UNIVERSITAT DE BARCELONA
Facultat de Matemàtiques i Informàtica

GRAU DE MATEMÀTIQUES

Treball final de grau

Coloración del plano: el problema de Hadwiger-Nelson

Autor: Sergio Barril Pizarro

Director: Dr. F. Javier Soria de Diego
Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, 20 de juny de 2019

Índice

Abstract **III**

Agradecimientos **IV**

1. Introducción **1**

2. Teoría de grafos **3**

2.1. Grafos 3

2.2. Coloración 4

3. El problema de Hadwiger-Nelson **7**

3.1. Introducción al problema 7

3.2. Aubrey de Grey 12

3.3. Cota inferior de 5 13

3.3.1. Definiciones previas 13

3.3.2. El grafo H 15

3.3.3. El grafo L 16

3.3.4. El grafo M 21

3.3.5. El grafo N 26

4. Reducción del grafo 5-cromático **27**

4.1. Comprobación de k-cromaticidad 27

4.1.1. Fundamentos de lógica proposicional 27

4.1.2. Forma normal conjuntiva y derivación 28

4.1.3. Formulación de la k-colorabilidad de un grafo 30

4.1.4. Solucionadores SAT 31

4.2. Reducción de de Grey 31

4.2.1. El grafo G 34

4.3. Reducción de Heule 36

5. Estado del problema **40**

5.1. Mejora de las cotas del plano 41

5.2. Cotas para dimensiones superiores 42

6. Conclusiones **44**

Apéndice **45**

Vertex.py 45

Graphs.py 46
Color.py 48
Sat.py 50
Tikz.py 51
Main.py 52

Referencias **58**

Abstract

How many colors are needed in order to color the Euclidean plane in a way such that no two points at unit distance have the same color?

This problem was first posed by Edward Nelson in 1950, and shortly after the first bounds were found: a lower bound of 4 and an upper bound of 7.

These bounds were the best known for almost 70 years, until in April 2018 the biomedical gerontologist Aubrey de Grey shared a graph with 1585 vertices and 7909 edges of length 1, that could not be colored with 4 colors, thus improving the lower bound to 5.

Besides looking for graphs with edges of length 1 that can't be colored with 5 colors (which would further improve the lower bound to 6), one of the current goals is finding a smaller graph that can't be 4-colored. To that end, two propositional logic tools have proven useful: SAT solvers and DRAT checkers. Currently, the smallest 5-chromatic graph found is due to Marijn Heule, with only 553 vertices.

An improvement in both the number of vertices of the smallest 5-chromatic graph and the upper or lower bounds is still being actively pursued.

Resumen

¿Cuántos colores hacen falta para colorear el plano euclídeo de forma que no haya dos puntos a distancia 1 con el mismo color?

Esta pregunta la planteó por primera vez Edward Nelson en 1950, y poco después se obtuvieron las primeras cotas: una cota inferior de 4 y una cota superior de 7.

Estas cotas fueron las mejores conocidas durante casi 70 años, hasta que en abril de 2018 el gerontólogo Aubrey de Grey compartió un grafo, con 1585 vértices y 7909 aristas de longitud 1, que no puede ser coloreado con 4 colores, aumentando así la cota inferior a 5.

Además de buscar un grafo como el anterior que no pueda ser coloreado con 5 colores (que subiría la cota inferior a 6), los esfuerzos se han centrado también en encontrar grafos 5-cromáticos cada vez más pequeños. Para ello, se han demostrado útiles dos herramientas de lógica proposicional: los solucionadores SAT y los comprobadores DRAT. Actualmente, el grafo más pequeño, que fue encontrado por Marijn Heule, es de 553 vértices.

A día de hoy se sigue intentado encontrar una mejora tanto del número de vértices del grafo 5-cromático como de las cotas superior e inferior.

2010 Mathematics Subject Classification. 05C15, 05C85

Agradecimientos

Principalmente quiero dar las gracias a Javier Soria. En primer lugar, por su gran trabajo haciendo la asignatura de grafos: da gusto que el primer contacto con la teoría de grafos sea así.

Después, gracias por proponerme este trabajo: ha acabado encajando a la perfección. Con él he podido poner en práctica conocimientos de casi todas las optativas que he hecho.

Finalmente, gracias por ir revisando el progreso pese a mi ritmo de trabajo irregular.

1. Introducción

El problema de Hadwiger-Nelson busca encontrar la siguiente propiedad de $\mathbb{R}^2$:

Problema 1.1. ¿Cuántos colores hacen falta para colorear el plano euclídeo de forma que no haya dos puntos a distancia 1 con el mismo color?

Ese número, al que llamaremos número cromático del plano (o NCP), sigue siendo desconocido a día de hoy. Prácticamente desde su planteamiento en 1950, el problema ha tenido dos cotas claras: $NCP \geq 4$ y $NCP \leq 7$.

Desde entonces, y ante las dificultades de mejorar esas cotas, se han ido estudiando problemas alternativos. Por ejemplo:

- Se ha planteado el problema para otros espacios métricos.
- Se han encontrado cotas del número cromático fraccionario del plano, es decir, coloreando los vértices con más de un color a la vez.
- Se han encontrado cotas del número cromático del plano evitando que haya dos puntos con el mismo color cuya distancia esté en un intervalo $I \subset \mathbb{R}$.

El primer avance en la resolución del problema original se dio en abril de 2018, cuando el gerontólogo Aubrey de Grey construyó un grafo con aristas de longitud 1 que necesitaba 5 colores para cumplir la condición del problema, demostrando así que $NCP \geq 5$.

A partir de este descubrimiento, la investigación alrededor del problema de Hadwiger-Nelson se ha reactivado con la creación de un proyecto Polymath dedicado a él: un proyecto abierto de colaboración matemática para avanzar en el problema.

Este trabajo tiene dos objetivos principales:

- El primero es estudiar los dos resultados más importantes obtenidos hasta ahora: el descubrimiento del grafo de de Grey, y la obtención del grafo más pequeño encontrado hasta la fecha que requiera 5 colores.
- El segundo es el desarrollo de una aplicación que permita construir los diferentes grafos y pueda comprobar el número de colores que requieren.

Estructura de la Memoria

- En la Sección 2 se dan las definiciones básicas de la teoría de grafos, así como una pequeña introducción a la coloración de grafos.
- En la Sección 3 se introduce el problema de Hadwiger-Nelson de coloración del plano.
    - En la Sección 3.1 se ve el problema para el caso de una dimensión: el número cromático de $\mathbb{R}$. También se ve una pequeña introducción histórica del problema de Hadwiger-Nelson, así como las diferentes cotas inferiores y superiores clásicas.

- En la Sección 3.2 presentamos brevemente a Aubrey de Grey, quien logró mejorar la cota inferior del problema a 5.
- En la Sección 3.3 vemos la demostración de que el número cromático del plano es mayor o igual a 5.

  Empezamos definiendo unas funciones que nos servirán para construir los diferentes grafos que iremos viendo: rotación de un grafo, la suma de Minkowski entre grafos y el recorte de un grafo.

  Después se construyen los diferentes grafos H, L y M, observando sus propiedades.

  Finalmente, se construye el grafo N, que es 5-cromático, demostrando así que $NCP \geq 5$.

- En la Sección 4 se ven diferentes estrategias para encontrar grafos 5-cromáticos más pequeños.

  - En la Sección 4.1 vemos una manera más eficiente de comprobar la propiedad de k-colorabilidad de un grafo: usando solucionadores SAT.

    Para poder utilizarlos, en primer lugar damos conceptos básicos de lógica proposicional, seguidos de los conceptos de fórmula en forma normal conjuntiva y derivación.

    A continuación, escribimos la fórmula lógica que determina si un grafo es k-coloreable, y finalmente presentamos el concepto de solucionador SAT.
  - En la Sección 4.2 vemos diferentes estrategias usadas por Aubrey de Grey para reducir su grafo N de más de 20 mil vértices a 1581.
  - En la Sección 4.3 vemos la estrategia de reducción de grafos de Marijn Heule, junto a los diferentes grafos 5-cromáticos que logró, incluido el récord actual de 553 vértices.

- En la Sección 5 se ve la existencia del proyecto Polymath, y se anuncian resultados para las diferentes cotas. También se ven las cotas superior e inferior en espacios euclídeos de diferentes dimensiones.

- Finalmente, en el Apéndice se explica la implementación de la aplicación *UDGraphs*, que construye y colorea grafos de distancia unitaria.

2. Teoría de grafos

En esta sección se dan las definiciones básicas de la teoría de grafos, que se usarán en el resto de secciones. También se incluye una introducción a la coloración de grafos, con problemas clásicos de esta rama de la teoría de grafos.

2.1. Grafos

Un **grafo** es una colección de puntos (llamados *vértices* o *nodos*) y líneas uniendo algunos de esos puntos (llamadas *aristas*). Se puede formalizar esta definición:

Definición 2.1. *Un grafo es un par $G = (V, E)$ formado por un conjunto de vértices, V, y un conjunto de aristas, $E \subseteq V \times V$.*

Nos referiremos al conjunto de vértices del grafo G como $V(G)$, y al conjunto de aristas como $E(G)$. A menudo trataremos a G indistintamente de $V(G)$ o $E(G)$. Así, diremos $v \in G$ y $e \in G$ en vez de $v \in V(G)$ o $e \in E(G)$.

La arista $e = \{x, y\}$ la escribiremos como xy.

Definición 2.2. *Dos vértices $x, y \in G$ son **adyacentes** (o vecinos) si están unidos por una arista, es decir, si la arista xy pertenece a G.*

Definición 2.3. *Un vértice $v \in G$ y una arista $e \in G$ son **incidentes** si $v \in e$. Dos aristas e_1, e_2 diferentes son incidentes si son incidentes a un mismo vértice, es decir, $e_1 \cap e_2 \neq \emptyset$.*

Definición 2.4. *El **orden** de G, denotado por $|G|$, es el número de vértices de G.*

Ejemplo 2.5. El grafo G de la Figura 1 tiene como conjunto de vértices $V = \{A, B, C, D, E, F\}$, y como aristas $E = \{AB, AC, CD, CE, CF, EF\}$. Los vértices A y C son vecinos, pues la arista AC los une. La arista CF es incidente con el vértice F. Las aristas CF y CE son incidentes, ya que tienen C como vértice en común.

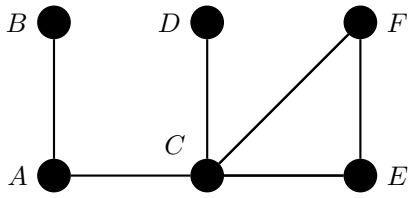


Figura 1: Ejemplo de grafo G.

Los grafos quedan determinados por V y E salvo isomorfismo:

Definición 2.6. *Dos grafos G y H son **isomorfos** si existe una biyección entre sus vértices que conserva las aristas (es decir, $xy \in E(G) \iff \varphi(x)\varphi(y) \in E(H)$ para todo $x, y \in V$)*

Definición 2.7. *Sean $G = (V, E)$ y $G' = (V', E')$. Decimos que G' es un **subgrafo** de G, $G' \subseteq G$, si $V' \subseteq V$ y $E' \subseteq E$. Si, además, $G \neq G'$, G' es un **subgrafo propio**.*

Podemos definir operaciones básicas entre grafos:

Definición 2.8. *Dados dos grafos $G = (V, E)$ y $G' = (V', E')$:*

- *Definimos la **unión** de dos grafos, $G \cup G' \coloneqq (V \cup V', E \cup E')$. Claramente, G y G' son subgrafos de $G \cup G'$.*
- *Definimos la **intersección** de dos grafos, $G \cap G' \coloneqq (V \cap V', E \cap E')$. Claramente, $G \cap G'$ es un subgrafo de G y de G'.*

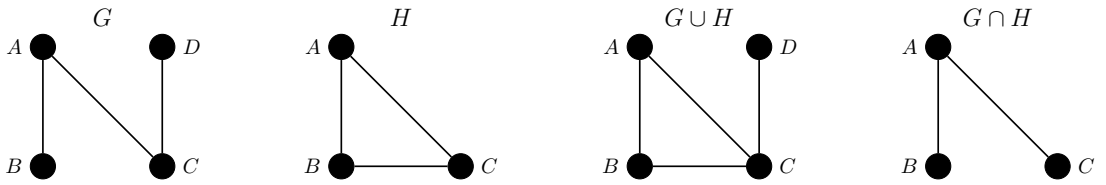


Figura 2: Unión e intersección de dos grafos, G y H.

Definición 2.9. *El **grado** de un vértice $v \in G$, denotado por $d(v)$, es el número de aristas incidentes con v.*

Definición 2.10. *Se denota el **grado mínimo** como $\delta(G) \coloneqq \min\{d(v) | v \in V(G)\}$, y el **grado máximo** como $\Delta(G) \coloneqq \max\{d(v) | v \in V(G)\}$.*

Definición 2.11. *Se define el **grado medio** de un grafo G como $d(G) = \frac{1}{|G|} \sum_{v \in G} d(v)$.*

Definición 2.12. *Si el grado de todos los vértices de G es igual a k, decimos que G es* **k-*regular***.

Ejemplo 2.13. Observando los grafos de la Figura 2, en el grafo G se tiene $d(A) = 2$, y $d(B) = 1$. Los grados mínimo y máximo de $G \cup H$ son $\delta(G \cup H) = 1$ y $\Delta(G \cup H) = 3$. El grafo H tiene $\delta(H) = \Delta(H) = 2$, y por lo tanto es un grafo 2-regular.

Definición 2.14. *Dado un grafo G y un conjunto de vértices, U, se define $G - U$ como el grafo resultante de eliminar los vértices de $V(G) \cap U$ y las aristas incidentes en esos vértices.*

Definición 2.15. *Un grafo G es **maximal** respecto a una propiedad si G tiene esa propiedad, pero no hay ningún grafo H con $G \subsetneq H$ que la tenga.*

2.2. Coloración

La coloración de grafos se basa en etiquetar los vértices de forma que dos vértices vecinos no compartan la misma etiqueta.

Definición 2.16. *Una **coloración** de vértices de un grafo $G = (V, E)$ es una aplicación $c\colon V \to S$ tal que $c(v) \neq c(w)$ para v y w adyacentes. Los elementos de S son los colores disponibles. Una **k-coloración** de un grafo G es una coloración $c\colon V \to \{1, \ldots, k\}$, es decir, una coloración con hasta k colores diferentes.*

A menudo, el objetivo de los problemas de coloración consiste en minimizar el número de colores diferentes usados.

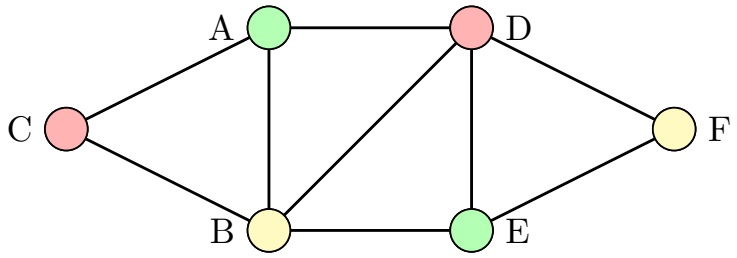


Figura 3: Ejemplo de 3-coloración de un grafo

Definición 2.17. *El* ***número cromático*** *de un grafo G, denotado por χ, es el entero k más pequeño tal que hay una k-coloración de G. Si $\chi(G) \leq k$, decimos que G es* ***k-coloreable****. Si además $\chi(G) = k$, decimos que G es* ***k-cromático****.*

El siguiente problema, llamado *Problema del Almacenamiento*, es un ejemplo clásico de problema que se puede solucionar usando la coloración de grafos:

Problema 2.18. Se quiere almacenar n productos explosivos en contenedores teniendo en cuenta que hay productos que no se pueden almacenar en el mismo contenedor, pues reaccionarían. El objetivo es hacerlo minimizando el número de contenedores a usar.

Para solucionarlo, se puede plantear el siguiente grafo G: los vértices de G serán los productos que queremos almacenar, y dos vértices son adyacentes si los productos reaccionarían (es decir, no se pueden poner en el mismo contenedor). Entonces, basta con dar una k-coloración del grafo G, con $k = \chi(G)$, y cada color representaría un contenedor diferente.

Se puede definir también la coloración de aristas:

Definición 2.19. *Una* ***coloración de aristas*** *de $G = (V, E)$ es una aplicación $c\colon E \to S$ tal que $c(e) \neq c(f)$ para e y f adyacentes. Los elementos de S son los colores disponibles. Se define el* ***índice cromático****, denotado por $\chi'(G)$, al entero k más pequeño tal que hay una k-coloración de aristas de G.*

Todo problema de coloración de aristas se puede traducir a uno de coloración de vértices. Considerando su grafo de línea $L(G)$, esto es, el grafo cuyos vértices son las aristas de G, y dos vértices son vecinos si las aristas de G son incidentes.

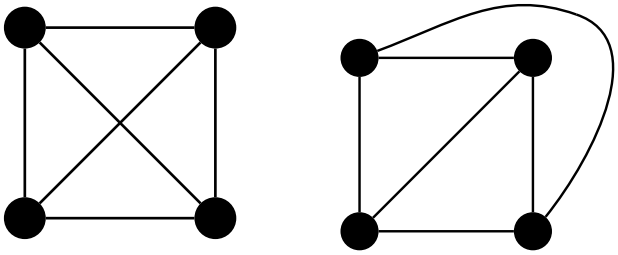
A no ser que se especifique lo contrario, con coloración de un grafo G nos referiremos a la coloración de sus vértices.

La terminología de "colores" se utiliza debido a que uno de los problemas clásicos de coloración de grafos es el de la coloración de mapas. Vemos una definición previa:

Definición 2.20. *Un grafo G es* ***plano*** *si sus aristas solo se cruzan en los vértices. Un grafo G es* ***planar*** *si es isomorfo a un grafo plano.*

El problema clásico de coloración de mapas es el siguiente:

Problema 2.21. ¿Cuántos colores hacen falta para colorear cualquier mapa, de modo que dos regiones adyacentes no tengan el mismo color?



(a) Grafo planar (b) Grafo plano

Figura 4: Grafo planar G y su grafo plano isomorfo H

Este problema queda resuelto con el siguiente teorema:

Teorema 2.22 (Teorema de los Cuatro Colores)**.** *Todo grafo planar es 4-coloreable.*

Todo mapa se puede reinterpretar como un grafo donde los vértices son las regiones y dos vértices son vecinos si las regiones son adyacentes. Así, el Teorema de los Cuatro Colores afirma que con cuatro colores se puede colorear cualquier mapa de forma que dos regiones no tengan el mismo color.

La primera demostración correcta del Teorema de los Cuatro Colores fue publicada por Kenneth Appel y Wolfgang Haken en 1977 [1]. En ella, Appel y Haken dan una lista de más de 1400 grafos que todo grafo planar maximal del plano debe contener. Luego, usando un ordenador, comprueban que todas esas configuraciones se pueden colorear con 4 colores.

Esta demostración tuvo bastante controversia, en gran parte por el uso del ordenador en la demostración. Aunque hoy en día las demostraciones que emplean un ordenador están mucho más aceptadas, se suele preferir una demostración que no dependa de los ordenadores, pues acostumbran a dar una mejor visión de la solución del problema.

Como veremos en la Sección 3, el avance que se ha hecho en el problema de Hadwiger-Nelson también requiere el uso del ordenador para hacer unas comprobaciones.

3. El problema de Hadwiger-Nelson

El problema del número cromático del plano fue planteado hace más de 60 años, y desde entonces había mantenido las mismas cotas inferior y superior: 4 y 7.

En 2018, el gerontólogo Aubrey De Grey demostró la cota inferior de 5, dando un grafo con aristas de distancia 1 coloreable con 5 colores pero no con 4.

En esta sección se ve el planteamiento del problema, con las demostraciones de las cotas clásicas. Luego se estudia la mejora de la cota inferior con el grafo de de Grey, y finalmente se estudian formas de reducir el número de vértices del grafo encontrado.

3.1. Introducción al problema

Antes de presentar el problema, conviene definir dos conceptos previos:

Definición 3.1. *Una **arista unitaria** es una arista cuya longitud en el plano euclídeo es 1. Un grafo cuyas aristas son todas unitarias se dice que es un **grafo de distancia unitaria**.*

El problema de Hadwiger-Nelson plantea la siguiente pregunta respecto al plano euclídeo:

Problema 3.2. ¿Cuántos colores hacen falta para colorear todos los puntos del plano de forma que dos puntos cualesquiera a distancia 1 tengan colores distintos?

Este problema se puede reinterpretar en términos de grafos:

Problema 3.3. Sea G el grafo infinito formado por los puntos del plano como vértices, y con aristas entre los vértices a distancia euclídea 1. ¿Cuál es el número cromático de G?

Así, el problema de Hadwiger-Nelson suele llamarse también *el problema del número cromático del plano (NCP).*

Podemos empezar planteándonos el problema en $\mathbb{R}$, que tiene una solución sencilla.

Proposición 3.4. *El número cromático de la recta real (NCR) es* 2.

Demostración. Que $NCR \geq 2$ es claro, ya que todo grafo con alguna arista requiere al menos 2 colores.

Vemos que $NCR \leq 2$. Podemos recubrir la recta real con los intervalos $A_n = [n, n+1)$ para $n \in \mathbb{N}$. Como no hay dos puntos en A_n a distancia 1, podemos colorear todos los puntos de A_n con el primer color. Los puntos a distancia 1 de A_n estarán en A_{n-1} y A_{n+1}, que habrá que colorear con el segundo color.

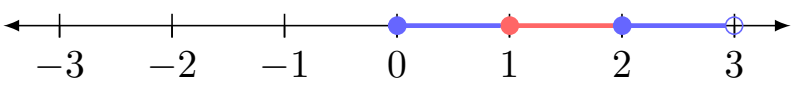


Figura 5: Coloración de la recta real con 2 colores.

