



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERÍA MATEMÁTICA

POTENCIAS Y POTENCIAS EXACTAS: ALGORITMOS Y COLOREOS

TESIS PARA OPTAR AL GRADO DE
MAGÍSTER EN CIENCIAS DE LA INGENIERÍA, MENCIÓN MATEMÁTICAS
APLICADAS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL MATEMÁTICO

PEDRO PABLO CORTÉS ROJAS

PROFESOR GUÍA:
DANIEL QUIROZ BRITO

PROFESORA CO-GUÍA:
MAYA STEIN

MIEMBROS DE LA COMISIÓN:
MARTÍN MATAMALA VÁSQUEZ
JOSÉ ZAMORA PONCE

Este trabajo ha sido parcialmente financiado por ANID/Fondecyt Iniciación en Investigación 11201251, MATHAMSUD MATH210008 y CMM ANID BASAL FB210005.

SANTIAGO DE CHILE
2023

RESUMEN DE LA TESIS PARA OPTAR AL
GRADO DE MAGÍSTER EN CIENCIAS DE LA INGENIERÍA,
MENCION MATEMÁTICAS APLICADAS Y
TÍTULO DE INGENIERO CIVIL MATEMÁTICO
POR: PEDRO PABLO CORTÉS ROJAS
FECHA: 2023
PROFESOR GUÍA: DANIEL A. QUIROZ BRITO
PROFESORA CO-GUÍA: MAYA STEIN

POTENCIAS Y POTENCIAS EXACTAS: ALGORITMOS Y COLOREOS

En esta tesis se trabaja con los siguientes conceptos de teoría de grafos:

Dado un grafo G y un entero positivo p , la potencia exacta p -ésima de G denotada por $G^{[\#p]}$ es el grafo con el mismo conjunto de vértices de G ; y entre dos vértices u y v hay una arista en $G^{[\#p]}$ si y solamente si los vértices u y v están a distancia *exactamente* p en G . Cuando $p = 2$ se dice que $G^{[\#2]}$ es el cuadrado exacto de G o que G es una raíz cuadrada exacta de $G^{[\#2]}$. Y cuando $p = 3$ se habla de cubos exactos y raíces cúbicas exactas respectivamente.

Dado un grafo G y un entero positivo p , la potencia p -ésima de G denotada por G^p es el grafo con el mismo conjunto de vértices de G ; y entre dos vértices u y v hay una arista en G^p si y solamente si los vértices u y v están a distancia a lo más p en G .

El *número subcromático* de un grafo G , denotado por $\chi_s(G)$, es el menor número de colores necesarios para colorear los vértices del grafo, de manera que cada clase de color induce una unión disjunta de cliques.

La noción clásica de número de coloreo se define como el menor entero k tal que G tiene un ordenamiento lineal de vértices en la que cada vértice tiene, a lo más, $k - 1$ vecinos menores que él. Existen diversas generalizaciones de este concepto.

En la primera parte de esta tesis se resuelve el problema de encontrar una caracterización para cuadrados exactos de árboles la cual conlleva a encontrar un algoritmo de reconocimiento que en tiempo polinomial determina si un grafo es el cuadrado exacto de un árbol. De igual manera se resuelve la caracterización, en dos de los tres casos posibles, para los cubos exactos de árboles obteniendo un esbozo de un algoritmo polinomial de reconocimiento. Y se resuelve el problema de encontrar una caracterización para los grafos que tienen raíces cuadradas exactas que son triangle-free.

En la segunda parte se estudia el número subcromático de potencias de grafos, logrando mejorar la cota de dicho número para algunas familias de grafos poco densas. Para ello se introducen y estudian nuevos parámetros de números generalizados de coloreo.

Para mi familia.

Agradecimientos

Quiero partir agradeciendo a mi familia, en especial a mis padres, quienes han estado siempre apoyándome. Mis logros han sido fruto de su esfuerzo y amor. Estaré eternamente agradecido de todo lo que me han entregado y por estar siempre conmigo.

Agradecer a mis amigos y compañeros de Kung Fu, he pasado muy buenos momentos y me he divertido mucho con todos ustedes, desde los entrenamientos hasta las juntas y cervezas que hemos compartido. En especial al profe Jaime por siempre preocuparse de todos sus alumnos.

Durante mi paso por el DIM logré conocer grandes personas y amigos con los que he compartido muchos momentos valiosos, las peripecias de este viaje las viví bien acompañado y estoy muy agradecido del apoyo, las risas, las juntas y experiencias que compartí con ustedes: Gonza, Leo, Fabian, Caro, Aldo, Benja, Manu, Mariano, Jota, Catherine y Javier.

Agradecer a mi profesor guía, Daniel, por aceptarme como su tesista, por enseñarme cómo es el mundo académico, por ser una gran persona, por las oportunidades que me dió de investigar temas tan interesantes y la posibilidad de discutir matemática con personas de diferentes partes (a pesar de que mi inglés no era el mejor). Agradecer a mi profesora co-guía, Maya, cuyas clases me motivaron a adentrarme en el mundo de la teoría de grafos. También agradecer a Martín y José por su disposición para formar parte de la comisión, y por el tiempo que se dieron para leer y entregarme sus correcciones de la tesis.

Finalmente agradecer a todas las personas que han dejado alguna huella en mí y me han hecho crecer como persona.

Tabla de Contenido

Introducción	1
1. Cuadrados y Cubos Exactos de Árboles	5
1.1. Cuadrados exactos de árboles	6
1.1.1. Algoritmo de reconocimiento de cuadrados exactos de árboles	13
1.2. Caracterización de grafos con raíz cuadrada exacta triangle-free	20
1.3. Cubos exactos de árboles	21
1.3.1. Cuando $T^{[\#3]}$ tiene 3 componentes conexas	22
1.3.2. Cuando $T^{[\#3]}$ tiene 2 componentes conexas	23
1.3.3. Cuando $T^{[\#3]}$ tiene 1 componente conexas	47
2. Número subcromático de potencias de grafos	48
2.1. Números generalizados de Coloreo	48
2.2. Acotamiento del número subcromático de potencias de grafos usando el número de coloreo semidébil	51
2.3. Cotas explícitas de $gcol$ para grafos con treewidth acotado o minors excluidos	53
Conclusión	57
Bibliografía	63
Anexo Versión con $gcol$ del Teorema 2.3	64

Índice de Ilustraciones

1.1.	Ejemplo de un árbol con $\text{diam}(T) \leq 2p - 4$ cuya potencia exacta p tiene más de p componentes conexas.	6
1.2.	Un árbol T y su cuadrado exacto con componentes C_1 y C_2 . Aquí C_1 no es isomorfo a ninguna componente conexas de $T_{C_2}^{[\#2]}$ y C_2 no es isomorfo a ninguna componente conexas de $T_{C_1}^{[\#2]}$	10
1.3.	Construcción de un árbol T (usando Procedimiento 1) cuyo cuadrado exacto es G	14
1.4.	P_5^+	21
1.5.	$B(n, m_1, m_2)$	22
1.6.	Caso $m_1 = 2$, $m_2 = 3$ y $n = 6, 7, 8$	23
1.7.	$P_m^{3, m-2}$	23
1.8.	Árboles Canónicos de C_R	28
1.9.	Tipo de árboles T en la Familia Excepción y sus cubos exactos.	30
1.10.	Árbol T con segmentos marcados.	32
1.11.	Cubo Exacto de T con segmentos marcados.	32
1.12.	C_R , C_{NR} , C_{NR}^{tw} y \hat{T}	38
1.13.	Isomorfismo φ	39
1.14.	Procedimiento 2.	40
2.1.	Ejemplos para $k = 1$, $k = 2$ y $k = 3$	61
1.	Esquema del Walk	65

Introducción

La teoría de grafos es un área muy activa que tiene gran impacto tanto dentro como fuera de las matemáticas.

Una noción importante dentro de esta área es la de potencia de grafos. Dado un grafo G y un entero positivo p , la potencia p -ésima de G denotada por G^p es el grafo con el mismo conjunto de vértices de G ; y entre dos vértices u y v hay una arista en G^p si y solamente si los vértices u y v están a distancia a lo más p en G . Cuando $p = 2$ se dice que G^2 es el cuadrado de G o que G es una raíz cuadrada de G^2 . Cuando $p = 3$ se habla de cubo y raíz cúbica, respectivamente.

El artículo de Ross y Harary [14] fue uno de los primeros en trabajar con cuadrados de grafos y en dicho artículo lograron caracterizar los cuadrados de árboles. Dicha caracterización permitió encontrar algoritmos de tiempo polinomial que resuelven el problema de reconocer grafos que son cuadrados de árboles [8, 9].

Por otra parte, existe la noción de potencias exactas de grafos, conocidas en la literatura como ‘exact distance graph’ [12]. De forma similar dado un grafo G y un entero positivo p , la potencia exacta p -ésima de G denotada por $G^{[p]}$ es el grafo con el mismo conjunto de vértices de G ; y entre dos vértices u y v hay una arista en $G^{[p]}$ si y solamente si los vértices u y v están a distancia *exactamente* p en G . Para $p = 2$ y $p = 3$ se habla de cuadrados exactos y cubos exactos, respectivamente.

Motivados por los resultados de potencias normales, surge la problemática de estudiar caracterizaciones de potencias exactas de árboles. Una motivación adicional es que los árboles son casos particulares de grafos outerplanar ¹ y estudiar los cubos exactos de grafos outerplanar tiene relevancia por lo siguiente:

Un resultado muy conocido es que determinar si un grafo es k -coloreable para $k \geq 3$ es un problema NP-completo. Resultados generales de Van den Heuvel, Kierstead y Quiroz [17] implican que el cubo exacto de un grafo outerplanar tiene número cromático a lo más 10. Por lo tanto si el problema de decidir si un grafo es un cubo exacto de un grafo outerplanar se pudiera resolver en tiempo polinomial, entonces se tendría una condición suficiente (pero no necesaria) para identificar grafos que son 10-coloreable en tiempo polinomial.

En el capítulo 1 de esta tesis se presentarán los resultados obtenidos con respecto a los

¹Un grafo G es outerplanar si y solamente si no contiene ni al completo K_4 ni al bipartito completo $K_{2,3}$ como menor.

cuadrados exactos y cubos exactos de árboles:

- En la sección 1.1 se mostrarán los resultados de los cuadrados exacto donde se dará una caracterización que permite encontrar un algoritmo de tiempo polinomial que resuelve el problema de reconocer cuadrados exactos de árboles.
- En la sección 1.2 se mostrará un resultado que caracteriza los grafos que tienen raíces cuadradas exactas que son triangle-free.
- En la sección 1.3 se presentará una caracterización de los grafos que son cubos exactos de árboles, en dos de los tres casos posibles (que serán explicados en dicha sección), permitiendo el esbozo de un algoritmo de reconocimiento en tiempo polinomial para dichos casos.

En el Capítulo 2 tornamos las atención desde las potencias exactas a las potencias. En particular estudiaremos la noción de subcoloreo que nos permite ‘colorear’ grafos densos con pocos colores. Este concepto se introdujo por primera vez en el artículo de Albertson et al. [1].

Recordemos que un k -coloreo propio de un grafo G es una función $f : V(G) \rightarrow \{0, \dots, k-1\}$ tal que para toda arista $uv \in E(G)$ se cumple que $f(u) \neq f(v)$. Un k -subcoloreo de un grafo G es una función $f : V(G) \rightarrow \{0, \dots, k-1\}$ tal que cada conjunto de vértices coloreados con $i \in \{0, \dots, k-1\}$ induce una unión disjunta de cliques. El *número subcromático*, denotado por $\chi_s(G)$, es el menor entero k tal que G admite un k -subcoloreo.

Un resultado reciente de Nešetřil, Ossona de Mendez, Pilipczuk y Zhu [11] muestra una cota superior constante para el número subcromático de potencia de grafos que provienen de familia de grafos que son ‘sparse’. Para obtener dichos resultados se usan generalizaciones del número de coloreo, que se define como el menor entero k tal que G tiene un ordenamiento lineal de vértices en la que cada vértice tiene a lo más $k-1$ vecinos menores que él. Existen diversas generalizaciones de este concepto, pero una que ha sido de mucha utilidad en diversos aspectos es la que introdujeron Kierstead y Yang en [7], los números generalizados de coloreo $wcol_d$ y col_d , conocidos como el *número de d -coloreo débil* y *número de d -coloreo*, respectivamente. Con estas nociones en [11] se obtuvo el siguiente resultado:

Teorema ([11]) Para todo grafo G y entero positivo d se tiene que

$$\chi_s(G^d) \leq wcol_{2d}(G).$$

Por otra parte Van den Heuvel, Ossona de Mendez, Quiroz, Rabinovich y Siebertz en [16] obtuvieron la siguiente cota para $wcol_d$:

Teorema ([16]) Si G es un grafo planar se tiene que

$$wcol_d(G) \leq \binom{d+2}{2}(2d+1).$$

Por lo que estos dos teoremas permiten concluir que $\chi_s(G^3) \leq 364$ para el caso en que G sea un grafo planar.

En el capítulo 2 se mostrará una mejora de dicha cota bajándola a 95. Por otra parte se presentará que para grafos con treewidth 2 se tiene una cota superior de 6 para el número subcromático de sus cuadrados y de 10 para sus cubos, lo que implica la misma cota superior para los cuadrados y cubos de grafos outerplanar, respectivamente. Por otra parte, se presentarán resultados que implican que el número subcromático de cuadrados de grafos K_5 -minor free está acotado por 180.

De forma más detallada, en el capítulo 2 presentaremos:

- En la sección 2.1 se presentarán dos nuevos parámetros de coloreo similares a $wcol_d$ y col_d . Uno de ellos llamado número de coloreo semidébil, el cuál permitirá mejorar la cota para el número subcromático de cubos de grafos planares. Y el otro concepto es el $gcol$, donde éste parámetro generaliza las nociones de col_d y $wcol_d$.
- En la sección 2.2 se mostrarán cotas del número subcromático de potencia de grafos usando el número de coloreo semidébil. Y se mostrará una mejor cota para el número subcromático de cubos de grafos planares.
- En la sección 2.3 se presentarán resultados de acotamiento para el número $gcol$ los cuales implicarán resultados tales como una cota superior para el número subcromático de cuadrados de grafos con treewidth 2 y una cota del número subcromático de cuadrados de grafos con minors excluidos.

