido vértices en la operación, de todos los grafos completos finales el de mayor orden es  $K_n$  que además sólo aparecerá una vez. De este modo  $P_G(k)$  tendrá grado  $n$  y además el coeficiente del término de mayor grado será 1. ■

Puede asimismo demostrarse que el coeficiente del término de grado  $n-1$  es igual al número de aristas del grafo cambiado de signo. De otro lado como por definición  $P_G(0)=0$  el término independiente del polinomio cromático de cualquier grafo es 0.



## Capítulo 3. Optimización combinatoria

Las cotas y resultados obtenidos en capítulos anteriores no siempre permiten acotar convenientemente el número cromático de un grafo ni mucho menos obtener coloreados ajustados. En el siguiente capítulo se define la complejidad del problema del coloreado en relación con otros problemas llamados NP-completos y se describen una serie de técnicas algorítmicas que permiten obtener coloreados de forma eficiente.

También se describen los algoritmos Hormigas y Simulated Annealing con el objeto de aplicarlos en redes telefónicas y obtener distribuciones frecuenciales que minimicen el número de frecuencias empleadas. Finalmente, se describen una serie de algoritmos combinatorios, entre ellos el algoritmo Hormigas, para la resolución aproximada del problema del coloreado de grafos.

### 3.1 Complejidad del coloreado de grafos

Se dice que una función  $G(n)$  es de orden superior a otra función  $F(n)$ , si existe algún entero  $n_0$  tal que  $\forall n > n_0$  se cumple que  $G(n) > k \cdot F(n)$  para un número  $k$  positivo. En la tabla 3.1 se comparan valores de distintas expresiones en función de  $n$ . Puede observarse que las expresiones polinómicas crecen a un ritmo mucho menor que las funciones exponenciales o la función factorial. Se dice asimismo que una función  $f(n)$  es una  $O(g(n))$  de la función  $f(n)$  si el cociente  $f(n)/g(n)$  puede acotarse para todo  $n$  arbitrariamente grande.

En la práctica se considera que un algoritmo es eficiente cuando es capaz de encontrar una solución óptima del problema en función de tiempo polinomial. En ocasiones esto no es posible. En estos casos se considera entonces que un algoritmo de resolución es aceptable si es capaz de dar una aproximación de la solución óptima en un intervalo de tiempo polinómico (es decir tras un número de operaciones que aumenta polinómicamente en relación al tamaño del problema) o bien si en la mayoría de casos es capaz de encontrar la solución óptima en un tiempo polinómico de ejecución.

Función	$n = 2$	$n = 8$	$n = 128$	$n = 1024$
$n$	2	$2^3$	$2^7$	$2^{10}$
$n \log_2 n$	2	$3 \cdot 2^3$	$7 \cdot 2^7$	$10 \cdot 2^{10}$
$n^2$	$2^2$	$2^6$	$2^{14}$	$2^{20}$
$n^3$	$2^3$	$2^9$	$2^{21}$	$2^{30}$
$2^n$	$2^2$	$2^8$	$2^{128}$	$2^{1024}$
$n!$	2	$4.9 \cdot 2^{13}$	$4.5 \cdot 2^{714}$	$2^{8769}$

Tabla 3.1: orden de crecimiento de algunas funciones básicas

Sea un grafo  $G$  de orden  $n$  y número cromático  $\chi(G) = k$  tal que quiere obtenerse un coloreado óptimo de  $G$ . Un proceso de búsqueda exhaustiva de un coloreado óptimo de  $G$  requiere en el mejor de los casos y suponiendo el número cromático desconocido un número de iteraciones igual a:

$$N_{iter} = \sum_{i=2}^{k-1} i^n$$

En este caso el algoritmo comenzaría explorando todos los posibles 2-coloreados del grafo y comprobando en cada caso si el coloreado cumple las restricciones (dos vértices adyacentes deben tener asignados distintos colores). A continuación, si no se hubiera encontrado ningún 2-coloreado correcto, el algoritmo repetiría la operación para tres colores y así sucesivamente. El número de coloreados distintos que admite un grafo de orden  $n$  (cumplan o no las restricciones) con un número de colores igual a  $i$  es  $i^n$ . En caso de que alcanzado el número cromático el primer coloreado explorado fuera correcto se tendría el número de operaciones calculado.

De este modo, dado un grafo  $G$  de orden 50 cuyo número cromático es  $\chi(G)=4$  el número de operaciones necesarias para encontrar un coloreado óptimo de  $G$  mediante el método de búsqueda exhaustivo es igual a  $1.27 \cdot 30^{30}$ . En el caso en que se partiera de una cota  $k_m$  del número cromático el número de iteraciones sería

$$N_{iter} \sum_{i=k_m}^{k-1} i^n$$

en caso de que la cota fuera inferior o

$$N_{iter} \sum_{i=k-1}^{k_m} i^n$$

en el caso en que se partiera de una cota superior y se fuera rebajando el número de colores empleados hasta encontrar un número de colores con el que no fuera posible colorear el grafo. La expresión final es en los tres casos exponencial. De este modo no puede resolverse en general un problema del coloreado de grafos de forma exhaustiva. Más en general no se conoce ningún algoritmo capaz de devolver la solución óptima de un problema de coloreado en un tiempo polinómico de modo que deberán describirse técnicas algorítmicas capaces de aproximar una solución del problema en intervalos de ejecución polinomiales.

## 3.2 Problemas NP-completos

Un problema combinatorio cuya resolución requiere un algoritmo de complejidad  $O(p)$  (es decir un algoritmo que realice un número de operaciones proporcional a una función polinómica del tamaño del problema) se dice que es un problema de tipo **P**. De otro lado, un problema es de tipo **NP** cuando puede ser resuelto polinómicamente por una máquina de Turing no determinista, que puede definirse como un conjunto de máquinas de Turing procesando la información en paralelo. Una solución de un problema **NP** puede ser comprobada en un intervalo de tiempo polinomial.

Un problema de tipo **P** pertenece por definición a la clase **NP**. Se ha conjeturado de otro lado que **P**  $\neq$  **NP**.

Se dice que un problema combinatorio se puede transformar polinómicamente en otro problema si dada una solución de uno de los problemas puede construirse en un tiempo polinomial una solución del segundo problema. Se define de este modo la clase de problemas NP-completos como el subconjunto de los problemas de tipo **NP** tales que pueden ser traducidos unos en términos de los otros en un tiempo polinomial. Si pudiera resolverse eficientemente (es decir, mediante un algoritmo de complejidad polinómica) uno de los problemas de tipo NP-completo se demostraría que toda la familia puede ser resuelta en intervalos de tiempo polinomiales.

Algunos de los problemas clásicos de tipo NP-completo pertenecen a la teoría de grafos, como el problema del conjunto independiente máximo, el aplanamiento de grafos o el problema del coloreado. Estos problemas no pueden ser resueltos genéricamente con algoritmos de orden polinómico (o para ser más precisos, no se han encontrado por el momento técnicas polinómicas de resolución). En este caso deben definirse algoritmos capaces de encontrar soluciones casi óptimas en intervalos de ejecución polinómicos. Estos métodos, que suelen denominarse métodos de optimización combinatoria, aprovechan en cada caso las características del espacio de soluciones para acercarse eficientemente a las zonas en donde existen soluciones óptimas del problema. Algunos de estos algoritmos, como por ejemplo los Algoritmos Genéticos, Simulated Annealing o el algoritmo Hormigas, responden además a esquemas generales y probabilistas (con el objeto de evitar extremos locales de la función de coste) que pueden ser aplicados sobre una amplia gama de problemas combinatorios.

## 3.3 Simulated Annealing

Si se reduce bruscamente la temperatura de un líquido por debajo del punto de fusión el resultado es un estado desordenado de energía mayor que la que corresponde al compuesto en estado cristalino. En este caso, las moléculas han alcanzado un mínimo local de energía. Si, por el contrario, la temperatura del líquido fuera reducida lentamente y de acuerdo con una pauta adecuada de

enfriamiento, el líquido evolucionaría hacia el equilibrio y de esta forma su configuración, de energía mínima, hubiese sido la de un compuesto ordenado y cristalino.

El algoritmo de Simulated Annealing se define según la pauta de enfriamiento descrita para un compuesto cristalino. De este modo el algoritmo cambia en cada iteración (normalmente de forma restringida en el sentido de que modifica sólo una de las posiciones del vector de soluciones) y de forma aleatoria la última solución calculada. Una vez obtenida la nueva solución Simulated Annealing compara las energías (es decir las funciones de coste) de ambas soluciones.

Si la energía de la nueva solución es menor que la energía de la antigua solución (supondremos un problema de minimización sin pérdida de generalidad) entonces la nueva solución es aceptada y sustituye a la anterior. Si, por el contrario, la energía ha aumentado la solución no se acepta con una probabilidad determinada por el factor de Boltzman  $e^{\frac{\Delta f}{T}}$  donde  $\Delta f$  es la diferencia de energías entre el nuevo estado  $j$  y el estado anterior  $i$ , y donde  $T$  es la temperatura actual del sistema.

$$P_{\text{aceptar}}(j) = \begin{cases} 1 & \text{si } f(j) \leq f(i) \\ e^{-\frac{f(j)-f(i)}{T}} & \text{si } f(j) > f(i) \end{cases}$$

De este modo, la probabilidad de aceptar la nueva solución es menor cuanto mayor es la diferencia de energías, y mayor cuanto mayor es la temperatura del sistema.

Este proceso de generación y aceptación o refutación se repite un cierto número de veces para cada temperatura. Una vez se han completado el número de iteraciones el sistema se enfría (i.e. se disminuye la temperatura del sistema), repitiéndose a continuación el mismo proceso hasta que se alcanza la temperatura final. Por otro lado puede determinarse un número mínimo de aceptaciones para cada temperatura, de tal forma que si al llegar a la última iteración no se han aceptado un número determinado de cambios el proceso continúa hasta alcanzar la cota mínima de aceptaciones requerida o bien hasta superar un número máximo de iteraciones. En este punto el algoritmo puede cambiar a una nueva temperatura o dar por finalizada la búsqueda.

Conforme el algoritmo se aproxima a la temperatura mínima la probabilidad de aceptar un aumento de la energía se aproxima a cero. Esta dependencia con respecto a la temperatura permite escapar de los mínimos locales y asegurar la convergencia del algoritmo. De hecho, se puede demostrar que Simulated Annealing conduce al máximo global cuando el número de iteraciones crece indefinidamente.

Uno de los parámetros que determinan el comportamiento del algoritmo es la temperatura inicial. Una temperatura inicial elevada obliga al algoritmo a aceptar empeoramientos de la función de coste con una frecuencia alta de manera que la energía oscilará mientras la temperatura no haya superado un determinado umbral. De esta forma, aunque la solución encontrada finalmente pueda ser aproximadamente la misma, el número de iteraciones habrá aumentado con el consiguiente empeoramiento de la eficiencia del algoritmo. En cambio, la elección de una temperatura inicial demasiado baja comportará la no aceptación (o aceptación con una probabilidad casi nula) de empeoramientos de la función de coste, aumentando la probabilidad de estancamiento en mínimos locales. La temperatura final debe ser tal que un determinado empeoramiento de la función de coste sea aceptado con una probabilidad comparable a cero. Esta temperatura debe asegurar un número mínimo de estados explorados.