□

El problema del número cromático del plano fue planteado (aunque no publicado) por primera vez por Edward Nelson en 1950, quien se lo mencionó a John Isbell. Nelson encontró la cota inferior de 4, mientras que Isbell encontró la cota superior de 7.

La primera publicación del problema fue por parte de Martin Gardner en 1960 [4]. En 1961, Hugo Hadwiger publicó el problema y las demostraciones de las dos cotas [7], apuntando que el problema es de Nelson, y las cotas de Isbell. Aun así, debido a la confusión en la literatura sobre quién propuso el problema, y como Hadwiger contribuyó en resultados similares, se le ha acabado llamando problema de Hadwiger-Nelson, aunque todo apunta a que es originalmente de Edward Nelson.

Una primera observación permite ver que para colorear el plano se necesitan al menos 3 colores:

Proposición 3.5. *El plano no es 2-coloreable. Es decir,* $NCP \geq 3$.

Demostración. Sea T un triángulo equilátero de lado 1. T tiene 3 vértices, y para colorearlos con 2 colores, habrá que repetir al menos un color. Por lo tanto, $NCP \geq 3$.

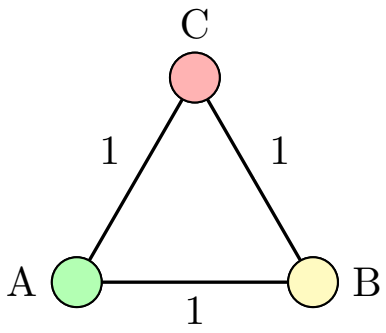


Figura 6: Coloración de un triángulo equilátero de lado 1 con 3 colores.

□

Los hermanos Leo y William Moser dieron en 1961 una demostración de que se necesitan al menos 4 colores, usando un grafo llamado **huso de Moser** (Figura 7) (en inglés, *Moser's spindle*) [10]. El huso de Moser es un grafo con 7 vértices y 11 aristas unitarias, construido como sigue:

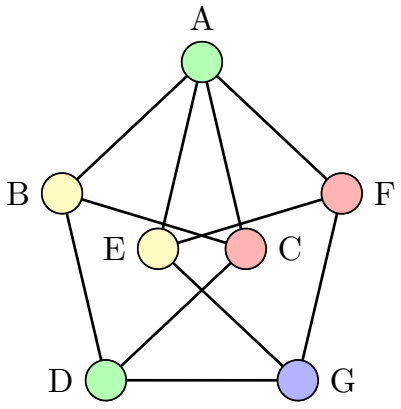


Figura 7: Huso de Moser coloreado con 4 colores.

Consideramos el rombo R_1 formado por dos triángulos equiláteros de lado 1 como en la Figura 7, $ABDCA$. Sea R_2 el mismo rombo R_1 tras aplicarle una rotación de centro A, hasta que el vértice D esté a una distancia de 1 de su posición original. Esto es, una rotación de $\arccos(\frac{5}{6})$ con centro A. Finalmente, el huso de Moser será la unión de ambos rombos, uniendo con una arista los vértices D y G.

Observación 3.6. El vértice E está a una distancia menor de 1 del vértice B, puesto que está contenido dentro del triángulo equilátero BCD. Del mismo modo, el vértice F está a una distancia menor de 1 del vértice C.

Vemos que el huso no se puede colorear con 3 colores:

Proposición 3.7. *El plano no es 3-coloreable. Es decir,* $NCP \geq 4$.

Demostración. Suponemos que el huso es coloreable con 3 colores (verde, amarillo y azul). Empezamos coloreando el vértice A de verde. Los vértices B y C tendrán que ser coloreados amarillo y rojo, ya que ABC forman un triángulo. Por el mismo razonamiento, E y F tendrán que ser coloreados con amarillo y rojo, ya que AEF forman un triángulo. Pero entonces D y G, que son vecinos, tienen que ser coloreados de verde, y por lo tanto el huso no es 3-coloreable. □

Por lo que respecta a las cotas superiores, seguiremos un razonamiento similar al del caso de la recta real, $\mathbb{R}$, visto en la Proposición 3.4.

Proposición 3.8. *Se puede colorear el plano con 9 colores. Es decir,* $NCP \leq 9$.

Demostración. Para verlo, cubriremos el plano con cuadrados de lado d como en la Figura 8. Empezamos coloreando un cuadrado con el color 1, y los ocho cuadrados que lo rodean, con los 8 colores restantes. Repitiendo el mismo procedimiento, cubrimos el plano.

Vemos ahora la condición sobre d para que no haya puntos a distancia 1 del mismo color.

- No puede haber puntos a distancia 1 dentro del mismo cuadrado. La distancia máxima dentro de un cuadrado es $\sqrt{2}d$, así que la condición será $d < \frac{\sqrt{2}}{2}$.
- No puede haber dos cuadrados del mismo color a menos de distancia 1 entre ellos. La distancia entre dos cuadrados con el mismo color es $2d$, así que la condición será $d > \frac{1}{2}$.

Así pues, cogiendo $\frac{1}{2} < d < \frac{\sqrt{2}}{2}$, tendremos una coloración del plano con 9 colores.

Como tenemos un intervalo para d, los bordes de los cuadrados se pueden colorear de diferentes formas. Un criterio, por ejemplo, sería que los colores con número más bajo tienen preferencia sobre los números más altos. □

Esta cota se puede mejorar usando hexágonos en vez de cuadrados.

Proposición 3.9. *Se puede colorear el plano con 7 colores. Es decir,* $NCP \leq 7$.

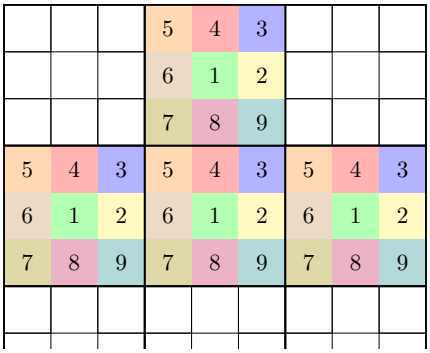


Figura 8: Coloración del plano con 9 colores.

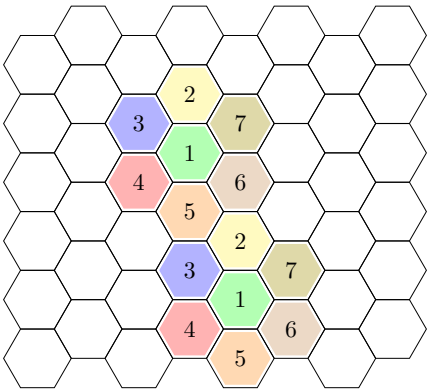


Figura 9: Coloración del plano con 7 colores usando hexágonos.

Demostración. Para verlo, cubriremos el plano con hexágonos regulares de lado d, como en la Figura 9. Empezamos coloreando un hexágono con el color 1, y los seis hexágonos que le rodean, con los 6 colores restantes. Copiando la unión de estos hexágonos, recubrimos el plano.

Vemos ahora la condición sobre d para que no haya puntos a distancia 1 del mismo color:

- No puede haber puntos a distancia 1 dentro del mismo hexágono. La distancia máxima dentro de un hexágono regular es $2d$, así que la condición será $d < \frac{1}{2}$.
- No puede haber dos hexágonos del mismo color a menos de distancia 1 entre ellos. Calculamos la distancia mínima entre dos hexágonos usando la Figura 10.

  En primer lugar, a es el apotema del hexágono regular, que es $a = \frac{\sqrt{3}}{2}d$. Además, usando los triángulos verdes vemos que la base del rectángulo cumplirá $b^2 = d^2 - a^2$. Finalmente, para encontrar la distancia mínima tenemos:

  $$x^2 = (3a)^2 + b^2 = 8a^2 + d^2 = 7d^2$$

  Por lo tanto, la distancia mínima entre dos hexágonos del mismo color es $x = \sqrt{7}d$, y la condición $x > 1$ se traduce como $d > \frac{\sqrt{7}}{7}$.

Así pues, cogiendo $\frac{\sqrt{7}}{7} < d < \frac{1}{2}$, tendremos una coloración del plano con 7 colores.

Como tenemos un intervalo para d, los bordes de los hexágonos se pueden colorear de diferentes formas. Un criterio, por ejemplo, sería que los colores con número más bajo tienen preferencia sobre los números más altos.

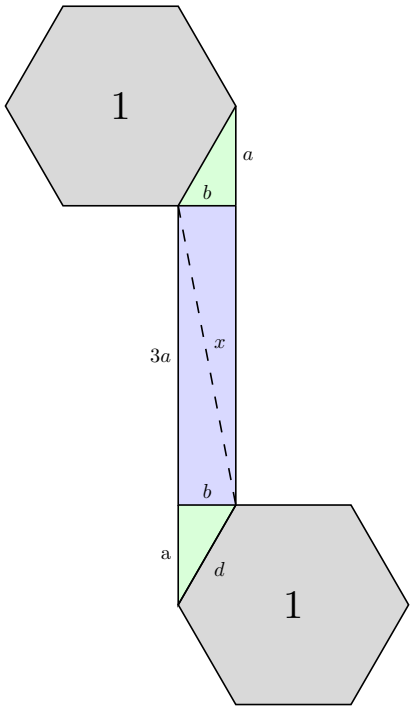


Figura 10: Distancia mínima entre dos hexágonos del mismo color.

□

Esta es la demostración clásica de la cota superior $NCP \leq 7$, pero no es la única sencilla. En 1982, el matemático húngaro László A. Székely demostró la cota superior de 7 usando un embaldosado con cuadrados [17] como en la Figura 11.

Demostración. Consideramos una fila de cuadrados con diagonal 1, que coloreamos de forma ordenada con los colores $1, 2, \ldots, 7$. Las siguientes filas se construyen desplazando 2.5 cuadrados la fila anterior.

Las aristas superior y derecha de los cuadrados se pintan del color del cuadrado, excepto las esquinas superior izquierda e inferior derecha.

Vemos que se cumplen las condiciones:

- No hay dos puntos a distancia 1 en un cuadrado que estén coloreados del mismo color, ya que solo se podría dar en las diagonales pero las esquinas tienen colores diferentes.
- La distancia mínima entre dos cuadrados del mismo color es de 1.5 cuadrados. El lado de los cuadrados es de $d = \frac{\sqrt{2}}{2}$, y por lo tanto la distancia mínima entre dos cuadrados es $x = \frac{3\sqrt{2}}{4} > 1$.

□

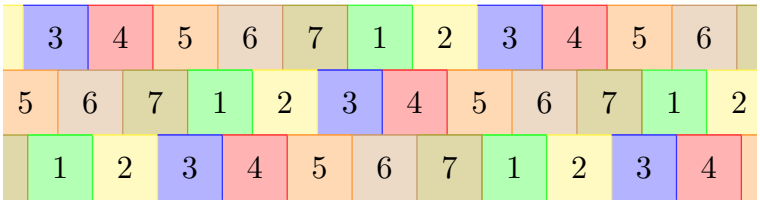


Figura 11: Coloración del plano con 7 colores usando cuadrados.

Las cotas inferior y superior de 4 y 7 para el problema del número cromático del plano no habían sido mejoradas desde el planteamiento del problema.

Esto es, hasta que el 8 de abril de 2018 Aubrey de Grey construyó un grafo de distancia unitaria cuyo número cromático es $\chi = 5$. Así, de Grey demostró que $NCP \geq 5$.

3.2. Aubrey de Grey

Aubrey de Grey es un gerontólogo inglés conocido por su investigación sobre técnicas para ralentizar el envejecimiento humano. De Grey sostiene, de hecho, que llegará un momento en el que tales técnicas se irán perfeccionando suficientemente rápido para que se pueda llegar a vivir indefinidamente [6].

De Grey se graduó en Ciencias de la Computación por la Universidad de Cambridge. Fue su ahora exmujer, la genetista Adelaide Carpenter, la que le introdujo a la biología y la genética. En diciembre del 2000, de Grey obtuvo el doctorado en biología.

Figura 12: Aubrey de Grey.

Aubrey de Grey es el cofundador de la SENS Research Foundation[2], una organización sin ánimo de lucro que investiga estrategias para ralentizar el envejecimiento en las personas. Se conocen las fuentes principales del daño celular relacionado con la edad, pero sus mecanismos concretos y cómo evitarlos son procesos más complicados. En su lugar, en SENS se centran más en procedimientos y terapias para revertir ese daño, evitando así enfermedades ligadas a la edad.

[2] `https://www.sens.org`

Aubrey de Grey ha sido también jugador profesional del juego de mesa *Othello*. La faceta matemática de de Grey viene motivada por otros jugadores de este juego, que eran matemáticos, y que le introdujeron a la teoría de grafos.

3.3. Cota inferior de 5

Aubrey de Grey logró un avance muy importante en el problema de Hadwiger-Nelson, aumentando la cota inferior del número cromático del plano hasta 5.

Teorema 3.10. *El plano no es 4-coloreable. Es decir,* $NCP \geq 5$.

Para la demostración, de Grey sigue los siguientes pasos:

1. Construye un grafo de distancia unitaria, H, y observa que solo hay cuatro coloraciones diferentes de H con cuatro colores o menos. De estas, dos coloraciones tienen tres vértices de un mismo color, y las otras dos coloraciones no.

2. Construye un grafo de distancia unitaria, L, formado por la unión de 52 copias de H. Observa que en cualquier 4-coloración de L hay alguna copia de H con tres vértices del mismo color.

3. Construye un grafo de distancia unitaria, M, que contiene una copia de H, y observa que ninguna 4-coloración de M hace que la copia de H tenga tres vértices del mismo color.

4. Construye un grafo de distancia unitaria, N, formado por la unión de 52 copias de M, de forma que las respectivas copias de H estén distribuidas como en el grafo L.

   Como N tiene una copia de L, toda 4-coloración hace que alguna copia de H tenga tres vértices del mismo color. Pero por construcción de M, ninguna copia de H tendrá una terna monocromática al ser coloreada con cuatro colores, concluyendo así que N no es 4-coloreable.

3.3.1. Definiciones previas

Antes de pasar a la demostración, conviene considerar tres operaciones que nos ayudarán a construir los diferentes grafos.

Definición 3.11. *Definimos* $\mathbf{rot}(G, \alpha, v)$ *como la rotación de ángulo* α *del grafo* G *respecto al en el vértice* v*. Habitualmente aplicaremos la rotación respecto al vértice central, y escribiremos* $\mathbf{rot}(G, \alpha)$.

La siguiente rotación resultará especialmente útil:

Definición 3.12. *Denotamos como* $\theta_i^k(G, v)$ *la rotación de ángulo* $k \cdot \arccos(\frac{2i-1}{2i})$ *respecto al vértice* v*. Habitualmente aplicaremos la rotación respecto al vértice central, y escribiremos* $\theta_i^k(G)$*. Además, si* $k = 1$ *escribiremos* $\theta_i(G, v)$.

Proposición 3.13. *Sea* A *un vértice a distancia* $\sqrt{i}$ *del punto* O*. Entonces,* A *y* $\theta_i(A, O)$ *están a distancia 1 entre sí.*

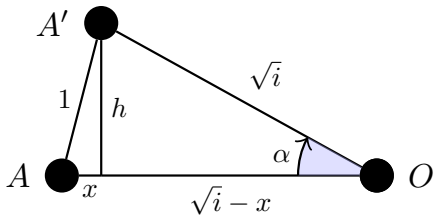


Figura 13: Rotación de un punto a distancia $\sqrt{i}$ del centro.

Demostración. En la Figura 13 se representan el vértice central, O, un vértice a distancia $\sqrt{i}$ del centro, A, y el vértice rotado, $A' = \theta_i(A, O)$. Vemos que entonces el ángulo es $\alpha = \arccos(\frac{2i-1}{2i})$:

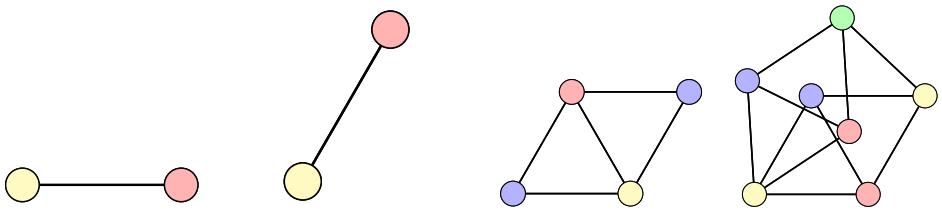
$$h^2 + x^2 = 1$$
$$i = h^2 + (\sqrt{i} - x)^2 \Rightarrow x = \frac{1}{2\sqrt{i}}$$
$$\cos(\alpha) = \frac{\sqrt{i} - x}{\sqrt{i}} = \frac{\sqrt{i} - \frac{1}{2\sqrt{i}}}{\sqrt{i}} \Rightarrow \cos(\alpha) = \frac{2i-1}{2i}$$

Así pues, para que un punto a distancia $\sqrt{i}$ del centro y su rotado estén a distancia 1 entre sí, el ángulo de rotación tendrá que ser $\arccos(\frac{2i-1}{2i})$. □

Definición 3.14. *Definimos la* ***suma de Minkowski*** *de dos grafos A y B, $A \oplus B$, como el grafo dado por los vértices $\{a + b : a \in A, b \in B\}$ cuyas aristas son entre los vértices a distancia 1.*

Ejemplo 3.15. Sean $A = \{(0,0), (1,0)\}$ y $B = \{(0,0), (\frac{1}{2}, \frac{\sqrt{3}}{2})\}$. Entonces, su suma de Minkowski es $A \oplus B = \{(0,0), (1,0), (\frac{1}{2}, \frac{\sqrt{3}}{2}), (\frac{3}{2}, \frac{\sqrt{3}}{2})\}$.

Observamos que los vértices $(0,0)$ y $(\frac{3}{2}, \frac{\sqrt{3}}{2})$ están a distancia $\sqrt{3}$. Así pues, el grafo $(A \oplus B) \cup \theta_3(A \oplus B)$ estará formado por dos rombos unidos en un vértice, con uno de ellos rotado de forma que el vértice opuesto esté a distancia 1 del mismo vértice sin rotar. Es decir: un huso de Moser.



(a) Grafo A. (b) Grafo B. (c) Grafo $A \oplus B$. (d) Huso de Moser.

Figura 14: Ejemplo de la suma de Minkowski.

Definición 3.16. *Definimos el* ***grafo recortado*** *de G, $\text{trim}(G, r)$ como el grafo $G - U$, donde U es el conjunto de vértices que distan más de r del vértice central.*

En el contexto que nos ocupa, consideraremos que las aristas del grafo resultante de la unión de dos grafos serán todas aquellas entre vértices a distancia 1 entre sí.

3.3.2. El grafo H

Definición 3.17. *Se define el* ***grafo H*** *como el grafo de distancia unitaria de 7 vértices y 12 aristas unitarias formado por un hexágono regular de lado 1 y su centro.*

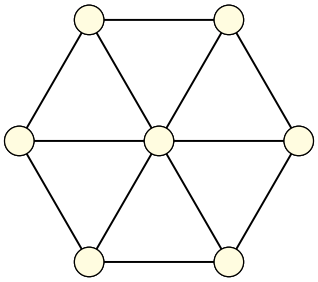


Figura 15: Grafo H.

El grafo H será el grafo básico a partir del cual construiremos el resto de grafos. Estudiamos sus diferentes coloraciones.

Lema 3.18. *Solo hay cuatro coloraciones distintas (salvo rotación o reflexión del grafo, o trasposición de los colores) del grafo H con hasta 4 colores.*

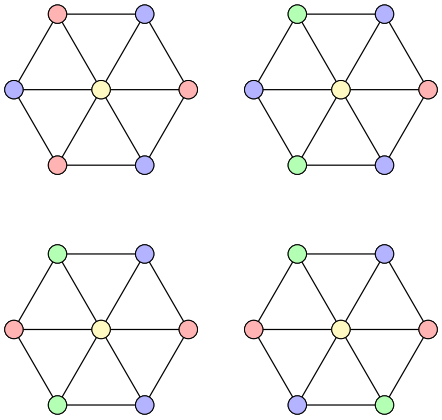


Figura 16: Las cuatro coloraciones distintas del grafo H usando tres o cuatro colores.

Demostración. Empezamos observando que el vértice central es adyacente al resto de vértices, y así, toda coloración tendrá un color exclusivo para él (amarillo).

Vemos que H es 3-coloreable. Coloreamos el vértice derecho con el segundo color (rojo). Finalmente, coloreamos el resto de vértices alternando entre un tercer color (azul) y el segundo color. Esta es la única 3-coloración de H.

Estudiamos ahora sus 4-coloraciones según la distribución de los colores en los 6 vértices del hexágono.

- Si hay un color con tres vértices (azul), un color con dos vértices (verde), y un color con un vértice (rojo), hay una única coloración salvo rotación o trasposición de colores.

Sea el vértice derecho rojo. Entonces, sus vecinos deben ser verdes o azules. Los tres vértices azules deben estar intercalados por los dos vértices verdes, dando así la única coloración.

- Si los tres colores tienen dos vértices cada uno, hay dos coloraciones diferentes posibles.

  Sea el vértice derecho rojo. Entonces, sus vecinos deben ser verdes o azules.

  - Si ambos vecinos son azules, los dos vértices verdes restantes deben estar intercalados por el otro vértice rojo.
  - Si un vecino es azul y el otro verde, los tres vértices restantes quedan determinados. Cualquier otra coloración resultaría equivalente bajo una rotación de H y trasposición de colores.

- No puede haber más de tres vértices del mismo color, pues no habría suficientes vértices de otros colores para separarlos.

□

Observación 3.19. De las cuatro coloraciones de H, hay dos con tres vértices del mismo color (las dos superiores de la Figura 16) y dos que no (las dos inferiores).