Preliminares

En la siguiente tesis trabajaremos solamente con grafos finitos, simples y no dirigidos. Usaremos como referencia [5] para la notación básica de grafos.

La *distancia* entre dos vértices $u, v \in V(G)$ es la cantidad de aristas de un camino más corto entre u y v , la cual denotamos por $d_G(u, v)$, o $d(u, v)$ cuando el grafo esta claro por el contexto.

El *diámetro* de un grafo G , denotado por $\text{diam}(G)$, es la mayor distancia entre dos vértices en un grafo.

El *complemento* de un grafo G , denotado por \overline{G} , es el grafo con el mismo conjunto de vértices que G y entre dos vértices hay una arista si y solamente si en G no existe dicha arista.

La *vecindad* de un vértice $v \in V(G)$, se define como $N_G(v) := \{u \in V(G) | d_G(u, v) = 1\}$. La *vecindad cerrada* de $v \in V(G)$ se define como $N_G[v] := N_G(v) \cup \{v\}$.

El *grado* de un vértice $v \in V(G)$, denotado $d_G(v)$, se define como la cantidad de vecinos de v , es decir, $d_G(v) := |N_G(v)|$.

Para un grafo G y un entero positivo p se define la *potencia p de G* , denotada por G^p , como el grafo con el mismo conjunto de vértices de G y entre dos vértices, u y v , hay una arista si y solamente si $d_G(u, v) \leq p$. Por otra parte, la *potencia exacta p de un grafo G* ,

denotada $G^{\lfloor p \rfloor}$, se define de manera similar pero cambiando la condición de las aristas, donde ahora hay una arista entre dos vértices u y v si y solamente si $d_G(u, v) = p$.

El camino de n vértices se denota por P_n .

El *grafo completo* en n vértices, denotado por K_n , es un grafo con todas las aristas posibles. Un subgrafo se dirá *clique* si es un grafo completo.

Sean $G_1 = (V_1, E_1)$ y $G_2 = (V_2, E_2)$ dos grafos. Un *isomorfismo* de G_1 hacia G_2 es una función biyectiva φ de V_1 a V_2 que cumple que $uv \in E_1$ si y solamente si $\varphi(u)\varphi(v) \in E_2$.

Dado un grafo G y un subconjunto de vértices $V' \subseteq V(G)$, se tiene que $G[V']$ denota el *subgrafo inducido* por V' , es decir $V(G[V']) = V'$ y $E(G[V']) = \{uv \in E(G) \mid u, v \in V'\}$.

Dado H_1 y H_2 subgrafos de G , denotaremos el conjunto de aristas en G entre H_1 y H_2 como $E(H_1, H_2) := \{uv \in E(G) \mid u \in H_1 \text{ y } v \in H_2\}$.

Cuando decimos que un subgrafo H de G , es un grafo maximal que cumple cierta propiedad, nos estamos refiriendo a que H cumple la propiedad pero al agregar aristas y/o vértices de $G \setminus H$ al grafo H se tiene que ya no cumple dicha propiedad. En otras palabras no existe otro subgrafo diferente que contenga a H y cumpla la propiedad.

Cuando hablamos de una familia de grafos nos referimos a un conjunto de grafos (puede ser un conjunto infinito de grafos).

Un *grafo bipartito* es un grafo G en el que se puede particionar los vértices $V(G)$ en dos partes disjuntas A_1, A_2 tal que $V(G) = A_1 \cup A_2$, $E(G) = E(A_1, A_2)$. Equivalentemente un grafo es bipartito si no contiene ciclos impares.

Un *grafo bipartito completo* es un grafo bipartito G donde la bipartición A_1, A_2 satisface que $E(G) = \{uv \mid u \in A_1 \text{ y } v \in A_2\}$, es decir para cualquier vértice en A_1 se tiene que este vértice está conectado con todos los vértices de A_2 .

Dado un conjunto de vértices A_1 y A_2 , denotaremos por K_{A_1, A_2} al grafo bipartito completo que tiene como bipartición A_1, A_2 .

Un *matching* $M \subseteq E(G)$ es un subconjunto de aristas independientes. Un matching maximal es un matching $M \subseteq E(G)$ tal que para toda arista $e \in E(G) \setminus M$ se tiene que $M \cup \{e\}$ deja de ser matching.

Un *bloque* es un subgrafo maximal conexo y sin vértices de corte.

Un grafo conexo se dirá que es un *clique-tree* si cada uno de sus bloques es un clique.

Dado dos grafos G_1 y G_2 que no comparten vértices, se define la unión disjunta de dichos grafos, denotada por $G_1 + G_2$, como el grafo tal que $V(G_1 + G_2) := V(G_1) \cup V(G_2)$ y $E(G_1 + G_2) := E(G_1) \cup E(G_2)$.

Capítulo 1

Cuadrados y Cubos Exactos de Árboles

En este capítulo se presentará una caracterización de los cuadrados exactos de árboles la cual permite crear un algoritmo que en tiempo polinomial permite resolver el problema de reconocer los cuadrados exactos de árboles. También se presentará una caracterización de los cubos exactos de árboles, en dos de los tres casos posibles, la cual permite esbozar algoritmos que en tiempo polinomial resuelven el problema de reconocer cubos exactos de árboles.

A continuación se presenta un lema simple, pero importante, para el estudio que se hará posteriormente.

Lema 1.1 *Sea $p \geq 2$ un entero. Si T es un árbol con $\text{diam}(T) > 2p - 4$, entonces $T^{[\#p]}$ tiene a lo más p componentes conexas.*

DEMOSTRACIÓN. Sea $P = v_0 \dots v_d$ un camino de tamaño máximo en T , es decir $d = \text{diam}(T)$. Para cada $i \in \{0, \dots, p-1\}$ definimos el conjunto $B_i := \{v_j \in V(P) \mid j = kp + i \text{ para algún } k \in \mathbb{N}\}$. Notar que $V(P) = \cup_{i=0}^{p-1} B_i$. Dado que para cualquier par de vértices en B_i se tiene que están a distancia kp para algún $k \in \mathbb{N}$, entonces existe un camino en $T^{[\#p]}$ entre ese par de vértices y por ende pertenecen a la misma componente conexa. Esto prueba el resultado si es que T fuera solamente el camino P .

Sea $w \in V(T) \setminus V(P)$ y sea $v_c \in V(P)$ el vértice final en P , del único w - P camino en T . Sea $m := d_T(w, v_c)$. Es suficiente mostrar que w está conectado en $T^{[\#p]}$ a algún B_i .

Primero consideramos el caso en que $m \leq p$. Dado que T es un árbol, si uno entre v_{c+p-m} o $v_{c-(p-m)}$ pertenece al camino P , entonces dicho vértice será adyacente a w en $T^{[\#p]}$. Supongamos por contradicción que este no es el caso, es decir tenemos que $c+p-m \notin \{0, \dots, d\}$ y que $c-(p-m) \notin \{0, \dots, d\}$. Dado que $m \leq p$ se tiene que $c+p-m \geq d+1$ y $c-(p-m) \leq -1$. Con esta última observación y sumado a la hipótesis de que $\text{diam}(T) > 2p - 4$ obtenemos que $2p - 2m \geq d + 2 > 2p - 2$ lo que implica $m < 1$, una contradicción. Por lo tanto existe un vértice en P adyacente a w en $T^{[\#p]}$. En particular, w está en la misma componente conexa que algún B_i .

El caso $m > p$ se deduce del caso anterior porque existe algún vértice u en el $w-v_c$ camino tal que $d_T(u, w) = kp$ para algún $k \in \mathbb{N}$ y $d_T(u, v_c) \leq p$. Por eso w está en la misma componente conexa que u en $T^{[\#p]}$, y u está en la misma componente conexa que algún B_i . ■

Veamos un caso sencillo de un grafo que no cumple la hipótesis y su potencia exacta p tiene más de p componentes conexas. Consideremos $p \geq 3$ y $n \geq p$, entonces el grafo $K_{1,n}$, conocido como la estrella en $n + 1$ vértices, es un árbol cuyo diámetro es 2 y su potencia exacta p tiene $n + 1$ componentes conexas. En la Figura 1.1 se muestra un ejemplo.

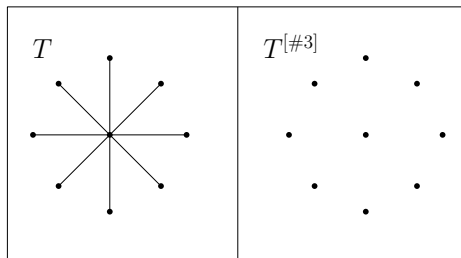


Figura 1.1: Ejemplo de un árbol con $\text{diam}(T) \leq 2p - 4$ cuya potencia exacta p tiene más de p componentes conexas.

1.1. Cuadrados exactos de árboles

A continuación vamos a estudiar los cuadrados exactos de árboles. De ahora en adelante T será un árbol no trivial (tiene al menos una arista).

Los resultados de esta sección 1.1 y los resultados de 1.2 han sido aceptados para su publicación, junto con otros resultados, en el artículo [3].

En [3] demostramos que el cuadrado exacto de T es un grafo con dos componentes conexas donde cada una de ellas es un clique-tree. En esta tesis daremos una demostración alternativa a dicho resultado usando una metodología diferente. Para ello veamos los siguientes resultados:

Lema 1.2 *Sea B un grafo bipartito, con bipartición (X, Y) . Si tenemos una arista en el cuadrado exacto de B , es decir $uv \in E(B^{[\#2]})$, entonces los vértices u y v pertenecen al mismo conjunto de la bipartición, es decir $u, v \in X$ o $u, v \in Y$.*

DEMOSTRACIÓN. Por contradicción supongamos que los vértices u y v están en conjuntos diferentes, digamos $u \in X$ y $v \in Y$. Como $uv \in E(B^{[\#2]})$ entonces se tiene que la distancia en B entre dichos vértices es dos. Es decir existe un vértice $w \in V(B)$ tal que uww es un u, v -camino en B y $uv \notin E(B)$. Si $w \in X$ se tendría que $uw \in E(B)$ sería una arista entre vértices de la misma parte de la bipartición, lo que contradice la definición de bipartición. Se tiene la misma contradicción si $w \in Y$. En conclusión u y v pertenecen al mismo conjunto de la bipartición. ■

Lema 1.3 *Sea T un árbol con al menos 2 vértices. Entonces el cuadrado exacto de T tiene exactamente dos componentes conexas.*

DEMOSTRACIÓN. Como T es un grafo bipartito (por ser árbol) se tiene que tiene una bipartición (X, Y) .

Primero notemos que no puede tener una sola componente conexa. Si $T^{[\#2]}$ fuera conexo entonces por el Lema 1.2 todos los vértices de T estarían en la misma parte de la bipartición de T , y se tendrían aristas entre vértices del mismo conjunto de la bipartición lo que es una contradicción.

Ahora supongamos que $T^{[\#2]}$ tiene $k \geq 3$ componentes conexas C_1, \dots, C_k . Notar que el Lema 1.2 permite concluir que para $i \in [k]$ se tiene que $V(C_i) \subseteq X$ o que $V(C_i) \subseteq Y$.

El caso en que las todas las componentes conexas sean vértices de la misma clase, es decir $\cup_{i=1}^k V(C_i) \subseteq X$ o $\cup_{i=1}^k V(C_i) \subseteq Y$, no puede ocurrir ya que entonces T no tendría aristas.

Ahora supongamos el otro caso en que existen dos componentes conexas con vértices de clases diferentes de la bipartición de T . Sin pérdida de generalidad podemos asumir que $V(C_1) \subseteq X$ y $V(C_2) \subseteq Y$. Si tomamos un vértice $v_1 \in C_1$ y un vértice $v_2 \in C_2$ entonces como T es conexo se tiene que existe un v_1, v_2 -camino en T . Recordar que entre vértices de un mismo C_i no pueden tener una arista en T y además entre vértices de X (lo mismo para Y) no pueden haber aristas en T . Con las observaciones anteriores se puede concluir que el v_1, v_2 -camino en T debe contener un subcamino $u_1 u_2 u_3$ donde estos tres vértices pertenecen a tres componentes conexas diferentes en $T^{[\#2]}$. Pero esto significa que u_1 y u_3 están a distancia dos en T y por lo tanto deberían estar en la misma componente conexa, lo que es una contradicción.

En conclusión el cuadrado exacto de T contiene dos componentes conexas. ■

Una observación importante que mostraremos a continuación es que la vecindad en T de un vértice no hoja induce un bloque en el cuadrado exacto. También es importante señalar que los vértices de corte en $T^{[\#2]}$ toman un rol estructural importante.

A continuación tenemos los siguientes lemas:

Lema 1.4 *Sea T un árbol. Para todo vértice $v \in T$, la vecindad $N_T(v)$ induce un clique en $T^{[\#2]}$.*

DEMOSTRACIÓN. Esto es porque para todo $u, w \in N_T(v)$ se tiene que $d_T(u, w) = 2$, entonces $uw \in T^{[\#2]}$. ■

Lema 1.5 *Sea T un árbol con $|T| \geq 2$ y sean C_1 y C_2 las componentes conexas de $T^{[\#2]}$. Si $|C_1| \geq 2$, las siguientes afirmaciones son equivalentes*

- (a) $B \subseteq C_1$ es un bloque de $T^{[\#2]}$
- (b) Existe $v \in V(C_2)$ con $d_T(v) \geq 2$ tal que $B = C_1[N_T(v)]$.

DEMOSTRACIÓN. Primero probaremos que (b) implica (a). Por el Lema 1.4 el subgrafo inducido por $N_T(v)$ es un clique en $T^{\{\#2\}}$, y como $B = C_1[N_T(v)]$, entonces se tiene que B es un clique. Como $d_T(v) \geq 2$, entonces $|B| \geq 2$.

Como es un subgrafo completo con al menos dos vértices, entonces es un grafo conexo sin vértices de corte. Veamos que es un subgrafo maximal con esta propiedad, es decir, que es un bloque.

Si B no es maximal, entonces existe algún conjunto de vértices $W = \{w_1, \dots, w_m\} \subseteq C_1 \setminus B$ tal que $C_1[V(B) \cup W]$ es un subgrafo conexo sin vértice de corte.