### Inicio

Inicializa temperatura  $T = T_0$   
 busca una solución al azar  
 calcula función de coste

### para cada iteración

genera una nueva solución  
 calcula función de coste  
**si** nueva solución mejor antigua solución  
 guarda nueva solución

### si no

guarda nueva solución con probabilidad  $e^{-\frac{\Delta f}{T}}$   
**si** nueva solución menor que mejor solución  
 guarda nueva solución como mejor solución  
**si** número de iteraciones igual a cambio de temperatura  
 cambia temperatura  $T_k = rT_{k-1}$

### fin para

**hasta** máximo número de iteraciones o solución encontrada

Figura 3.1: Estructura básica de Simulated Annealing

Entre la primera y la última temperatura debe establecerse una pauta de enfriamiento. Si la probabilidad de aceptación de estados disminuye de forma exponencial en función de la temperatura y se desea que el cambio de probabilidad de aceptación al enfriarse el sistema sea significativo, la temperatura debe ser reducida según pautas exponenciales. De esta forma, el esquema de enfriamiento sigue la fórmula:

$$T_k = T_0 r^k$$

donde  $T_k$  es el valor de la temperatura una vez se ha enfriado  $k$  veces,  $T_0$  es la temperatura inicial y  $r$  es la razón de enfriamiento, que acostumbra a tener valores próximos a la unidad.

Finalmente el número de iteraciones que el algoritmo debe realizar para cada una de las temperaturas debe garantizar una exploración adecuada del espacio de soluciones. Es necesario un número mínimo de exploraciones para que el algoritmo no quede encerrado en algún mínimo local.

### 3.4 Algoritmo Hormigas

Se describe en el presente apartado un algoritmo de optimización combinatoria para el problema del coloreado de grafos que puede no obstante generalizarse para cualquier clase de problema de asignación con restricciones y más en general para cualquier tipo de problema combinatorio.

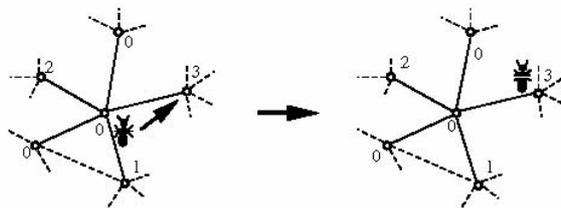


Figura 3.2: Ejemplo de movimiento de una hormiga en el coloreado de un grafo

El algoritmo Hormigas distribuye inicialmente una serie de agentes (las llamadas hormigas) a lo largo de los vértices del grafo de forma aleatoria. A continuación cada una de las hormigas modifica el coloreado del grafo conforme un criterio de optimización local. De este modo, cada hormiga se desplaza en cada iteración hasta el vértice adyacente con una función de coste mayor (en este caso, al vértice cuyo color es utilizado por un número mayor de vecinos) y sustituye su color por un nuevo color con el objeto de minimizar la función de coste del vértice, es decir, el número de violaciones cometidas (que no siempre puede ser cero) en la asignación del nuevo color.

#### Inicio

colorea aleatoriamente el grafo  
distribuye aleatoriamente las hormigas

#### para cada hormiga

**si** probabilidad  $p_n$  mueve al peor vértice adyacente

**si no** mueve a otro vértice adyacente

**si** probabilidad  $p_c$  asigna el mejor color

**si no** asigna cualquier color

**actualiza** función de coste

**si** nueva solución menor que mejor solución

guarda nueva solución como mejor solución

#### fin para

**hasta** máximo número de iteraciones o solución encontrada

Figura 3.3: Estructura básica del algoritmo Hormigas.

Este proceso es repetido en cada iteración de forma aleatoria: cada agente u hormiga se desplaza al peor vértice adyacente con una determinada probabilidad  $p_n$  (en otro caso puede desplazarse a cualquier otro vértice adyacente) y le asigna el mejor color también bajo otra probabilidad  $p_c$  (en otro caso puede asignarle cualquier color escogido de forma equiprobable). Ambas probabilidades son parámetros ajustables. El proceso de movimiento (y coloreado) de cada agente u hormiga se lleva a cabo de forma simultánea a la acción de otras hormigas. El número de hormigas que se desplaza a lo largo del grafo es un parámetro ajustable en función del diámetro del grafo.

### 3.5 Coloreado de grafos y generalización de Hormigas

En el presente apartado se generaliza el uso del algoritmo Hormigas a un problema de coloreado de grafos (y más en general se explica cómo aplicarlo a un problema genérico de optimización combinatoria) y se comparan los resultados con los coloreados obtenidos mediante un segundo algoritmo descrito por Costa y Hertz (ver [7]). En este caso, un coloreado de  $k$  colores de un grafo  $G$  de orden  $n$  se representa por un vector de  $n$  coordenadas tal que cada coordenada toma un valor entre 0 y  $k-1$  igual al color del vértice correspondiente.

Las tablas 3.2, 3.3 y 3.4 muestran los resultados obtenidos sobre grafos aleatorios de orden 100, 300, 500 y 1000 respectivamente, al aplicar el algoritmo Hormigas y el algoritmo de Costa y Hertz. El mejor comportamiento de Hormigas en relación al algoritmo de Costa se deduce de la distinta naturaleza de los algoritmos (que sólo comparten el mecanismo de funcionamiento en paralelo). Así el algoritmo Hormigas se desplaza a lo largo del grafo (contrariamente al algoritmo de Costa que se desplaza a través del espacio de soluciones) procesando cada hormiga la información mediante técnicas implícitas (en contraste con las grandes listas que requiere el algoritmo de Costa).

orden	100	300	500	1000
Costa	12.65	29.20	46.40	91.50
Horm	12.05	27.50	42.55	78.25

Tabla 3.2 : Comparación entre el algoritmo Hormigas y el algoritmo de Costa y Hertz. En cada caso indica la media de los colores necesarios sobre un promedio de 20 grafos aleatorios con una probabilidad igual a 0.4 a incluir una arista.

orden	100	300	500	1000
Costa	15.40	35.70	55.60	111.00
Horm	14.85	34.60	53.10	99.15

Tabla 3.3: Comparación entre el algoritmo Hormigas y el algoritmo de Costa para grafos aleatorios con probabilidad 0.5 de incluir una arista.

orden	100	300	500	1000
Costa	18.65	44.40	68.80	127.50
Horm	17.95	43.00	66.60	123.05

Tabla 3.4: Comparación entre el algoritmo Hormigas y el algoritmo de Costa para grafos aleatorios con probabilidad 0.6 de incluir una arista.

El algoritmo Hormigas puede entenderse de este modo como un algoritmo de Simulated Annealing “dirigido” o una versión particular de un algoritmo de Tabu Search (clase particular de algoritmo que realiza una búsqueda similar a Simulated Annealing con el apoyo de listas que registran las soluciones anteriores e impiden que el algoritmo entre en bucles cerrados). En cada iteración el algoritmo Hormigas escoge, para cada una de las hormigas, un vértice y le asigna un nuevo color. Esta operación se corresponde en el espacio de soluciones a la selección de una de las coordenadas del vector solución y a su redefinición conforme a un criterio de optimización local (el algoritmo Hormigas intenta minimizar en cada movimiento el número de violaciones cometidas en cada vértice). Estas dos operaciones son formalmente equivalentes al procedimiento seguido por Simulated Annealing para generar una nueva solución (escoge una coordenada del vector solución y cambia su valor) con la diferencia de que en el caso de Hormigas la selección se hace no de forma aleatoria sino conforme un criterio de optimización. El algoritmo Hormigas puede verse asimismo como una variación del algoritmo de Tabu Search que genera soluciones (mediante el mismo procedimiento que Simulated Annealing) que luego son aceptadas o rechazadas con arreglo a una lista que contiene un historial de la evolución del algoritmo y debe actualizarse en cada iteración. Al contrario que Tabu Search, el algoritmo Hormigas basa sus decisiones en un estudio local del grafo que en general (y dependiendo de cómo se haya definido la función de coste local) requiere un número de operaciones menor que el proceso y actualización de la lista con la que trabaja Tabu Search.

La “memoria” de un algoritmo puede considerarse (contrariamente al modelo clásico) no desde la perspectiva de una matriz de datos que registra informaciones explícitas de la evolución del algoritmo, sino como una estrategia capaz de beneficiarse de la historia reciente del algoritmo (y, como se ha dicho, sin manejar por ello registros explícitos de información). El algoritmo Hormigas funciona con arreglo a este segundo esquema de memoria que no reproduce los acontecimientos pasados en listas explícitas de información. Como se ha

dicho, cada vez que una hormiga se desplaza de un vértice  $i$  a un vértice  $j$  y cambia su color la hormiga permanece en el vértice  $j$  hasta la siguiente iteración, en la cual examina los vértices adyacentes a  $j$ . De esta manera el algoritmo Hormigas incluye un tipo de memoria implícita, también llamada *stigmergy*, dado que tras haber sido cambiado el color del vértice  $j$  es posible que alguno de los vértices adyacentes a  $j$  haya resultado afectado y convenga en consecuencia modificar su asignación. De este modo cuando en la siguiente iteración la hormiga, en vez de desplazarse arbitrariamente a cualquier otro punto del grafo, explora la zona contigua al vértice  $j$  está teniendo en cuenta la propia historia del algoritmo, información que sin embargo no debe ser registrada en ninguna matriz ya que está contenida en el propio grafo (o medio) y en las posiciones ocupadas por las diferentes hormigas. Las hormigas manejan así información de los acontecimientos pasados del algoritmo sin necesidad de realizar cálculos ni de evaluar enormes matrices de datos, particularidad que se traduce finalmente en un aumento de la eficiencia del algoritmo.

Alg	descriptor	funciona	falta operador
a-alg.	cada hormiga salta al azar hasta un vértice adyacente y cambia al mejor color (cmc)	no funciona	1a (y así 2)
b-alg	cada hormiga salta al azar a otro vértice y cmc	no funciona	1a (y así 2)
c-alg	cada hormiga se desplaza al peor vértice adyacente o al azar a otro vecino y cmc	ok	ninguno
cI-alg	cada hormiga se desplaza al peor vértice adyacente o al azar a otro vecino y cambia por cualquier color	no funciona	1b
cII-alg	como c, pero no en paralelo	regular	3
d-alg	cada hormiga se desplaza al peor vértice adyacente o al azar a cualquier otro vértice y cmc	ok, pero peor que c	2 es atenuado
e-alg	cada hormiga se desplaza aleatoriamente a cualquier vértice, busca el peor vértice adyacente y cmc	muy pobre	2

Tabla 3.5 : Comparación entre los distintos algoritmos obtenidos de eliminar alguno de los operadores que definen el algoritmo Hormigas.

Con objeto de aplicar el algoritmo Hormigas a cualquier tipo de problema de asignación de recursos se han separado en un conjunto genérico de operadores las acciones ejecutadas por Hormigas en cada iteración:

- **Op 1.a.** Desde el vértice actual (item) desplazarse al peor vértice adyacente.
- **Op 1.b.** Asignar el mejor color (recurso) posible al nuevo vértice (item)
- **Op 2.** Continuar la búsqueda entre los vértices adyacentes al último vértice afectado (memoria a corto plazo)
- **Op 3.** Operador de paralelismo (memoria a largo plazo).

Se han realizado asimismo modificaciones sobre el algoritmo Hormigas con el propósito de evaluar la importancia de los distintos operadores, habiéndose eliminado en cada caso uno de ellos, tras lo cual se han intentado colorear distintos tipos de grafos con el algoritmo resultante. La tabla 3.5 contiene una descripción de los algoritmos obtenidos, el operador sustraído en cada caso y

los resultados de aplicar los algoritmos a un problema de coloreado de grafos (se ofrece una valoración cualitativa dado que en aquellos casos en que el algoritmo no obtiene resultados no se han realizado estadísticas).

Como puede observarse el mejor algoritmo de la tabla es el algoritmo original c-*alg* mientras que si se elimina alguno de los operadores 1a, 1b o 2 el algoritmo resultante no ofrece buenos coloreados. De este modo para obtener un algoritmo de asignación eficiente es necesario combinar los tres primeros operadores: mientras 1a y 1b se benefician de una información de carácter puntual (lo que ve cada hormiga en cada iteración) el operador 2 contribuye a la formación de la memoria a corto plazo.