Además, en las dos 4-coloraciones sin ternas monocromáticas, una tiene vértices del mismo color a distancia $\sqrt{3}$ (inferior izquierda) y la otra tiene todos los vértices del mismo color a distancia 2 (inferior derecha).

3.3.3. El grafo L

Para construir el grafo L, construiremos y estudiaremos antes varios subgrafos de L.

Definición 3.20. *Se define el* ***grafo J*** *como el grafo de distancia unitaria de 31 vértices, formado por la unión de 13 copias del grafo H.*

Las copias se unen de la siguiente forma: una copia de H en el centro, seis copias centradas en los vértices a distancia 1, y otras seis centradas en los vértices a distancia $\sqrt{3}$ del centro.

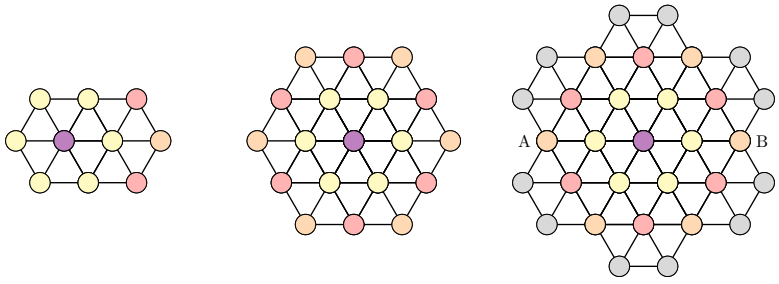


Figura 17: Construcción del grafo J.

En la Figura 17 se representa la construcción de J, donde el vértice central se representa en morado y el resto según la distancia a este. En la figura de la izquierda hay 2 copias de H: una copia central y otra centrada en el vértice derecho de esta. En la figura central

hay 7 copias de H: una copia central, y otras 6 centradas en los vértices a distancia 1 del vértice central. Finalmente, en la figura de la derecha está el grafo J con las 13 copias de H.

Definición 3.21. *Nombraremos* ***vértices de enlace*** *a los vértices a distancia 2 del vértice central. Una* ***diagonal de enlace*** *será un par de vértices de enlace alineados con el centro. Etiquetaremos con A y B una diagonal de enlace de J.*

Para el estudio de las coloraciones de J, nos centraremos en aquellas tales que ninguna de las 13 copias de H tengan 3 vértices del mismo color. Concretamente, nos centraremos en aquellos vértices a una distancia menor o igual a 2 del vértice central: el resto no juega un papel importante.

Los vértices a distancia mayor de 2 serán coloreados de negro, representando que se les puede asignar cualquier color que no genere ternas monocromáticas en ninguna copia de H.

Lema 3.22. *Hay seis 4-coloraciones distintas salvo rotación o reflexión del grafo, trasposición de los colores y salvo los colores de los vértices a distancia mayor de 2, del grafo J tales que ninguna de las 13 copias de H tiene tres vértices del mismo color.*

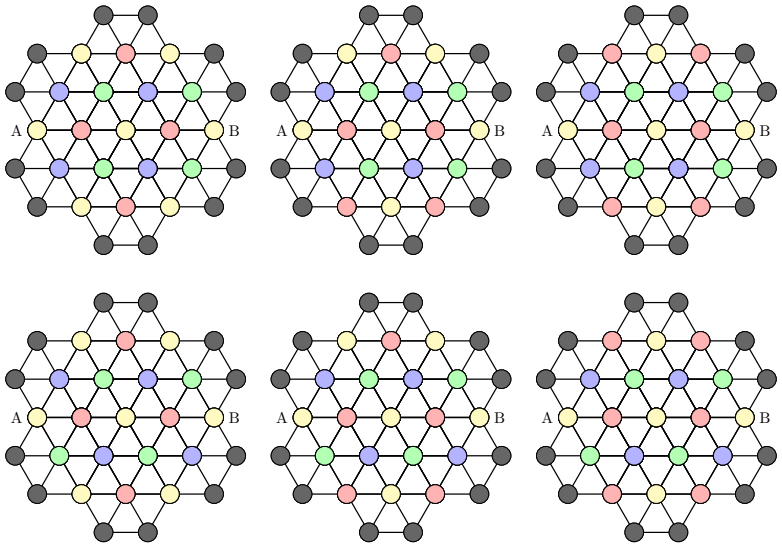


Figura 18: Diferentes 4-coloraciones de J sin ternas monocromáticas en ninguna copia de H.

Demostración. Hemos visto en la Observación 3.19 que solo hay dos 4-coloraciones de H distintas en las que no hay ternas monocromáticas. Así, cogeremos como H central una de las dos coloraciones inferiores de la Figura 16. Nos referiremos a la coloración inferior izquierda como coloración 1, y a la inferior derecha como coloración 2.

Empezamos observando que los colores de los vértices de la fila central y las filas superior e inferior a esta quedan determinados por la coloración del H central (Figura 19).

- Supongamos que la copia central de H está coloreada con la coloración 1. Entonces, en la copia de H a la izquierda del H central quedan tres vértices por colorear. Esta

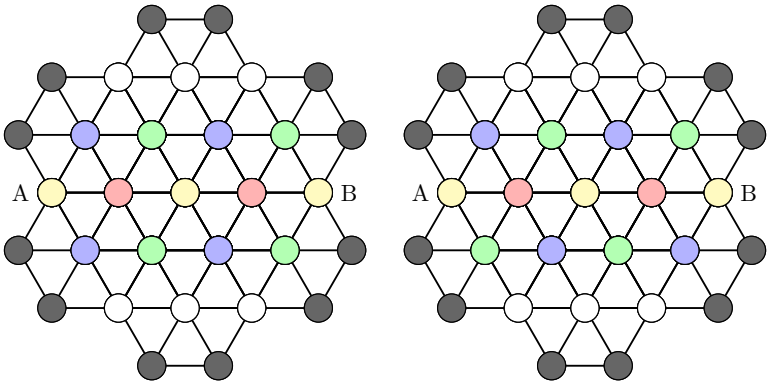


Figura 19: Coloraciones diferentes de las tres filas centrales de J.

copia de H ya tiene 2 vértices verdes, y su vértice central es rojo, así que falta uno amarillo y dos azules. Así pues, el vértice A tendrá que ser el amarillo, y los otros dos azules.

Usando el mismo razonamiento, en la copia de H a la derecha del H faltan 2 vértices verdes y uno amarillo, así que el vértice B tendrá que ser el amarillo, y los otros dos verdes.

- Supongamos que la copia central de H está coloreada con la coloración 2. Entonces, en la copia de H a la izquierda del H central quedan por colorear un vértice azul, uno verde y uno amarillo. Veamos que si coloreamos el vértice A de azul o verde, acabamos con una coloración equivalente.

  Supongamos que coloreamos A de azul (con verde obtenemos un resultado análogo). Entonces los otros dos vértices restantes quedan determinados (amarillo el de arriba, verde el de abajo). Observamos el H centrado en este vértice verde. Faltan por colorear dos vértices amarillos y uno rojo, que se colorean intercalados.

  A la copia de H centrada en el vértice azul a la derecha del vértice verde solo le falta un vértice por colorear, que coloreamos en rojo.

  Nos fijamos ahora en la copia de H centrada en el vértice verde superior del H central. Hay dos vértices a colorear, de forma única, con azul y rojo.

  En la copia de H centrada en este último vértice rojo, faltan por colorear dos vértices amarillos y uno verde, que se colorean intercalados.

  Finalmente, a la copia de H centrada en el vértice azul superior del H central solo le falta un vértice por colorear, que coloreamos en verde.

  Rotando el grafo resultante $\frac{\pi}{3}$, logramos una coloración con las tres filas centrales como en la Figura 19.

Una vez visto que toda 4-coloración de J tiene que tener las filas centrales como en la Figura 19, estudiamos los 6 vértices restantes. Observamos que todos los vértices restantes son vecinos de vértices azules y verdes, así que tendrán que ser amarillos o rojos.

Las tres coloraciones distintas de los 6 vértices se muestran en la Figura 21.

Las dos formas de colorear las tres filas centrales y las tres formas de colorear las filas restantes dan lugar a las seis coloraciones de la Figura 18.

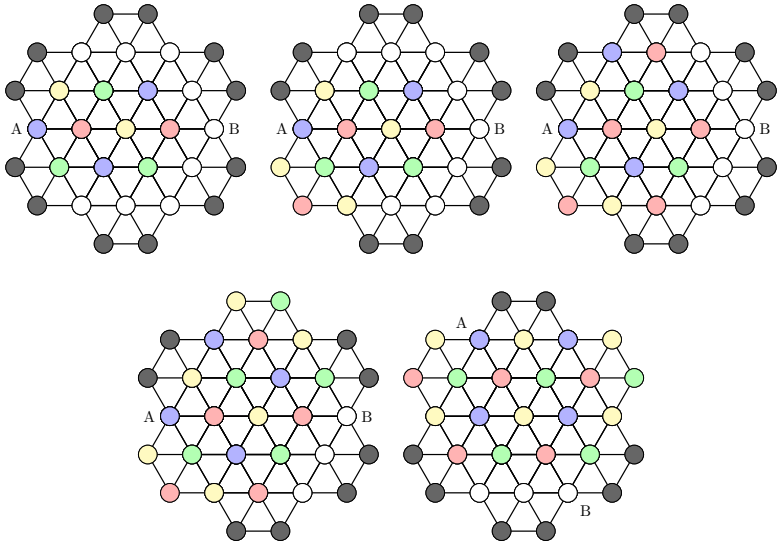


Figura 20: Unicidad de la coloración de las tres filas centrales de J con H central con coloración 2.

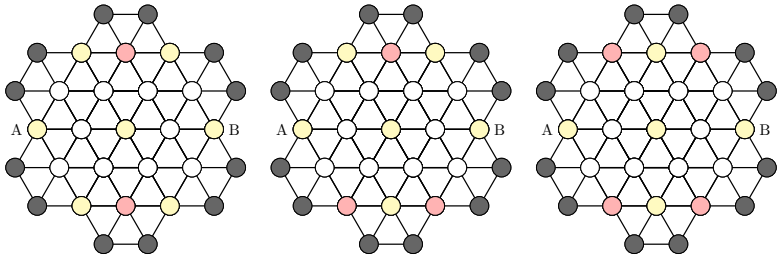


Figura 21: Tres coloraciones diferentes de los vértices de enlace de J.

Finalmente, se puede ver en la Figura 22 que existe alguna coloración de los vértices negros para cada tipo diferente de coloración. □

Observación 3.23. Pese a no importarnos sus colores concretos, los vértices a distancia mayor de 2 son necesarios. En la demostración de la Figura 20 se colorean algunos de esos vértices, y sin ellos habría más formas distintas de colorear J sin ternas monocromáticas en ninguna copia de H.

Observación 3.24. Respecto a la coloración de los vértices de enlace de J de la Figura 18, observamos que hay tres opciones:

a) Los seis vértices de enlace tienen el mismo color del vértice central (las dos coloraciones de la izquierda).

b) Cuatro vértices de enlace consecutivos tienen el mismo color que el central, y los otros dos tienen un segundo color (las dos coloraciones centrales).

c) Una diagonal de enlace tiene el color del vértice central, y las otras dos diagonales tienen un segundo color (las dos coloraciones de la derecha).

Definición 3.25. *Se define el* ***grafo K*** *como el grafo de distancia unitaria de 61 vértices, formado por la unión de dos copias del grafo J centradas en el mismo vértice, con una de ellas rotada* $\arccos(\frac{7}{8})$ *respecto al centro.*

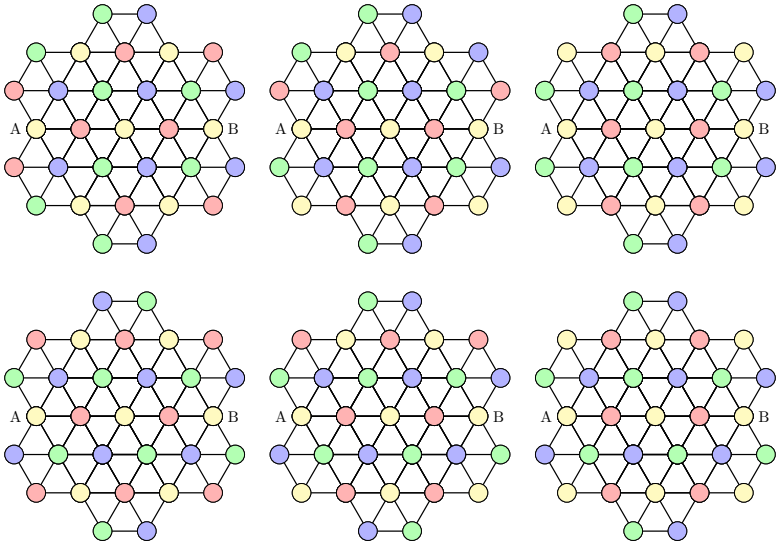


Figura 22: Ejemplos de cada tipo de coloración de J.

Es decir, $K = J \cup \theta_4(J)$.

Observación 3.26. Los vértices de enlace de J están a distancia 2 del centro. Así, usando la Proposición 3.13, vemos que todo vértice de enlace de J está a distancia 1 de su rotado en $\theta_4(J)$.

Es decir, en K un vértice de enlace y su rotado son vecinos.

Observación 3.27. Sean P y Q dos vértices de enlace consecutivos de J, y P' la contraparte de P en $\theta_4(J)$. Entonces, P' y Q no son vecinos.

Demostración. El ángulo entre P y Q es de $\frac{\pi}{3}$, puesto que forman un triángulo equilátero con el centro de K. Hemos demostrado que la rotación θ_4 deja los puntos a distancia 2, a distancia 1 de su posición original.

Como $2 \arccos(\frac{7}{8}) < \frac{\pi}{3}$, entonces $\theta_4^2(P) = \theta_4(P') \neq B$, y por lo tanto P' y Q no son vecinos. □

Vemos ahora una propiedad importante del grafo K:

Proposición 3.28. *Todas las diagonales de enlace de K son monocromáticas.*

Demostración. Vemos que en cualquier 4-coloración de K donde ninguna copia de H tenga tres vértices del mismo color, los vértices de enlace de ambas copias de J tienen que estar coloreados como la opción c de la Observación 3.24

Las opciones a y b implicarían tener varios vértices de enlace consecutivos del color del centro, que haría que algún vértice de enlace y su contraparte de la copia rotada tuvieran el mismo color.

Por tanto, se tienen que colorear las copias de J usando c, y así todas las diagonales de enlace del grafo K serán monocromáticas. □

Finalmente, definimos el grafo L.

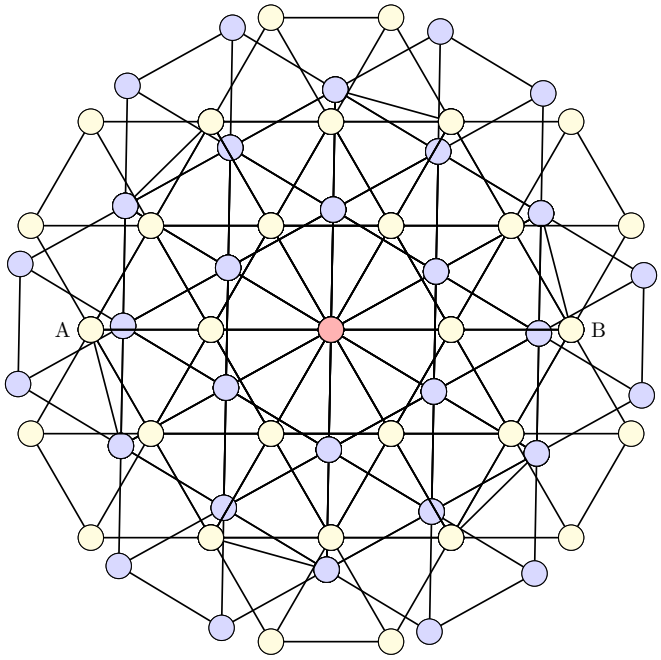


Figura 23: El grafo K, con 26 copias de H.

Definición 3.29. *Sean A y B una diagonal de enlace de K. Se define el* ***grafo L*** *(Figura 24) como el grafo de distancia unitaria de 121 vértices, formado por la unión del grafo K con una copia de K rotada* $\arccos(\frac{31}{32})$ *respecto al vértice de enlace A. Es decir, $L = K \cup \theta_{16}(K, A)$. En total, tiene 52 copias de H.*

Observación 3.30. Esta rotación hará que el otro vértice de la diagonal de enlace de $\{A, B\}$, que está a distancia 4 de A, esté a distancia 1 de $\theta_{16}(B, A)$.

Vemos ahora la propiedad importante del grafo L, que es la que usaremos para encontrar el grafo 5-cromático.

Proposición 3.31. *En toda 4-coloración de L hay alguna copia de H con tres vértices del mismo color.*

Demostración. Suponemos que hay una 4-coloración de L sin copias de H con tres vértices del mismo color.

Sea $B' = \theta_{16}(B, A)$. Por la Proposición 3.28, todas las diagonales de enlace de K tienen que ser monocromáticas. Esto quiere decir que A y B tienen el mismo color. Como A y B' también son una diagonal de enlace, A y B' tienen el mismo color.

Pero B y B' son adyacentes y no pueden tener el mismo color, llegando así a una contradicción. □

3.3.4. El grafo M

El siguiente paso es encontrar un grafo, al que llamaremos M, que tenga una copia de H tal que ninguna 4-coloración de M permita que H tenga tres vértices del mismo color.

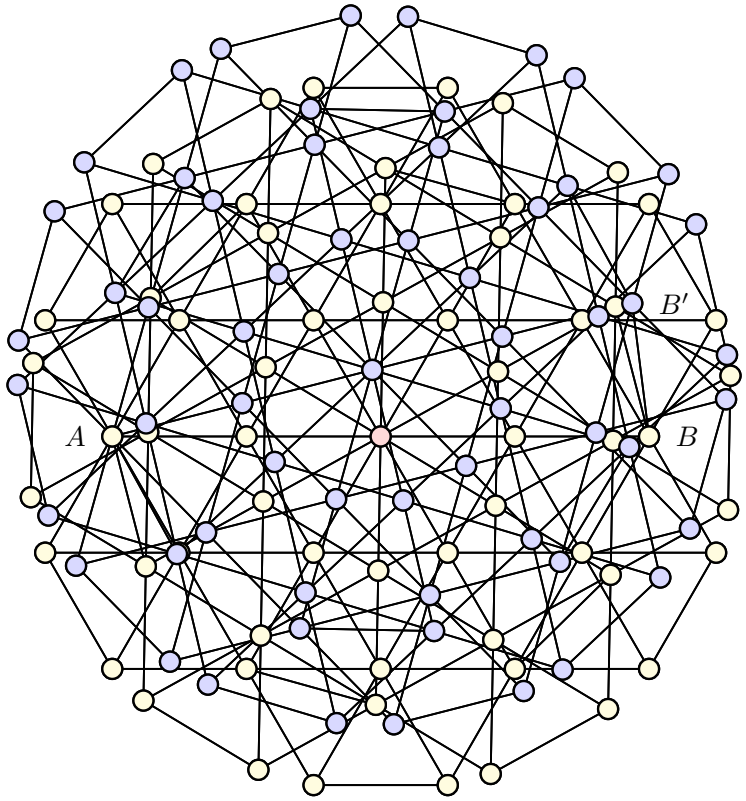


Figura 24: El grafo L, con 52 copias de H. Se representan en amarillo los vértices de K, en azul los de $\theta_{16}(K, A)$.

Observación 3.32. En cualquier coloración del grafo H, si hay 3 vértices del mismo color, estos forman un triángulo equilátero de lado $\sqrt{3}$.

Para buscar ese grafo M, de Grey se fijó en el huso de Moser (Figura 7). El huso de Moser tiene dos pares de vértices que distan $\sqrt{3}$ tales que ambos pares no pueden ser del mismo color simultáneamente. Siguiendo la nomenclatura usada en Figura 7, se trata de $\{A, D\}$ y $\{A, G\}$.

Así, de Grey esperaba que un grafo formado uniendo muchos husos tuviera los pares de vértices monocromáticos a distancia $\sqrt{3}$ distribuidos uniformemente, impidiendo la existencia de triángulos equiláteros monocromáticos. Además, al estar formado por triángulos equiláteros de lado 1, fácilmente podrían dar lugar a hexágonos regulares como H.

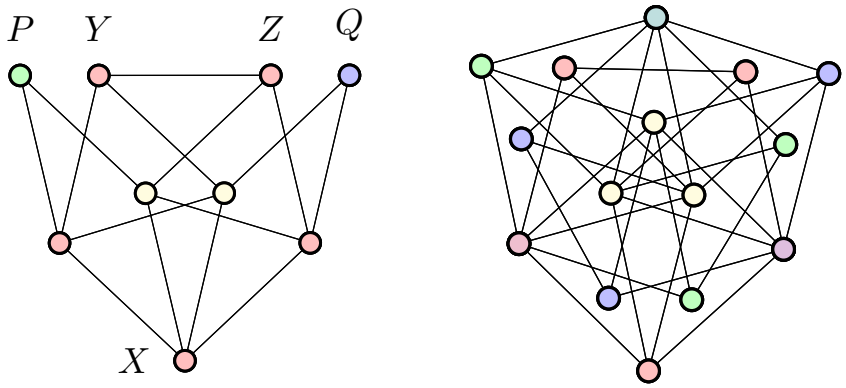
El siguiente grafo nos será aún más útil que el huso de Moser:

Definición 3.33. *Se define el* ***grafo T*** *como el grafo de distancia unitaria de 9 vértices formado por un huso de Moser, añadiéndole dos vértices, P y Q, y dos aristas.*

Estos dos vértices adicionales le dan al grafo T las siguientes propiedades:

- Los vértices P, Q y X forman un triángulo equilátero.
- Los vértices P y Q están alineados con Y y Z.