Notar que ningún vértice $w \in W$ puede estar conectado a dos o más vértices de B ya que si existe un $w \in W$ y dos vértices diferentes $u_1, u_2 \in V(B)$ tal que u_1w y u_2w son aristas de C_1 entonces w está a distancia dos de u_1 y u_2 en T . Se tienen dos posibilidades: la primera es que w es un vecino de v lo que implica que $w \in B$ pero $w \in C_1 \setminus B$. Y la segunda opción es que existen los vértices s_1, s_2 (podrían ser iguales entre sí) que son diferentes de v tal que u_1s_1w y u_2s_2w son caminos en T , pero u_1vu_2 también lo es, por lo que existiría un ciclo en el árbol.

Por lo anterior y como $C_1[V(B) \cup W]$ es un subgrafo conexo sin vértice de corte se tiene que existen dos vértices $u_1, u_2 \in V(B)$ y un camino $u_1w_{i_1} \dots w_{i_p}u_2$, con $p \geq 2$, donde los vértices internos del camino son vértices de W . Pero eso implica que en T existe un camino entre u_1 y u_2 diferente al camino u_1vu_2 , lo que generaría un ciclo en T , contradiciendo que T es árbol. Por lo tanto, B es un bloque.

Ahora vamos a probar que (a) implica (b). Notar que $|B| \geq 2$ porque $|C_1| \geq 2$ y B es un bloque de C_1 . En particular, existe una arista $uw \in E(B)$, y un vértice $v \in T$ tal que $u, w \in N_T(v)$. Vamos a mostrar que $C_1[N_T(v)] = B$.

Primero notemos que $C_1[N_T(v)] \subseteq B$. Ya vimos que $u, w \in B$, ahora si tomamos un vértice $t \in C_1[N_T(v)] \setminus \{u, w\}$, si es que existe, se tiene que tu y tw son aristas en C_1 . Por la maximalidad de B como subgrafo conexo sin vértices de corte, se tiene que $t \in B$ y por lo tanto $C_1[N_T(v)] \subseteq B$.

Ahora veamos que $B \subseteq C_1[N_T(v)]$. Por contradicción supongamos que existe un vértice $x \in B \setminus C_1[N_T(v)]$. Como B es bloque, existe un vértice $y \in V(B)$ tal que xy es una arista de B . Como x, y son vecinos en B significa que en T existe un vértice z tal que xzy es un camino de T . Pero como B es bloque existe un ciclo en B que contiene a las aristas uw y xy , pero eso significa que entre u y w hay un camino en T diferente al camino uvw , lo que contradice que T es árbol. En conclusión $B = C_1[N_T(v)]$. ■

Teorema 1.6 *Sea T un árbol con al menos dos vértices. Entonces el cuadrado exacto de T es un grafo con dos componentes conexas, cada una de ellas un clique-tree.*

DEMOSTRACIÓN. Por el Lema 1.3 se tiene que $T^{\{\#2\}}$ tiene dos componentes conexas. Falta ver que cada una de esas componentes conexas es un clique-tree.

Sean C_1 y C_2 las componentes conexas de $T^{[\#2]}$. El caso en que $|C_1| = 1$ es trivial ya que T sería una estrella y C_2 sería un clique. Luego en el caso en que $|C_1| \geq 2$ se tiene directo del Lema 1.5. ■

El resultado del Teorema 1.6 nos hace plantear la siguiente pregunta: Dado un grafo G con dos componentes conexas clique-tree ¿existe un árbol cuyo cuadrado exacto es G ?

Para responder esta pregunta vamos a necesitar algunos lemas y algunas construcciones nuevas.

Primero veamos un lema y su corolario que serán útiles en el trabajo posterior.

Lema 1.7 *Un vértice v en T es vértice de corte de $T^{[\#2]}$ si y solamente si v tiene al menos dos vecinos no hojas en T .*

DEMOSTRACIÓN. Si v es un vértice de corte en $T^{[\#2]}$, hay al menos dos bloques B_1 y B_2 tal que $v \in B_1 \cap B_2$. Por el Lema 1.5, B_1 y B_2 son inducidos por las vecindades de dos vértices diferentes, w_1 y w_2 respectivamente, tal que $d(w_1) \geq 2$ y $d(w_2) \geq 2$. En particular, se tiene que $w_1, w_2 \in N_T(v)$ y no son hojas.

Para la otra dirección, asumimos que existen w_1 y w_2 vecinos no hojas de v en T . El Lema 1.5 nos dice que las vecindades de w_1 y w_2 inducen dos bloques, B_1 y B_2 respectivamente, en $T^{[\#2]}$. Notar que estos bloques no pueden ser iguales. Dado que se tiene $v \in N_T(w_1) \cap N_T(w_2)$ también se tiene que $v \in B_1 \cap B_2$, lo que significa que v es un vértice de corte de $T^{[\#2]}$. ■

Corolario 1.8 *Si $v \in V(T^{[\#2]})$ pertenece a k bloques de $T^{[\#2]}$, entonces v tiene al menos k vecinos en T .*

Ahora veremos una nueva construcción.

Sea C un clique-tree. Definimos el árbol canónico de C , denotado T_C , de la siguiente manera: Para cada bloque B en C creamos un nuevo vértice v_B y definimos el conjunto de vértices y aristas de T_C como $V(T_C) = V(C) \cup \{v_B | B \text{ es un bloque de } C\}$ y $E(T_C) = \{\{u, v_B\} | B \text{ es un bloque de } C \text{ y } u \in V(B)\}$. Si v_B es el vértice en T_C correspondiente al bloque B de C , diremos que v_B surge de B en T_C .

Veamos que T_C efectivamente es un árbol.

Lema 1.9 *Sea C un clique-tree. Entonces T_C es un árbol.*

DEMOSTRACIÓN. Sea $\{u_1, \dots, u_n\}$ los vértices de C y $\{B_1, \dots, B_m\}$ sus bloques. Dado que C es conexo es fácil notar que por la definición del árbol canónico se tiene que T_C también es conexo.

Ahora por contradicción supongamos que T_C tiene al menos un ciclo. Tomemos el menor ciclo W en T_C . Notar que el ciclo W sería una secuencia alternante de vértices de la forma $u_{i_1} v_{B_{j_1}} u_{i_2} v_{B_{j_2}} \dots u_{i_k} v_{B_{j_k}} u_{i_1}$ con $k \geq 2$. Esto implica que $u_{i_1} u_{i_2} \dots u_{i_k} u_{i_1}$ es un ciclo de C , eso significa que están en el mismo bloque contradiciendo que W es un ciclo en T_C . ■

Con esta construcción podemos concluir que todo grafo C clique-tree es una componente conexa del cuadrado de algún árbol. En particular de su árbol canónico T_C .

Teorema 1.10 *Un grafo es un clique-tree si y solamente si el grafo es isomorfo a una componente conexa del cuadrado exacto de un árbol.*

DEMOSTRACIÓN. Si un grafo C es un clique-tree no es difícil notar que es la componente conexa de $T_C^{[\#2]}$, que por el lema anterior sabemos que es un árbol. Para la otra dirección si el grafo es isomorfo a una componente conexa del cuadrado exacto de un árbol, se tiene que por el Teorema 1.6 cada una de las dos componentes conexas es un clique-tree. ■

Ahora responderemos la pregunta formulada anteriormente sobre cuándo la unión disjunta de dos clique-trees C_1 y C_2 es el cuadrado exacto de algún árbol.

Como veremos abajo, la idea es asignar a cada clique de C_1 un vértice v en C_2 tal que el clique represente la vecindad de v en un árbol. Esto claramente nos recuerda la forma en que se construye el árbol canónico de un clique-tree. En efecto, si consideramos el cuadrado exacto de T_{C_1} , una de sus componentes conexas será C_1 , y si la otra resulta ser isomorfa a C_2 entonces inmediatamente podemos decir que C_1 y C_2 son el cuadrado exacto de un árbol. Pero puede pasar que C_1 y C_2 son el cuadrado exacto de algún árbol pero C_1 no es isomorfo a ninguna de las componentes conexas de $T_{C_2}^{[\#2]}$ y C_2 tampoco es isomorfo a ninguna de las componentes conexas de $T_{C_1}^{[\#2]}$ (ver Figura 1.2). En este caso, sin embargo, mostraremos que la componente de $T_{C_1}^{[\#2]}$ que no es C_1 siempre será un subgrafo de C_2 . En tal caso, si encontramos un isomorfismo adecuado (que cumpla ciertas propiedades) entre esta componente y algún subgrafo de C_2 , entonces podremos completar (usando el Procedimiento 1 que se mostrará más adelante) el árbol canónico de C_1 de manera que dicha completación sea un árbol cuyo cuadrado exacto sea la unión disjunta de C_1 y C_2 .

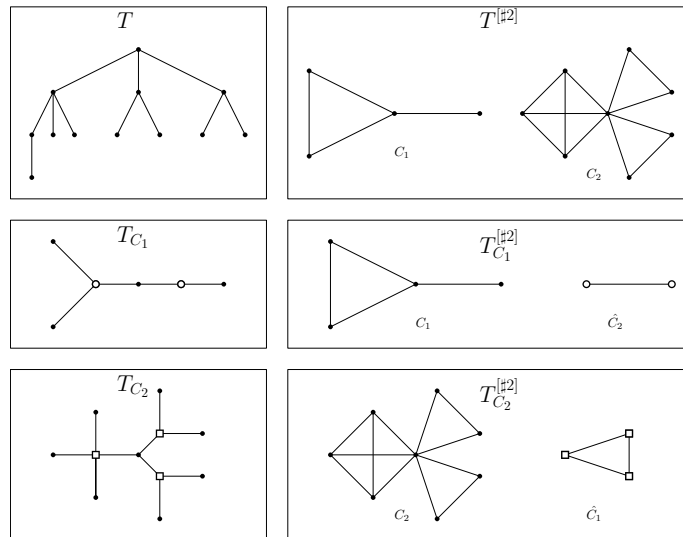


Figura 1.2: Un árbol T y su cuadrado exacto con componentes C_1 y C_2 . Aquí C_1 no es isomorfo a ninguna componente conexa de $T_{C_2}^{[\#2]}$ y C_2 no es isomorfo a ninguna componente conexa de $T_{C_1}^{[\#2]}$.

Un poco de notación. Vamos a denotar el conjunto de todos los vértices de corte de un grafo H , como $\text{VC}(H)$.

A continuación mostraremos una caracterización que nos permitirá responder nuestra pregunta.

Teorema 1.11 *Sea G un grafo con dos componentes conexas C_1 y C_2 cada una de ellas un clique-tree. Sea \hat{C}_2 la componente conexa de $T_{C_1}^{[\#2]}$ que no es C_1 . Entonces existe un árbol T tal que $T^{[\#2]} = G$ si y solamente si existe un isomorfismo $\varphi : V(\hat{C}_2) \rightarrow V(S) \subseteq V(C_2)$ tal que*

- (1) *Para todo vértice $v \in V(\hat{C}_2)$ se tiene que $d_{T_{C_1}}(v)$ es mayor o igual al número de bloques al cual $\varphi(v)$ pertenece en C_2 .*
- (2) *Si $x \in \text{VC}(C_2)$ entonces $x \in \varphi(\hat{C}_2)$.*

DEMOSTRACIÓN. Notar que el caso en que C_1 tiene un solo vértice la equivalencia se tiene fácilmente, concluyendo que C_2 tiene un solo bloque, es decir C_2 es un clique. Así que ahora asumiremos que C_1 tiene por lo menos dos vértices.

Primero vamos a asumir que existe un árbol T tal que $T^{[\#2]} = G$. Consideremos un vértice arbitrario $v \in V(\hat{C}_2) \subseteq V(T_{C_1})$. Este vértice es un vértice que surge en T_{C_1} a partir de un bloque B_v de C_1 . Por el Lema 1.5 existe $x \in V(C_2)$ tal que $B_v = C_1[N_T(x)]$. Sea $V_S = \{x \in V(C_2) | \exists v \in \hat{C}_2 \text{ que surge de } C_1[N_T(x)] \text{ en } T_{C_1}\}$ (es decir las no hojas de T que están en C_2) y sea $S = C_2[V_S]$. Definimos $\varphi : V(\hat{C}_2) \rightarrow V(S)$ tal que

$$\varphi(v) = x \text{ si y solo si } v \text{ surge de } C_1[N_T(x)] \text{ en } T_{C_1}.$$

Dado que los bloques son inducidos por la vecindad de un único vértice, se tiene que φ está bien definida y es una biyección.

Primero veamos que φ es un isomorfismo. Tener $uv \in E(\hat{C}_2)$ es equivalente a tener dos bloques B_u y B_v en C_1 , tal que u y v surgen en T_{C_1} por B_u y B_v respectivamente, y satisfacen que $B_u \cap B_v \neq \emptyset$. Pero B_u y B_v son inducidos en C_1 por las vecindades en T de $\varphi(u)$ y $\varphi(v)$, respectivamente, y como se intersectan, entonces se tiene que $\varphi(u)\varphi(v) \in E(C_2)$. Por lo tanto $uv \in E(\hat{C}_2)$ es equivalente a $\varphi(u)\varphi(v) \in E(C_2)$.

Ahora veamos que se cumple (1), sea $v \in V(\hat{C}_2)$. Notar que $d_{T_{C_1}}(v) = |V(\hat{B}_v)|$, donde \hat{B}_v es el bloque en C_1 tal que v surge de \hat{B}_v en T_{C_1} . Por la definición de φ , $\hat{B}_v = C_1[N_T(\varphi(v))]$. Esto nos entrega que $d_T(\varphi(v)) = |V(\hat{B}_v)|$. Dado que tenemos que $d_{T_{C_1}}(v) = d_T(\varphi(v))$, por el Corolario 1.8 tenemos (1). Para probar (2) tomamos un vértice $x \in \text{VC}(C_2)$. Notar que la vecindad de x en T induce un bloque B en C_1 , y por lo tanto existe un vértice $v_B \in \hat{C}_2$ que surge de B en T_{C_1} . Por lo tanto, por la definición de φ , se tiene que $\varphi(v_B) = x$.

Ahora vamos a asumir que existe un isomorfismo $\varphi : V(\hat{C}_2) \rightarrow V(S) \subseteq V(C_2)$ tal que satisface las condiciones (1) y (2). Vamos a construir a partir de T_{C_1} un árbol T tal que $T^{[\#2]} = G$.