Los cuatro operadores descritos pueden ser aplicados sin pérdida de generalidad sobre cualquier problema combinatorio de asignación. Para ello deberá definirse un área de influencia para cada uno de los ítems del problema (normalmente se establecerán adyacencias con aquellos ítems con los que guarda alguna relación de incompatibilidad) así como una función de coste local que permita tomar decisiones sin tener que analizar el estado completo de la asignación. Aunque como función de coste local puede tomarse la función global resulta conveniente definir criterios locales de optimización con el objeto de reducir el número de cálculos y aumentar la eficiencia del algoritmo.

## Capítulo 4. Multicoloreado de grafos.

### Introducción

El problema de la asignación de frecuencias, tal y como se ha visto en los capítulos anteriores, puede ser planteado en términos de la teoría de grafos como un problema de coloreado y ser resuelto así mediante el empleo de técnicas algorítmicas como Hormigas o Simulated Annealing. De otro lado, en las redes celulares de telefonía móvil es habitual trabajar con más de una frecuencia por celda debido a los requerimientos de tráfico. Esto implica, en términos de la teoría de grafos, tener que colorear con más de una frecuencia cada vértice (o celda) del grafo resultante. En estos casos se ha llevado a cabo en primer término una adaptación del problema (descrita en el primer capítulo) con el objeto de resolverlo como si se tratara de un coloreado simple, desdoblado para ello cada celda en una serie de vértices (tantos vértices como frecuencias tiene asignadas la celda). Las adyacencias de los vértices del grafo construido se describen así en función de las interferencias entre celdas, de modo que dos vértices son adyacentes si pertenecen a celdas interferentes, siendo el peso de la arista que los une igual a la distancia que deben guardar las frecuencias asignadas en cada celda.

En el presente capítulo se desarrolla un modelo teórico que, combinado con las técnicas algorítmicas descritas en los capítulos anteriores, permite colorear un grafo con más de un color por vértice sin necesidad de transformar el grafo ni desdoblar los vértices. Los resultados y métodos descritos son en cualquier caso extrapolables para un número mayor de colores e incluso para el caso genérico en que cada vértice lleve asociado un número variable de colores. En nuestro caso, explicaremos como realizar un 3-coloreado partiendo de un 2-coloreado realizado en un proyecto anterior.

### 4.1 Multicoloreado de grafos

Sea un grafo  $G = (V, E)$ , se define un  $r$ -multicoloreado o un  $r$ -tuple coloreado de  $G$  como una asignación de colores (o enteros) a los vértices de  $G$ , de forma que cada vértice tenga asignados  $r$  colores diferentes y dos vértices adyacentes no tengan ningún color en común.

Se define asimismo el número  $r$ -multicromático o más genéricamente número multicromático  $X_r(G)$  de un grafo  $G$  como el cardinal del mínimo conjunto de colores necesarios para  $r$ -multicolorear  $G$ . En ocasiones, cuando no haya posibilidad de ambigüedad, nos referiremos al número multicromático como número cromático  $X_r(G)$ .

**Proposición 4.1** Sea  $X(G) = k \Rightarrow X_r(G) \leq r \cdot k$

**Demostración.** Se puede  $r$ -multicolorar el grafo repitiendo  $r$  veces el primer coloreado con  $k$  colores (de 1 a  $k$  el primer coloreado, de  $k+1$  a  $2k$  el segundo coloreado, y así sucesivamente). ■

Es sencillo demostrar además la existencia de grafos para los cuales  $X_r(G) < r \cdot k$ . Así, por ejemplo, el ciclo de orden 5 (y en general cualquier ciclo  $C_n$  con  $n$  impar), que tiene número cromático  $X(C_5) = 3$ , puede ser 2-multicolorado con 5 colores, tal y como ilustra la figura 4.1.

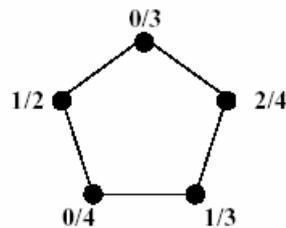


Figura 4.1: Ejemplo de 2-multicolorado de un ciclo de orden 5.

De otro lado, si se tiene un  $r$ -multicolorado con  $X_r(G)$  colores, se puede conseguir un  $(r+1)$ -multicolorado añadiendo a cada vértice un nuevo color (escogido entre  $X_r(G)+1$  y  $X_r(G)+k$ ) y distribuido según un coloreado simple de  $X(G) = k$  colores. De este modo

$$\begin{aligned} X(G) &= k \\ X_2(G) &\leq 2k & X_2(G) &= 2k - \Delta_2 & \Delta_2 &\geq 0 \\ X_3(G) &\leq 3k - \Delta_2 & X_3(G) &= 2k - \Delta_2 - \Delta_3 & \Delta_3 &\geq 0 \end{aligned}$$

y en general

$$X_r(G) \leq X_{r-1}(G) + k \quad X_r(G) = rk - \sum_{i=2}^r \Delta_i \quad \Delta_i \geq 0$$

donde  $\Delta_i$  puede ser mayor, igual o menor que  $\Delta_{i+1}$ , según se demuestra más adelante en el teorema 4.10.

#### 4.1.1 Cotas generales

Se ofrecen a continuación algunos resultados simples sobre las cotas del número multicromático de un grafo  $G$ .

**Proposición 4.2** *Sea un grafo  $G$  tal que contiene un clique de orden  $c$ . En este caso se cumple  $X_r(G) \geq r \cdot c$*

**Demostración.** Inmediata dado que ninguno de los  $r$  coloreados podrá utilizar menos de  $c$  colores distintos en cada caso. ■

**Proposición 4.3** *Dado un grafo  $G$  de orden  $n$  tal que  $S$  es el orden del conjunto independiente máximo de  $G$ . En este caso*

$$X_r(G) \geq \left\lceil \frac{r \cdot n}{S} \right\rceil$$

**Demostración.** Dado un color  $j$  cualquiera,  $j$  puede ser asignado en como máximo  $S$  vértices diferentes. De otro lado, si queremos multicolorear  $G$  con  $r$  colores en cada vértice tendremos un total de  $n \cdot r$  colores asignados a lo largo de todo el grafo (contando las repeticiones). De aquí  $X_r(G) \geq \left\lceil \frac{r \cdot n}{S} \right\rceil$ . ■

**Proposición 4.4** *Dado un grafo  $G$  conexo y no completo de grado máximo  $\Delta \geq 3$  se cumple  $X_r(G) \leq r \cdot \Delta$*

**Demostración.** Por el teorema de Brooks el número cromático de un grafo  $G$  cumple  $X(G) \leq \Delta$  si  $G$  es conexo y no es completo y tiene grado máximo  $\Delta \geq 3$ . Entonces aplicando la proposición 4.1 se obtiene  $X_r(G) = r \cdot X(G) \leq r\Delta$  ■

**Proposición 4.5.** *Dado un grafo  $G$  de número  $r$ -multicromático  $X_r(G)$  se cumple  $X_r(G) \leq \min\{X_i(G) + X_j(G) \mid i + j = r\} \leq rX(G)$*

**Demostración.** La primera desigualdad se deduce del hecho de que para obtener un multicoloreado con  $r$  colores pueden superponerse dos multicoloreados con  $i$  y  $j$  colores por vértice respectivamente, si  $i + j = r$ . De otro lado, de la proposición 4.1 y para  $i + j = r$ , queda

$$X_i(G) + X_j(G) \leq iX(G) + jX(G) = rX(G) \quad \blacksquare$$

#### 4.1.2 Número multicromático de algunas familias de grafos

Se calcula a continuación el número cromático de algunas familias de grafos.

**Proposición 4.6** *El número  $r$ -multicromático de un grafo completo de orden  $n$  es*

$$X_r(K_n) = n \cdot r$$

**Proposición 4.7** *Sea  $G = (V, E)$  un grafo bipartito. En este caso*

$$X_r(G) = 2r$$

**Demostración.** Dado un grafo  $G=(V,E)$  bipartito, existe al menos un multicolorado de  $G$  con  $2r$  colores (superponiendo  $r$  veces el primer coloreado con dos colores) y por lo tanto  $X_r(G) \leq 2r$ . De otro lado de (4.2) se tiene que  $X_r(G) \geq 2r$  ya que todo grafo bipartito contiene un clique de orden 2. De esta forma  $X_r(G) = 2r$ . ■

Teniendo en cuenta que un árbol es un caso particular de grafo bipartito no nulo se obtiene el siguiente corolario.

**Corolario 4.8** El número  $r$ -multicromático de un árbol  $A$  es  $X_r(A) = 2r$

En estos casos, tanto si  $G$  es un grafo completo como si es bipartito, se obtiene para la expresión (4.1) un incremento del número  $r$ -multicromático según aumenta  $r$  tal que  $\Delta_r = 0 \forall r \geq 2$ .

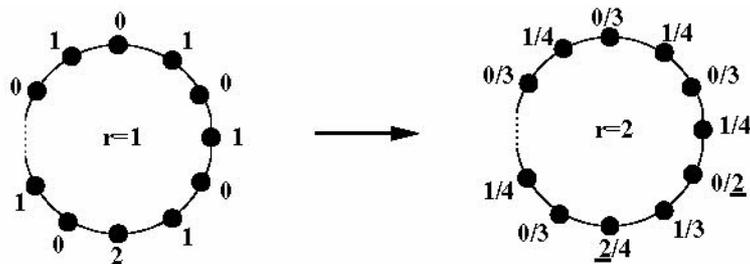


Figura 4.2: Ejemplo de ciclo de orden  $n$  con  $n$  impar.

**Proposición 4.9** Sea  $C_n$  un ciclo de orden  $n$ . Entonces

$$X_r(C_n) = \begin{cases} 2r & \text{si } n \text{ par} \\ 2r + \left\lceil \frac{2r}{n-1} \right\rceil & \text{si } n \text{ impar} \end{cases}$$

**Demostración.** En el caso en que el orden  $n$  es par el ciclo  $C_n$  es bipartito. El número cromático se deduce entonces de la proposición 4.7. De otro lado, para colorear un circuito con  $n$  vértices para  $n$  impar, se necesitan como mínimo 3 colores. Ahora bien, para 2-multicolorar este grafo sólo se necesitan 5 colores, dado que se puede aprovechar uno de los colores  $c$  del primer coloreado (supóngase que  $c$  sólo se ha empleado una vez en la primera asignación) en un vértice no adyacente y colorear los vértices restantes según el esquema del primer coloreado, con dos nuevos colores (figura 4.2). Este esquema podría repetirse para el tercer y cuarto coloreado (aprovechando siempre el color  $c$  del primer coloreado) y así sucesivamente hasta que no fuera posible reutilizar el color  $c$ . Como el conjunto independiente máximo de

$C_n$  tiene orden  $\frac{n-1}{2}$ , el color  $c$  puede repetirse un número de veces igual a  $\frac{n-1}{2}$ , pudiéndose reutilizar en suma en  $\frac{n-1}{2} - 1$  ocasiones. De este modo

$$X_1(C_n) = 3$$

$$X_2(C_n) = 5$$

$$X_3(C_n) = 7$$

hasta

$$X_{\frac{n-1}{2}}(C_n) = 3 + 2\left(\frac{n-1}{2} - 1\right) = n$$

Para el siguiente coloreado se necesitarían entonces 3 nuevos colores, dado que no podría utilizarse de nuevo el color  $c$  ni ninguno de los anteriores, repitiéndose en este punto el mismo proceso anterior. Así, en cada incremento de colores (es decir cada vez que pasemos de un  $r$ -multicoloreado a un  $r+1$ -multicoloreado) se necesitarán dos nuevos colores, excepto cuando se pasa de  $r = \frac{n-1}{2} \mathbb{1}(1 \in \phi)$  a  $r+1$ , donde el incremento es de 3 unidades. De este modo se deduce la siguiente cota del número cromático

$$X_r(C_n) \leq 2r + \left\lceil \frac{r}{(n-1)/2} \right\rceil = 2r + \left\lceil \frac{2r}{(n-1)} \right\rceil$$