Definición 3.34. *Se define el* ***grafo U*** *como el grafo de distancia unitaria de 15 vértices que contiene tres copias del grafo T, rotadas $\frac{2\pi}{3}$.*



(a) Grafo T. (b) Grafo U.

Figura 25: Grafos T y U. Los colores permiten diferenciar mejor los tres pentágonos de U, cada uno correspondiente a un grafo T.

Este grafo U tiene simetría triangular. Así, de Grey supuso que rotaciones y traslaciones de U podrían inducir una alta densidad de husos de Moser.

Mediante la unión de rotaciones y traslaciones de U, de Grey obtuvo grafos con una gran densidad de husos de Moser, pero no pudo encontrar a partir de ellos ningún grafo con la propiedad necesaria para M.

El siguiente grafo, que construimos a continuación, sí que le sirvió:

Definición 3.35. *Sea MS un huso de Moser. Construimos el grafo B uniendo tres husos de Moser, de modo que $B = \bigcup_{k=0}^{2} \theta_3^{\frac{k}{2}}(MS)$ (Figura 26a). A continuación unimos tres copias del grafo B, rotadas múltiplos de $\frac{2\pi}{3}$ como en la Figura 26b, obteniendo así el grafo C.*

Definición 3.36. *Se define el* ***grafo V*** *como el grafo de distancia unitaria de 31 vértices, formado por los vértices de C a distancia menor o igual que 1 del vértice central.*

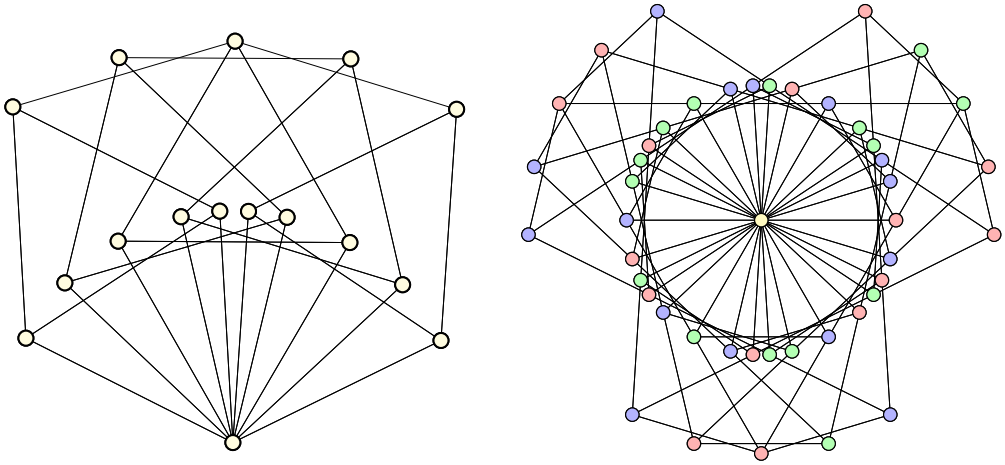
Sea $A = (1, 0)$. El grafo V también se puede construir de la siguiente forma:

$$V = (0,0) \cup \bigcup_{j=0}^{5} \bigcup_{k=-2}^{2} \theta_1^j \theta_3^{\frac{k}{2}}(A)$$

Observación 3.37. El grafo V contiene el grafo H.

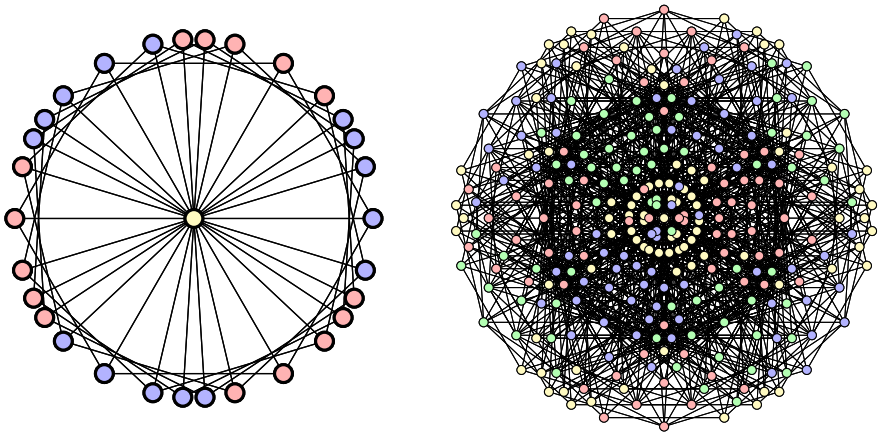
Definición 3.38. *Se define el* ***grafo W*** *como el grafo de distancia unitaria de 301 vértices, formado por la suma de Minkowski de dos copias de V, eliminando los vértices a distancia mayor de $\sqrt{3}$ del centro. Es decir, $W = \mathrm{trim}(V \oplus V, \sqrt{3})$*

Finalmente, construimos el grafo que servirá como M.



(a) Grafo B, con tres husos de Moser. (b) Grafo C, con tres copias del grafo B.

Figura 26: Construcción del grafo V.



(a) Grafo V. (b) Grafo W.

Figura 27: Grafos V y W.

Definición 3.39. *Se define el* ***grafo M*** *como el grafo de distancia unitaria de 1345 vértices, formado al unir 6 copias de W, centradas cada una en un vértice de H. Es decir, $M = W \oplus H$.*

Comprobamos ahora que el grafo M tiene la propiedad que necesitábamos:

Proposición 3.40. *No existe ninguna 4-coloración de M en la que el H central tenga 3 vértices del mismo color.*

Demostración. Es en este punto de la demostración en el que se requiere el uso de un ordenador. Para ello, de Grey implementó el siguiente algoritmo de coloración:

1. Primero, coloreamos el H central de M con una de las dos coloraciones con tres vértices del mismo color. Queremos ver que entonces no podremos colorear el resto de M solo con cuatro colores.

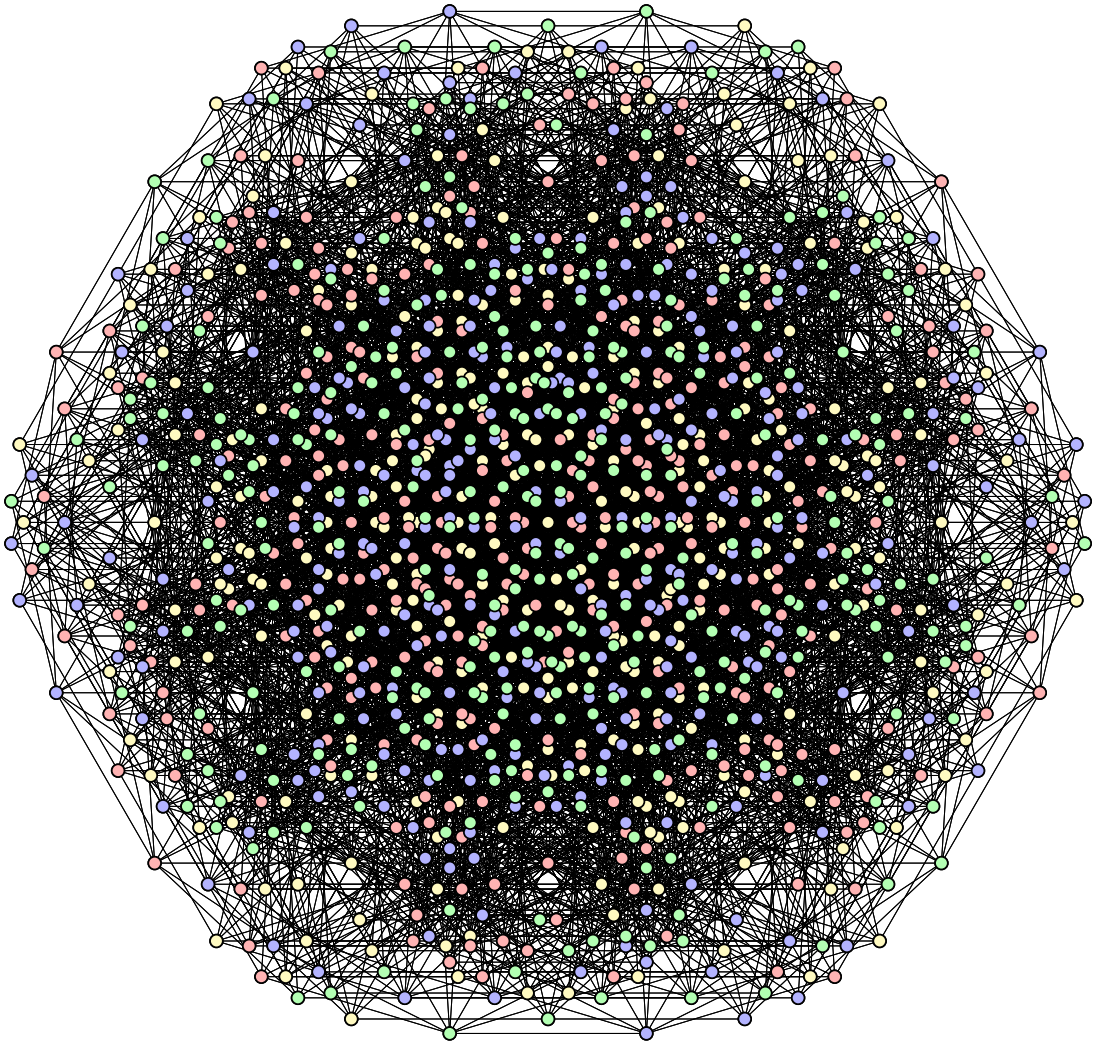


Figura 28: 4-coloración del grafo M, con 1345 vértices.

2. Ordenamos el resto de vértices según los siguientes parámetros, de mayor a menor importancia:
   - El número de husos de Moser que contienen el vértice.
   - El grado del vértice.
   - El número de triángulos equiláteros de lado 1 que contienen el vértice.
3. Coloreamos el primer vértice sin colorear, con el primer color que no hayamos probado aún.
4. Comprobamos si ese vértice tiene vecinos sin colorear que:
   - Ya tengan vecinos con los cuatro colores. En este caso, pasamos al paso 5.
   - Tengan vecinos con tres colores. En este caso, asignamos el cuarto color a este vértice y repetimos este paso 4 con él.
5. Si encontramos un vértice sin colores disponibles, revertimos todas las coloraciones

hechas en la última iteración de los pasos 3 y 4, y repetimos el paso 3 con el siguiente color. Si no quedan colores disponibles, descoloramos el vértice anterior a este.

6. Si por el contrario no hemos encontrado ningún vértice sin colores disponibles, aplicamos el paso 3 al siguiente vértice.

7. El algoritmo termina cuando o bien todos los vértices han sido coloreados (y por lo tanto, el grafo es 4-coloreable), o bien se tiene que descolorar algún vértice de H (y por lo tanto, no hay ninguna 4-coloración con esa coloración inicial de H).

Repitiendo el algoritmo con las dos posibles coloraciones de H con tres vértices del mismo color, obtenemos que en ninguno de los dos casos existe una 4-coloración de M con una terna monocromática. □

Este algoritmo está también implementado en el programa *UDGraphs* (ver Apéndice).

3.3.5. El grafo N

Finalmente, construimos un grafo 5-cromático al que llamaremos N.

Definición 3.41. *Se define el* ***grafo N*** *como el grafo de distancia unitaria de 20425 vértices, formado al unir 52 copias de M rotadas y desplazadas de forma que cada H central coincida con una copia diferente de H en L.*

Teorema 3.42. *El plano no es 4-coloreable. Es decir, $NCP \geq 5$.*

Demostración. Supongamos que N es 4-coloreable.

La Proposición 3.31 dice que toda 4-coloración de L tiene alguna copia de H con tres vértices del mismo color. Por otro lado, la Proposición 3.40 dice que ninguna 4-coloración de M tiene algún H con tres vértices del mismo color.

Como $L \subset N$, algún H tendrá que tener una terna monocromática. Pero como todas las copias de H forman parte de una copia de M, en ninguna 4-coloración habrá un H con tres vértices del mismo color. Por lo tanto, N no puede ser coloreado con 4 colores y $NCP \geq 5$. □

4. Reducción del grafo 5-cromático

En la sección anterior hemos demostrado que el número cromático del plano es mayor o igual que 5. Sin embargo, de esta demostración se pueden destacar dos cosas a mejorar:

- En una parte de la demostración se ha usado un ordenador para comprobar una propiedad del grafo M. Pese a ser esto válido, siempre se prefiere una demostración sin usar el ordenador, pues suele dar una visión más clara de por qué es cierto aquello que se quiere demostrar.
- El grafo N, que es 5-cromático, tiene 20425 vértices y 151311 aristas. Este grafo es válido para demostrar que el plano no es 4-coloreable, pero encontrar un grafo más simple (con menos vértices o aristas) sería de interés de cara a buscar un grafo 6-cromático.

En esta sección veremos estrategias para la construcción de dos grafos 5-cromáticos con menos vértices que N.

- El primero, de 1581 vértices y construido también por Aubrey de Grey, se construye mediante una serie de reducciones sobre el grafo N [5].
- El segundo, de 553 vértices y construido por Marijn Heule. Se obtiene con reducciones mediante solucionadores SAT [8].

4.1. Comprobación de k-cromaticidad

Para demostrar la Proposición 3.40 hemos utilizado un algoritmo de coloración mediante *backtracking*. Este consistía en colorear los vértices en orden y, al llegar a un vértice que no podía ser coloreado, deshacer el último paso y probar con otra coloración.

Pese a ser un algoritmo muy poco eficiente en general, para este caso concreto resultó suficientemente rápido. Esto se debe, sobre todo, a que con la coloración inicial del H central muchos vértices quedaban fijados. Sin embargo, en los grafos que trataremos en esta sección no tendremos este escenario.

Es por esto que conviene ver otra forma de demostrar que un grafo G es k-cromático: mediante **solucionadores SAT**. Los introducimos a continuación, junto a una base de lógica proposicional.

4.1.1. Fundamentos de lógica proposicional

A continuación se presentan unas definiciones básicas de lógica.

Definición 4.1. *Una **variable proposicional** x es una variable que puede ser verdadera o falsa.*

Se construyen las **fórmulas proposicionales** mediante el uso de los conectores $\neg$, $\wedge$, $\vee$, $\rightarrow$ y $\leftrightarrow$ mediante las siguientes reglas:

- Si p es una variable proposicional, p es una fórmula.

- Si φ es una fórmula proposicional, su negación $\neg\varphi$ también lo es.
- Si φ y ψ son fórmulas y $*$ es uno de los conectores $\wedge, \vee, \rightarrow$ o $\leftrightarrow$, entonces $(\varphi * \psi)$ también lo es.

Nos centraremos únicamente en los conectores $\neg$, $\wedge$ y $\vee$: negación, conjunción y disyunción. Se puede ver que los otros dos conectores, condicional y bicondicional, se siguen de estos tres.

Definición 4.2. *Sea* Var *un conjunto de variables proposicionales. Una* ***valoración*** *s es una función $s\colon \text{Var} \to \{0,1\}$ que asigna a cada variable los valores 0 (falso) o 1 (verdadero).*

Toda valoración s puede extenderse a una función s^* que asigne a cada fórmula un elemento de 0 o 1, siguiendo las siguientes reglas:

- $s^*(p) = s(p)$ si p es una variable proposicional.
- $s^*(\neg\varphi) = 1 - s^*(\varphi)$ si φ es una fórmula proposicional.
- $s^*(\varphi \wedge \psi) = \text{mín}\{s^*(\varphi), s^*(\psi)\}$ si φ y ψ son fórmulas proposicionales.
- $s^*(\varphi \vee \psi) = \text{máx}\{s^*(\varphi), s^*(\psi)\}$ si φ y ψ son fórmulas proposicionales.

Se puede representar los diferentes valores de las fórmulas según los valores de las subfórmulas que la forman mediante **tablas de verdad**.

φ	$\neg\varphi$
0	1
1	0

φ	ψ	$(\varphi \wedge \psi)$
0	0	0
1	0	0
0	1	0
1	1	1

φ	ψ	$(\varphi \vee \psi)$
0	0	0
1	0	1
0	1	1
1	1	1

Tabla 1: Tablas de verdad de la negación, conjunción y disyunción.

Observación 4.3. De las tablas se observa que:

- La conjunción de dos fórmulas es cierta si y solo si ambas fórmulas son ciertas.
- La disyunción de dos fórmulas es falsa si y solo si ambas fórmulas son falsas.

4.1.2. Forma normal conjuntiva y derivación

Notación 4.4. *Si x es una variable proposicional, representaremos la fórmula $\neg x$ como $\overline{x}$.*

Definición 4.5. *Un* ***literal*** *l es una variable proposicional x (literal* ***positivo****) o la negación de una variable proposicional, $\overline{x}$ (literal* ***negativo****). La variable proposicional del literal l se denota como* $\text{var}(l)$.

Definición 4.6. *Sea $x = \text{var}(l)$. Se define el* ***complemento*** *de un literal l como $\overline{l} = \overline{x}$ si l es positivo, y $\overline{l} = x$ si l es negativo.*

Definición 4.7. *Sea l un literal, y $x = \mathrm{var}(l)$. Una valoración s* ***satisface*** *el literal l si:*

- *l es positivo y $s(x) = 1$.*
- *l es negativo y $s(x) = 0$.*

En caso contrario, decimos que s ***no satisface*** *l.*

Definición 4.8. *Una* ***cláusula*** *C es una disyunción de literales, es decir, $C = l_1 \vee l_2 \vee \ldots \vee l_n$ para los literales l_i.*

Definición 4.9. *Una valoración s* ***satisface*** *la cláusula C si contiene un literal satisfecho por s. De lo contrario, decimos que v* ***no satisface*** *C.*

Observación 4.10. Como la cláusula $C = \emptyset$ no tiene ningún literal, en particular para toda valoración s, C no contiene ningún literal satisfecho y ninguna valoración satisfará C.

Introducimos ahora el concepto de forma normal conjuntiva:

Definición 4.11. *Una fórmula F está en* ***forma normal conjuntiva*** *(FNC) si es una conjunción de cláusulas. Es decir, $F = C_1 \wedge C_2 \wedge \ldots \wedge C_n$ para las cláusulas C_i.*

Definición 4.12. *Una valoración s* ***satisface*** *una fórmula F en forma normal conjuntiva si s satisface a todas las cláusulas de F.*

Definición 4.13. *Decimos que una fórmula F es* ***satisfacible*** *si existe alguna valoración s que satisfaga F. De lo contrario, diremos que F es* ***insatisfacible****.*

Definición 4.14. *Dadas dos fórmulas, F_1 y F_2, se dice que son* ***equisatisfacibles*** *si o bien son ambas satisfacibles, o bien son ambas insatisfacibles.*

Definición 4.15. *Una cláusula C es* ***redundante*** *respecto a una fórmula F si F y $F \wedge C$ son equisatisfacibles.*

Finalmente, definimos el concepto de derivación.

Definición 4.16. *Dada una fórmula $F = \{C_1, \ldots, C_m\}$, definimos la* ***derivación*** *de la cláusula C_n como la sucesión de cláusulas $C_{m+1}, \ldots, C_n$. Así, definimos la* ***fórmula acumulada*** *como $F_i = \{C_1, \ldots C_i\}$.*

Definición 4.17. *Una derivación es* ***correcta*** *si para todo $i > m$, C_i es redundante respecto a F_{i-1}.*

Definición 4.18. *Una derivación es una* ***demostración*** *de la fórmula F si de ella se deriva la cláusula vacía, $C = \emptyset$.*

Observación 4.19. Una demostración de una fórmula F certifica que esa fórmula es insatisfacible. En efecto, como el proceso de derivación resulta en fórmulas equisatisfacibles, F y $F \wedge \emptyset$ son equisatisfacibles. Como la cláusula vacía es insatisfacible, $F \wedge \emptyset$ será insatisfacible, y por equisatisfacibilidad, F también.

4.1.3. Formulación de la k-colorabilidad de un grafo

Podemos expresar la k-colorabilidad de un grafo G como una fórmula proposicional en forma normal conjuntiva, de modo que el grafo G sea k-coloreable si y solo si esta fórmula es satisfacible.

Empezamos considerando $k \cdot |G|$ variables proposicionales: una para cada vértice y cada posible color. Denotamos con $x_{v,c}$ a la variable proposicional del vértice v con el color c. Si $x_{v,c}$ es verdadera, v tendrá el color c, y si es falsa, v no tendrá el color c.

La fórmula tendrá dos tipos de cláusulas, que se corresponden a las dos condiciones de coloración de un grafo:

- Todo vértice tiene que estar coloreado. Es decir, para cada vértice $v \in G$:

$$C_v = x_{v,1} \vee x_{v,2} \vee \ldots \vee x_{v,k}$$

- No puede haber dos vértices adyacentes con el mismo color. Es decir, para cada par de vértices adyacentes v y w y para cada color c:

$$C_{v,w,c} = \overline{x}_{v,c} \vee \overline{x}_{w,c}$$

Definición 4.20. *Construimos la* ***fórmula de k-colorabilidad*** *de un grafo G como:*

$$G_k = \bigwedge_{v \in V(G)} C_v \wedge \bigwedge_{(v,w) \in E(G)} \bigwedge_{c \in \{1,\ldots k\}} C_{v,w,c}$$

Podríamos añadir un tercer tipo de cláusula:

- Cada vértice puede tener únicamente un color. Es decir, para cada vértice $v \in G$, y para cada par de colores c_1, c_2:

$$C_{v,c_1,c_2} = \overline{x}_{v,c_1} \vee \overline{x}_{v,c_2}$$

Vemos que estas cláusulas son redundantes:

Proposición 4.21. *Sea v un vértice de G y sean c_1 y c_2 dos colores distintos. Entonces, la cláusula C_{v,c_1,c_2} es redundante respecto a G_k.*

Demostración. Tenemos que ver que G_k y $F \coloneqq G_k \wedge C_{v,c_1,c_2}$ son equisatisfacibles.