Como φ es un isomorfismo y tanto C_2 como \hat{C}_2 son clique-tree, se tiene que $\varphi(\text{VC}(\hat{C}_2)) \subseteq \text{VC}(C_2)$. De este modo si C_2 no tiene vértices de corte, entonces \hat{C}_2 tampoco los tiene. Por lo

tanto C_2 y \hat{C}_2 tienen un único bloque y $|V(\hat{C}_2)| \leq |V(C_2)|$. Sea $w \in C_1$ el único vértice cuya vecindad en T_{C_1} induce \hat{C}_2 . Agregamos a T_{C_1} todos los vértices de $C_2 \setminus \varphi(\hat{C}_2)$ como vecinos hojas de w . El nuevo árbol satisface que su cuadrado exacto es isomorfo a G .

Ahora supongamos que C_2 tiene al menos un vértice de corte. Por (2) tenemos que para todo $x \in VC(C_2)$ existe $v \in \hat{C}_2$ tal que $\varphi(v) = x$. Para todo bloque B de C_2 definimos $A_B = \{w \in B | w \in C_2 \setminus \varphi(\hat{C}_2)\}$, es decir, A_B es el conjunto de vértices en B que no tienen preimagen por φ .

Ahora basados en la notación anterior vamos a introducir el Procedimiento 1 el cual aplicaremos para construir un árbol cuyo cuadrado exacto es G . Luego veremos su correctitud.

Procedimiento 1 COMPLETACIÓN RAIZ EXACTA ÁRBOL.

Input: T_{C_1}, G

Output: T árbol tal que $T^{[\#2]} \cong G$

$F \leftarrow \emptyset$

$T \leftarrow T_{C_1}$

for $x \in VC(C_2)$ **do**

$v \leftarrow \varphi^{-1}(x)$

for B bloque en C_2 tal que $x \in B$ y $B \setminus (A_B \cup \{x\}) \neq \emptyset$ **do**

Sea $y \in B \setminus (A_B \cup \{x\})$

$w \leftarrow \varphi^{-1}(y)$

Sea $z \in T$ tal que $z \in N_{T_{C_1}}(v) \cap N_{T_{C_1}}(w)$

if $z \notin F$ **then**

$V(T) \leftarrow V(T) \cup A_B$

$E(T) \leftarrow E(T) \cup \{az | a \in A_B\}$

$F \leftarrow F \cup \{z\}$

end if

end for

for B bloque en C_2 tal que $x \in B$ y $B \setminus (A_B \cup \{x\}) = \emptyset$ **do**

Let $h \in N_{T_{C_1}}(v) \setminus F$

$V(T) \leftarrow V(T) \cup A_B$

$E(T) \leftarrow E(T) \cup \{ah | a \in A_B\}$

$F \leftarrow F \cup \{h\}$

end for

end for

return T

Sea $x \in VC(C_2)$ y v su preimagen por φ . En el caso donde B satisface $B \setminus (A_B \cup \{x\}) \neq \emptyset$ tomamos $y \in B \setminus (A_B \cup \{x\})$ con su preimagen w . Dado que φ es un isomorfismo tenemos que $vw \in E(\hat{C}_2)$ y luego podemos tomar el único vértice $z \in N_{T_{C_1}}(v) \cap N_{T_{C_1}}(w)$. Si $z \notin F$ agregamos los vértices de A_B a T , cada uno como un vecino hoja de z y luego agregamos z a F . Si T ya era un árbol entonces esta operación sigue manteniéndolo como árbol: en efecto dado que $z \in V(C_1)$, en este caso solo agregamos hojas al vértice de T_{C_1} .

Dado que agregamos a F a los más un vértice por bloque al que x pertenece, por la condición 1 del teorema, existe un vértice $h \in N_{T_{C_1}}(v) \setminus F$ en el caso en que $B \setminus (A_B \cup \{x\}) = \emptyset$.

A este vértice h le conectamos los vértices de A_B como hojas. Dado que tenemos $x \in \text{VC}(C_2) \subseteq V(C_2)$ y $h \in N_{T_{C_1}}$, se debe tener que $h \in V(C_1)$. Por lo tanto todo lo que hacemos en este paso es agregar hojas a vértices de T_{C_1} . Es decir el resultado final (output) es en efecto un árbol. Más aún dado que asumimos que $\text{VC}(C_2) \neq \emptyset$, tenemos que cada bloque en C_2 contiene al menos un vértice de corte, y por ende en este proceso agregamos a T_{C_1} todos los vértices de $C_2 \setminus \varphi(\hat{C}_2)$ como hojas.

Para finalizar la prueba, debemos probar que $T^{[\#2]}$ es isomorfo a G . Definimos $\psi : V(T^{[\#2]}) \rightarrow V(G)$ tal que $\psi(v) = v$ si $v \in V(C_1) \cup V(C_2) \setminus (V(\varphi(\hat{C}_2)))$, o $\psi(v) = \varphi(v)$ si $v \in V(\hat{C}_2)$. Veamos que ψ es un isomorfismo. Es claro que ψ es una biyección. Ahora tomemos una arista $\psi(u)\psi(v) \in E(G)$. Hay solamente 4 posibles casos:

Caso 1: $\psi(u), \psi(v) \in V(C_1)$. Dado que T fue obtenido a partir de T_{C_1} simplemente agregándole hojas a los vértices en C_1 , tenemos que $\varphi(u)\varphi(v) = uv \in E(C_1)$ es equivalente a $uv \in E(T^{[\#2]})$.

Caso 2: $\psi(u), \psi(v) \in V(C_2) \setminus V(\varphi(\hat{C}_2))$. En este caso $\varphi(u)\varphi(v) = uv \in E(C_2 \setminus \varphi(\hat{C}_2))$ que es equivalente a que exista un bloque B en C_2 tal que $u, v \in A_B$. Por el Procedimiento 1, esto es equivalente a que $uv \in E(T^{[\#2]})$.

Caso 3: $\psi(u), \psi(v) \in V(\varphi(\hat{C}_2))$. En este caso tenemos que $\psi(u)\psi(v) = \varphi(u)\varphi(v) \in E(\varphi(\hat{C}_2))$ y dado que φ es un isomorfismo, esto es equivalente a tener que uv es una arista en \hat{C}_2 . Esto último es equivalente a tener que $uv \in E(T^{[\#2]})$.

Caso 4: $\psi(u) \in V(C_2) \setminus V(\varphi(\hat{C}_2))$ y $\psi(v) \in V(\varphi(\hat{C}_2))$. En este caso $\psi(u)\psi(v) = u\varphi(v) \in E(C_2)$. Esto es equivalente a que exista un bloque B en C_2 que contiene a $u\varphi(v)$ y $u \in A_B$. Pero en el algoritmo u es agregado a T como vecino de un vértice z que es vecino de v . En otras palabras, $u\varphi(v) \in E(C_2)$ es equivalente a $uv \in E(T^{[\#2]})$.

Así se tiene el resultado. ■

Corolario 1.12 *Un grafo es el cuadrado exacto de algún árbol no trivial si y solamente si tiene dos componentes conexas C_1 y C_2 , cada una de ellas un clique-tree, tal que existe un isomorfismo $\varphi : V(T_{C_1}^{[\#2]}) \setminus V(C_1) \rightarrow V(S) \subseteq V(C_2)$ tal que*

- (1) *Para todo vértice $v \in V(T_{C_1}) \setminus V(C_1)$ tenemos que $d_{T_{C_1}}(v)$ es al menos el número de bloques a los que $\varphi(v)$ pertenece en C_2 .*
- (2) *Si $x \in \text{VC}(C_2)$, entonces x pertenece a la imagen de φ .*

En la Figura 1.3 se muestra un ejemplo del Procedimiento 1:

1.1.1. Algoritmo de reconocimiento de cuadrados exactos de árboles

En esta subsección mostraremos un algoritmo que dado un grafo G decide si G tiene una raíz cuadrada exacta que es árbol.

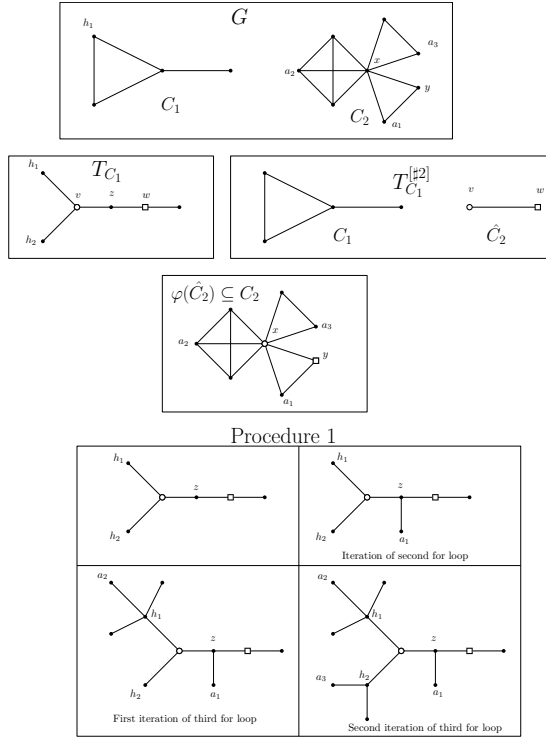


Figura 1.3: Construcción de un árbol T (usando Procedimiento 1) cuyo cuadrado exacto es G .

En [9] Lin y Skiena dieron un algoritmo que en tiempo lineal encuentra la raíz cuadrada (normal) árbol de los grafos que son cuadrados de árboles. En [8] Lau dió un algoritmo diferente y más simple que en tiempo lineal reconoce cuadrados (normales) de árboles. Nosotros construimos un algoritmo de decisión que en tiempo polinomial determina si dado un grafo G existe algún árbol T tal que $T^{[\#2]} = G$ y en caso positivo retorna dicho árbol.

En [10] Matula muestra un algoritmo que en tiempo polinomial decide si un árbol S con n_S vértices es isomorfo a algún subárbol de un árbol T con n_T vértices. Nosotros modificamos el Algoritmo A de [10] para obtener un algoritmo que dado un grafo G decide si G tiene una raíz cuadrada exacta que es árbol. Nuestro algoritmo se puede extender para que retorne dicho árbol en caso de existir.

Primero vamos a necesitar revisar el esquema general del algoritmo de Matula y para ello necesitaremos algunas definiciones y resultados claves de [10]. Para un árbol T y $r \in V(T)$, denotamos por $T[r]$ al árbol T con raíz en r . Para u, v tal que $uv \in E(T)$ definimos la *rama* o *limb* $T[u, v]$ como el subárbol maximal de T que contiene a la arista uv y u es una hoja, la cuál asignamos como la raíz de dicho subárbol. La *altura* de una rama $T[u, v]$ denota la mayor distancia de un vértice de $T[u, v]$ a la raíz u . Para un árbol enraizado $T[u]$, las ramas (o limbs) de $T[u]$ son las ramas $T[v, w]$ tal que w está más lejos de la raíz que v .

Sean S y T árboles con $x, y \in V(S)$ y $u, v \in V(T)$. El árbol enraizado $S[x]$ es isomorfo al árbol enraizado $T[u]$ si existe un isomorfismo entre S y T tal que la imagen de x es u . Un isomorfismo de la rama $S[x, y]$ a un subárbol de la rama $T[u, v]$ que tenga como raíz a u , se llama *limb embedding* de $S[x, y]$ en $T[u, v]$ (Notar que la imagen de x e y son u y v , respectivamente). Si existe dicho limb embedding, decimos que $S[x, y]$ puede ser embedded

en $T[u, v]$. Sea a_1, a_2, \dots, a_p los vecinos de y en $S[x, y] - x$, y b_1, b_2, \dots, b_q , los vecinos de v en $T[u, v] - u$. La *highest-limbs matrix* asociada con $S[x, y]$ y $T[u, v]$ es una matriz de $p \times q$ tal que las fila corresponden a los limbs $S[y, a_i]$, $1 \leq i \leq p$, las columnas corresponden a los limbs $T[v, b_j]$, $1 \leq j \leq q$, y la posición $S[y, a_i], T[v, b_j]$ tiene valor 1 si $S[y, a_i]$ puede ser embedded en $T[v, b_j]$ y 0 en otro caso.

Para una 0, 1-matriz de $p \times q$ entradas, con $p \leq q$, un *matching bipartito* es un conjunto de entradas tal que ninguna entrada del conjunto esta en la misma fila o columna que otra entrada del conjunto. Un matching bipartito máximo es *completo para las filas* si hay p entradas en el matching bipartito, una por cada fila.

El siguiente teorema es fundamental para el algoritmo del subarbol-isomorfismo en [10].

Teorema 1.13 (Matula [10]) *La rama $S[x, y]$ es isomorfa a un subarbol enraizado de la rama $T[u, v]$ (con u como la raíz) si y solamente si la highest-limbs matrix asociada tiene un matching bipartito máximo que es completo para las filas.*

Para completar los ingredientes del algoritmo de Matula necesitamos la siguiente definición. La *limb embedding matrix* $M(S[x, y], T)$ es una matriz que tiene una fila por cada una de las $n_S - 1$ ramas del árbol enraizado $S[x, y]$ y una columna por cada una de las $2(n_T - 1)$ ramas del árbol (sin raíz) T . Más específicamente, vamos a ordenar las ramas de $S[x, y]$ de forma no-decreciente según la altura y ordenamos las filas de acuerdo a esto. Para todos los pares de ramas $S[a, b]$ de $S[x, y]$ y $T[u, v]$ de T , la entrada para la posición $S[a, b], T[u, v]$ es 1 si $S[a, b]$ puede ser embedded en $T[u, v]$ y 0 en caso contrario.