De otro lado, teniendo en cuenta que para un ciclo  $C_n$  impar el orden del conjunto independiente máximo es  $S = (n-1)/2$ , puede obtenerse según (4.3) la siguiente cota inferior del número cromático de  $C_n$  con  $n$  impar

$$X_r(C_n) \geq \left\lceil \frac{m}{(n-1)/2} \right\rceil = 2r + \left\lceil \frac{2m}{(n-1)} \right\rceil$$

De este modo, combinando las dos cotas así obtenidas se tiene que

$$\left\lceil \frac{2m}{(n-1)} \right\rceil \leq X_r(C_n) \leq 2r + \left\lceil \frac{2r}{(n-1)} \right\rceil$$

Ahora bien, dado que  $r \in \phi$

$$\left\lceil \frac{2m}{(n-1)} \right\rceil = \left\lceil \frac{2r(n-1) + 2r}{n-1} \right\rceil = \left\lceil 2r + \frac{2r}{n-1} \right\rceil = 2r + \left\lceil \frac{2r}{n-1} \right\rceil$$

quedando finalmente

$$X_r(C_n) = 2r + \left\lceil \frac{2r}{(n-1)} \right\rceil = \left\lceil \frac{2rn}{(n-1)} \right\rceil \quad (4.4) \quad \blacksquare$$

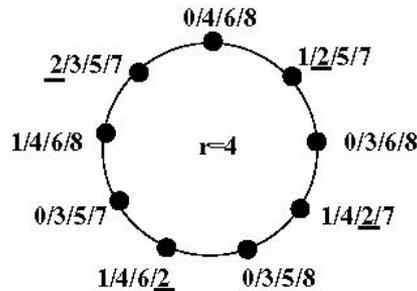


Figura 4.3: Ejemplo de 4-multicoloreado de un ciclo de orden 9.

La figura 4.3 muestra el esquema de multicoloreado descrito en la demostración anterior para el caso de un ciclo  $C_9$ . En este caso se cumplirían las siguientes desigualdades

$$\begin{array}{lll} X_1(C_9) \leq 3 & X_5(C_9) \leq 12 & X_9(C_9) \leq 21 \\ X_2(C_9) \leq 5 & X_6(C_9) \leq 14 & \dots \\ X_3(C_9) \leq 7 & X_7(C_9) \leq 16 & \\ X_4(C_9) \leq 9 & X_8(C_9) \leq 18 & \end{array}$$

En el caso de los ciclos impares puede observarse además que cuando  $r$  es múltiplo del orden  $S$  del conjunto independiente máximo,  $r = 1 \cdot S = 1 \cdot \frac{n-1}{2}$ , entonces  $X_r(C_n) = n \cdot 1$ . En estos casos es lo mismo  $r$ -multicolorear  $C_n$  que superponer 1 veces un  $\frac{n-1}{2}$ -multicoloreado, ya que  $X_{\frac{n-1}{2}}(C_n) = n$ .

**Teorema 4.10** Dado un grafo  $G$  de número  $r$ -multicromático  $X_r$ . Entonces, según la expresión 4.1 se cumple

$$X_r(G) = rk - \sum_{i=2}^r \Delta_i$$

donde  $\Delta_i$  puede ser mayor, igual o menor que  $\Delta_{i+1}$ .

**Demostración.** El sumatorio es el mismo que el de la expresión 4.1. De otro lado para demostrar que  $\Delta_i$  puede ser mayor, igual o menor que  $\Delta_{i+1}$  basta tomar el multicoloreado de  $C_9$  descrito en la figura 4.3, y calcular  $\Delta_i$  para cada valor  $r$ .

$$\begin{array}{ll}
X_1(C_9) \leq 3 & \\
X_2(C_9) \leq 5 & \Delta_2 = 1 \\
X_3(C_9) \leq 7 & \Delta_3 = 1 \\
X_4(C_9) \leq 9 & \Delta_4 = 1 \\
X_5(C_9) \leq 12 & \Delta_5 = 0 \\
X_6(C_9) \leq 14 & \Delta_5 = 1
\end{array}$$

De este modo se deducen las tres posibilidades descritas

$$\begin{array}{ll}
\Delta_i = \Delta_{i+1} & (\Delta_2 = \Delta_3) \\
\Delta_i > \Delta_{i+1} & (\Delta_4 > \Delta_5) \\
\Delta_i < \Delta_{i+1} & (\Delta_5 < \Delta_6)
\end{array}$$

Puede verse asimismo que para el ciclo  $C_n$  con  $n$  impar  $\Delta_i = 0$  para  $i = \frac{n+1}{2} \cdot 1$  y  $\Delta_i = 1$  en el resto de los casos. De este modo se tendrá, para  $i = \frac{n+1}{2} \cdot 1$  y para un ciclo de grado impar,  $\Delta_i > \Delta_{i\pm 1}$ .

### 4.1.3 Grafos ajustables y grafos perfectos

Se define un grafo ajustable como un grafo  $G$  tal que  $X(G) = c$  donde  $c$  es el orden del clique máximo de  $G$ . De otro lado, un grafo es  $r$ -ajustable si  $X_r(G) = rc$  ( $r \in \phi$ ). De esta forma, y según la expresión (4.1), si un grafo es  $r$ -ajustable entonces  $\Delta_i = 0 \quad \forall i \geq 2$ .

**Proposición 4.11** Si  $G$  es ajustable  $\Rightarrow G$  es  $r$ -ajustable ( $r \geq 1$ )

**Demostración.** Si  $G$  es ajustable  $\Rightarrow X(G) = c$ . Entonces, según (4.2) y considerando la proposición 4.1, resulta  $rc \leq X_r(G) \leq rc$  y por tanto  $X_r(G) = rc$ .

■

Se dice que un grafo  $G$  es perfecto si todo subgrafo inducido de  $G$  se puede colorear con un número de colores igual al orden del clique máximo del subgrafo inducido.

**Proposición 4.12** Todo grafo perfecto es, por definición, ajustable pero no todo grafo ajustable es perfecto.

La figura 4.4 ilustra un ejemplo de grafo ajustable no perfecto. Si en este grafo, que es ajustable (ya que se puede colorear con un número de colores igual al orden del clique máximo, en este caso 3), se elimina el vértice  $w$ , el subgrafo inducido resultante no es coloreable con un número de colores igual al orden del clique máximo (ahora igual a 2) y en consecuencia no es un grafo perfecto.

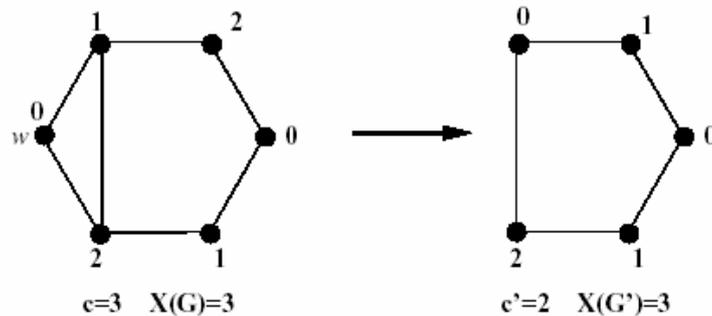


Figura 4.4: Ejemplo de grafo ajustable no perfecto.

La caracterización de un grafo  $r$ -ajustable tiene especial interés en el coloreado y multicoloreado de grafos dado que permite establecer cotas inferiores del número cromático.

## 4.2 Reutilización de colores en el multicoloreado de grafos

Según la proposición 4.1 es posible, en algunos casos,  $r$ -multicolorear un grafo con un número de colores menor que  $r$  veces el número cromático. Como se ha visto este ahorro se produce gracias a la reutilización de colores. En la figura 4.5 se ha 2-multicoloreado (o doblecoloreado) un grafo, aprovechando en la segunda asignación de algunos vértices colores del primer coloreado. Una vez coloreados estos vértices debe colorearse (para obtener un doblecoloreado final del grafo) el subgrafo inducido formado por los vértices restantes, es decir los vértices que no reutilizan colores. De esta forma, el 2-multicoloreado de un grafo  $G(V,E)$  puede dividirse en tres coloreados simples:

1. Coloreado de  $G$
2. Coloreado (compatible con el primero) de un subconjunto  $S \subset V$  de vértices de  $G$  reutilizando colores del primer coloreado.
3. Coloreado del subgrafo inducido resultante de eliminar los vértices de  $S$ .

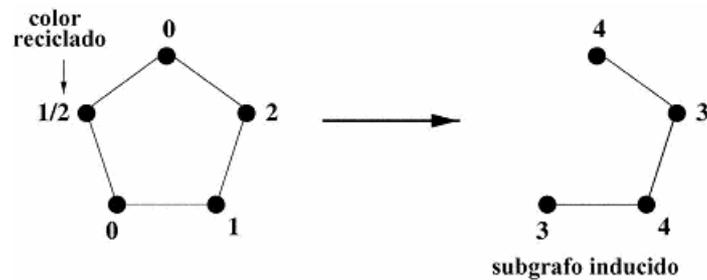


Figura 4.5: Ejemplo de subgrafo inducido tras reutilizar colores.

En cualquier caso, el proceso de reasignación y construcción del subgrafo inducido, así como su posterior coloreado, no es un problema sencillo. De este modo, el proceso anterior debe plantearse en términos tales que pueda ser resuelto algorítmicamente y aplicando los mismos métodos combinatorios descritos en los capítulos anteriores. En primer lugar, se sabe que la forma óptima de colorear el subgrafo inducido es un problema NP-completo, a pesar de lo cual se conocen herramientas algorítmicas capaces de obtener soluciones cuasi-óptimas, como Hormigas o Simulated Annealing. De otro lado, la reutilización en el segundo coloreado de colores del primer coloreado debería dar como resultado un subgrafo inducido lo más sencillo posible en términos del coloreado del grafo. De este modo, la construcción del subgrafo inducido debe considerar ciertos parámetros relacionados con la dificultad de coloreado, como por ejemplo la densidad del grafo o el grado medio. El subgrafo inducido dependerá además de la distribución de los colores del primer coloreado, de modo que distribuciones distintas (aun en el caso en que empleen el mismo número de colores) pueden dar lugar a subgrafos inducidos distintos.

En el presente apartado se describe un método algebraico de reutilización de colores, proponiéndose a continuación distintos esquemas de selección en función de la naturaleza del problema.

**Definición 4.13** Sea un grafo  $G(V,E)$  de orden  $n$  y un coloreado simple  $\Phi : V(G) \rightarrow \{1, \dots, k\}$  de  $G$ , con un número  $k$  de colores. Se define la matriz  $C$  de  $n$  filas y  $k$  columnas, donde las filas representan los  $n$  vértices de un grafo  $G$ , y las columnas los  $k$  colores utilizados en  $\Phi(G)$ , como la matriz de elementos  $c_{ij}$  tales que

$$c_{ij} = \begin{cases} 1 & \text{si ni el vértice } i \text{ ni ningún vértice adyacente a } i \text{ tienen el color } j \\ 0 & \text{en caso contrario} \end{cases}$$

De este modo, un valor  $c_{ij} = 1$  significa que puede cambiarse el color del vértice  $i$  por el color  $j$  (dado que ningún vértice vecino ni tampoco el mismo vértice  $i$  lo emplean) o más sencillamente que puede añadirse el color  $j$  al vértice  $i$  además del color inicial. A pesar de ello, debe notarse que no todos los cambios (o

adiciones) correspondientes a  $c_{ij}$  no nulos pueden realizarse simultáneamente, ya que de este modo podría asignarse el mismo color a vértices adyacentes.