Si F es satisfacible, entonces existe una valoración s tal que el vértice v tiene asignado un único color. Claramente, esa misma valoración satisface G_k.

Si G_k es satisfacible, entonces existe una valoración s tal que todo vértice tiene asignados uno o más colores. Sea c_0 un color asignado al vértice v con la valoración s. Definimos la siguiente valoración:

$$\tilde{s}(x_{w,c}) = \begin{cases} s(x_{w,c}) & \text{si } w \neq v \\ 1 & \text{si } w = v \text{ y } c = c_0 \\ 0 & \text{si } w = v \text{ y } c \neq c_0 \end{cases}$$

Entonces, $\tilde{s}$ satisface a F, demostrando así que G_k y F son equisatisfacibles. □

Observación 4.22. Si la fórmula G_k es satisfacible, la valoración s que la satisfaga nos da una k-coloración de G. En efecto, coloreando cada vértice w con el color c si y solo si $s(x_{w,c}) = 1$, se obtiene la k-coloración de G.

Observación 4.23. Dado un vértice v y un color c_0, podemos fijar su color añadiendo la siguiente fórmula a la fórmula G_k:

$$F_{v,c_0} = x_{v,c_0} \wedge \bigwedge_{c \neq c_0} \overline{x}_{v,c}$$

Para evitar simetrías de colores, podemos fijar el color de algunos vértices. Si hay tres vértices que formen un triángulo, por ejemplo, le podemos asignar hasta tres colores diferentes.

Observación 4.24. Si G_k es satisfacible pero G_{k-1} es insatisfacible, concluiremos que el grafo G es k-cromático.

4.1.4. Solucionadores SAT

Hemos reinterpretado el problema de encontrar el número cromático de un grafo por un problema de satisfacibilidad Booleana.

Definición 4.25. *Un **problema de satisfacibilidad Booleana** (o problema SAT) consiste en determinar si una fórmula proposicional es satisfacible o no.*

El problema de satisfacibilidad Booleana, igual que el problema de coloración de grafos, es un problema **NP-completo**. Esto es, no se conocen algoritmos que en un tiempo polinomial decidan si una fórmula es satisfacible o no.

Los problemas SAT han sido muy estudiados en las últimas décadas. Para resolverlos, tenemos herramientas más eficientes y optimizadas que el algoritmo de coloración de *backtracking*: los **solucionadores SAT**.

Además de servir para determinar el número cromático de los grafos, también podemos utilizar los solucionadores SAT para encontrar una k-coloración de un grafo G, y como veremos en la subsección 4.3, para encontrar grafos k-cromáticos más pequeños.

4.2. Reducción de de Grey

La construcción del grafo N se basa en aplicar operaciones a diferentes grafos, aprovechando sus propiedades. Aún así, los grafos utilizados en la sección anterior no son minimales: se les puede quitar algunos vértices manteniendo las propiedades que permiten asegurar que el grafo resultante es 5-cromático.

Ir eliminando vértices de N uno a uno y comprobando que el grafo sigue siendo 5-cromático es poco práctico. Por eso, de Grey usó diferentes estrategias para reducir el grafo N.

Reducción de M

El grafo N está formado por 52 copias del grafo M, usando la propiedad de que M no tiene 4-coloraciones con tres vértices del mismo color en el H central.

Así, una primera estrategia para reducir N sería deshacerse de aquellos vértices de M cuya eliminación no genere 4-coloraciones con ternas monocromáticas en el H central.

Reducción de S

Definición 4.26. *Definimos el* ***grafo*** S *como la unión de 13 copias del grafo* M*, de modo que los grafos* H *centrales de* M *formen el grafo* J*.*

Observación 4.27. El grafo N está formado por 4 copias de S.

La única propiedad del grafo J que se ha usado en la construcción de L es la recogida en la Observación 3.24, que fija las tres únicas configuraciones de colores de los vértices de enlace.

Así, podemos reducir el grafo S (y en consecuencia, el grafo N) prescindiendo de aquellos vértices de S que al ser eliminados no introduzcan nuevas coloraciones de los vértices de enlace, aunque introduzcan ternas monocromáticas.

En las estrategias que siguen a continuación, también permitiremos coloraciones con ternas monocromáticas.

Relajación de la condición de J

No hace falta que las coloraciones de los vértices de enlace sean exactamente los de la Observación 3.24.

Recordamos que en la demostración, las coloraciones de los vértices de enlace de J se usan para ver que en el grafo K todas las diagonales de enlace tienen que ser monocromáticas.

Podemos usar una condición más débil mediante la siguiente proposición:

Proposición 4.28. *Para que las diagonales de enlace del grafo* K *sean monocromáticas, basta con ver que tanto para* J *como para* S*, las únicas posibles coloraciones de los vértices de enlace cumplan sean:*

(a) Al menos cinco de los vértices de enlace son del mismo color que el vértice central.

(b) Exactamente cuatro vértices de enlace son del mismo color que el centro, y no todas las diagonales de enlace son monocromáticas.

(c) Dos vértices de enlace son del color del centro, y todas las diagonales de enlace son monocromáticas.

Demostración. En primer lugar, observamos que las dos copias de J usadas en construir K están centradas en el mismo punto, así que compartirán color de vértice central.

Para construir una coloración de K con estas reglas, intentamos asignar a cada vértice de J del color del centro, un vértice de $\theta_4(J)$ de otro color, que será su vecino.

No se puede colorear K usando en alguna de sus copias la coloración (a), pues hay al menos 5 vértices del color central, y ninguna de las tres posibles coloraciones de J tiene más de 4 vértices de otro color.

Tampoco se puede colorear K usando en alguna de sus copias la coloración (b). Esta coloración tiene 4 vértices del color central, así que necesita que la otra copia tenga 4 vértices de otro color. Pero la coloración (c) tiene todas las diagonales monocromáticas, y la (b) no.

Así, solo queda la opción de colorear ambas copias con la coloración (c), y entonces todas las diagonales de enlace de K son monocromáticas. □

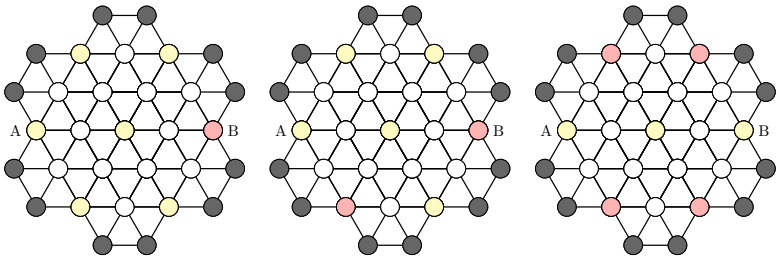


Figura 29: Coloraciones adicionales de los vértices de enlace de J.

Así, podremos eliminar de S vértices cuya ausencia mantenga una de las tres coloraciones de la Figura 21 o de las tres de Figura 29 para los vértices de enlace.

Relajación de la construcción de N

En la construcción de N, usamos que el grafo K tiene todas las diagonales de enlace monocromáticas, y llegamos a una contradicción al construir L.

En particular, teníamos dos diagonales de enlace monocromáticas unidas por un vértice, y los otros dos vértices eran vecinos. Vemos ahora que para que eso suceda no es necesario que todas las diagonales de enlace de K sean monocromáticas.

El grafo N está formado por cuatro copias de S. Lo construimos explícitamente:

Definición 4.29. *Sea el grafo $S_{a,1}$ una copia del grafo S centrada en $(0,0)$ con una de las diagonales de enlace siendo A y B. Definimos entonces $S_{b,1} \coloneqq \theta_4(S_{a,1})$.*

Definimos también $S_{a,2} \coloneqq \theta_{16}(S_{a,1}, A)$, y $S_{b,2} \coloneqq \theta_4(S_{a,2})$.

Con esta notación, $N = S_{a,1} \cup S_{b,1} \cup S_{a,2} \cup S_{b,2}$.

La condición de que K tuviera todas las diagonales monocromáticas se puede cambiar por estas dos condiciones:

1. Solo una de las tres diagonales de enlace del grafo $S_{a,1}$ tiene que ser monocromática. Sea $D = \{A, B\}$ esta diagonal de enlace. Entonces, $\theta_{16}(D, A) \subset S_{a,2}$ también tiene que ser monocromática.

2. Las 4-coloraciones de los grafos S_a y S_b deben ser incompatibles si la diagonal D no es monocromática.

Así, por ejemplo, si el grafo $S_{a,i}$ no tiene la diagonal D monocromática, pero tiene al menos un vértice de cada diagonal de enlace del color del centro, y además los $S_{b,i}$ tienen una diagonal de enlace monocromática del color del centro de $S_{a,i}$, entonces se llega a contradicción.

Adición de vértices

El grafo S contiene varios conjuntos de cuatro o más vértices situados en una circunferencia de radio 1 cuyo centro no pertenece a S.

Sea X un conjunto de vértices como el descrito. Supongamos que para todo vértice $v \in X$, si eliminamos v, el grafo resultante es 4-coloreable. Además, supongamos que existe un conjunto de vértices de al menos 2 vértices, $Y \subset S$, tal que al eliminar todos sus vértices, fuerza que se usen los 4 colores para colorear los vértices de X.

Entonces, eliminando los vértices de Y y añadiendo un vértice en el centro de la circunferencia de los vértices de X, mantendremos la 5-cromaticidad reduciendo el número de vértices en al menos uno.

4.2.1. El grafo G

Usando las estrategias descritas anteriormente, de Grey redujo el grafo N y obtuvo el grafo G, de 1581 vértices. Detallamos la construcción del grafo a continuación.

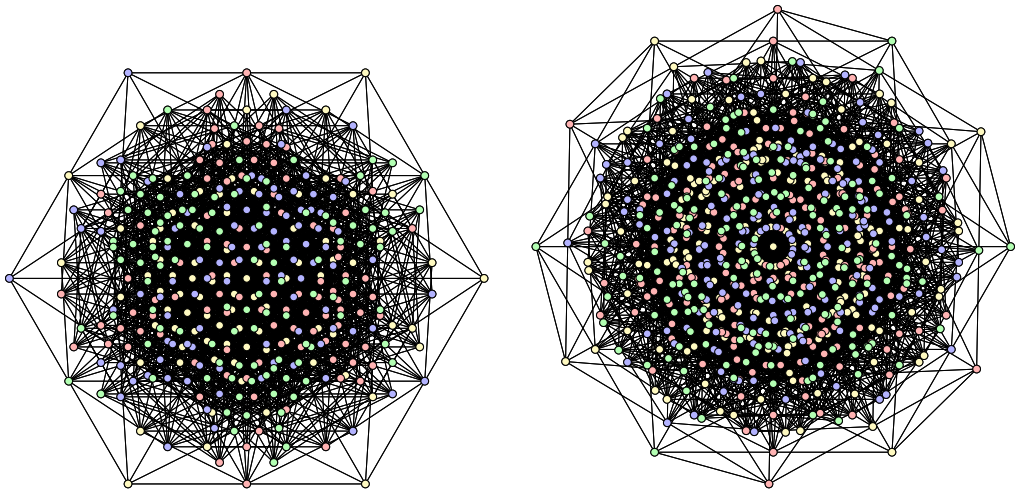
Sea C el conjunto de vértices del plano siguiente:

$$(0,0), \left(\frac{1}{3}, 0\right), (1,0), (2,0), \left(\frac{\sqrt{33}-3}{6}, 0\right), \left(\frac{1}{2}, \frac{1}{\sqrt{12}}\right), \left(1, \frac{1}{\sqrt{3}}\right), \left(\frac{3}{2}, \frac{\sqrt{3}}{2}\right),$$
$$\left(\frac{7}{6}, \frac{\sqrt{11}}{6}\right), \left(\frac{1}{6}, \frac{\sqrt{12}-\sqrt{11}}{6}\right), \left(\frac{5}{6}, \frac{\sqrt{12}-\sqrt{11}}{6}\right),$$
$$\left(\frac{2}{3}, \frac{\sqrt{11}-\sqrt{3}}{6}\right), \left(\frac{2}{3}, \frac{3\sqrt{3}-\sqrt{11}}{6}\right), \left(\frac{\sqrt{33}}{6}, \frac{1}{\sqrt{12}}\right),$$
$$\left(\frac{\sqrt{33}+3}{6}, \frac{1}{\sqrt{3}}\right), \left(\frac{\sqrt{33}+1}{6}, \frac{3\sqrt{3}-\sqrt{11}}{6}\right), \left(\frac{\sqrt{33}-1}{6}, \frac{3\sqrt{3}-\sqrt{11}}{6}\right),$$
$$\left(\frac{\sqrt{33}+1}{6}, \frac{\sqrt{11}-\sqrt{3}}{6}\right), \left(\frac{\sqrt{33}-1}{6}, \frac{\sqrt{11}-\sqrt{3}}{6}\right),$$
$$\left(\frac{\sqrt{33}-2}{6}, \frac{2\sqrt{3}-\sqrt{11}}{6}\right), \left(\frac{\sqrt{33}-4}{6}, \frac{2\sqrt{3}-\sqrt{11}}{6}\right),$$
$$\left(\frac{\sqrt{33}+13}{12}, \frac{\sqrt{11}-\sqrt{3}}{12}\right), \left(\frac{\sqrt{33}+11}{12}, \frac{\sqrt{3}+\sqrt{11}}{12}\right),$$
$$\left(\frac{\sqrt{33}+9}{12}, \frac{\sqrt{11}-\sqrt{3}}{4}\right), \left(\frac{\sqrt{33}+9}{12}, \frac{3\sqrt{3}+\sqrt{11}}{12}\right),$$
$$\left(\frac{\sqrt{33}+7}{12}, \frac{\sqrt{3}+\sqrt{11}}{12}\right), \left(\frac{\sqrt{33}+7}{12}, \frac{3\sqrt{3}-\sqrt{11}}{12}\right),$$
$$\left(\frac{\sqrt{33}+5}{12}, \frac{5\sqrt{3}-\sqrt{11}}{12}\right), \left(\frac{\sqrt{33}+5}{12}, \frac{\sqrt{11}-\sqrt{3}}{12}\right),$$
$$\left(\frac{\sqrt{33}+3}{12}, \frac{3\sqrt{11}-5\sqrt{3}}{12}\right), \left(\frac{\sqrt{33}+3}{12}, \frac{\sqrt{3}+\sqrt{11}}{12}\right),$$

$$\left(\frac{\sqrt{33}+3}{12}, \frac{3\sqrt{3}-\sqrt{11}}{12}\right), \left(\frac{\sqrt{33}-3}{12}, \frac{\sqrt{11}-\sqrt{3}}{12}\right),$$
$$\left(\frac{\sqrt{33}+1}{12}, \frac{\sqrt{11}-\sqrt{3}}{12}\right), \left(\frac{\sqrt{33}-1}{12}, \frac{3\sqrt{3}-\sqrt{11}}{12}\right),$$
$$\left(\frac{15-\sqrt{33}}{12}, \frac{\sqrt{11}-\sqrt{3}}{4}\right), \left(\frac{15-\sqrt{33}}{12}, \frac{7\sqrt{3}-3\sqrt{11}}{12}\right),$$
$$\left(\frac{13-\sqrt{33}}{12}, \frac{3\sqrt{3}-\sqrt{11}}{12}\right), \left(\frac{11-\sqrt{33}}{12}, \frac{\sqrt{11}-\sqrt{3}}{12}\right).$$

Definición 4.30. *Construimos el grafo S_a, de 397 vértices y 1974 aristas, considerando los vértices de C, rotándolos respecto a $(0,0)$ múltiplos de $\frac{\pi}{3}$ y reflejándolos respecto el eje x. Las aristas de S_a serán las que surjan entre vértices a distancia 1.*

Definimos también el grafo S_b como $S_b = \theta_4(S_a)$.



(a) El grafo S_a. (b) El grafo Y.

Figura 30: 4-coloración de los grafos S_a e Y.

Definición 4.31. *Construimos el grafo Y, de 791 vértices y 3938 aristas, como la unión de S_a y S_b, eliminando los vértices $\left(\frac{1}{3}, 0\right)$ y $\left(-\frac{1}{3}, 0\right)$.*

Definición 4.32. *Sea A el vértice $(-2, 0)$. Definimos los grafos $Y_a = \theta_{16}^{\frac{1}{2}}(Y, A)$ y $Y_b = \theta_{16}^{-\frac{1}{2}}(Y, A)$.*

Definición 4.33. *Finalmente, definimos el grafo 5-cromático G, de 1581 vértices y 7877 aristas como $G = Y_a \cup Y_b$.*

Mediante el uso de un solucionador SAT (por ejemplo, Glucose 4.0[3]) podemos comprobar que el grafo G es, en efecto, no coloreable con 4 colores.

[3] `http://www.labri.fr/perso/lsimon/glucose/`

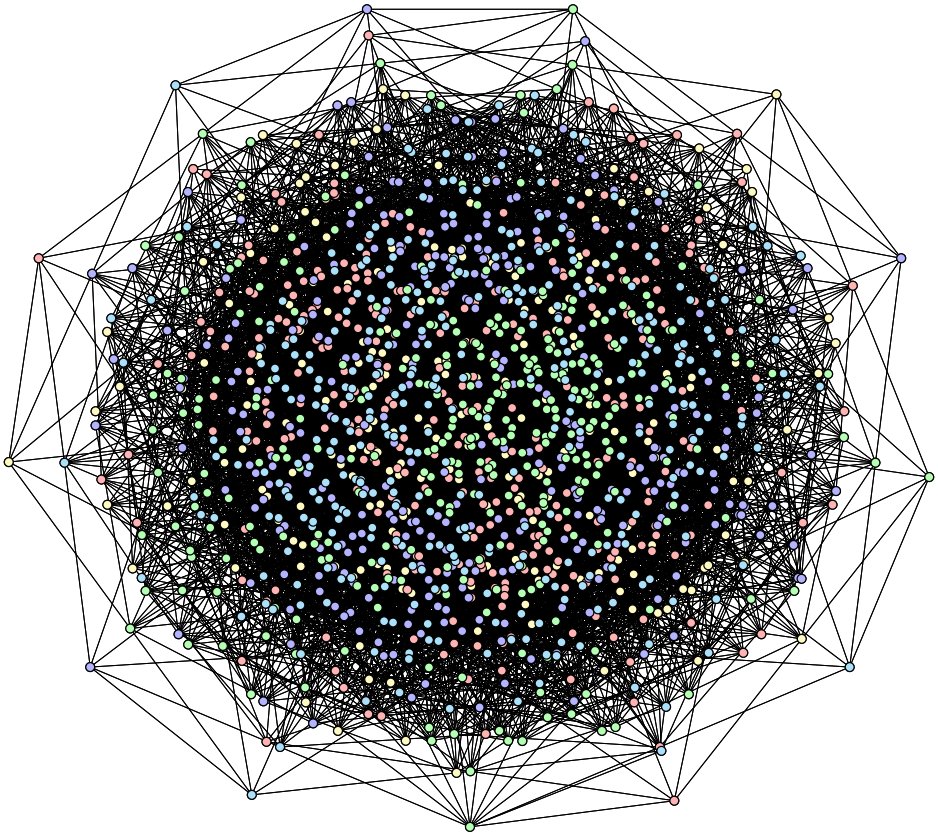


Figura 31: 5-coloración del grafo G.

4.3. Reducción de Heule

La estrategia de Marijn Heule se basa en el uso de solucionadores SAT para, partiendo de grafos 5-cromáticos, encontrar grafos más pequeños que tampoco puedan ser coloreados con 4 colores.

La mayoría de solucionadores SAT son capaces de generar certificados de insatisfacibilidad, en forma de una demostración como la definida en la Sección 4.1.2. Otro tipo de herramienta, los **comprobadores DRAT**, verifican la insatisfacibilidad de una fórmula F a partir de esa demostración. Algunos de ellos, además, pueden encontrar una subfórmula de F que siga siendo insatisfacible (por ejemplo, DRAT-Trim[4]), que dará lugar a un subgrafo del grafo original.

Esta subfórmula tendrá menos cláusulas que la fórmula original F. En el contexto de las fórmulas sobre la k-colorabilidad de un grafo, podemos decir lo siguiente:

- Si la cláusula $x_{v,1} \vee x_{v,2} \vee \ldots \vee x_{v,k}$ ha sido eliminada, se elimina el vértice v.
- Si la cláusula $\overline{x}_{v,c} \vee \overline{x}_{w,c}$ ha sido eliminada, se elimina la arista $\{v, w\}$.

La demostración que obtengamos del solucionador SAT y en consecuencia, el subgrafo resultante, dependerán del orden en el que se presenten las cláusulas en la fórmula inicial.

[4] `https://www.cs.utexas.edu/~marijn/drat-trim/`

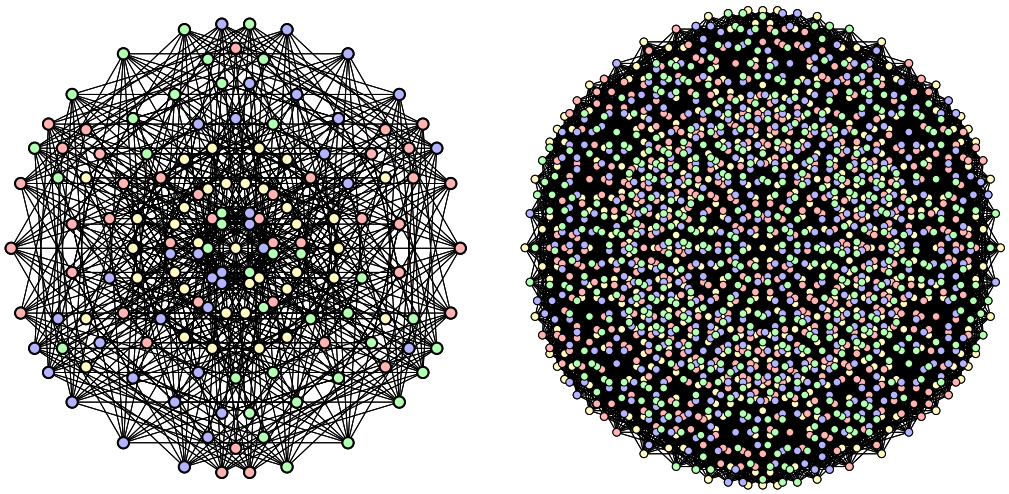
Así pues, la estrategia consistirá en ordenar aleatoriamente las cláusulas de la fórmula, y aplicar la reducción.