Notar que decidir si existe un subárbol de T isomorfo a S es equivalente a decidir si la rama $S[l, y]$ enraizada en una hoja l de S puede ser embedded en alguna rama de T . No es difícil ver que la rama $S[l, y]$ puede ser embedded en alguna rama de T si y solamente si hay un 1 en alguna entrada de la última fila de la correspondiente limb embedding matrix. Por lo tanto necesitamos una manera eficiente de computar dicha matriz. Para todas las filas de la matriz correspondientes a ramas de altura 1 de $S[l, y]$ se fijan todas las entradas como 1. Ahora, asumiendo que la matriz ha sido computada correctamente hasta alguna fila, se construye (llena) la siguiente fila, correspondiente a la rama, digamos, S' de $S[l, y]$, como sigue: Para cada entrada construimos la highest-limbs matrix de S' y T' , donde T' es la rama de T correspondiente a la columna de dicha entrada. Esta matriz solo necesita información de las ramas de menor altura, y esa información la podemos obtener de las filas previas de la limb embedding matrix. Decidir el valor de una entrada se reduce, por el Teorema 1.13, a un problema de matching bipartito. Dado que solo necesitamos resolver una cantidad cuadrática (en $\max\{|V(S)|, |V(T)|\}$) de dichos problemas, el algoritmo corre en tiempo polinomial. De hecho, un isomorfismo, si existe, se puede recuperar en tiempo polinomial, como mencionaremos posteriormente.

Algoritmo de Decisión

Ahora mostraremos el algoritmo que se puede dividir en dos grandes etapas. Como input recibe un grafo G , y decide si existe algún árbol T tal que $T^{\{\#2\}} = G$.

Etapa I. Inicialización

En esta etapa computamos algunos grafos e información importante que será usada en la segunda etapa.

- I.1 Decidir si G tiene exactamente dos componentes conexas C_1, C_2 , cada una de ellas un clique-tree. Si no se tiene, entonces retorna que no existe un árbol cuyo cuadrado exacto es G .
- I.2 Para cada vértice v en C_2 se guarda el número $b_{C_2}(v)$ de bloques a los que pertenece en C_2 .
- I.3 A partir de C_1 y C_2 se construye los árboles canónicos T_{C_1} y T_{C_2} .
- I.4 A partir de T_{C_1} se construye $T_{C_1}^{[\#2]}$. Notar que $T_{C_1}^{[\#2]}$ consiste en C_1 y otro clique-tree \hat{C}_2 . Para cada vértice v en \hat{C}_2 se guarda $d_{T_{C_1}}(v)$.
- I.5 A partir de \hat{C}_2 se construye $T_{\hat{C}_2}$.

Etapa II. Obtener el isomorfismo

En esta etapa se determina si existe un isomorfismo de $T_{\hat{C}_2}$ a algún subárbol de T_{C_2} del cual podamos construir un isomorfismo entre \hat{C}_2 y algún subgrafo de C_2 que satisfaga las condiciones del Teorema 1.11.

- II.1 Enraizar $T_{\hat{C}_2}$ en una hoja l cuyo vecino se llamará y . Luego ordenar las $|V(T_{\hat{C}_2})| - 1$ ramas de $T_{\hat{C}_2}[l, y]$ en orden no-decreciente según su altura. Después se crea la matriz $M := M(T_{\hat{C}_2}[l, y], T_{C_2})$ que tiene una fila por cada rama de $T_{\hat{C}_2}[l, y]$ (las filas se ordenan de acuerdo a su correspondiente altura) y una columna por cada una de las ramas de T_{C_2} . Note que M juega el rol de la limb embedding matrix.
- II.2 Por cada una de las ramas $T_{\hat{C}_2}[a, b]$ de $T_{\hat{C}_2}[l, y]$ de altura 1, y por cada rama $T_{C_2}[u, v]$ de T_{C_2} se fijarán las entradas asociadas con estos pares de ramas en M como 1 si alguna de las siguientes dos condiciones se cumple:
 - (i) $a \in V(\hat{C}_2)$, $u \in V(C_2)$ y $d_{T_{C_1}}(a) \geq b_{C_2}(u)$,
 - (ii) $b \in V(\hat{C}_2)$, $v \in V(C_2)$ y $d_{T_{C_1}}(b) \geq b_{C_2}(v)$.

En caso contrario la entrada se fija como 0.

- II.3 Para h que corre entre 2 y la altura de $T_{\hat{C}_2}[l, y]$:
Para cada rama $T_{\hat{C}_2}[a, b]$ de $T_{\hat{C}_2}[l, y]$ de altura h , y para cada rama $T_{C_2}[u, v]$ de T_{C_2} , se crea la matriz $H := H(T_{\hat{C}_2}[a, b], T_{C_2}[u, v])$, que tiene una fila por cada vecino de b en $T_{\hat{C}_2}[a, b] - a$, y una columna por cada vecino de v en $T_{C_2}[u, v] - u$ (H juega el rol de la highest-limbs matrix). Para cada vecino $c \neq a$ de b , y cada vecino $w \neq u$ de v , se fija la entrada correspondiente de H como 1 si la entrada $T_{\hat{C}_2}[b, c], T_{C_2}[v, w]$ de M es 1, y 0 en caso contrario. Determinar un matching bipartito máximo para H . Si es completo para las filas y se cumple alguna de las condiciones (i) o (ii) entonces se fija el valor de la entrada $T_{\hat{C}_2}[a, b], T_{C_2}[u, v]$ de M como 1. En caso contrario se fija como 0.

II.4 Para cada 1 de la última fila se computa el isomorfismo asociado, de la siguiente manera. Si se preservan las soluciones de todos los problemas de matching máximo por cada entrada con un 1 en M , entonces uno se puede devolver, partiendo desde el matching correspondiente a ese 1, el subárbol de T_{C_2} isomorfo a $T_{\hat{C}_2}[a, b]$ y su correspondiente isomorfismo. Notar que no es necesario guardar estos matching dado que se pueden computar nuevamente a partir de M .

Si hay un 1 en la última fila de M tal que el subárbol asociado de T_{C_2} que es isomorfo a $T_{\hat{C}_2}$ contiene todos los vértices de corte de C_2 , entonces retorna que existe un árbol cuyo cuadrado exacto es G . En caso contrario, retorna que no existe dicho árbol.

Para ver que la etapa I puede ser computada en tiempo polinomial ayuda tener en mente que los bloques y vertices de corte de un grafo pueden ser computados en tiempo cuadrático [13]. La etapa II puede ser computada en tiempo polinomial dado que el número de problemas de matching bipartito que hay que resolver es a lo más cuadrático en el número de vértices de G .

Ahora hay que revisar la correctitud del algoritmo. Antes de eso serán necesarios algunos lemas.

Lema 1.14 *Sean C y D dos clique-trees. Si $\varphi : T_C \rightarrow S \subseteq T_D$ es un isomorfismo tal que $\varphi(V(T_C) \cap V(C)) \subseteq V(S) \cap V(D)$, entonces se tiene que $\varphi(C) \subseteq V(D)$ y que $\psi : C \rightarrow D[\varphi(C)]$ tal que $v \in C$ $\psi(v) = \varphi(v)$ es un isomorfismo.*

DEMOSTRACIÓN. Sea $v \in C$. Por la definición de T_C se tiene que $v \in T_C$. Por hipótesis se tiene que $\varphi(v) \in V(D)$.

Ahora veamos que ψ es un isomorfismo. Claramente esta función es una biyección. Ahora veamos que es un isomorfismo, mostrando que $uv \in E(C)$ es equivalente a que $\varphi(u)\varphi(v) \in D$. Es claro que $uv \in E(C)$ si y solamente si existe algún $w \in V(T_C) \setminus V(C)$ tal que uvw es un camino en T_C . Pero dado que φ es un isomorfismo esto es equivalente a que $\varphi(u)\varphi(w)\varphi(v)$ es un camino en S con $\varphi(w) \in S \setminus V(D)$. Y su vez esto es equivalente a $\varphi(u)\varphi(v) \in D$, que es lo que se buscaba. ■

Sea G un grafo con dos componentes conexas C_1 y C_2 tal que cada una de ellas es un clique-tree. Sea \hat{C}_2 la componente conexa de $T_{C_1}^{[\#2]}$ que no es C_1 . Decimos que la rama $T_{\hat{C}_2}[a, b]$ puede ser *well embedded* en $T_{C_2}[u, v]$ si $T_{\hat{C}_2}[a, b]$ puede ser embedded en $T_{C_2}[u, v]$ a través de un isomorfismo que mapea cada vértice de $s \in V(T_{\hat{C}_2}[a, b]) \cap V(\hat{C}_2)$ a un vértice $f_s \in V(C_2)$ tal que $d_{T_{C_1}}(s) \geq b_{C_2}(f_s)$. Decimos que un isomorfismo entre $T_{\hat{C}_2}$ y un subárbol de T_{C_2} es un *buen isomorfismo* si cada vértice $s \in V(T_{\hat{C}_2}) \cap V(\hat{C}_2)$ es mapeado a un vértice $f_s \in V(C_2)$ tal que $d_{T_{C_1}}(s) \geq b_{C_2}(f_s)$. Decimos que un buen isomorfismo es *perfecto* si el subárbol de T_{C_2} al que $T_{\hat{C}_2}$ es mapeado contiene todos los vértices de corte de C_2 .

El ingrediente final de nuestra demostración es nuestra versión del Teorema 1.13.

Lema 1.15 *Sea $h \geq 2$ un entero. Supongamos que las entradas de M han sido computadas de tal manera que para una rama de $T_{\hat{C}_2}[l, y]$ de altura $h - 1$ y una rama de T_{C_2} , la posición*

de M correspondiente a esas ramas es 1 si y solamente si la rama de $T_{\hat{C}_2}[l, y]$ puede ser well embedded en la de T_{C_2} . Una rama $T_{\hat{C}_2}[a, b]$ de $T_{\hat{C}_2}[l, y]$ de altura h puede ser well embedded a una rama $T_{C_2}[u, v]$ de T_{C_2} si y solamente si la matriz asociada $H := H(T_{\hat{C}_2}[a, b], T_{C_2}[u, v])$ tiene un matching bipartito máximo que es completo para las filas y se satisface alguna de las condiciones (i) o (ii).

DEMOSTRACIÓN. Primero asumamos que $T_{\hat{C}_2}[a, b]$ puede ser well embedded en $T_{C_2}[u, v]$. Esto implica que hay un isomorfismo φ desde $T_{\hat{C}_2}[a, b]$ hacia $T_{C_2}[u, v]$ que es un buen isomorfismo y, en particular, mapea a hacia u . Si tenemos $a \in V(\hat{C}_2)$, esto nos dice que $u \in V(C_2)$ y $d_{T_{C_1}}(a) \geq b_{C_2}(u)$, en otras palabras, (i) se satisface. Si en vez de eso se tiene $b \in V(\hat{C}_2)$, entonces tenemos que (ii) se satisface.

Recordar que las entradas de la matriz H se fijan usando la información de la matriz M sobre las ramas de $T_{\hat{C}_2}[l, y]$ de altura $h - 1$. Para cada vecino c de b en $T_{\hat{C}_2}[a, b] - a$ se tiene que $T_{\hat{C}_2}[b, c]$ puede ser embedded hacia $T_{C_2}[v, \varphi(c)]$. Esto en conjunto con el supuesto en M implica que la entrada $c, \varphi(c)$ de H es 1. Dado que φ es un isomorfismo, tenemos que para cada fila en H asociada a la vecindad de b en $T_{\hat{C}_2}[a, b] - a$ tenemos una columna diferente en H con una entrada con un uno, i.e., H tiene un matching bipartito máximo que es completo para las filas.

Para la otra dirección, notar que el supuesto en M implica que si H tiene un matching bipartito máximo que es completo para las filas, entonces para cada vecino c_i de b en $T_{\hat{C}_2}[a, b] - a$ existe un buen isomorfismo ϕ_i de $T_{\hat{C}_2}[b, c_i]$ hacia $T_{C_2}[v, \phi_i(c_i)]$. Definimos ϕ tal que $\phi(a) = u$ y $\phi(x) = \phi_i(x)$ para todo $x \in V(T_{\hat{C}_2}[b, c_i])$ y cada vecino c_i de b en $T_{\hat{C}_2}[a, b] - a$. Notar que ϕ es un isomorfismo de $T_{\hat{C}_2}[a, b]$ hacia una rama de $T_{C_2}[u, v]$, en otras palabras, $T_{\hat{C}_2}[a, b]$ puede ser embedded hacia $T_{C_2}[u, v]$. Más aún, si uno de entre (i) y (ii) se satisface, se tiene que dicho isomorfismo es un buen isomorfismo. ■

Teorema 1.16 *Las Etapas I y II deciden correctamente si el grafo G del input contiene una raíz cuadrada exacta que es árbol.*

DEMOSTRACIÓN. Primero vamos a mostrar que una vez que la Etapa I haya chequeado que G es un grafo con dos componentes conexas C_1 y C_2 , cada una de ellas un clique-tree, entonces en la Etapa II es suficiente decidir si $T_{\hat{C}_2}$ tiene un isomorfismo perfecto hacia un subgrafo de T_{C_2} . En efecto, si existe dicho isomorfismo φ , entonces por el Lema 1.14 (con $C = \hat{C}_2$ y $D = C_2$) se tiene un isomorfismo $\psi: V(\hat{C}_2) \rightarrow V(C_2)$ que, dado que φ es perfecto, satisface las condiciones del Teorema 1.12, y garantiza la existencia de una raíz exacta árbol. Para la inversa, si existe una raíz exacta árbol, entonces el Teorema 1.12 garantiza la existencia de un isomorfismo de $V(\hat{C}_2)$ hacia $V(C_2)$ que puede ser fácilmente extendido a un isomorfismo perfecto de $T_{\hat{C}_2}$ hacia algún subgrafo de T_{C_2} .

Ahora notar que decidir si $T_{\hat{C}_2}$ tiene un isomorfismo perfecto hacia un subgrafo de T_{C_2} es equivalente a decidir si la rama $T_{\hat{C}_2}[l, y]$ enraizada en una hoja l de $T_{\hat{C}_2}$ puede ser well embedded en alguna rama de T_{C_2} que contiene todos los vértices de corte de C_2 .

Mostraremos por inducción en h que los pasos II,2 y II,3 computan M de tal manera que para una rama $T_{\hat{C}_2}[a, b]$ de $T_{\hat{C}_2}[l, y]$ de altura h y una rama $T_{C_2}[u, v]$ de T_{C_2} , la posición

$T_{\hat{C}_2}[a, b], T_{C_2}[u, v]$ en M es 1 si y solamente si $T_{\hat{C}_2}[a, b]$ puede ser well embedded en $T_{C_2}[u, v]$. En particular, esto implica que hay un 1 en la última fila de M si y solamente si $T_{\hat{C}_2}[l, y]$ puede ser well embedded en algún subgrafo de T_{C_2} . Dado que la Etapa II retorna que hay una raíz exacta árbol para G si y solamente si para alguna de esas entradas con un 1 el isomorfismo asociado tiene todos los vértices de corte de C_2 en su imagen, por los argumentos dados en los dos párrafos previos, se tiene que esto implica la correctitud del algoritmo.