**Definición 4.14** *Sea un grafo  $G(V,E)$  y sea  $\Phi(G)$  un coloreado simple de  $G$ . Se define  $V_i \subset V$  como el subconjunto de vértices  $v \in V$  tales que  $\Phi(v) = i$ , es decir el conjunto de vértices que tienen asignados según  $\Phi(G)$  el color  $i$  (de este modo los vértices de  $V_i$  no son adyacentes). Un subconjunto  $V_i$  es completo si las filas de  $C$  correspondientes a cada uno de los vértices de  $V_i$  son no nulas, es decir, si todos sus vértices pueden aprovechar algún color del primer coloreado.*

**Proposición 4.15** *Sea un grafo  $G(V,E)$  y sea  $\Phi(G)$  un coloreado simple de  $G$  con un número  $k$  de colores. Si existe para  $\Phi(G)$  un conjunto  $V_i$  completo entonces  $G(V,E)$  es  $(k-1)$ -coloreable.*

**Demostración.** *En primer lugar  $G$  es  $k$ -coloreable puesto que puede colorearse según  $\Phi(G)$  con un número  $k$  de colores. De otro lado, según la definición puede añadirse a cada vértice  $v \in V_i$  un nuevo color (que puede o no repetirse) sin posibilidad de conflicto, dado que los vértices de  $V_i$  no son adyacentes. Por lo tanto, si existe un  $V_i$  completo se puede reemplazar, con arreglo a la matriz  $C$ , el color original  $i$  de cada uno de los vértices de  $V_i$  por otro color de  $\Phi(G)$  y colorear  $G$  con  $k-1$  colores, es decir, todos los colores de  $\Phi(G)$  menos el color  $i$ .*

En este caso además  $G$  podrá ser 2-multicoloreado según la proposición 4.1 con un número de colores igual a  $2k-2$ .

De otro lado, en el caso en que más de un conjunto  $V_i$  del grafo  $G$  sea completo, debería comprobarse la existencia de incompatibilidades. Así, si por ejemplo se tienen dos conjuntos  $V_i$  y  $V_j$  completos como los de la 4.6 es posible que existan dos subconjuntos de vértices  $V'_i$  y  $V'_j$ , incluidos en  $V_i$  y  $V_j$  respectivamente, que contengan adyacencias entre ellos y que, por lo tanto, incluyan vértices que no puedan tener el mismo color asignado. En este caso, si según la matriz  $C$  no existe ningún coloreado compatible de los vértices de  $V'_i$  y  $V'_j$ , entonces los subconjuntos de vértices  $V_i$  y  $V_j$  se llaman incompatibles. Si, contrariamente son compatibles  $G$  podrá ser coloreado con  $k-2$  colores (los  $k$  colores de la asignación inicial menos los colores  $i$  y  $j$ ), y de este modo podrá 2-multicolorearse  $G$  con al menos  $2k-4$  colores. En general,  $G$  podrá colorearse con  $k-v$  colores, donde  $v$  es el número de conjuntos  $V_i$  completos y compatibles. Asimismo, en este caso será posible  $r$ -multicolorear  $G$  al menos con  $r(k-v)$  colores.

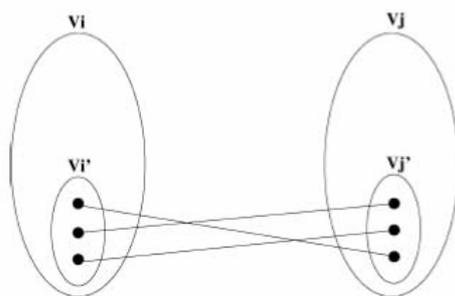


Figura 4.6: Ejemplo de subconjuntos  $V_i$  completos con vértices adyacentes.  $V_i$

Cuando el coloreado simple  $\Phi(G)$  emplea un número de colores  $k$  igual al número cromático no pueden darse casos de subconjuntos  $V_i$  completos. De otro modo  $k$  no podría ser el número cromático. Igualmente, cuando se trabaja con coloreados próximos a las cotas inferiores del número cromático, tampoco es sencillo encontrar subconjuntos  $V_i$  completos. Por otra parte, empleando en la práctica coloreados simples poco restrictivos (es decir coloreados con  $k+m$  colores, donde  $m \geq 1$ ), han podido encontrarse conjuntos  $V_i$  completos, si bien en ningún caso ha resultado que el número  $v$  de conjuntos  $V_i$  completos y compatibles fuera mayor que el número  $m$  de colores adicionales resultando en definitiva diseños con un número de colores igual a  $k+m-v \geq k$ . De este modo, no tiene sentido intentar obtener coloreados simples por el procedimiento de emplear diseños poco restrictivos y a continuación buscar conjuntos  $V_i$  completos, si bien no hay que olvidar que en algunos puede resultar más sencillo colorear un grafo  $G$  con  $k+1$  colores y encontrar a continuación un subconjunto  $V_i$  completo (problema que no reviste ninguna complicación y que es resoluble en un tiempo proporcional al cuadrado del orden del grafo) que colorear directamente  $G$  con  $k$  colores.

En el caso en que no existan subconjuntos  $V_i$  completos no se podrá colorear directamente el grafo con menos de  $k$  colores, a pesar de lo cual podrán reutilizarse colores del primer coloreado para así definir un subgrafo inducido que, según la proposición 4.1, pueda ser coloreado con menos de  $k$  colores. Así, si por ejemplo se consiguiera colorear el subgrafo inducido de  $G$  con  $k-1$  colores, se habría conseguido 2-multicolorear  $G$  con un número de colores igual a  $2k-1$ .

#### 4.2.1 Construcción del subgrafo inducido

En el presente apartado se describe un procedimiento para, dado un grafo  $G(V,E)$  y un coloreado simple  $\Phi(G)$ , construir un subgrafo inducido obtenido de eliminar de  $V$  los vértices que puedan emplear en el segundo coloreado alguno

de los colores del primer coloreado  $\Phi(G)$ . Para ello se parte de la matriz  $C$  definida en el apartado anterior.

Primero de todo, como los vértices correspondientes a filas nulas de la matriz  $C$  no pueden reutilizar ningún color, no serán objeto de estudio y por lo tanto las filas correspondientes pueden ser eliminadas. Se define entonces la matriz  $C'$  de  $n'$  filas y  $k$  columnas, resultado de eliminar las filas nulas de  $C$ .

A continuación, se construye una matriz  $D$  de  $n'$  filas y  $k$  columnas, donde cada elemento  $d_{ij}$  es el resultado de sumar a cada  $c'_{ij} \neq 0$  todos los  $c'_{kj}$  de la misma columna  $j$  que se corresponden con los vértices  $k$  adyacentes al vértice  $i$ , es decir

$$d_{ij} = c'_{ij} + c'_{ij} \sum_k c'_{kj} \quad \forall k \text{ adyacente a } i$$

De esta manera  $d_{ij}$  calcula el número de vértices que pudiendo reutilizar el color  $j$  son además adyacentes a  $i$ . La figura 4.7 muestra un ejemplo de construcción de la matriz  $D$ .

Una vez definida la matriz  $D$  se puede observar que si alguno de los elementos  $d_{ij}$  de una fila  $i$  es igual a 1, el vértice  $i$  puede recibir en un segundo coloreado el color  $j$ , ya que ningún vértice adyacente a  $i$  podrá reutilizar el color  $j$ . De este modo, todos los vértices  $i$  tales que  $d_{ij}=1$  podrán ser doble-coloreados con los respectivos colores  $j$  (es decir podrán recuperar el primer color de  $\Phi(G)$  más un segundo color  $j$ ).

$$C' = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \rightarrow \begin{cases} 0 \text{ adyacente a } 1 \\ 0 \text{ adyacente a } 2 \\ 0 \text{ adyacente a } 3 \\ 2 \text{ adyacente a } 3 \end{cases} \rightarrow D = \begin{pmatrix} 0 & 1 & 4 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 3 & 0 \end{pmatrix}$$

Figura 4.7: Ejemplo de construcción de la matriz  $D$ .

De otro lado, las filas que no contienen  $d_{ij}=1$  corresponden a vértices que presentan incompatibilidades con otros en el proceso de reutilizar los colores. Puede construirse entonces una nueva matriz  $D'$  resultante de eliminar las filas de  $D$  que contienen algún 1, y de actualizar los elementos  $d_{ij}$  de las filas restantes según las nuevas incompatibilidades (aquellos vértices que tengan asignado un segundo color ya no requerirán otros colores y por lo tanto podrá restarse una unidad en los vértices adyacentes a  $i$  que requirieran un mismo color  $j$  que  $i$ ). El hecho de actualizar los elementos  $d_{ij}$  de las filas restantes, puede provocar la aparición de nuevos elementos igual a 1 en  $D'$ . En este caso,

se ha de repetir el proceso anterior hasta que no queden 1's en la matriz  $D'$ , tal y como muestra la figura 4.8.

$$D = \begin{pmatrix} 0 & 1 & 4 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 3 & 0 \end{pmatrix} \rightarrow D' = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 \end{pmatrix} \rightarrow D'' = \begin{pmatrix} 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 \end{pmatrix}$$

Figura 4.8: Ejemplo de construcción de la matriz  $D'$  con las adyacencias de la figura 4.7

Una vez la matriz  $D'$  sólo tiene elementos nulos o mayores que uno, debe decidirse la combinación de colores que, dada la información contenida en  $D'$ , puede asignarse a los vértices del grafo. Para cada vértice  $i$  presente en  $D'$  existen varias posibilidades: asignar cualquiera de los colores posibles, correspondientes a elementos  $d'_{ij}$  no nulos, o no asignar ningún color. Si no se asigna ningún color, los colores requeridos por el vértice  $i$  quedan libres para otros vértices adyacentes. En el caso de la figura 4.8 se puede observar que los dos vértices representados, que son adyacentes, pueden ser coloreados con el mismo color. En cualquier caso, debe escogerse un coloreado tal que el subgrafo inducido sea simple de colorear (ya que no todos los vértices presentes en la matriz  $D'$  podrán finalmente reutilizar un color del primer coloreado). Esta combinación óptima o casi óptima puede llevarse a cabo de diversas formas:

1. Aplicando un algoritmo combinatorio.
2. Aplicando un algoritmo secuencial.
3. De forma exhaustiva, es decir, examinando todas las combinaciones posibles.

### Algoritmo combinatorio

El subgrafo de los vértices representados en la matriz  $D'$  puede colorearse mediante un algoritmo combinatorio como Hormigas o Simulated Annealing. En este caso, cada vértice tendría restringido el grupo de colores que puede utilizar con arreglo a la información contenida en la matriz  $D'$ . Esta solución no explora todos los posibles coloreados, pudiendo en consecuencia no calcular el mejor subgrafo inducido. En todo caso, su aplicación es aconsejable en aquellos casos en que el número de filas de la matriz  $D'$  sea elevado.



El proceso de evaluación de la función de coste se repite a continuación con los vértices restantes. A continuación se vuelve a eliminar el vértice con coste mínimo y así sucesivamente mientras queden filas en la matriz.

### Algoritmo exhaustivo

El algoritmo exhaustivo busca el coloreado de los vértices de  $D'$  que conduzca al subgrafo inducido más favorable según uno o varios de los siguientes criterios:

1. Minimizar el número de vértices del subgrafo inducido (es decir, maximizar el número de vértices de  $D'$  coloreados).
2. Minimizar el número de aristas del subgrafo inducido.
3. Minimizar la densidad del subgrafo inducido.