Estos subgrafos todavía se pueden reducir más: no todos sus vértices son necesarios para mantener la propiedad de ser 5-cromático. Para cada vértice v del subgrafo F, estudiamos el número cromático de $F \setminus v$, y si sigue siendo 5 lo eliminamos.

La construcción empieza con el mismo grafo V usado por de Grey. En esta sección nos referiremos a este por V_{31}, para diferenciarlo del resto de grafos que construyamos a continuación. A partir de él, construimos los siguientes grafos[5]:

Definición 4.34. *Se define el grafo de distancia unitaria V_{151}, de 151 vértices y 510 aristas, como el grafo recortado $V_{151} = \mathrm{trim}(V_{31} \oplus V_{31}, 1)$.*

Definición 4.35. *Se define el grafo de distancia unitaria V_{1939}, de 1939 vértices y 13374 aristas, como el grafo $V_{1939} = V_{31} \oplus V_{151}$.*



(a) Grafo V_{151}. (b) Grafo V_{1939}.

Figura 32: 4-coloración de los grafos V_{151} y V_{1939}.

Finalmente, construimos el primer grafo 5-cromático al que aplicaremos el método de reducción:

Observación 4.36. El grafo $V_{1939} \cup \theta_4(V_{1939})$, con 3877 vértices y 26814 aristas, es 5-cromático.

La mayoría de grafos obtenidos a partir de él tienen algo más de 800 vértices, y unos pocos tienen algo menos de 700. Heule observó que, de los grafos con menos de 700 vértices, casi 200 vértices pertenecían al cuerpo $\mathbb{Q}[\sqrt{3}, \sqrt{11}]$.

Los puntos en el cuerpo $\mathbb{Q}[\sqrt{3}, \sqrt{11}]$ presentan simetría: bajo reflexión respecto el eje horizontal, y bajo rotación de $\frac{\pi}{3}$.

Definición 4.37. *Construimos el grafo de distancia unitaria S_{199}, con 199 vértices y 888 aristas. Todos sus vértices pertenecen al cuerpo $\mathbb{Q}[\sqrt{3}, \sqrt{11}]$.*

[5]Los grafos 5-cromáticos encontrados por Marijn Heule resultantes de la reducción se encuentran en `https://github.com/marijnheule/CNP-SAT`

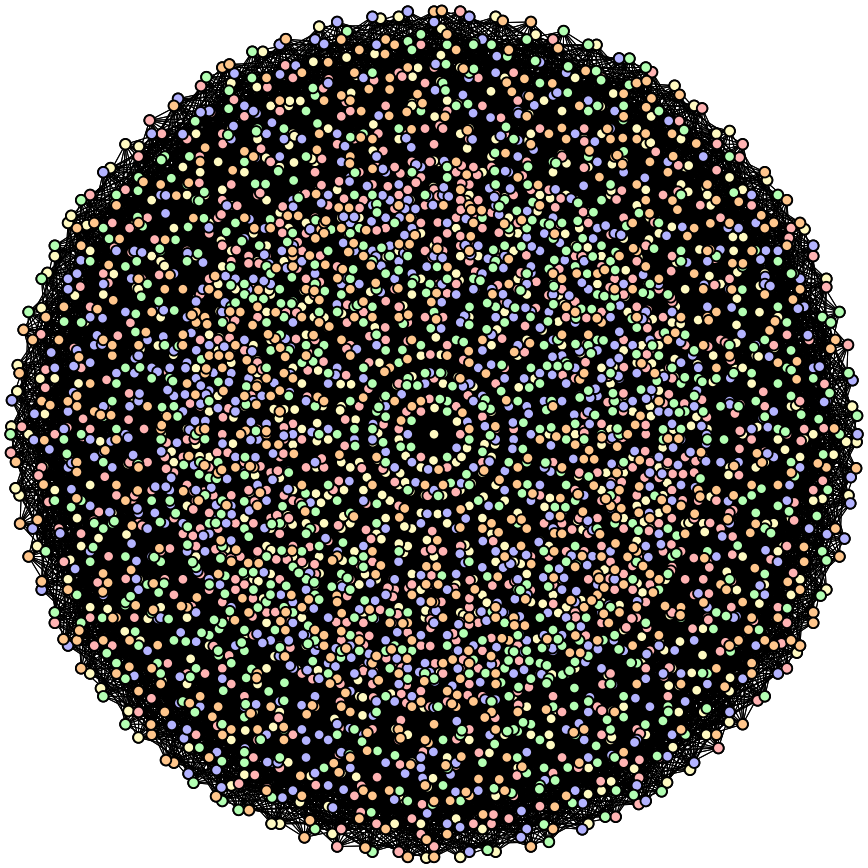


Figura 33: 5-coloración del grafo $V_{1939} \cup \theta_4(V_{1939})$.

Observación 4.38. Ninguno de los subgrafos reducidos contiene a S_{199} en su totalidad, pero les faltan unos pocos vértices.

Observación 4.39. S_{199} es un subgrafo de V_{1939}.

Estudiando las 4-coloraciones de $S_{199} \cup \theta_4(S_{199})$, se observan dos cosas:

- Vértices cercanos suelen tener el mismo color.
- Aproximadamente la mitad de vértices a distancia 2 del centro son del mismo color que el vértice central.

Realizando el proceso de reducción usando el grafo $V_{1939} \cup \theta_4(S_{199})$, que también es 5-cromático, obtenemos de forma consistente grafos con menos de 700 vértices.

El siguiente paso para reducirlos aún más consiste en unir dos de los grafos obtenidos mediante reducción (diferentes, o el mismo dos veces), rotar una de las dos copias, y volver a aplicar la reducción.

En general, esto daba lugar a grafos más grandes que sus dos componentes. Sin embargo algunas rotaciones en concreto dan lugar a muchas aristas nuevas, subiendo así el grado medio al grafo y permitiendo una mayor reducción.

Las rotaciones que mejor funcionaron fueron las de θ_1^k y θ_3^k, con k pequeños. Por ejemplo, la rotación $\theta_3^{\frac{1}{2}}$ creaba aristas entre los vértices a distancias 1, $\frac{\sqrt{33}+3}{6}$ y $\frac{\sqrt{33}-3}{6}$.

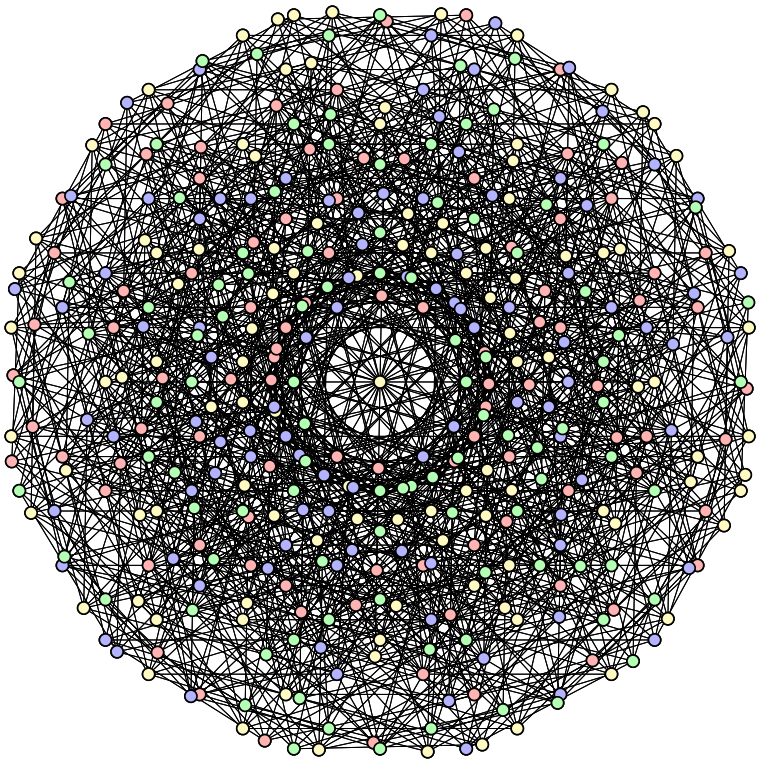


Figura 34: 4-coloración de $S_{199} \cup \theta_4(S_{199})$.

Usando este método, Heule logró un grafo con 610 vértices y 3000 aristas, al que llamaremos G_{610}.

Observación 4.40. Existe una 5-coloración de G_{610} vértices en la que el quinto color es usado únicamente para colorear el vértice central.

Los vértices de los grafos encontrados mediante estas reducciones se pueden separar en dos clases: los que pertenecen a un subgrafo de $\theta_4(S_{199})$ y los que no. Además, todos los vértices de los grafos más pequeños están a una distancia menor o igual que 2 del vértice central. Para lograr reducir aún más los grafos, Heule hizo lo siguiente:

- Primero, construye el grafo $\theta_4(S_{199}) \oplus \theta_4(S_{199})$.
- A este grafo, le elimina los vértices a distancia 2 o menos del centro, quedándose con 2028 vértices. Sea C el conjunto de vértices resultante.
- Finalmente, construye un grafo uniendo el grafo G_{610} con los vértices de C, y poniendo aristas entre los vértices a distancia 1.

Aplicándole la reducción a este grafo, Heule obtiene finalmente grafos 5-cromáticos con 553 vértices y unas 2720 aristas.

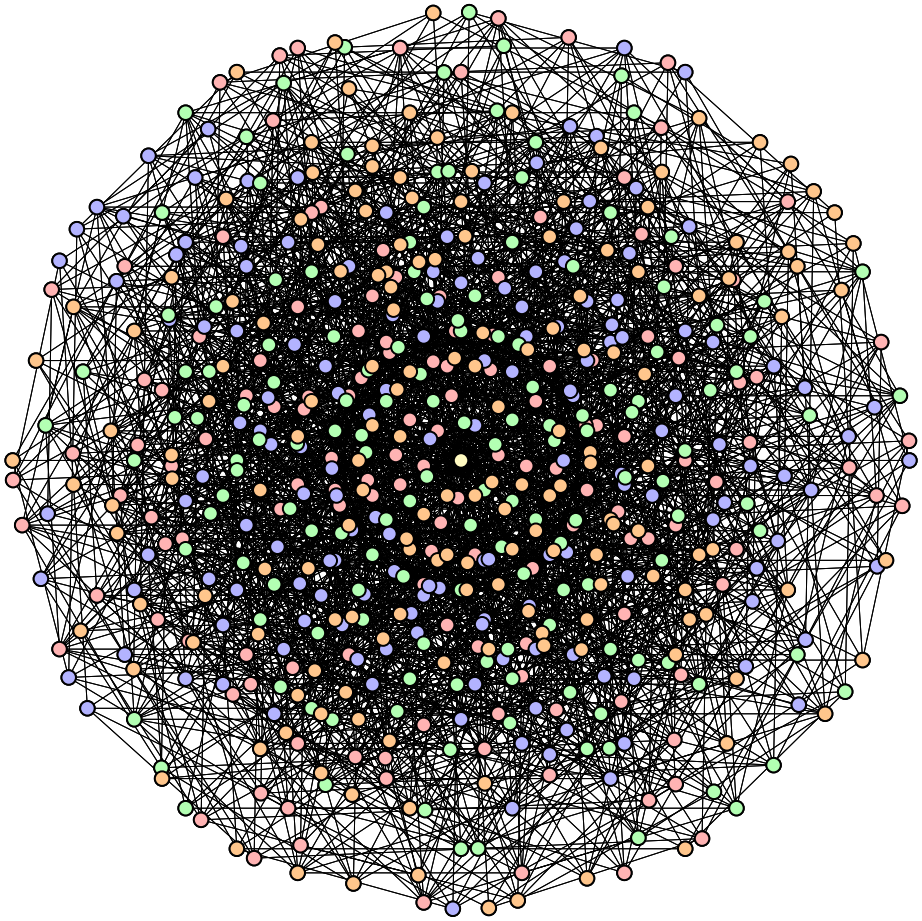


Figura 35: 5-coloración de G_{610}.

5. Estado del problema

Cuando en abril de 2018 Aubrey de Grey dio a conocer su grafo 5-cromático, se inició un Proyecto Polymath para el problema de Hadwiger-Nelson: el **proyecto Polymath16** [13].

Los proyectos Polymath consisten en la colaboración de un gran grupo de matemáticos para resolver conjuntamente problemas matemáticos abiertos.

El primer proyecto Polymath empezó en 2009, cuando Tim Gowers propuso como problema encontrar una nueva demostración del teorema de la densidad de Hales-Jewett. Gracias a las diferentes aportaciones, se llegaron a publicar artículos, que citaron a Polymath como D. H. J. Polymath (refiriéndose al problema *Density Hales-Jewett*)

El proyecto Polymath16, dirigido por Aubrey de Grey y Dustin Mixon, tiene tres objetivos principales:

- Encontrar grafos de distancia unitaria 5-cromáticos más pequeños.
- Reducir el uso del ordenador para la demostración de $NCP \geq 5$.
- Usar estos grafos 5-cromáticos para obtener diferentes resultados, como un grafo

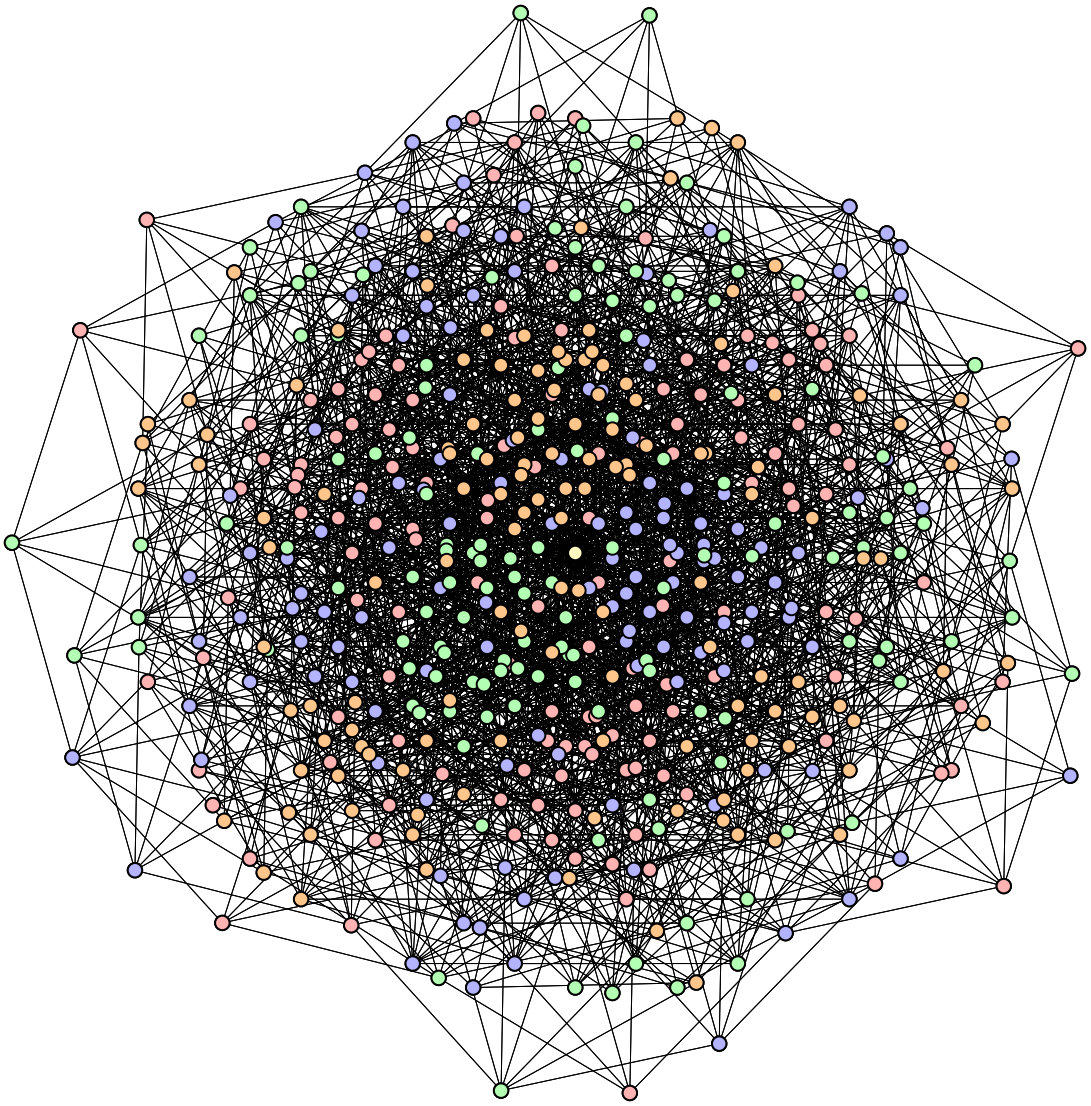


Figura 36: 5-coloración del grafo G_{553}.

6-cromático o mejorar las cotas en otras dimensiones.

Las diferentes reducciones de grafos 5-cromáticas vistas en el trabajo, por ejemplo, fueron comentadas en el blog de Dustin Mixon[6] (que concentra la actividad del Polymath) a medida que se iban descubriendo.

A continuación vemos algunos resultados referentes a la mejora de las cotas y las cotas en otras dimensiones.

5.1. Mejora de las cotas del plano

Para mejorar las cotas de NCP hay dos opciones: o bien mejorar la cota superior a 6, o mejorar la cota inferior a 6.

[6] `https://dustingmixon.wordpress.com/`

Para mejorar la cota inferior, una forma sería encontrar un grafo de distancia unitaria 6-cromático. A continuación vemos otra forma de demostrar que no puede haber una 5-coloración del plano.

Empezamos definiendo el concepto de coloración por embaldosado:

Definición 5.1. *Definimos el una coloración por* ***embaldosado*** *como una partición del plano en regiones con área positiva y acotadas por una curva cerrada simple.*

En 2005, Stephen P. Townsend publicó la demostración del siguiente teorema, corrigiendo un error en la demostración de Douglas R. Woodall [16, 19].

Teorema 5.2 (Teorema de Townsend-Woodall)**.** *Toda 5-coloración de un mapa planar contiene dos puntos a distancia 1 del mismo color.*

Este teorema implica el siguiente resultado:

Teorema 5.3. *No hay ninguna 5-coloración del plano mediante un embaldosado.*

Aun así, esto no descarta la existencia de una 5-coloración del plano que no sea mediante embaldosado. Si se refutara la existencia de tal coloración, la cota inferior del NCP subiría a 6, pese a no haber encontrado un grafo de distancia unitaria 6-cromático.

Respecto a reducir la cota superior, en 1999 Carsten Thomassen demostró el siguiente resultado [18]:

Teorema 5.4. *Dada una 6-coloración del plano por embaldosado:*

- *El diámetro máximo de cualquier región es exactamente 1.*
- *La distancia entre dos regiones con el mismo color es exactamente 1.*

En un embaldosado como este, se evita que haya puntos a distancia uno con el mismo color mediante la coloración adecuada de los bordes de las regiones.

Si el número cromático del plano resultara ser 7 (y por lo tanto existiera un grafo de distancia unitaria), este tendría que tener un número elevado de vértices. En 1998, Dan Pritikin publicó el siguiente resultado [14]:

Teorema 5.5. *Todo grafo de distancia unitaria con menos de 6197 vértices es 6-coloreable.*

En julio de 2018, Jaan Parts publicó en el Polymath una mejora de este resultado:

Teorema 5.6. *Todo grafo de distancia unitaria con menos de 6906 vértices es 6-coloreable.*

Los posibles grafos 6-cromáticos no tienen a priori por qué ser tan grandes. El siguiente resultado también fue publicado en el Polymath por Jaan Parts:

Teorema 5.7. *Todo grafo de distancia unitaria con menos de 24 vértices es 5-coloreable.*

5.2. Cotas para dimensiones superiores

El problema más conocido y estudiado es el de coloración del plano, pero el problema se puede plantear para otros espacios métricos.

En la Sección 3 vimos que el número cromático de $\mathbb{R}$ es 2, y que el número cromático de $\mathbb{R}^2$ está entre 5 y 7.

En 1970, Dmitry E. Raiskii demostró la siguiente cota inferior del número cromático de $\mathbb{R}^n$ [15].

Teorema 5.8. *Para $n \geq 2$, se tiene la siguiente cota inferior del número cromático de $\mathbb{R}^n$:*

$$n + 2 \leq \chi(\mathbb{R}^n)$$

Para el caso $\mathbb{R}^3$, en el año 2000 Oren Nechushtan dio una una cota inferior de 6 [11], y en 2002, David Coulson publicó la cota superior de 15 [2].

Para dimensiones superiores, las cotas son mucho peores:

- Para $\mathbb{R}^4$, se tiene una cota inferior de 9 y superior de 54.
- Para $\mathbb{R}^5$, se tiene una cota inferior de 9 y superior de 156.
- Para $\mathbb{R}^6$, se tiene una cota inferior de 12 y superior de 462.