Let $T_{\hat{C}_2}[a, b]$ be a limb of $T_{\hat{C}_2}[l, y]$ of height $h = 1$. Notice that $T_{\hat{C}_2}[a, b]$ can be embedded in any limb of T_{C_2} , and that one of the conditions *II,2.(i)* and *II,2.(ii)* is satisfied if and only if $T_{\hat{C}_2}[a, b]$ can be well embedded in the limb of T_{C_2} .

Ahora asumamos que las entradas de M han sido computadas preservando la propiedad deseada para todas las filas correspondiente a las ramas de $T_{\hat{C}_2}[l, y]$ de altura a lo más $h - 1$ para algún $h \geq 2$ (esta es nuestra hipótesis inductiva).

Sea $T_{\hat{C}_2}[a, b]$ una rama de $T_{\hat{C}_2}[l, y]$ de altura h y sea $T_{C_2}[u, v]$ una rama de T_{C_2} . Consideremos la matriz $H := H(T_{\hat{C}_2}[a, b], T_{C_2}[u, v])$ asociada a estas ramas. Notar que las ramas $T_{\hat{C}_2}[b, c]$ de $T_{\hat{C}_2}[a, b]$, donde $c \neq a$ es un vecino de $T_{\hat{C}_2}$, también son ramas de $T_{\hat{C}_2}[l, y]$ y tienen altura a lo más $h - 1$. Y las ramas $T_{C_2}[v, w]$ de T_{C_2} , donde $w \neq u$ es un vecino de v , también son ramas de T_{C_2} , por lo tanto la información necesaria para llenar las entradas de H están disponibles en la porción de M computada anteriormente. Ahora el paso *II,3* determina un matching máximo para H y, por el Lema 1.15, $T_{\hat{C}_2}[a, b]$ puede ser well embedded en $T_{C_2}[u, v]$ si y solamente si el matching es completo para las filas y una de las condiciones entre (i) y (ii) se cumple. Dado que este paso pone un 1 en la entrada de M correspondiente a $T_{\hat{C}_2}[a, b]$ y $T_{C_2}[u, v]$ si y solamente si el matching es completo para las filas y una de las condiciones entre (i) y (ii) se cumplen, entonces esta entrada de M es computada de forma que preserve la propiedad deseada.

Por lo tanto el resultado se tiene. ■

Al inicio de esta demostración se mostró que una vez que el algoritmo ha decidido que existe un isomorfismo perfecto desde $T_{\hat{C}_2}$ hacia un subgrafo de T_{C_2} , entonces a partir de este isomorfismo se puede obtener, usando el Lema 1.14, un isomorfismo que satisface las condiciones del Teorema 1.12. A partir de este último isomorfismo se puede obtener una raíz exacta árbol de G usando el procedimiento dado en la demostración del Teorema 1.11.

1.2. Caracterización de grafos con raíz cuadrada exacta triangle-free

Un grafo G se dice que es triangle-free si no contiene ningún ciclo de largo 3 como subgrafo.

A continuación se presenta una caracterización de los grafos que tienen raíz cuadrada exacta triangle-free.

Teorema 1.17 *Un grafo G con vértices v_1, v_2, \dots, v_n admite una raíz cuadrada exacta triangle-free si y solamente si G tiene una colección de cliques Cl_1, Cl_2, \dots, Cl_n que satisfacen las siguientes propiedades:*

- $v_i \notin Cl_i$, para cada i .
- Si $v_i \in Cl_j$, entonces $v_j \in Cl_i$ para cada par i, j .
- $v_i v_j \in E(G)$ si y solamente si $Cl_i \cap Cl_j \neq \emptyset$.
- Para cada $i \in [n]$, y para cada par v_j, v_k de vértices de Cl_i se tiene que $v_j \notin Cl_k$.

DEMOSTRACIÓN. Para la dirección fácil, asumimos que H es una raíz cuadrada exacta triangle-free de G . Por la definición se tiene que $V(H) = V(G)$. Basta con tomar $Cl_i = N_H(v_i)$. Notar que si H es triangle-free, entonces $N_H(v_i)$ induce un clique en G .

La primera condición se tiene por la definición de la vecindad, ya que $v_i \notin N_H(v_i) = Cl_i$.

La segunda condición también se tiene directamente ya que si $v_i \in Cl_j = N_H(v_j)$, entonces $v_j \in N_H(v_i) = Cl_j$.

Veamos la tercera condición, notar que si $v_i v_j \in E(G)$, entonces en H están a distancia 2. Sea x un vecino común de v_i y v_j en H , entonces $x \in Cl_i \cap Cl_j$. Inversamente, si hay un vértice x en $Cl_i \cap Cl_j$, entonces v_i y v_j están a distancia a lo más 2 en H , pero como H es triangle-free, entonces están a distancia exactamente 2 y por lo tanto son adyacentes en G .

Para chequear que la última condición se mantiene, notar que si v_j y v_k son ambos vecinos de v_i , dado que H es triangle-free, v_j no es adyacente a v_k en H y, por lo tanto, $v_j \notin Cl_k$.

Para la otra dirección, sean Cl_1, Cl_2, \dots, Cl_n una colección de cliques que satisfacen las propiedades. Vamos a construir un grafo H y se mostrará que dicho grafo es una raíz cuadrada exacta triangle-free de G . Para los vértices de H tendremos $V(H) = V(G)$. Para las aristas, cada vértice v_i es adyacente a todos los vértices en Cl_i . La primera condición implica que H no tiene loops. La segunda condición implica que $Cl_i = N_H(v_i)$. La tercera condición implica que si v_i y v_j están a distancia 2 en H , entonces son adyacentes en G ya que sus vecinos comunes estarán en Cl_i y en Cl_j , por lo tanto efectivamente $H^{[\#2]} = G$. Finalmente, usando la última condición mostraremos que H debe ser triangle-free. Esto se tiene porque, si v_i es adyacente a los vértices v_j y v_k en H (i.e., $v_j, v_k \in Cl_i$), entonces por la última condición $v_j \notin Cl_k$ y, por lo tanto, por la construcción de H , v_j no es adyacente a v_k en H . ■

1.3. Cubos exactos de árboles

El objetivo de esta sección es caracterizar los cubos exactos de árboles y determinar si el problema de su reconocimiento se puede resolver en tiempo polinomial.

Ahora consideremos T un árbol. Por el Lema 1.1 se tiene que si $\text{diam}(T) \geq 3$ entonces $T^{[\#3]}$ tiene a lo más 3 componentes conexas.

Es fácil notar que si $\text{diam}(T) \leq 2$ entonces T es una estrella (pensar en un camino de tamaño máximo). Pero si T es una estrella entonces $T^{[\#3]}$ tiene tantas componentes conexas como vértices, es decir es un grafo sin aristas, cuyo reconocimiento es trivial. Por lo tanto de ahora en adelante en esta sección vamos a asumir que $\text{diam}(T) \geq 3$.

Una estructura que será relevante para el estudio de los cubos exactos de árboles es la que denotaremos por P_5^+ . Se define la estructura P_5^+ de la siguiente manera: Al camino $P_5 = v_1v_2v_3v_4v_5$ se le agrega un vértice v_6 que esta unido al vértice central v_3 del P_5 .

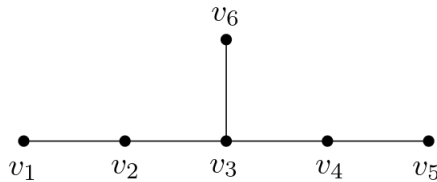


Figura 1.4: P_5^+ .

Esta estructura permite enunciar el siguiente resultado:

Lema 1.18 *Si T contiene a P_5^+ entonces $T^{[\#3]}$ tiene a lo más 2 componentes conexas.*

DEMOSTRACIÓN. Notar que si $T = P_5^+$ entonces los vértices v_1, v_2, v_4, v_5 y v_6 están en la misma componente conexa mientras que v_3 esta en una segunda componente conexa. Ahora notar que cualquier vértice que se le agregue a esta estructura necesariamente esta a distancia múltiplo de tres de alguno de los seis vértices del P_5^+ por lo tanto pertenecería a la componente conexa respectiva, incluso esas dos componentes se podrían conectar (formando una sola componente). ■

1.3.1. Cuando $T^{[\#3]}$ tiene 3 componentes conexas

Vamos a definir una nueva estructura que llamaremos *dobles escobas*. Para $n, m_1, m_2 \in \mathbb{N}$ se define el grafo *doble escoba* $B(n, m_1, m_2)$ como el grafo obtenido del camino P_n al agregarle m_1 nuevos vértices unidos a la primera hoja del camino y m_2 nuevos vértices unidos a la segunda hoja del camino.

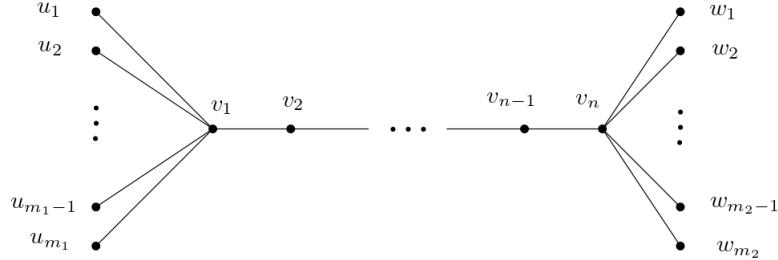


Figura 1.5: $B(n, m_1, m_2)$.

Cuando se tenga que $m_1 = 0$ y $m_2 \geq 2$ se hablará simplemente de *escoba*. Si $n = 1$ entonces $B(1, m_1, m_2) = K_{1, m_1 + m_2}$.

Por el Lema 1.18 se puede concluir que si el cubo exacto de un árbol tiene tres componentes conexas entonces el árbol no contiene a la estructura P_5^+ .

Podemos notar que los únicos árboles que no contienen a la estructura P_5^+ , son las doble escobas $B(n, m_1, m_2)$ con $n \geq 3$ y en el caso en que $n = 3$ se tiene que m_1 o m_2 es distinto de cero (para que se cumpla que $\text{diam}(T) \geq 3$). Esto ya que si el árbol no contiene a la estructura P_5^+ , significa que no contiene un vértice de grado mayor o igual que tres que tiene al menos dos vecinos de grado mayor o igual que dos.

Es fácil ver que el cubo exacto del árbol $B(n, m_1, m_2)$ tiene la siguiente forma:

- Si $n \equiv_3 0$ entonces $B(n, m_1, m_2)^{[\#3]} = B(\frac{n}{3}, m_1, 0) + B(\frac{n}{3}, m_2, 0) + P_{\frac{n}{3}}$
- Si $n \equiv_3 1$ entonces $B(n, m_1, m_2)^{[\#3]} = B(\frac{n-1}{3}, m_1, 0) + B(\frac{n-1}{3}, m_2, 0) + P_{\frac{n+2}{3}}$
- Si $n \equiv_3 2$ entonces $B(n, m_1, m_2)^{[\#3]} = P_{\frac{n+1}{3}} + P_{\frac{n+1}{3}} + B(\frac{n-2}{3}, m_1, m_2)$

En la figura 1.6 se muestra un ejemplo:

Dado que las únicas posibilidades son las recién mencionadas, es fácil pensar en algún algoritmo que en tiempo polinomial determine si un grafo de tres componentes conexas es el cubo exacto de algún árbol. El algoritmo primero tiene que corroborar que se tienen tres componentes conexas. Luego cuenta la cantidad de vértices n que tiene para así determinar si $n \equiv_3 0$, $n \equiv_3 1$ o $n \equiv_3 2$. Finalmente dependiendo de en qué caso estemos, hay que reconocer si la estructura de cada componente corresponde al cubo exacto de una doble escoba. Por lo que para el caso de tres componentes conexas el problema de reconocimiento se puede hacer en tiempo polinomial.

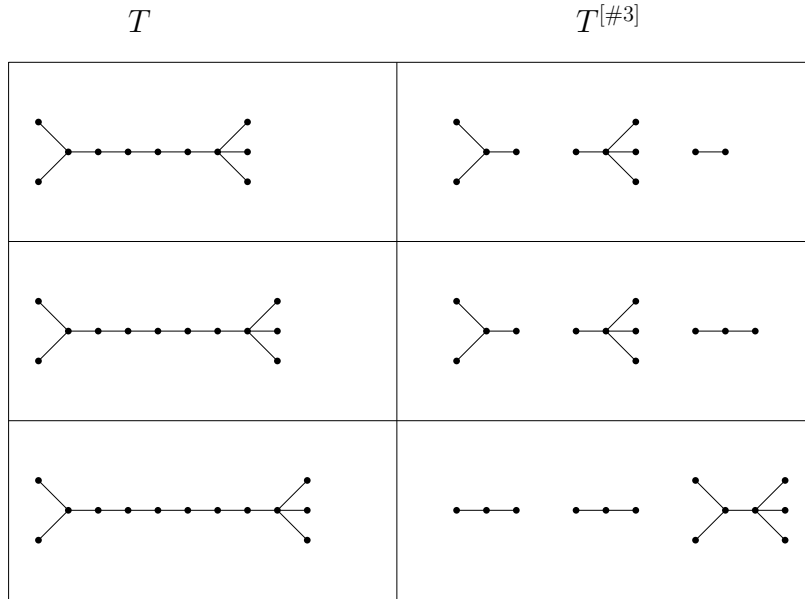


Figura 1.6: Caso $m_1 = 2$, $m_2 = 3$ y $n = 6, 7, 8$.

1.3.2. Cuando $T^{[\#3]}$ tiene 2 componentes conexas

Antes de analizar el caso de dos componentes conexas serán necesarios algunas definiciones y resultados que sirven para el caso de una y dos componentes conexas.

Para $k \in \mathbb{N}$ y $m \in \{3k + 6, 3k + 7\}$ definimos la siguiente estructura, que denotaremos $P_m^{3,m-2}$: Al camino $P_m = v_1 \dots v_m$ se le agrega un vértice u que esta conectado al vértice v_3 y un vértice w al vértice v_{m-2} del P_m .