En principio, y dado que la dificultad de colorear un grafo está relacionada con la densidad, el tercer criterio puede parecer el más conveniente. Ahora bien, si se escoge como criterio único la densidad, el algoritmo tiende a aumentar el número de vértices del subgrafo inducido a cambio de poder minimizar su densidad, añadiendo así vértices "redundantes" (de grado reducido) que en realidad no facilitan el coloreado del subgrafo, a pesar de disminuir la densidad. De este modo, se ha optado por escoger el coloreado que conduce al subgrafo inducido con mínima densidad del subgrafo de entre los que tienen mínimo número de vértices. De otro lado, dado que para un número fijo de vértices es lo mismo minimizar el número de aristas totales del subgrafo que minimizar la densidad, se ha optado por escoger el subgrafo con mínimo número de aristas, de entre los que tienen mínimo número de vértices.

Para analizar todos los posibles coloreados de los vértices de  $D'$ , el algoritmo ha de ser recursivo, de tal modo que evalúe la función de coste para cada uno de los coloreados, quedándose con la mejor combinación en cada iteración.

#### 4.2.2 Algoritmo de reciclado

Reuniendo lo descrito en los apartados anteriores se ha definido un algoritmo de reciclado capaz de 3-multicolorear (y más genéricamente de  $r$ -multicolorear) un subconjunto de vértices de un determinado grafo, reutilizando para ello colores incluidos en un 2-multicoloreado. El funcionamiento del algoritmo se resume en la figura 4.10.

En primer lugar, el algoritmo de reciclado construye la matriz  $C$  a partir de un coloreado doble  $\Phi(G)$  de  $G$ . A continuación se comprueba si existe algún subconjunto  $V_i$  completo. Si existe algún subconjunto  $V_i$  completo o varios subconjuntos  $V_i$  completos compatibles, se modifica el color de los vértices de

los distintos subconjuntos  $V_i$  obteniéndose un nuevo coloreado más restrictivo que el anterior y reconstruyendo la matriz  $C$  con arreglo al nuevo coloreado.

```

Inicio
  Construir matriz  $C$ 
    si (no existen subconjuntos de  $V_i$  completos)
      construir matriz  $C'$ 
      construir matriz  $D$ 
      construir matriz  $D'$ 
    fin si
  guardar vértices coloreados
  construir subgrafo inducido
  colorear subgrafo inducido

```

Figura 4.10: Algoritmo de reciclado.

A continuación se construyen las matrices  $C'$ ,  $D$  y  $D'$  a partir de la matriz  $C$ . Una vez refinada la matriz  $D'$  se colorean los vértices correspondientes con el método exhaustivo o con el algoritmo secuencial. Finalmente se construye el subgrafo inducido (de eliminar de  $G$  los vértices con un tercer color asignado) y se colorea con las técnicas algorítmicas descritas en los capítulos anteriores.

### 4.2.3 Algoritmo rcy para el 3-coloreado

Como se ha comentado al principio, el objetivo de este TFC es encontrar una serie de algoritmos que permitan conseguir un tercer coloreado a partir de la reasignación de frecuencias previamente definidas en la red. Es decir, obtener un grado más a partir del anterior. De hecho, el procedimiento es el mismo que el de la figura 4.10 pero matizando y redefiniendo algunos de los pasos internos en la programación de las matrices. Por lo tanto, podríamos generalizar este esquema para cualquier  $r$ -coloreado considerando los siguientes aspectos:

- Comprobar que cada vértice ha sido previamente coloreado con  $r-1$  colores.
- Guardar todos aquellos vértices que puedan ser reutilizados, a partir de la construcción de las matrices  $C$ ,  $C'$  y  $D$ .
- Construir la matriz  $D'$  para poder aislar aquellos vértices que no se pueden aprovechar directamente un color.
- Aplicar el algoritmo secuencial o el exhaustivo para colorear el resto de vértices posibles de tal manera que el subgrafo inducido tenga el menor número de ramas de entre los que tienen un mínimo número de vértices.
- Guardar estos vértices con el color reciclado.

## Capítulo 5. Asignación múltiple de frecuencias en redes GSM

### Introducción

Las redes GSM pueden modelarse por grafos cuyas aristas están etiquetadas con números enteros (en algunos casos números reales), correspondiéndose el peso o etiqueta de cada arista con los elementos de la matriz de restricciones. De otro lado cada celda trabaja normalmente con más de una portadora de modo que el proceso de asignación frecuencial se puede plantear en términos de un multicoloreado de grafos. De este modo, y así como en el capítulo anterior se ha traducido el problema de asignación múltiple en un problema simple de coloreado a través del desdoblamiento de cada celda en un número determinado de vértices, en el presente capítulo se ha resuelto el problema de modo directo, es decir, sin transformar el grafo inicial y trabajando con vértices a los cuales debe asignarse más de una frecuencia o color. Al final del capítulo se comparan ambos procedimientos.

### 5.1 Asignación de frecuencias mediante coloreado de grafos

Dado que el grafo que representa una red GSM tiene las aristas etiquetadas deberán en primer lugar redefinirse (si bien sólo de forma puntual) algunos de los operadores descritos en el apartado anterior. De los mecanismos estudiados deben considerarse los siguientes:

- El primer coloreado del grafo.
- La repetición de un coloreado  $r$  veces.
- La reutilización de colores.
- El coloreado del subgrafo inducido.

En cuanto al coloreado simple del grafo se seguirán los mismos procedimientos descritos en los capítulos anteriores. En el caso de repetir  $r$  veces un coloreado de  $k$  colores para obtener así un  $r$ -multicoloreado de un grafo  $G$  con pesos en las aristas, no se puede asegurar que el número total de colores utilizados sea  $rk$ . Es decir, si el primer coloreado va de 1 a  $k$  colores, el segundo de  $k+1$  a  $2k$  colores y así sucesivamente, puede suceder que vértices que tienen asignado el color  $1k$  (del coloreado 1) sean adyacentes a vértices que tienen asignado el color  $1k+1$  (del coloreado 1+1), y que la arista adyacente a ambos vértices tenga un peso  $p > 1$ , de modo que el multicoloreado final no sería correcto. De este modo, es necesario introducir un desplazamiento  $d$  entre los coloreados tal que  $d = p_m - 1$  donde  $p_m$  es el peso máximo de alguna de las aristas del grafo.

De esta manera, el primer coloreado irá del color 1 al color  $k$ , el segundo de  $k+1+d$  a  $2k+d$ , el tercero de  $2k+1+2d$  a  $3k+2d$  y así sucesivamente. En el presente proyecto, no obstante se han considerado inicialmente grafos con ramas de peso 1.

De otro lado, en el caso en que se multicoloree el grafo repitiendo el primer coloreado hay que considerar que el grafo no modela los elementos de la diagonal principal de la matriz de restricciones, que corresponden a las restricciones entre colores del mismo vértice.

Respecto a la reutilización de colores de un primer coloreado, el algoritmo de reciclado ha de tener en cuenta que:

- Al construir la matriz  $C$  los colores de un mismo vértice han de mantener una distancia mínima igual al elemento correspondiente de la diagonal principal de la matriz de restricciones. Asimismo los colores de vértices adyacentes han de mantenerse a una distancia mayor o igual al peso de la arista que los une.
- Al construir la matriz  $D$ , cada  $d_{ij}$  ha de contabilizar los elementos  $c_{ij}$  de vértices adyacentes de la misma columna y de columnas adyacentes hasta una distancia  $p-1$ , donde  $p$  es el peso que une los vértices. En este caso, cada  $d_{ij}$  refleja el número de vértices adyacentes que requieren un color del intervalo  $([j-(p-1), j+(p-1)])$ .
- Al construir y actualizar la matriz  $D'$ , cuando se elimina una fila y se libera un determinado color  $k$ , el cambio afecta a los vértices adyacentes que requieren un color del intervalo  $[j-(p-1), j+(p-1)]$ .

Finalmente, se ha implementado un método para 3-multicolorear grafos (que puede extrapolarse sin dificultades al caso de  $r$ -multicoloreado) que integra los algoritmos descritos en los apartados anteriores, es decir: algoritmos para el coloreado simple de grafos (Hormigas y Simulated Annealing), algoritmos de reutilización de colores (desarrollados a partir de la matriz  $D'$ ) y algoritmo de generación de subgrafos inducidos.

## 5.2. Implementación

A continuación se describen los algoritmos que se han utilizado en este TFC para 3-multicolorear grafos. Cada uno de estos algoritmos ha sido implementado de forma independiente para facilitar la ejecución final. Por lo tanto, la estructura global del procedimiento, según el orden de ejecución, es la que se describe en la figura 5.1.

Todos los módulos se han integrado mediante el intercambio de una serie de ficheros de datos según el procedimiento de la figura. En primer lugar, el algoritmo Hormigas o Simulated Annealing realizan el coloreado simple de  $G$ . Una vez se tiene el primer coloreado, el algoritmo de reciclado RCY asigna un tercer color a una serie de vértices presentes en la matriz  $D'$ , reutilizando para ello colores del doble coloreado inicial. A continuación MAKE\_NET genera el subgrafo inducido formado por los vértices que no han sido triplecoloreados. Finalmente el algoritmo JOIN une los distintos coloreados obtenidos formando así un 3-multicoloreado de  $G$ .

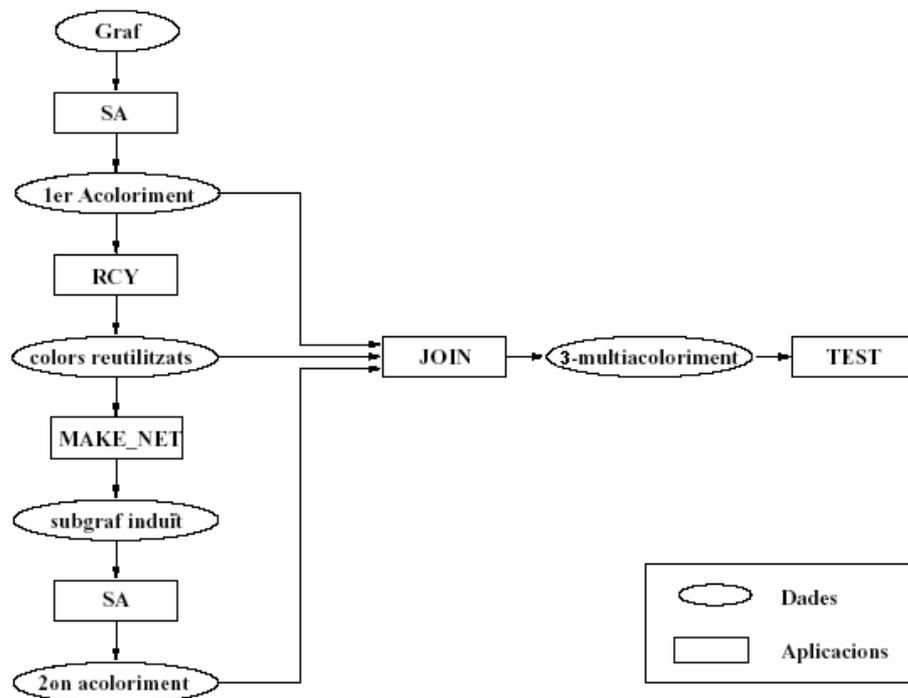


Figura 5.1: procedimiento para el 3-multicolorado de un grafo.

El algoritmo JOIN une de este modo el primer coloreado doble de  $G$ , un segundo coloreado (simple) parcial del mismo grafo reutilizando colores del primero y el coloreado del subgrafo inducido formado por los vértices restantes, para dar lugar al 3-multicolorado de  $G$ . Hormigas y Simulated Annealing colorean así en primer lugar  $G$  con  $k$  colores que van del 0 al  $k-1$ , además del subgrafo inducido con  $k'$  colores del 0 al  $k'-1$ . De este modo, antes de unir los dos coloreados el algoritmo suma  $k$  unidades a cada uno de los colores del subgrafo inducido de manera que este comience con un color  $k$  y termine con el color  $k+k'$ . De otro lado, si quisiera superponerse  $r$  veces el primer coloreado se ha de tener en cuenta la distancia  $d = p_m - 1$  entre coloreados.