Las dos últimas cotas superiores se encuentran en aportaciones a Polymath, siendo la cota de 156 de Aubrey de Grey y la de 462 de Philip Gibbs.

Alexander Soifer, un conocido matemático que ha mostrado un gran interés por este problema, conjeturó lo siguiente [16]:

Conjetura (de Alexander Soifer)**.**

- *El número cromático del plano es $NCP = 7$.*
- *El número cromático del espacio es $\chi(\mathbb{R}^3) = 15$.*
- *En general, el número cromático de espacio euclídeo de dimensión $n \geq 2$ es*

$$\chi(\mathbb{R}^n) = 2^{n+1} - 1$$

6. Conclusiones

La mejora en la cota inferior ha dado a conocer el problema de Hadwiger-Nelson a muchos matemáticos, invitándoles a participar en la resolución de este problema.

Un año después, la investigación sigue activa, aunque inevitablemente de forma más relajada. Para abordar el problema, ahora hay varios frentes abiertos, como por ejemplo:

- Formulación probabilística del problema de Hadwiger-Nelson.
- Formulación algebraica del problema de Hadwiger-Nelson.
- Búsqueda de una 6-coloración por embaldosados.
- Búsqueda de un grafo 6-cromático.
- Mejora del número cromático fraccionario del plano.
- Mejora de las cotas para otros espacios métricos.

Aunque el proyecto Polymath no haya dado una solución al problema del número cromático del plano, los resultados obtenidos para las diferentes ramas del problema permiten ver la eficacia de estas plataformas.

La pregunta ahora es: ¿veremos en un futuro cercano otra mejora de la cota de este interesante problema?

Apéndice: Programación de grafos de distancia unitaria

Para complementar este trabajo, se adjunta un programa para trabajar con grafos de distancia unitaria.

Esta aplicación ha sido programada usando Python 3.7. Se ha elegido este lenguaje de programación por varios motivos:

- Experiencia con el lenguaje.
- Comodidad a la hora de hacer pequeños proyectos.
- Fácil uso de librerías.

Los grafos han sido construidos mediante la librería *NetworkX*. Esta librería se encarga de la estructura del grafo: el almacenamiento del grafo en sí y los procesos de añadir y eliminar vértices y añadir aristas.

Además, la aplicación tiene una interficie gráfica hecha mediante la librería *PyQt5*.

Tanto el código como una versión ejecutable para Windows se encuentran disponibles en el siguiente repositorio de GitHub: `https://github.com/sergioBarril/UDGraphs/releases`

Para poder usar las funciones de dibujo, la aplicación necesitará usar los comandos "*pdflatex*" y/o "*lualatex*". Si no hay una distribución de LaTeX instalada, el programa generará el .tex pero no lo compilará.

Para su implementación, esta aplicación hace uso de diferentes módulos, que detallaremos a continuación:

Vertex.py

Este módulo define los vértices de los grafos. Los grafos de la librería *NetworkX* permiten que objetos de cualquier clase, siempre que sean hasheables, hagan de nodos.

En este módulo se define la clase `Vertex`, con los siguientes atributos:

- Coordenadas x e y del vértice.
- Color del vértice. Un vértice está coloreado si tiene el atributo color mayor que 0.

Para que dos vértices que representen el mismo punto no se añadan por duplicado, redefinimos la igualdad de vértices: dos vértices serán considerados el mismo si coinciden las coordenadas de ambos vértices redondeadas al tercer decimal.

Finalmente, se definen diferentes operaciones:

- Suma y resta de vértices.
- División de un vértice por un número.
- Cálculo de la distancia euclídea entre dos vértices.
- Rotación del vértice.

Graphs.py

Este módulo define los grafos de distancia unitaria con la clase `UnitDistanceGraph`, añadiendo diversas operaciones entre ellos. Además, construye los diferentes grafos vistos durante el trabajo en sus propias clases, que heredan de `UnitDistanceGraph`.

Los atributos de la clase `UnitDistanceGraph` serán:

- El número de vértices y aristas del grafo.
- El grafo en cuestión, usando la librería *NetworkX*.
- Una lista con los nodos ordenados según ciertos parámetros.

En primer lugar, tendremos la función `update(self)`, que actualizará el número de vértices y aristas del grafo.

Las siguientes funciones básicas son útiles a la hora de construir grafos:

- `add_node(self, v)` añade un vértice al grafo.
- `add_edge(self, v, w)` añade una arista entre dos vértices. Antes de añadirla, comprueba que, en efecto, ambos vértices están a distancia 1 entre sí.
- `copy_edge(self, v, w)` añade una arista entre dos vértices, ignorando si están a distancia 1 o no. Únicamente será usada para procesos de transformación de grafos donde sepamos con seguridad que los vértices acabarán siendo vecinos.
- `remove_node(self, v)` elimina un vértice del grafo (y todas las aristas incidentes en él).

A partir de un grafo, podemos usar las siguientes operaciones para obtener nuevos grafos:

- `rotate(self, i, k = None, center = None)` rota todos los vértices de un grafo un cierto ángulo. Con el argumento `center` se especifica el centro de rotación (si no hay, se usa el vértice $(0, 0)$). Si hay argumento `k`, se efectuará la rotación θ_i^k. De lo contrario, `i` representará el ángulo en radianes.
- `translate(self, diff)` efectúa una traslación de todos los vértices del grafo. `diff` será un objeto de la clase `Vertex`, que se sumará a todos los vértices del grafo.
- `reflection(self, c)` refleja el grafo respecto al eje vertical $x = c$.
- `trim(self, d)` elimina todos los vértices a distancia mayor que d.
- `union(self, G)` hace la unión de dos grafos, y uniendo con aristas todos los vértices a distancia 1 entre sí.
- `minkowskiSum(self, G)` hace la suma de Minkowski entre dos grafos. Primero comprueba si los grafos tienen el vértice $(0, 0)$, cosa que aceleraría el proceso: fijarían ciertos vértices y aristas.

  El resto de vértices se añaden como en la definición de la suma de Minkowski: sumando vértices de ambos grafos, y si el vértice resultante es nuevo, se mira si es vecino a alguno de los otros.

La aplicación implementa también el algoritmo de coloración usado en la Demostración 3.40. La implementación del algoritmo se da en *color.py*, pero la ordenación de los vértices se implementa en este módulo, pues también se usa para el módulo *sat.py*.

- `spindles(self)` retornará el número de husos de Moser diferentes contenidos en el grafo.
- `num_triangles(self, v)` calcula el número de triángulos equiláteros de lado 1 contienen al vértice v en el grafo.
- `get_rhombus(self, u, v, w)` toma como argumentos los tres vértices de un triángulo equilátero. Mediante rotaciones, encuentra los cuartos vértices que generen un rombo con todas las aristas 1.

  Retorna un diccionario que a cada vértice del triángulo inicial le asigna el vértice del rombo a distancia $\sqrt{3}$.

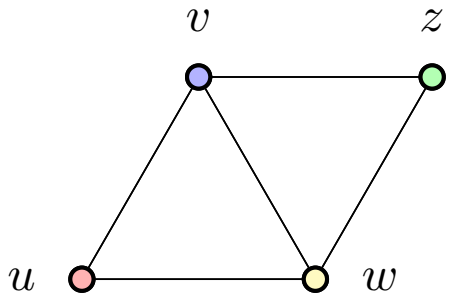


Figura 37: Ejemplo de rombo. En el diccionario se añadiría $u : z$.

- `is_valid_rhombus(self, sign, tip1, tip2, v, w)` comprueba si una rotación del rombo dado por los cuatro vértices con un ángulo en el sentido dado por `sign` da lugar a otro rombo que forme un huso de Moser con este.
- `spindles_in_rhombus(self, tip1, tip2, v, w, tip_mode=True)` retorna el número de husos de Moser contenidos en el grafo, y que contengan el rombo formado por `tip1, tip2, v` y `w`.

  Para ello, llama a la función `is_valid_rhombus`, aplicando una rotación de θ_3 o θ_3^{-1} al rombo respecto ambas puntas.

  Uno de los vértices del rombo será aquel vértice del que queremos saber en cuántos husos está contenido. Si ese vértice es una de las puntas, `tipMode` será `True`, y de lo contrario, `False`.

  Si no es una de las puntas, entonces ese mismo vértice del rombo formará parte de ambos triángulos. Como entonces será contado dos veces, dividimos la aportación total entre dos.

  Si es de una de las puntas, entonces al rotar con ese vértice como centro obtenemos nuevos rombos que lo contienen. Si rotáramos en ambos sentidos, estaríamos repitiendo los rombos. Por eso, si estamos en `tipMode` rotaremos respecto al vértice únicamente en sentido antihorario.

- `num_spindles(self, A)` retorna el número de husos de Moser que contienen el vértice A.

  Para ello, primero busca los diferentes triángulos que contienen el vértice A. Para cada uno, encuentra los diferentes rombos que contienen el vértice A y contenidos en el grafo. Finalmente, para cada rombo retorna el número de husos que lo contienen.

- `node_score(self, v)` retorna la puntuación del vértice `v` con la que se ordenarán. Esta puntuación vendrá dada por:

  $$\text{Puntuación} = 100 \cdot \text{Número de husos} + 10 \cdot \text{Grado del vértice} + \text{Número de triángulos}$$

Respecto a la coloración tenemos dos funciones:

- `color_graph(self, colors = 4, new = True)`, que mediante el módulo *color.py* colorea el grafo usando el algoritmo de *backtracking*.
- `sat_color(self, colors = 4)`, que mediante el módulo *sat.py* colorea el grafo usando fórmulas en forma normal conjuntiva y un solucionador SAT.

La función `draw_graph(self, fname, hard = False)` usa el módulo *tikz.py* para dibujar el grafo.

La función `cnf_trimming(self, cnfFile, dictFile)` toma como argumentos un fichero con una fórmula en FNC y un fichero diccionario que relaciona vértices e índices. Con ellos, se reduce el grafo eliminando los vértices v para los que la cláusula $x_{v,1} \vee \ldots \vee x_{v,k}$ no aparezca en la fórmula.

Finalmente, tiene las siguientes funciones guardar archivos:

- `save_graph(self, fname)` guarda los vértices del grafo en graphs\fname.v, y las aristas en graphs\fname.e.
- `save_dimacs(self, fname)` guarda las aristas del grafo en formato DIMACS (formato en el que se suelen representar los grafos) en el fichero dimacs\fname.col, y un fichero diccionario que relaciona vértices e índices en dimacs\fname.dict.
- `save_cnf(self, fname, colors, = 4, random = False)` guarda la fórmula en forma normal conjuntiva que describe la k-colorabilidad del grafo (donde $k =$ `colors`) en cnf\fname.cnf. Además, guarda un fichero diccionario que relaciona vértices e índices en cnf\fname.dict.

Además, este módulo contiene las construcciones de los diferentes grafos vistos en el trabajo.

Color.py

Este módulo implementa el algoritmo de coloración por *backtracking* explicado en el trabajo mediante la clase `ColoringGraph`.

El constructor de esta clase tiene los siguientes argumentos:

- `UDGraph`, que será el objeto de la clase `UnitDistanceGraph` a colorear.

- `colors`, que será el número máximo de colores que se pueden usar.
- `new`, que será un valor de Cierto o Falso. Si es cierto, se ignorará la coloración previa de `UDGraph` y se reiniciarán todos los vértices a -1 antes de colorear. Esto se utiliza a la hora de preservar la coloración del H central cuando se comprueba la propiedad del grafo M.

La clase tendrá los siguientes atributos:

- `self.graph` que contendrá el grafo equivalente a `UDGraph` pero expresado con índices.
- `self.vtoid` y `self.idtov` son diccionarios que emparejan cada vértice de `UDGraph` con un índice.
- `self.colors` y `self.new`.
- `self.banned_colors` es un diccionario que asigna a cada vértice una lista (inicialmente vacía). Esta lista guardará para cada vértice los colores que ya hemos probado, y que nos han hecho hacer *backtracking*.
- `self.decolorable_nodes` es un diccionario que asigna a cada vértice una lista (inicialmente vacía). Esta lista guardará para cada vértice v los nodos que hemos coloreado a partir de v. Así, cuando tengamos que decolorar el vértice v, también decoloraremos todos los vértices de la lista.

Observamos finalmente las funciones más importantes del módulo:

- `translate_graph(self, UDGraph)`, que genera los diccionarios que relacionan vértices e índices: `self.idtov` y `self.vtoid`.
- `copy_graph(self, UDGraph)`, que genera el grafo `self.graph` con los índices de `self.vtoid`.
- `available_colors(self, v)`, que retorna una lista con los colores disponibles para `v`: los colores que no estén presentes en los vecinos de v y que no estén en `self.banned_colors`.
- `color_node(self, v)`, que colorea el vértice v. Comprueba además todos sus vecinos: si hay alguno que no pueda ser coloreado, descolorea v y prueba con otro color. Si hay algún vecino con un único color disponible, se colorea con ese color usando esta misma función.

  Si se llegan a probar todos los colores disponibles de v sin encontrar un color que no dé lugar a nodos sin colores disponibles, retornará `False`: habrá que hacer *backtracking* y recolorear otro vértice.
- `uncolor_node(self, v)`, que decolorea el vértice v y todos los vértices coloreados a partir de él.
- `color_graph(self)`, que colorea el grafo. La lista `colored_nodes` va guardando los vértices que hemos coloreado directamente desde esta función (y no sus vecinos).

  Esta función recorre la lista de vértices, y los va coloreando mediante el método `color_node`. Si en algún momento se necesita hacer *backtracking* y recolorear un

vértice `v` anterior, se añade el color de ese vértice a `self.banned_colors[v]`, y se prueba con otro color.

Si ese no tuviera otros colores disponibles, se reinicia la lista y se recolorea un vértice anterior.

La función terminará cuando no se pueda hacer *backtracking* a ningún vértice (es decir, cuando `colored_nodes` esté vacía) o cuando todos los vértices hayan sido coloreados correctamente.

Retorna `True` si se ha podido colorear; de lo contrario retorna `False`.

- `color(self)`, que intenta colorear el grafo `self.graph`. Si lo logra, colorea también los vértices del grafo original.

Sat.py

Este módulo implementa la k-coloración de grafos mediante el uso de fórmulas lógicas en Forma Normal Conjuntiva (FNC) mediante la clase `UDGSat`.

Definición. *Sea G un grafo, n el número de vértices de G, y k el número máximo de colores a usar. Escribiremos la variable que determina que el vértice con índice i tiene color c, $x_{i,c}$, como $i + (c-1) \cdot n$.*

El constructor de esta clase tendrá los siguientes argumentos:

- `UDG`, que es un objeto de la clase `UnitDistanceGraph`.
- `colors`, que es el máximo número de colores a usar.

La clase tendrá los siguientes atributos:

- `self.UDG`, con el grafo a k-colorear.
- `self.k`, con el número máximo de colores que se pueden usar.
- `self.toId` y `self.toNode`, con los diccionarios que relacionan vértices e índices.
- `self.cnf`, que es una lista (que representa la fórmula en FNC) cuyos elementos son listas (que representan sus cláusulas).

La primera función es `translate(self)`, que genera los diccionarios `self.toId` y `self.toNode`.

Las siguientes funciones se encargan de escribir las cláusulas:

- `hasColor_clause(self)` añade las cláusulas que determinan que todo vértice tiene algún color. Esto es,
$$x_{i,1} \vee x_{i,2} \vee \ldots \vee x_{i,k}$$
- `triangle_clause(self)` añade las cláusulas que determinan hasta 3 colores iniciales para el grafo. Para ello, usando el orden de *graphs.py*, se localiza el vértice con mayor puntuación.

  Se mira cada triángulo que contenga a este vértice, y se colorea el triángulo cuyos vértices tengan más puntos. Es decir, se añaden las cláusulas $x_{i,c}$ para los vértices y colores que queremos fijar, y $\overline{x}_{i,c}$ para los colores que no queremos fijar.

- `edgeColor_clause(self)` añade las cláusulas que determinan que dos vértices vecinos no pueden tener el mismo color.

La función `solve(self, color = False)` desordena las cláusulas (para poder llegar a tener diferentes coloraciones), y usando la librería *pycosat*, determina la satisfacibilidad de la fórmula. Si es satisfacible y `color` es `True`, colorea el grafo.

Finalmente, la función `save_cnf(self, fname, randomMode = False)` escribe la fórmula FNC en el fichero cnf\fname.cnf y el diccionario de vértices e índices en el fichero cnf\fname.dict. Si `randomMode` es `True`, se reordenan las cláusulas antes de guardarlas en el fichero.

Tikz.py

Este módulo implementa la opción de dibujar los grafos. Para ello, se usa la librería de LaTeX *tkz-graph*, incluida con el programa.

Todas las imágenes de grafos en el trabajo están hechas usando o bien la librería *tkz-graph* manualmente, o bien usando este programa.

Esta funcionalidad está implementada en la clase `TikzDocument`.

El constructor de esta clase tendrá los siguientes argumentos:

- `fname`, que es el nombre del archivo a crear.
- `UDGraph`, que es un objeto de la clase `UnitDistanceGraph`.
- `factor`, que ajusta la longitud de las aristas. Así, haciendo el número más grande, los vértices se dispersan manteniendo todas las aristas con la longitud dada factor y se puede apreciar mejor el grafo.

La clase tendrá los siguientes atributos:

- `self.G`, con el grafo a dibujar.
- `self.fname`, con el camino hacia el fichero .tex a escribir.
- `self.pdfname`, con el camino hacia el fichero .pdf a generar.
- `self.factor`, con el factor por el que multiplicar la longitud de las aristas.
- `self.vtoid`, con el diccionario que relaciona vértices e índice.

Las siguientes funciones escribirán el archivo .tex:

- `open_new(self)` genera un nuevo archivo .tex, escribiendo su preámbulo y los primeros comandos.
- `add_nodes(self)` añade a ese archivo .tex los vértices del grafo. Introduce el comando \Vertex[x = x, y = y]{i}, donde x e y son las coordenadas del vértice, e i es su índice.
- `add_edges(self)` añade a ese archivo .tex las aristas del grafo. Introduce el comando \Edge(i)(j), donde i y j son los índices de dos vértices vecinos.

- `add_colors(self)` añade a ese archivo .tex los colores de los vértices. Introduce el comando \AddVertexColor$\{c\}\{i, j, \ldots\}$ para $i, j, \ldots$ vértices con el color c

El método `run(self, hard = False)` compilará el archivo .tex usando el comando `pdflatex`. Si se trata de un grafo difícil de dibujar (indicado por `hard = True`), se usará el comando `lualatex`.

Una vez compilado, se abrirá el archivo pdf en el lector de pdf.

Finalmente, la función `draw(self, hard = False)` llamará en orden al resto de métodos.

Main.py

Este módulo implementa la interficie gráfica de la aplicación, usando la librería de Python *PyQt5*. Esta interficie tiene tres ventanas:

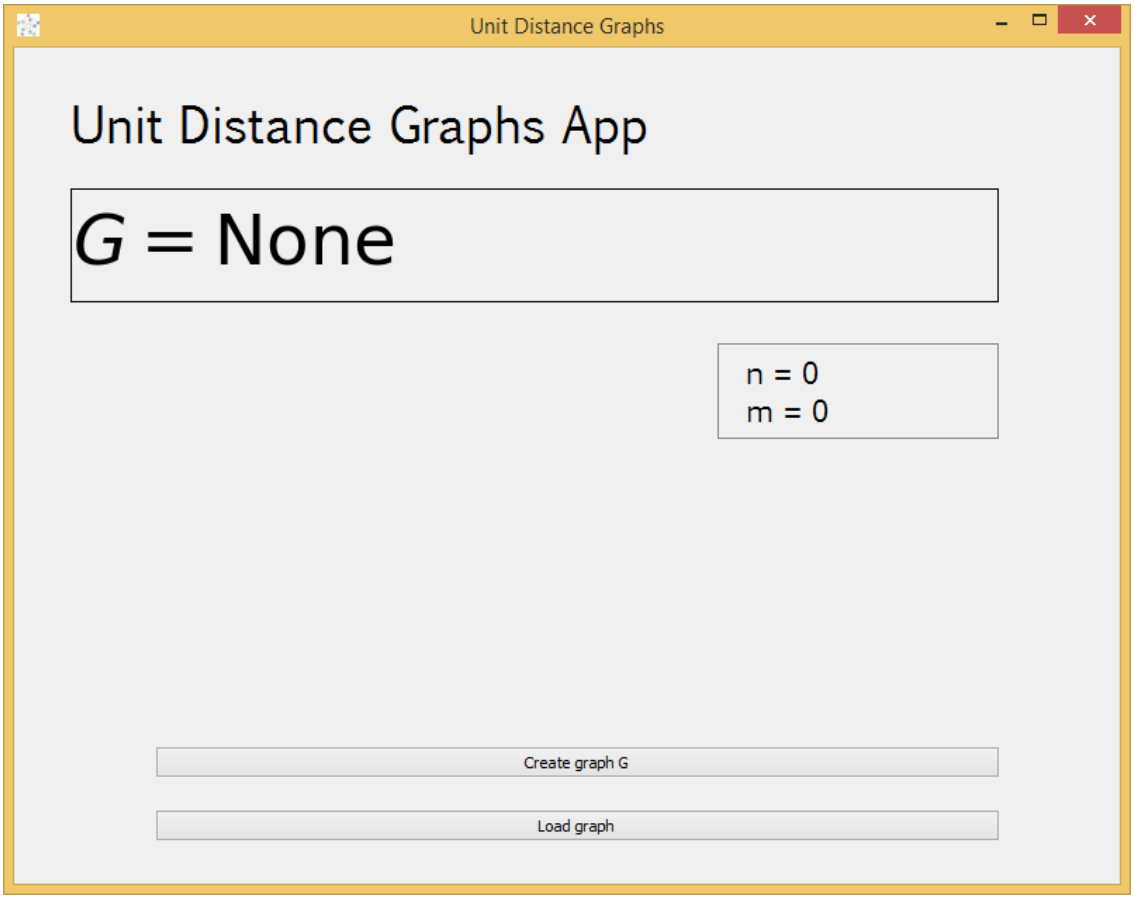


Figura 38: Ventana principal.