Figura 1.7: $P_m^{3,m-2}$.

Lema 1.19 $\forall k \in \mathbb{N}$ si T contiene a $P_m^{3,m-2}$ con $m \in \{3k + 6, 3k + 7\}$, entonces $T^{[\#3]}$ tiene una sola componente conexa.

DEMOSTRACIÓN. Vamos a asumir que T contiene al $P_m^{3,m-2}$ para algún $k \in \mathbb{N}$. Notar que para demostrar que $T^{[\#3]}$ tiene una sola componente conexa basta con demostrar que todos los vértices de la estructura $P_m^{3,m-2}$ pertenecen a la misma componente conexa, ya que cualquier otro vértice fuera de la estructura va a estar a distancia múltiplo de tres de algún vértice de $P_m^{3,m-2}$ y por ende va a estar en la misma componente conexa.

El primer caso es que $m = 3k + 6$. Notar que en la Figura 1.7 si no consideramos los vértices u y w , se obtiene que tenemos tres componentes conexas en el cubo exacto de ese camino:

1. $C1 = \{v_{3l-2} | l \in [k+2]\}$
2. $C2 = \{v_{3l-1} | l \in \cup[k+2]\}$
3. $C3 = \{v_{3l} | l \in \cup[k+2]\}$

Ahora veamos que los v3rtices u y w unen estas tres componentes conexas. Notar que u esta conectado en $T^{[\#3]}$ a v_1 y a v_5 por lo tanto la componente $C1$ y C_2 se conectan. Por otra parte el v3rtice w se conecta con v_{3k+6} y con v_{3k+2} por lo tanto C_2 se conecta con la componente C_3 por lo que est1 conectado a la misma componente de C_1 .

De forma similar se obtiene que para el caso en que $m = 3k + 7$ todos los v3rtices de la estructura pertenecen a la misma componente conexa

En conclusi3n el cubo exacto de T tiene una sola componente conexa. ■

Vamos a decir que un v3rtice $v \in V(T)$ cumple con la propiedad de *relevancia*, 3 que v es *relevante*, si $d_T(v) \geq 3$ y tiene al menos dos vecinos u, w tal que $d_T(u) \geq 2$ y $d_T(w) \geq 2$.

Con esta nueva definici3n hay que notar que la estructura $P_m^{3,m-2}$ de la Figura 1.7 tiene dos v3rtices relevantes, que son el v_3 y el v_{m-2} . Mientras que el P_5^+ de la Figura 1.4 tiene un solo v3rtice relevante que es el v_3 .

Observaci3n: Los Lemas 1.18 y 1.19 se pueden reescribir usando la definici3n de v3rtices relevantes de la siguiente manera respectivamente:

1. Si T contiene al menos un v3rtice relevante entonces $T^{[\#3]}$ tiene a lo m1s dos componentes conexas.
2. Si T contiene al menos dos v3rtices relevantes u y v tal que est1n a distancia que no es m3ltiplo de 3, entonces $T^{[\#3]}$ tiene una sola componente conexa.

Ahora vamos a demostrar los siguientes resultados:

Lema 1.20 *Todos los v3rtices relevantes de T pertenecen a la misma componente conexa en $T^{[\#3]}$.*

DEMOSTRACI3N. Primero supongamos que existen dos v3rtices relevantes en T que est1n distancia $3k+1$ o $3k+2$ con $k \in \mathbb{N}$, entonces hay que notar que el 1rbol contiene a la estructura $P_m^{3,m-2}$ con $m = 3k+6$ o $3k+7$ respectivamente. Por el Lema 1.19 se tiene que T contiene una sola componente conexa y por ende todos los v3rtices relevantes pertenecen a la misma componente.

El otro caso es que cualquier par de v3rtices relevantes est1n a distancia m3ltiplo de 3 entre ellos. Eso implica que todos los v3rtices relevantes est1n en la misma componente. ■

Lema 1.21 *Si T contiene un v3rtice v relevante tal que v pertenece a la misma componente conexa en $T^{[\#3]}$ que alguno de sus vecinos $N_T(v)$, entonces $T^{[\#3]}$ contiene una sola componente conexa.*

DEMOSTRACIÓN. Notar que v pertenece a un P_5^+ y representaría el v_3 de la Figura 1.4. En dicha estructura todos los vértices, salvo v_3 pertenecerían a la misma componente y cualquier otro vértice está a distancia múltiplo de tres de alguno de ellos. ■

Con los resultados anteriores podemos enunciar la siguiente equivalencia:

Proposición 1.22 *Existe un número natural $k \in \mathbb{N}$ tal que el árbol T contiene al grafo $P_m^{3,m-2}$ con $m \in \{3k+6, 3k+7\}$ si y solamente si $T^{[\#3]}$ tiene una sola componente conexa.*

DEMOSTRACIÓN. La implicancia de izquierda a derecha se tiene por el Lema 1.19.

Ahora vamos a demostrar que si $T^{[\#3]}$ tiene una sola componente conexa entonces T contiene la estructura de la figura 1.7. Para ello, demostraremos la contrarecíproca: Asumimos que T no contiene al grafo $P_m^{3,m-2}$ con $m \in \{3k+6, 3k+7\}$ y $k \in \mathbb{N}$. Para demostrar que $T^{[\#3]}$ tiene más de una componente conexa, lo haremos por inducción en la cantidad κ de vértices relevantes.

El caso base con $\kappa = 0$ implica que T no contiene la estructura de la Figura 1.4 y por lo tanto T es una doble escoba donde su cubo exacto ya sabemos que tiene 3 componentes conexas. Nuestra hipótesis inductiva es que todo árbol con $\kappa = l$ cumple que su cubo exacto tiene más de una componente conexa. Ahora para el paso inductivo consideramos un árbol T con $\kappa = l+1$. Escogemos un vértice $x \in V(T)$ cualquiera y lo dejamos como raíz. Aplicando el algoritmo BFS sobre $T[x]$ encontramos el vértice $l+1$ que es relevante, llamémoslo w . Podamos todos los nietos de w . El árbol T' que queda, tiene l vértices relevantes y por hipótesis inductiva su cubo exacto tiene más de una componente conexa. Notar que los nietos de w sólo están a distancia tres de los vecinos de w y al agregar los nietos al árbol ellos se agregarán a la componente de T' en la que están los vecinos de w . Pero por el Lema 1.21 w no está en dicha componente de T' , es decir no se unen las dos componentes conexas. ■

Al igual que antes, la Proposición 1.22 se puede reescribir de la siguiente manera:

- T contiene al menos dos vértices relevantes u y v que están a distancia que no es múltiplo de 3 si y solamente si $T^{[\#3]}$ tiene una sola componente conexa.

Ahora que se han introducido varios resultados y definiciones, podemos analizar el caso de las dos componentes conexas:

Por el Lema 1.18 y por la Proposición 1.22 se tiene que los árboles T cuyo cubo exacto tienen dos componentes conexas cumplen que T debe contener la estructura P_5^+ pero no contienen la estructura $P_m^{3,m-2}$ con $m \in \{3k+6, 3k+7\}$ y $k \in \mathbb{N}$.

Dado que T contiene a la estructura P_5^+ , entonces contiene al menos un vértice relevante. Si enraizamos el árbol T en uno de dichos vértices, el cual llamaremos r , se pueden definir los niveles asociados a la raíz como:

- $N_0 := \{r\}$

- Dado un entero positivo l se define $N_l := \{v \in V(T) \mid d(r, v) = l\}$

Por el Lema 1.20 se tiene que una de las componentes es la que contiene a todos los vértices relevantes. La pregunta que surge es ¿cuál es la estructura de dicha componente? en particular, ¿tiene ciclos?. La respuesta a esta última pregunta es ‘no’ y la veremos a continuación:

De ahora en adelante si T es un árbol cuyo cubo exacto tiene dos componentes conexas, entonces a la componente que contiene a los vértices relevantes la llamaremos C_R y a la otra componente la llamaremos C_{NR} .

Lema 1.23 *Si T es un árbol tal que $T^{[\#3]}$ tiene dos componentes conexas, entonces la componente C_R , que contiene a los vértices relevantes, no contiene ciclos.*

DEMOSTRACIÓN. Como T contiene a la estructura P_5^+ entonces T tiene al menos un vértice relevante. Tomemos un vértice relevante x de T y enraizamos el árbol en dicho vértice. Usaremos los niveles asociados a dicha raíz, es decir los niveles $N_l := \{v \in V(T) \mid d_T(x, v) = l\}$.

Primero vamos a probar que $V(C_R) = \cup_{r \in \mathbb{N}} N_{3l}$. La inclusión $\cup_{l \in \mathbb{N}} N_{3l} \subseteq V(C_R)$ es directa ya que x es un vértice que está en C_R y todo vértice a distancia múltiplo de 3 de x estará también en C_R . Para la otra inclusión si por contradicción asumimos que existe un vértice w en C_R que pertenece a un nivel de la forma N_{3l+1} o N_{3l+2} se tiene que w estaría a distancia múltiplo de 3 de algún vecino de x , por lo tanto x estaría en la misma componente que alguno de sus vecinos y por el Lema 1.21 se tendría que T contiene una sola componente conexa lo que es una contradicción.

Ahora que sabemos que todos los vértices de C_R están en niveles de la forma N_{3l} veamos que no tiene ciclos:

Por contradicción, vamos a asumir que C_R contiene al menos un ciclo. De ese ciclo vamos a tomar el vértice w que pertenezca al nivel más lejano, es decir $w \in N_{3\bar{l}}$ y si $w' \in N_{3l}$ está en el ciclo entonces $\bar{l} \geq l$. Ahora sean w_1 y w_2 los vecinos en el ciclo de w , eso implica que $d_T(w_1, w) = 3$ y $d_T(w_2, w) = 3$.

El primer caso es que w_1 o w_2 está en el nivel $N_{3\bar{l}}$. Supongamos que $w_1 \in N_{3\bar{l}}$, eso implica que existe un camino $wu_1u_2w_1$ en T , pero hay que notar que u_1 y u_2 están en niveles menores a $3\bar{l}$ ya que si alguno estuviera en un nivel mayor o igual, se generaría un ciclo en T . Pero eso implica que u_1 y u_2 tienen que ser vértices de $N_{3\bar{l}-1}$ que están conectados por una arista, lo que también generaría un ciclo en T .

El segundo caso es que w_1 y w_2 están en el nivel $N_{3\bar{l}-3}$, es decir existen u_1, u_2, u_3 y u_4 tal que $wu_1u_2w_1$ y $wu_3u_4w_2$ son caminos en T . Notar que u_1 y u_3 están en el nivel $N_{3\bar{l}-1}$ mientras que u_2 y u_4 están en el nivel $N_{3\bar{l}-2}$ (si no estuvieran en dichos niveles no sería posible la existencia de dichos caminos). Notar que los vértices u_1, u_2, u_3 y u_4 tienen que ser todos diferentes entre sí, porque si no se generaría un ciclo, por ejemplo si $u_2 = u_4$ significa que $wu_1u_2w_2TxTw_1$ formaría un ciclo. Pero aunque sean todos distintos también se genera el ciclo $wu_1u_2w_1TxTw_2u_4u_3w$. ■

Por lo tanto sabemos que una de las componentes conexas de $T^{[\#3]}$ es un árbol.

Lema 1.24 *Sea T un árbol cuyo cuadrado exacto tiene dos componentes conexas. Y dado un vértice relevante $r \in T$, para $l \in \mathbb{N}$ consideraremos N_l los niveles asociados a la raíz r . Entonces se cumple que:*

1. C_R está compuesto solamente por los vértices de los niveles N_{3k} .
2. Si $v \in N_{3k+1} \cup N_{3k+2}$ y $d_T(v) \geq 3$ entonces v no tiene nietos.

DEMOSTRACIÓN.

Para ver 1, es fácil notar que todo vértice de los niveles N_{3k} están a distancia múltiplo de 3 de la raíz r en T y por ende existe un camino en $T^{[\#3]}$ que une el vértice con r . Y por el Lema 1.21 se tiene que no puede haber vértices de otros niveles, porque si lo hubiera entonces habría solo una componente conexa. Para 2, si v tuviera nietos entonces sería un vértice relevante a una distancia no múltiplo de tres, es decir existe un entero c tal que T contendría a $P_m^{3,m-2}$ con $m \in \{3c + 6, 3c + 7\}$, y por la Proposición 1.22 tendría una sola componente conexa. ■

Lema 1.25 *Si $v \in V(C_R)$ es un vértice relevante en C_R , entonces es relevante en T .*

DEMOSTRACIÓN. Como v es relevante en C_R , entonces C_R contiene a la estructura P_5^+ . Basta con demostrarlo para el caso $C_R = P_5^+$. Enumeramos los vértices de C_R como en la Figura 1.4 y enraizamos C_R en el vértice relevante v_3 . Los vértices v_2, v_4 y v_6 no puede compartir ni padre ni abuelo en T por el Lema 1.24 y los vértices v_1 y v_5 tienen que ser bisnietos de v_2 y v_4 respectivamente. Por lo tanto se tiene que v_3 también es relevante en T . ■

A continuación veremos una definición que será importante para el resto de la sección. Vamos a introducir la noción de twin graph de un grafo:

Sea G un grafo, se dice que $u, v \in V(G)$ son *twins* si comparten la misma vecindad en G es decir $N_G(u) = N_G(v)$. Esta noción nos permite definir una relación de equivalencia $u \equiv v \iff u$ y v son twins. El *twin graph* de G , denotado por G^{tw} , tiene como vértices las clases de equivalencia de \equiv y hay una arista entre dos vértices $[u]_{\equiv} [v]_{\equiv}$ si y solamente si uv es una arista en G . Haciendo un abuso de notación cuando hablemos de un vértice $[u]_{\equiv} \in V(G^{tw})$ con $u \in V(G)$, simplemente diremos que $u \in V(G^{tw})$ entendiéndose que estamos hablando de la clase de equivalencia de u .

Volviendo a nuestro problema, consideremos C_R un árbol con al menos un vértice relevante. Enraizamos C_R en un vértice relevante $r \in V(C_R)$. Vamos a construir dos árboles canónicos T_1 y T_2 tal que sus cubos exactos tengan dos componentes conexas y una de ellas sea el árbol C_R .