Este conjunto de aplicaciones puede ampliarse al caso de  $r$ -multicolorados para  $r > 3$  si en vez de partir de un primer coloreado doble se dispone de un  $r-1$  coloreados, cuyos colores pueden ser reutilizados en la asignación del  $r$ -ésimo color del grafo. Otra posibilidad consistiría en combinar  $r-1$  colores simples, proceso que se ilustra en la figura 5.2 y que no puede asegurar un  $r$ -multicolorado óptimo ya que los conjuntos de vértices eliminados para cada uno de los primeros coloreados pueden ser no disjuntos o no estar repartidos de forma óptima (en el sentido de proporcionar un subgrafo inducido de orden y tamaño mínimos).

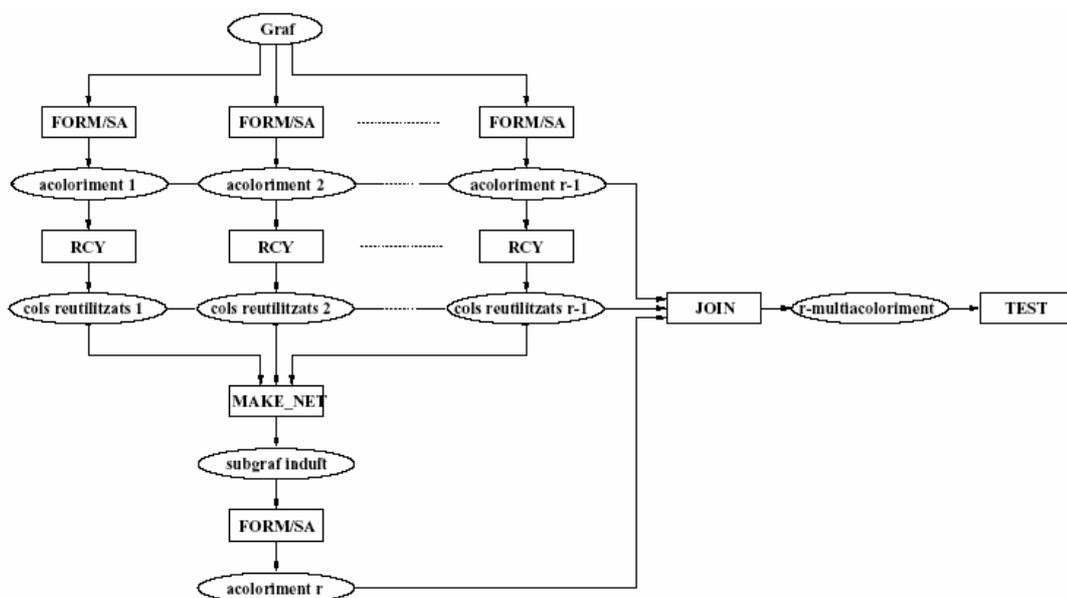


Figura 5.2: Procedimiento para el  $r$ -multicolorado de un grafo.

El ejemplo de la figura 5.3 desarrolla el 4-multicolorado del ciclo  $C_9$ , cuyo número multicromático es  $X_4(C_9) = 9$ . Puede así observarse que el número de colores necesarios para multicolorar  $C_9$  según el esquema de la figura 5.3 depende de la distribución de los primeros coloreados. De este modo, si los primeros diseños son los mismos, los colores reutilizables de cada uno de ellos caen siempre en los mismos vértices. Esta coincidencia impide que el multicolorado de  $C_9$  sea óptimo. Por otra parte, si los primeros coloreados de  $C_9$  fueran diferentes y, además, los colores reutilizados en cada uno de los diseños recayeran sobre vértices distintos, entonces el multicolorado de  $C_9$  podría ser óptimo.

De otro lado, cuando se trabaja con grafos de una cierta complejidad, como los que modelan redes GSM, es difícil determinar la configuración de los coloreados simples obtenidos mediante algoritmos combinatorios. De este modo, en general no podrá garantizarse en estos casos una correcta redistribución de los vértices que reutilizan colores, en el sentido de obtener subconjuntos lo más disjuntos posibles y disminuir así el orden y tamaño del subgrafo inducido resultante.

Es posible asimismo como se ha dicho obtener  $r$ -multicolorados de grafos a partir de la superposición de multicolorados de grado menor que  $r$ . De este modo, puede determinarse un 11-multicolorado de un grafo  $G$  a través de la superposición de un 6-multicolorado y un 4-multicolorado (o dos 5-multicolorados) y la reutilización de colores en un determinado conjunto de vértices (lo mayor posible). Podría generarse así un subgrafo inducido (formado por los vértices que no reutilizarán ningún color) que, una vez coloreado y

superponiendo convenientemente el 6-multicolorado, el 4-multicolorado y el coloreado simple del subgrafo inducido, devolviera un 11-multicolorado de  $G$ .

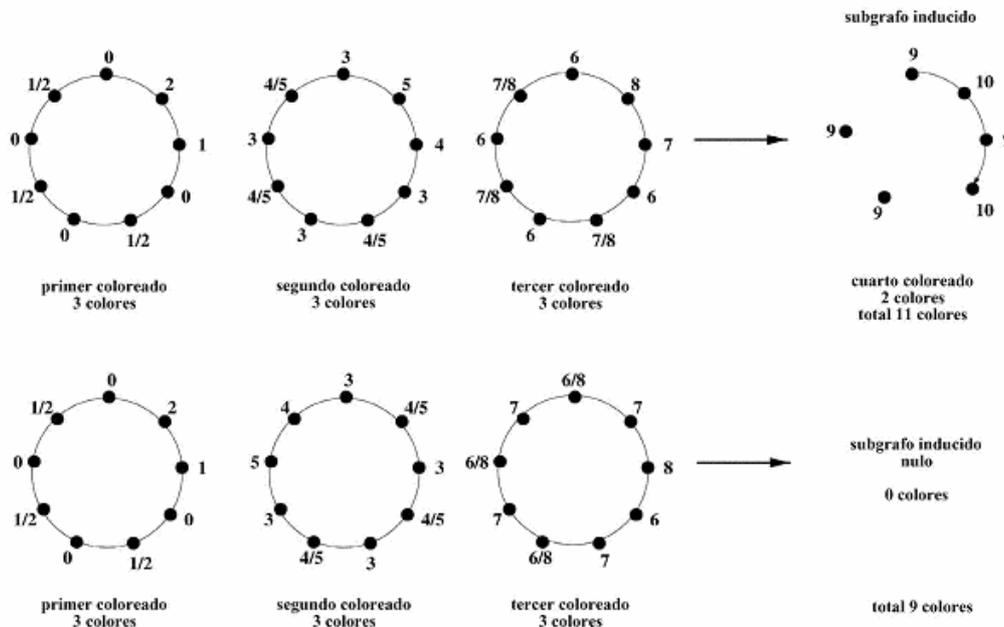


Figura 5.3: Dos ejemplos de 4-multicolorado de un ciclo de orden 9. En cada uno de los coloreados parciales se indica, en los vértices correspondientes y separado por una barra, el color reutilizado en el cuarto coloreado.

### 5.3 Resultados y comparaciones

A continuación se exponen los resultados de las simulaciones realizadas durante el curso de este proyecto. Han sido simulaciones aplicadas sobre diferentes redes GSM proporcionadas por la compañía Vodafone y que, previamente, ya habían sido estudiadas para el caso del doble-coloreado. Por lo tanto, en este apartado se presentan los resultados de un 3-multicolorado, bien realizado directamente con uno de los dos algoritmos combinatorios vistos anteriormente (en nuestro caso el algoritmo de Simulated Annealing), bien reciclando colores de los dos primeros coloreados.

Posteriormente se han tomado otras redes aleatorias con un número de celdas elevado para estudiar el comportamiento de estos algoritmos en casos más generales. Hay que decir que las redes se han modelado teniendo en cuenta sólo las interferencias cocanal, por lo que los grafos estudiados no tienen pesos en las ramas.

Por simplicidad, se ha supuesto el caso en que cada celda requiere tres frecuencias aunque, como se ha visto, las técnicas descritas son extrapolables al caso genérico en que cada celda requiere un número variable de frecuencias. Dado que el principal objeto del presente proyecto es la posibilidad

de reutilizar colores, en las pruebas se ha empleado el algoritmo Simulated Annealing –en detrimento de Hormigas– debido a su mayor velocidad de convergencia.

### 5.3.1 Redes GSM reales

En primer lugar, representamos la cota inferior obtenida en el proyecto anterior mediante el algoritmo *Gusanos* (ver [10]).

	celdas	Cota (un color)	Cota (tres colores)
Red 1	43	13	<b>39</b>
Red 2	45	15	<b>45</b>
Red 3	90	16	<b>48</b>
Red 4	99	15	<b>45</b>
Red 5	315	13	<b>39</b>

Tabla 5.1: Cota inferior para las redes GSM reales

A continuación, en la tabla 5.2 vemos los resultados obtenidos para nuestras simulaciones realizando directamente un tercer coloreado (es decir, sin reciclar) y realizando un primer doble-coloreado, un reciclado, un nuevo coloreado simple y la unión de ambos.

	celdas	RECICLANDO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Red 1	43	27	14	<b>41</b>	<b>41</b>
Red 2	45	30	15	<b>45</b>	<b>45</b>
Red 3	90	33	16	<b>49</b>	<b>50</b>
Red 4	99	32	15	<b>47</b>	<b>48</b>
Red 5	315	27	13	<b>40</b>	<b>42</b>

Tabla 5.2: Resultados de simulación de redes GSM reales durante un periodo de 20 horas.

Para obtener unos resultados coherentes –es decir, ejecutados bajo las mismas condiciones de simulación– se realiza para cada red el triple-coloreado durante 20 horas. Para el caso de un primer doble-coloreado y un coloreado simple debemos emplear en conjunto el mismo tiempo. De esta forma, y teniendo en cuenta que el algoritmo es más lento cuantos más colores ha de asignar, empleamos 14 horas para el doble coloreado y 6 horas para el coloreado simple. En este caso, hemos considerado despreciable el tiempo utilizado en los algoritmos RCY, MAKE NET y JOIN ya que es muy pequeño frente a las 20 horas de proceso total (la ejecución de los tres no ocupa más de 5 minutos de cálculo).

En la tabla anterior se observa que el algoritmo encuentra coloreados cercanos a la cota inferior (o clique máximo hallado por el algoritmo *Gusanos*). En particular, se consigue coloreado igual a la cota inferior en la red 2 (siendo por

lo tanto ajustable el grafo que modela la red). En las tres últimas redes, además, el algoritmo de reciclado ha necesitado un número menor de frecuencias que los métodos directos. De otro lado, cuanto más cerca esté de la cota inferior el coloreado del grafo, más difícil será la reutilización de colores y por eso los resultados que se obtienen reciclando y sin reciclar no presentan grandes diferencias.