- La ventana principal. Esta tiene la información del grafo actual, y se encarga de las opciones de guardar y cargar grafos, así como de dibujarlos y colorearlos.
- La ventana de selección de grafo. Permite seleccionar un grafo mediante botones.
- La ventana de operaciones con grafos. Permite efectuar algunas operaciones implementadas en el módulo *graphs.py*.

La función `texToPng(formula, fname, fontsize=12, dpi=300)` se encargará de, dada una expresión de LaTeX, compilarla en una imagen png en la carpeta texLabels dentro de la carpeta GUI. Estas imágenes se usarán en las ventanas principal y de operaciones para mostrar información del grafo actual.

Cuando aún no hay ningún grafo seleccionado, la ventana principal tiene dos opciones:

- "*Create graph G*", que abre la ventana de selección de grafo.
- "*Load graph*", que hace escoger un fichero .e con la información del grafo.

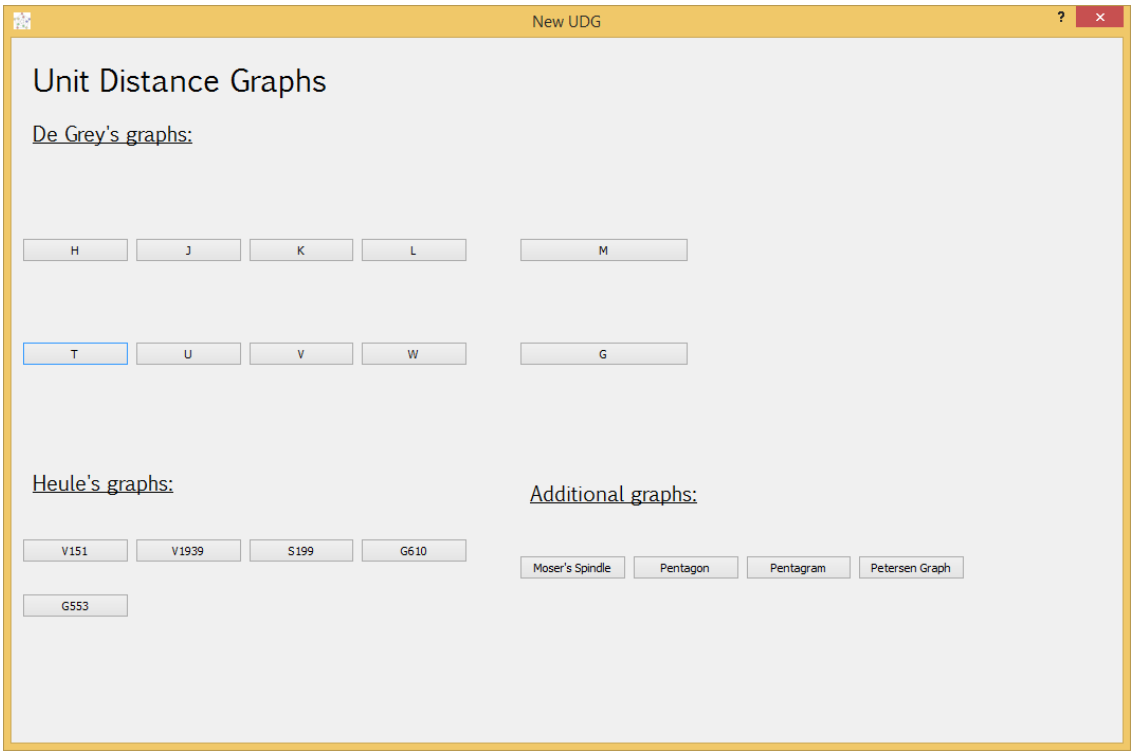


Figura 39: Ventana de selección de grafo.

La ventana de selección de grafo tiene botones con los diferentes grafos vistos en el trabajo, repartidos en tres secciones: los grafos usados por de Grey, los grafos usados por Heule y otros grafos. Todos estos grafos están definidos en *graphs.py*, a excepción de los grafos S_{199}, G_{610} y G_{553} de la demostración de Marijn Heule. Estos tres grafos se encuentran en forma de fichero .e en la carpeta heule dentro de la carpeta graphs[7]

Una vez seleccionado un grafo, la ventana principal se abre a nuevas opciones:

- En primer lugar, en la parte superior de la ventana tendremos el grafo seleccionado. Un poco más abajo estará la información relativa al número de vértices (n) y aristas (m).
- El botón "*Draw G*" dibujará el grafo seleccionado, usando el módulo *tikz.py*. El número al lado del botón es el factor que multiplica la longitud de las aristas, separando más o menos los vértices para apreciar mejor el grafo.

[7]Estos grafos, y otros obtenidos durante el trabajo de Heule, están subidos por él en `https://github.com/marijnheule/CNP-SAT`.

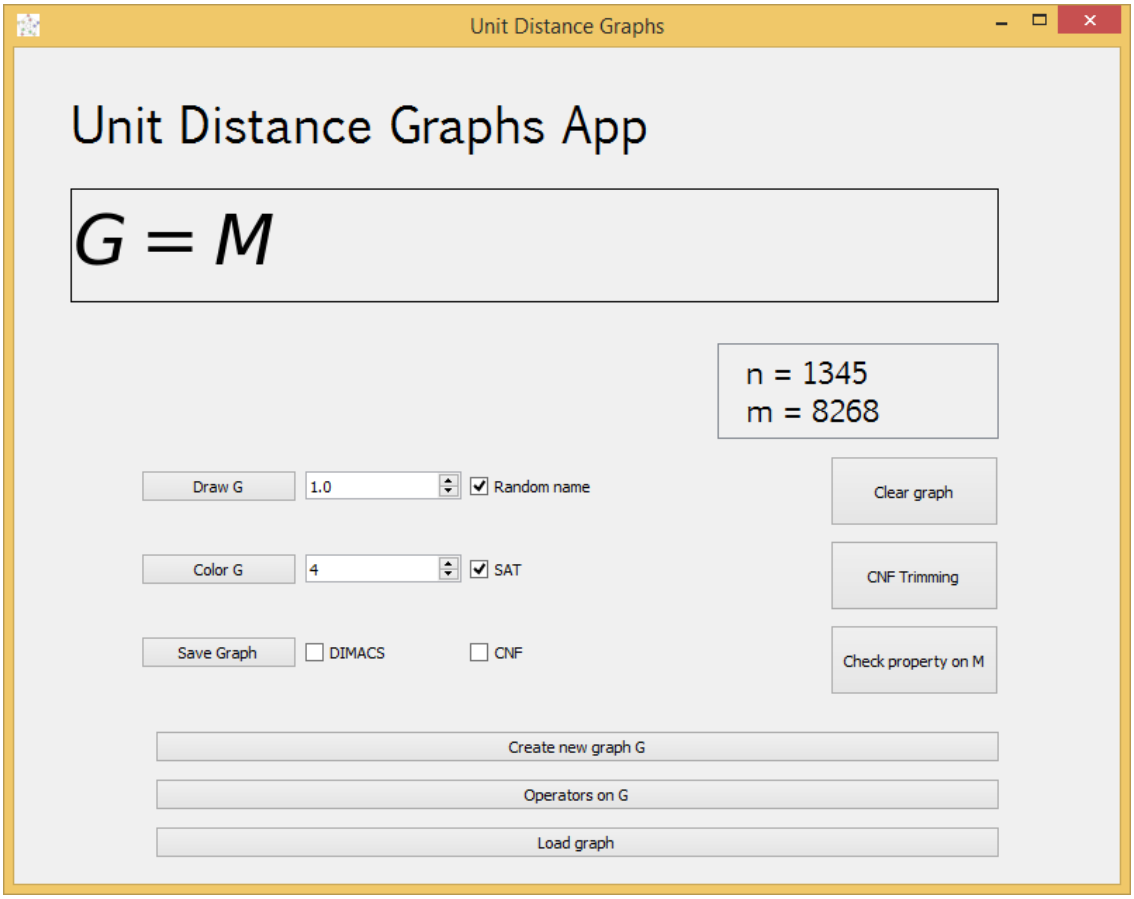


Figura 40: Ventana principal con el grafo M seleccionado.

Atención: Para poder usar esta funcionalidad hay que tener instalado LaTeX de forma que se puedan ejecutar los comandos "*pdflatex*" y "*lualatex*".

- La opción "*Random name*", que estará marcada por defecto, hace que los archivos .tex tengan como nombre un número entero aleatorio entre 1 y 10^9. De lo contrario, los archivos tendrán un nombre que haga referencia al grafo seleccionado (por ejemplo V_{155} sería V_{155}).

  Se ha hecho así porque LaTeX no podrá compilar el dibujo si ya hay un pdf con el mismo nombre abierto. Así pues, tener los nombres aleatorios permite poder dibujar el mismo grafo dos veces sin tener que cerrar el lector de pdf.

- El botón "*Color G*" coloreará el grafo actual. El número al lado del botón es el número máximo de colores a usar. Si la opción SAT está seleccionada, utilizará el módulo *sat.py*. De lo contrario, intentará colorear con el algoritmo de *backtracking* usando el módulo *color.py*.

- El botón "*Save Graph*" guardará ficheros .v y .e con la información de los vértices y aristas respectivamente. Si además están seleccionadas las opciones DIMACS o CNF, guardará también las aristas en formato DIMACS en la carpeta dimacs o la fórmula en forma normal conjuntiva en la carpeta cnf.

  La fórmula se hará teniendo en cuenta el número máximo de colores indicado al lado del botón de colorea (así, para demostrar que el grafo no es 4-coloreable, el

número máximo de colores tendrá que estar puesto en 4).

- El botón "*Clear graph*" elimina el grafo actual. Esto hace desaparecer todas las opciones menos las dos iniciales: abrir la ventana de selección de grafo y carga de grafo por fichero.
- El botón "*CNF Trimming*" abre una ventana para seleccionar dos ficheros: un fichero .cnf con una fórmula en forma normal conjuntiva que describa la k-colorabilidad del grafo actual, y un archivo .dict que relacione los vértices del grafo actual con los índices usados en la fórmula.

  Si hay algún vértice del grafo actual que no sea utilizado en la fórmula, se elimina del grafo. Esto permite reducir grafos manteniendo el número cromático, usando solucionadores SAT y comprobadores DRAT.
- Si el grafo es el grafo M, aparece un botón adicional: "*Check property on M*". Usando el algoritmo de backtracking, comprobará que si el H central de M tiene tres vértices del mismo color, M no es 4-coloreable.
- Finalmente, el botón "*Operators on G*" abre la ventana con los diferentes operaciones implementadas.

Las operaciones con grafos permitidas son 5:

- Unión de dos grafos.
- Suma de Minkowski de dos grafos.
- El grafo recortado hasta distancia d, dada por la expresión al lado del botón "*Trim*".
- Rotación de ángulo α respecto al vértice $(0, 0)$, dado por la expresión al lado del botón "*Rotate*".
- Rotación θ_i^k, con i y k enteros, respecto al vértice $(0, 0)$. El primer número al lado del botón "θ" representa la i, el segundo la k.

Los campos relativos a la rotación y el grafo recortado pueden usar expresiones usando *sqrt()* para raíces cuadradas, la palabra *pi* para π y ^ para exponentes.

Esta ventana tiene los mismos grafos disponibles para operar que la ventana de selección de grafos, así como un botón para cargar grafos de un fichero .e. Aun así, estarán desactivados hasta que se pulse el botón de unión o el de suma de Minkowski.

La construcción del grafo resultante no se inicia hasta que no se pulse el botón "*Build graph*". Eso permite la funcionalidad de **deshacer las operaciones** pulsando el botón "*Undo*".

Esto ha sido implementado usando un **deque** (del inglés, *Double-Ended Queue*, o cola doblemente terminada). A medida que se van introduciendo operaciones a hacer, se van añadiendo al deque, modificando la expresión de LaTeX mostrada.

Si en algún momento queremos rectificar la última operación realizada, pulsando el botón "*Undo*" eliminaremos la última operación añadida al deque, revirtiendo la expresión de LaTeX a su estado anterior. Cuando finalmente construyamos el nuevo grafo, recorreremos el deque ejecutando las operaciones de más antigua a más reciente.

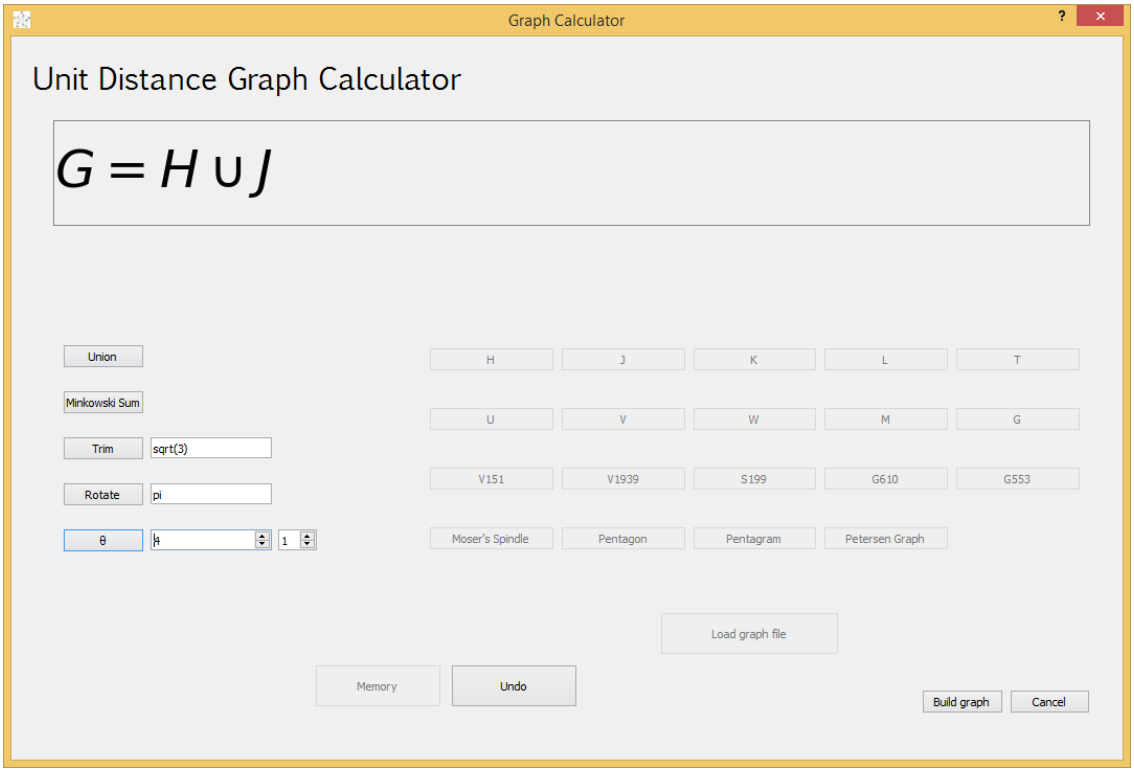


Figura 41: Ventana de la calculadora de grafos.

La falta de paréntesis hace que no sea posible la construcción de ciertos grafos, donde es necesario especificar un orden concreto en las operaciones. Para ello, después de pulsar los botones de unión o suma, se abre la opción de **crear un grafo auxiliar** pulsando el botón "*Memory*". Este apartará el grafo construido hasta el momento, y a partir de entonces todos los comandos afectarán al grafo auxiliar. Justo después de usar el botón "*Memory*" habrá que introducir un grafo sobre el que operar.

Una vez acabado el grafo auxiliar, hay que pulsar el botón de "*Merge*", que unirá o sumará (dependiendo de la opción elegida) el grafo auxiliar al grafo que se tenía hasta entonces.

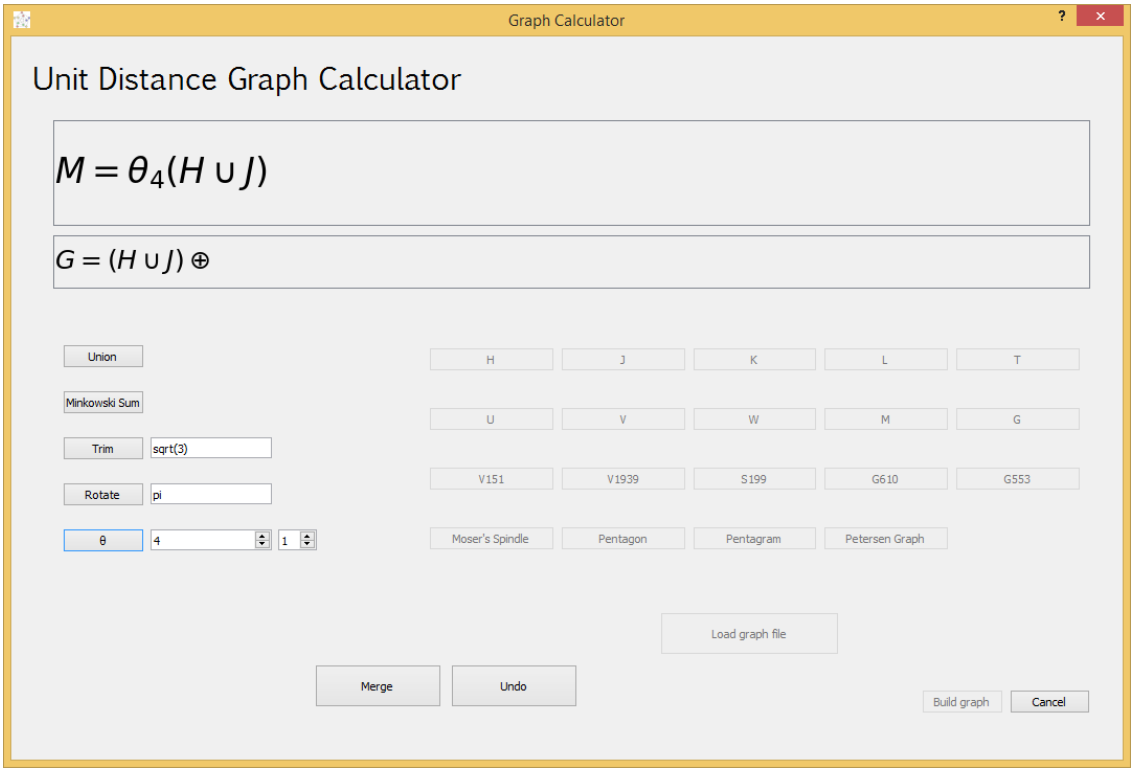


Figura 42: Ejemplo de uso de la opción *Memory*.

Referencias

[1] K. Appel y W. Haken, *The Solution of the Four-Color-Map Problem*, Scientific American **237** (1977), 108–121.

[2] D. Coulson, *A 15-colouring of 3-space omitting distance one*, Discrete Mathematics **256 (1-2)** (2002), 83–90.

[3] R. Diestel, *Graph Theory*, Berlin: Springer, 2016.

[4] M. Gardner, *A new collection of "brain teasers"*, Scientific American **206** (1960), 172–180.

[5] A.D.N.J. de Grey, *The chromatic number of the plane is at least 5*, Geombinatorics **28** (2018), 18–31.

[6] A.D.N.J. de Grey, *Escape Velocity: Why the Prospect of Extreme Human Life Extension Matters Now.*, PLoS Biol **2(6): e187** (2004). `https://doi.org/10.1371/journal.pbio.0020187`

[7] H. Hadwiger, *Ungelöste Probleme, Nr. 11*, Elemente der Mathematik **16** (1961), 103–104.

[8] M. Heule, *Computing Small Unit-Distance Graphs with Chromatic Number 5*, Geombinatorics **28** (2018), 32–50.

[9] E. Lamb, *Decades-Old Graph Problem Yields to Amateur Mathematician*, QuantaMagazine, 2018. `https://www.quantamagazine.org/decades-old-graph-problem-yields-to-amateur-mathematician-20180417/`

[10] L. Moser y W. Moser, *Solution to problem 10*, Canadian Mathematical Bulletin **4** (1961), 187–189.

[11] O. Nechushtan, *On the space chromatic number*, Discrete Mathematics **256** (2002), 499–507.

[12] D. H. J. Polymath, *Polymath16: Simplifying the lower bound proof for the Hadwiger-Nelson problem*, `https://dustingmixon.wordpress.com/2018/04/14/polymath16-first-thread-simplifying-de-greys-graph`

[13] D. H. J. Polymath, *Polymath16: Simplifying the lower bound proof for the Hadwiger-Nelson problem*, `http://michaelnielsen.org/polymath1/index.php?title=Hadwiger-Nelson_problem`

[14] D. Pritikin, *All unit-distance graphs of order 6197 are 6-colorable*, Journal of Combinatorial Theory, Series B **73.2** (1998), 159–163.

[15] D. E. Raiskii, *Realizing of all distances in a decomposition of the space $\mathbb{R}^n$ into n+1 parts*, Matematicheskie Zametki **7** (1970), 319–323.

[16] A. Soifer, *The Mathematical Coloring Book: Mathematics of Coloring and the Colorful Life of Its Creators*, Springer, 2008.

[17] A. L. Székely, *Remarks on the chromatic number of geometric graphs*, Graphs and other combinatorial topics: proceedings of the Third Czechoslovak Symposium on Graph Theory held in Prague, August 24th to 27th, 1982, Ed. por M. Fiedler, Leipzig: Teubner, 1983, 312–315.

[18] C. Thomassen, *On the Nelson unit distance coloring problem*, American Mathematical Monthly **106** (1999), 850–853.

[19] S. P. Townsend, *Colouring the plane with no monochrome unit*, Geombinatorics **14 (4)** (2005), 181–193.