### 5.3.2 Redes generadas aleatoriamente

En este caso realizamos diferentes pruebas con redes aleatorias. Estas redes pueden tener diferentes probabilidades de interferencia en la matriz de restricciones, de forma que a mayor probabilidad más 1's tendremos en la matriz, y por lo tanto, mayor será el grado de restricciones. Nos hemos basado en redes de 500 y 300 celdas y tres probabilidades de interferencia entre celdas: 0.1, 0.25 y 0.5. Primero simulamos dos redes para cada una de las tres probabilidades diferentes y para un tiempo de 20 horas tanto para el triple-coloreado como para el doble-coloreado y coloreado simple con reciclado. Los resultados son los siguientes:

	celdas	CON RECICLADO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Probabilidad 0,5	500	105	41	<b>146</b>	<b>149</b>
	500	100	48	<b>148</b>	<b>150</b>
Probabilidad 0,25	500	51	26	<b>77</b>	<b>79</b>
	500	50	25	<b>75</b>	<b>76</b>
Probabilidad 0,1	500	25	12	<b>37</b>	<b>37</b>
	500	24	12	<b>36</b>	<b>36</b>

Tabla 5.3: Resultados para redes aleatorias de 500 celdas en un tiempo de 20 horas

En la tabla 5.3 se observa que para periodos largos de simulación los resultados son mejores empleando el método de reciclado, si bien las diferencias no son muy significativas. En las redes con baja probabilidad de interferencia los resultados son iguales para ambos mecanismos.

En las tablas siguientes se muestran pruebas similares desarrolladas, no obstante, sobre un número mayor de redes y un tiempo de ejecución de 30 minutos, con el fin de observar las mejoras que se pueden deducir del hecho de no disponer de periodos prolongados de ejecución.

	celdas	CON RECICLADO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Red 1	500	26	12	<b>38</b>	<b>37</b>
Red 2	500	25	13	<b>38</b>	<b>38</b>
Red 3	500	26	12	<b>38</b>	<b>38</b>
Red 4	500	25	13	<b>38</b>	<b>37</b>
Red 5	500	26	12	<b>38</b>	<b>38</b>

Tabla 5.4: Resultados en un tiempo de ejecución de 30 minutos para redes aleatorias de 500 celdas con una probabilidad de 0.1

Según los resultados de la tabla 5.4 no resulta rentable emplear mucho tiempo de ejecución en redes con pocas interferencias ya que con sólo 30 minutos se obtienen prácticamente los mismos resultados de la tabla 5.3.

	celdas	CON RECICLADO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Red 1	500	57	23	<b>80</b>	<b>111</b>
Red 2	500	58	23	<b>81</b>	<b>109</b>
Red 3	500	66	20	<b>86</b>	<b>114</b>
Red 4	500	61	22	<b>83</b>	<b>108</b>
Red 5	500	61	22	<b>83</b>	<b>111</b>

Tabla 5.5: Resultados en un tiempo de ejecución de 30 minutos para redes aleatorias de 500 celdas con una probabilidad de 0.25

	celdas	CON RECICLADO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Red 1	500	131	40	<b>171</b>	<b>223</b>
Red 2	500	137	33	<b>170</b>	<b>223</b>
Red 3	500	139	34	<b>173</b>	<b>218</b>
Red 4	500	143	31	<b>174</b>	<b>227</b>
Red 5	500	136	37	<b>173</b>	<b>227</b>

Tabla 5.6: Resultados en un tiempo de ejecución de 30 minutos para redes aleatorias de 500 celdas con una probabilidad de 0.5

En las tablas 5.5 y 5.6, por el contrario, puede apreciarse el mejor comportamiento del algoritmo de reciclado en comparación con Simulated Annealing cuando se dispone de periodos cortos de simulación y la probabilidad de interferencia es considerable. La mejora es de aproximadamente un 25% en promedio. Si comparamos estas dos tablas con la tabla 5.4 se observa que cuando las interferencias superan un determinado grado el algoritmo de reciclado es más eficiente. En las tablas siguientes se

repiten las mismas pruebas –con un tiempo de simulación igualmente reducido a treinta minutos– para redes aleatorias de 300 celdas.

	celdas	CON RECICLADO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Red 1	300	18	9	<b>27</b>	<b>25</b>
Red 2	300	17	9	<b>26</b>	<b>26</b>
Red 3	300	17	9	<b>26</b>	<b>26</b>
Red 4	300	18	9	<b>27</b>	<b>26</b>
Red 5	300	18	9	<b>27</b>	<b>26</b>

Tabla 5.7: Resultados en un tiempo de ejecución de 30 minutos para redes aleatorias de 300 celdas con una probabilidad de 0.1

Los resultados de la tabla 5.7 son muy similares a los de la tabla 5.4, con la lógica diferencia de que el número de colores empleados es menor debido a la cantidad diferente de celdas. En este caso, además, y dado que las redes son más simples (tienen un menor número de celdas), los diseños con un coloreado directo son mejores que los que emplean el reciclado, tanto para una probabilidad de interferencia de 0.1 como para 0.25 (tabla 5.8). Sin embargo cuando la probabilidad aumenta a 0.5 otra vez los resultados obtenidos de aplicar el algoritmo de reciclado son mejores.

	celdas	CON RECICLADO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Red 1	300	35	18	<b>53</b>	<b>52</b>
Red 2	300	35	17	<b>52</b>	<b>52</b>
Red 3	300	35	18	<b>53</b>	<b>52</b>
Red 4	300	34	17	<b>51</b>	<b>51</b>
Red 5	300	35	18	<b>53</b>	<b>51</b>

Tabla 5.8: Resultados en un tiempo de ejecución de 30 minutos para redes aleatorias de 300 celdas con una probabilidad de 0.25

	celdas	CON RECICLADO			SIN RECICLAR
		coloreado doble	coloreado simple	total	triple coloreado
Red 1	300	69	33	<b>102</b>	<b>115</b>
Red 2	300	69	34	<b>103</b>	<b>110</b>
Red 3	300	68	34	<b>102</b>	<b>109</b>
Red 4	300	69	34	<b>103</b>	<b>110</b>
Red 5	300	69	35	<b>104</b>	<b>111</b>

Tabla 5.9: Resultados en un tiempo de ejecución de 30 minutos para redes aleatorias de 300 celdas con una probabilidad de 0.5

De este modo, en general, cuanto mayor es el número de celdas, mayor es el número de interferencias y menor es el período de simulación –es decir, cuanto mayor es la complejidad del problema– más eficaz resulta el mecanismo de reciclado de colores. Por tanto, el método descrito en el presente proyecto resulta conveniente en el diseño frecuencial de redes GSM, que precisamente responden al modelo de gran número de celdas e interferencias descrito en las tablas anteriores.

## Capítulo 6. Conclusiones

El principal objetivo de este proyecto ha sido la asignación de frecuencias en redes GSM, problema que tiene una importancia considerable en telefonía móvil debido al alto coste de utilización del espectro radioeléctrico. Para ello se ha partido de una serie de algoritmos capaces de colorear redes simples: tanto el algoritmo Hormigas como Simulated Annealing –que se han empleado en otros proyectos para la asignación de frecuencias– pueden ser aplicados, con leves modificaciones, para asignar más de una frecuencia en cada una de las celdas de una red, aunque en estos casos el tiempo de ejecución de los algoritmos aumenta considerablemente. Es por ello que la parte central de este proyecto ha consistido en la redefinición de un algoritmo determinista y de ejecución rápida, llamado RCY, para la asignación múltiple de frecuencias en redes GSM, aplicado en combinación con el algoritmo Simulated Annealing.

En el capítulo de resultados se ha visto que el algoritmo RCY ofrece buenos resultados para intervalos cortos de simulación. Por el contrario, cuando el período de ejecución aumenta los resultados obtenidos mediante un triple-coloreado directo o mediante un doble-coloreado y un reciclado con RCY son prácticamente los mismos. Se ha observado que la mayor eficiencia de reciclado se obtiene con las redes de mayor número de celdas y con probabilidades de interferencia elevadas. Por otro lado, a medida que aumenta el número de frecuencias asignadas en cada celda aumenta el número total de frecuencias necesarias para evitar las interferencias, ya que es mucho más complicado reutilizar las mismas frecuencias en celdas distintas.

De este modo, cabe remarcar la eficiencia del algoritmo de reciclaje dado que, aunque precisa de un número mayor de algoritmos para multicolorear las redes (como pueden ser TEST, JOIN o MAKE NET, además del algoritmo Simulated Annealing), obtiene los mismos valores de forma mucho más rápida, sobre todo cuando el número de celdas es elevado.

La reducción del número de frecuencias permite, de esta forma, aprovechar mejor el espectro radioeléctrico y dar servicio a un mayor número de clientes en igualdad de recursos, disminuyendo el volumen de interferencias. Dicha reducción puede tener también una traducción en el número de equipos de transmisión y por lo tanto en la disminución de los costes. Todo ello, por otro lado, es independiente de la emisión de potencia de las estaciones base –y por lo tanto de los posibles efectos que las radiaciones puedan tener en la salud del consumidor– ya que en el presente proyecto se ha tratado de minimizar el número de portadoras comunes en celdas contiguas y no la emisión total de señales, parámetro que interesa de hecho maximizar con el objeto de dar servicio al mayor número posible de usuarios.

## 6.1 Líneas futuras

En el presente proyecto, y por motivos de restricción de tiempo, se han efectuado pruebas sobre diseños frecuenciales con tres colores por celda. De este modo cabe la posibilidad de ampliar estas pruebas para coloreados más amplios y programar así una rutina general para reciclar las frecuencias y realizar un  $r$ -coloreado a partir de coloreados más simples. También puede plantearse la implementación del sistema para celdas con un número diferente de frecuencias cada una.

Por otra parte, se puede aprovechar este proyecto para estudiar otras redes GSM, conocidas como benchmarks, o nuevos modelos aleatorios, incluso establecer estos mismos casos estudiando el comportamiento de las redes con pesos en las ramas. Igualmente, se podría plantear el problema desde un punto de vista más teórico, enfatizando los capítulos relacionados con los grafos ajustables y  $r$ -ajustables y con los  $r$ -multicoloreados y  $(r+1)$ -multicoloreados en función de las propiedades topológicas –algunas de las cuales podrían determinarse en el curso del propio proyecto– de las redes estudiadas.

## BIBLIOGRAFÍA

- [1] E. Aarts, J. K. Lenstra (ed.), *Local Search in Combinatorial Optimization*, John Wiley & Sons, Chicester, 1997.
- [2] E. Aarts, J. Korst, *Simulated Annealing and Boltzmann Machines*, John Wiley & Sons, Chicester, 1989.
- [3] J. Abril, F. Comellas, A. Cortés, J. Ozón, M. Vaquer, A multi-agent system for frequency assignment in cellular radio networks, *IEEE Trans. Vehic. Tech.*, vol. 49 (No. 5), pp.1558-1565, 2000.
- [4] S.M. Allen, D.H. Smith, S. Hurley, Lower bounding techniques for frequency assignment, *Discrete Mathematics*, vol. 197/198, pp. 41-52, 1999.
- [5] A. Colorni, M. Dorigo, V. Maniezzo, Distributed optimization by ant colonies, *Proceedings of the first European Conference on Artificial Life Paris*, pp. 134-142, ed. by F.J. Varela and P. Bourguine, MIT-Press-Bradford Books, Massachusetts, 1991.
- [6] F. Comellas, J. Ozón, Graph Couloing Algorithms for Assignment Problems in Radio Networks, *Proceedings of the International Workshop on Aplications of Neural Networks to Telecommunications*, pp. 49-56, Lawrence Erlbaum Associates Publishers, 1995.
- [7] D. Costa, A. Hertz, Ants can colour graphs, *Journal of the Operational Research Society*, vol.48, pp. 295-305, 1997.
- [8] A. Gibbons, *Algorithmic Graph Theory*, Cambridge University Press, Cambridge, 1985.
- [9] J. Ozón, Contribución al coloreado de grafos y las redes pequeño-mundo, Tesis Doctoral, Departament de Matemàtica Aplicada IV, UPC, Barcelona, 2001.
- [10] E. Palà, Assignación de frecuencias hopping en redes GSM de Airtel, Projecte Final de Carrera ETSETB, Barcelona, 2001.