Primer Árbol Canónico: Dado el árbol C_R definimos el árbol $T_1(C_R)$ formado al subdividir dos veces las aristas de C_R .

Segundo Árbol Canónico: El árbol $T_2(C_2)$ se obtiene al subdividir dos veces las aristas del twin graph de C_R (notar que los únicos twins en C_R pueden ser hojas unidas al mismo vértice). Luego para cada vértice que representa una clase de equivalencia con más de un vértice, digamos $[u] = \{u_1, \dots, u_h\}$, quitamos $[u]$ del árbol y agregamos los vértices u_1, \dots, u_h como hojas hermanas conectadas al mismo vértice que estaba conectado $[u]$ (ya que este era hoja).

A continuación se puede ver un ejemplo para entender mejor dichas construcciones:

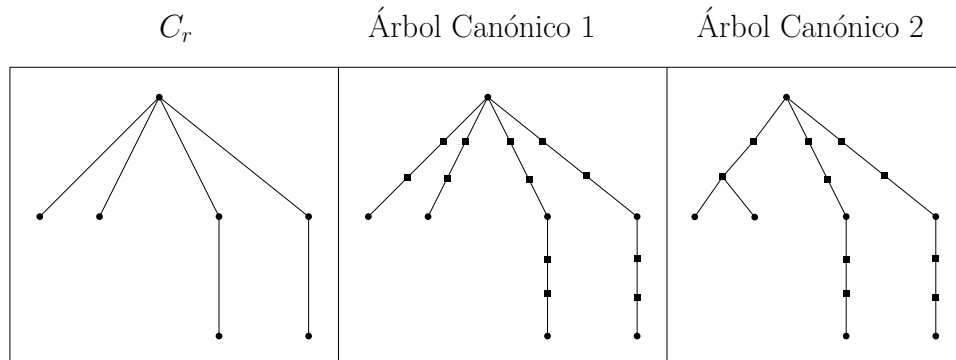


Figura 1.8: Árboles Canónicos de C_R .

Lema 1.26 *Todo árbol C_R con al menos un vértice relevante es una de las dos componente conexas del cubo exacto de algún árbol.*

DEMOSTRACIÓN. Basta con usar cualquiera de los dos árboles canónicos definidos anteriormente. ■

Para la caracterización que estamos buscando, vamos a tener que analizar la otra componente, es decir C_{NR} . Recordemos que nuestro objetivo es encontrar una caracterización, pero también encontrar un algoritmo que en tiempo polinomial pueda reconocer si un grafo es el cubo exacto de algún árbol. Para ello veamos algunos resultados que serán útiles.

Una observación importante, cuya demostración es directa, es la siguiente: Si G es un grafo bipartito entonces $G^{[\#3]}$ también lo es y más aún la misma bipartición de G es bipartición de $G^{[\#3]}$. No necesariamente es la única bipartición, lo que se tiene es que una bipartición de G también lo es de $G^{[\#3]}$.

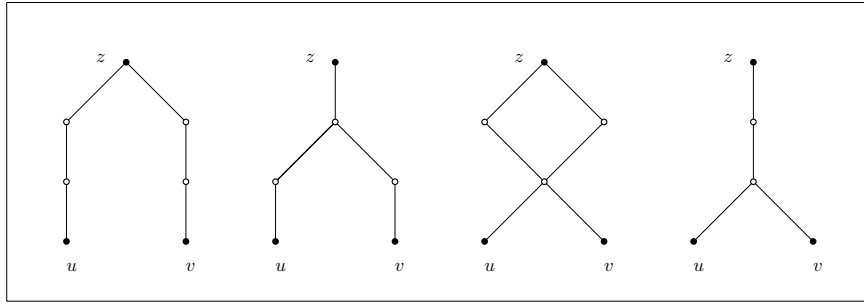
Notemos que los vértices que son twins en la componente de $T^{[\#3]}$ que no contiene a los relevantes tienen una propiedad interesante:

Lema 1.27 *Sea $i \in \mathbb{N}$ y $u, v \in N_{3i+1} \cup N_{3i+2}$. Entonces se tiene que $N_{T^{[\#3]}}(u) = N_{T^{[\#3]}}(v)$ si y solamente si existe un vértice w tal que u y v son hojas conectadas a w en T . Más aún en ambos casos se tiene que u y v están en el mismo nivel.*

DEMOSTRACIÓN. Sea $i \in \mathbb{N}$ y $u, v \in N_{3i+1} \cup N_{3i+2}$. En el caso en que $\exists w$ tal que u y v son hojas conectadas a w en T , entonces es directo ver que las vecindades de u y v en $T^{[\#3]}$ son las mismas.

Ahora para ver la otra implicancia supongamos que las vecindades de u y v en $T^{[\#3]}$ son las mismas. Primero hay que notar que u y v pertenecen al mismo nivel, es decir $u, v \in N_{3i+1}$ o $u, v \in N_{3i+2}$. Esto se tiene ya que el cubo exacto de T es bipartito y como u, v tienen la misma vecindad en el cubo, tienen que estar en la misma parte de la bipartición de $T^{[\#3]}$, pero dicha bipartición también es bipartición de T , por lo tanto están en la misma parte de la bipartición de T . Luego hay que notar que los niveles N_{3i+1} y N_{3i+2} pertenecen a partes diferentes de dicha bipartición de T , lo que en este caso implica que u y v están en el mismo nivel. De ahora en adelante m será el nivel al cuál pertenecen, es decir $u, v \in N_m$.

Como $N_{T^{[\#3]}}(u) = N_{T^{[\#3]}}(v)$ supongamos que existe un vértice $z \in N_{T^{[\#3]}}(u)$ tal que $z \in N_j$ con $j < m$. Es fácil notar que en este caso se tiene que $j \in \{m-1, m-3\}$. A continuación se muestran las posibles 4 configuraciones en T para los vértices u, v y z



Ahora veamos que la única opción factible es la cuarta.

La primera configuración puede implicar dos cosas:

- Si $z \in N_{m-3}$ entonces es un vértice con grado mayor o igual que tres en T , que está en un nivel que no es múltiplo de 3 y además tiene nietos, por lo que se contradice con el Lema 1.24.
- Si $z \in N_{m-1}$ se tendría un ciclo en T que es árbol, ya que los dos vértices blancos que están conectados a z en la figura, necesariamente son vértices en el nivel N_{m-2} , pero dichos vértices tienen un camino entre ellos que pasa por la raíz del árbol y que no usa el vértice z generando así el ciclo.

La segunda configuración no se puede tener ya que el padre de u en T sería vecino de v en $T^{[\#3]}$ pero no de u , por lo tanto sus vecindades no serían iguales, pero por hipótesis tienen que ser iguales. La tercera configuración genera un ciclo en T . Por lo tanto la única opción factible es la opción 4.

Ahora veamos que u y v son hojas. Dado que la única configuración posible era la 4, significa que u y v tienen el mismo padre en T . Si alguno de los vértices no fuera hoja, por ejemplo u , entonces los hijos de u en T serían vecinos de v en $T^{[\#3]}$ pero no de u , es decir no tendrían la misma vecindad contradiciendo la hipótesis.

En el caso en que no existe un vértice $z \in N_{T^{[\#3]}}(u)$ tal que $z \in N_j$ con $j < m$. Significa que para todo $z \in N_{T^{[\#3]}}(u)$ se tiene que $z \in N_j$ con $j > m$, pero eso implica que $m = 1$ y es fácil ver que u y v son hojas del mismo padre en T . ■

Ahora que se sabe que dos vértices del mismo nivel comparten vecindad en $T^{[\#3]}$ únicamente si son hojas con el mismo padre, surge la pregunta de si pueden haber casos en que dos vértices de niveles diferentes compartan la misma vecindad en $T^{[\#3]}$ en la componente C_{NR} . Esto solo ocurre en un caso particular de árbol, que son los árboles que pertenecen a la familia *Excepcion*, que se definirá a continuación.

Antes de definir la familia necesitamos la siguiente definición: para dos enteros positivos k, l con $l \leq k$ definiremos la estructura P_k^l que se forma al tomar el camino en $P_k = v_1 \dots v_k$ y se le agrega un vértice w que se conecta al vértice v_l de P_k .

La familia *Excepcion* está conformada por todos los árboles T tal que su twin graph, T^{tw} , pertenece al conjunto $\{P_8^6, P_9^7, P_{10}^8, P_9^6, P_{10}^7, P_{11}^8\}$ y además los vértices que no son hojas en el twin graph representan un solo vértice en T , es decir cada uno de los vértices no hojas del twin graph es una clase de equivalencia con un solo elemento. En otras palabras los vértices que son hojas en el twin graph son los únicos vértices que pueden representar más de un vértice en el grafo original.

En la Figura 1.9, se muestra los grafos de la familia *Excepcion*, donde los vértices rojos representan que en dicha posición hay una o más hojas conectadas al vértice negro. Como pueden haber tantas hojas como uno quiera, esta familia es infinita, pero en la figura se muestra sus posibles formas y la forma de sus cubos exactos.

Notar que los vértices cuadrados de la Figura 1.9 representan vértices de niveles diferentes en T que comparten la misma vecindad en $T^{[\#3]}$. Este tipo de árboles son los únicos árboles en donde el cubo exacto contiene vértices con la misma vecindad pero dichos vértices pertenecen a niveles diferentes, por ello se le llama familia excepción.

Esencialmente, los grafos en esta familia son lo mas parecido a una doble escoba que puede ser un grafo con vértice relevante.

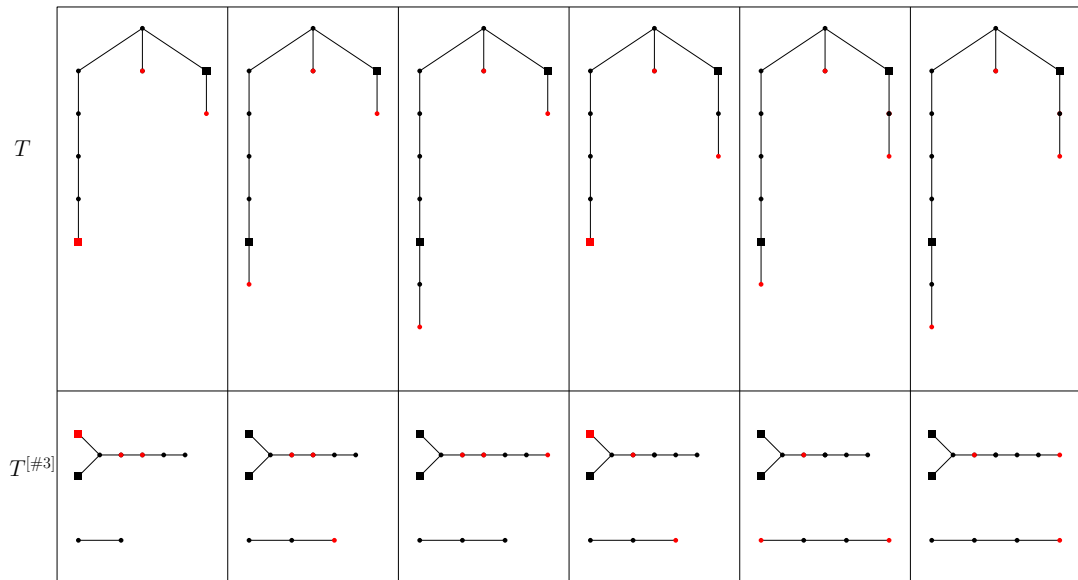


Figura 1.9: Tipo de árboles T en la Familia Excepción y sus cubos exactos.

Armar un algoritmo de reconocimiento en tiempo polinomial para determinar si un grafo

es el cubo de un árbol $T \in Excepcion$ es sencillo ya que solo hay que verificar que un grafo G de dos componentes conexas cumple que su twin graph consiste en dos árboles como los de los cubos exactos en la Figura 1.9.

Lema 1.28 Sean u, v vértices en niveles no múltiplos de tres. Si $N_{T[\#3]}(u) = N_{T[\#3]}(v)$ entonces se cumple alguna de las dos condiciones:

1. u y v son hojas en T con el mismo padre.
2. T es de la familia *Excepcion* y más aún, uno de los vértices está en el nivel N_1 y el otro en el nivel N_5 .

DEMOSTRACIÓN. El caso en que u y v estén en el mismo nivel se tiene por el Lema 1.27 que se cumple la primera condición.

Ahora supongamos que u y v no están en el mismo nivel, sin pérdida de generalidad diremos que u está en un nivel menor que v . Es fácil ver que si T pertenece a la familia *Excepcion* entonces los vértices u y v se encuentran en el nivel N_1 y en N_5 respectivamente y son los vértices cuadrados de la Figura 1.9.

Veamos que no existe otro caso. Supongamos que T no pertenece a la familia *Excepcion* y que u y v están en niveles diferentes, digamos $l(u)$ y $l(v)$ respectivamente, con $l(u) < l(v)$. Considerando que r es la raíz de T hablaremos de las ramas de T como los árboles $T[r, s]$ donde s es el hijo de r (recordar notación de la sección pasada), es decir el subárbol formado por s y su descendencia. Primero hay que notar que u y v no pueden estar en la misma rama, es decir el u - v camino en T debe pasar por r , ya que si están en la misma rama, el vértice u que está en el nivel $l(u)$ está a distancia 3 de un vértice en el nivel $l(u) + 3$ de la misma rama, pero u y v tienen la misma vecindad en $T^{[\#3]}$ por lo tanto al estar en la misma rama que u se tiene que $l(v) = l(u) + 6$, pero u está a distancia 3 de un vértice w el cual está a una distancia mayor de v (w puede ser el bisabuelo de u en el caso en que $l(u) \geq 4$ o un vértice de otra rama en el caso en $l(u) < 4$ que es hijo o nieto de r). Y el caso en que u y v estén en ramas diferentes tampoco puede ocurrir ya que como $T \notin Excepcion$, entonces también podemos encontrar un vértice que está a distancia tres de uno de ellos pero a una distancia mayor del otro. ■

Lema 1.29 Si T es un árbol tal que su cubo exacto tiene dos componentes conexas, $T^{[\#3]} = C_R + C_{NR}$, entonces el cubo exacto del twin graph de T también tiene dos componentes conexas, que corresponden al twin graph de C_R y C_{NR} , respectivamente. Y más aún se cumple que $|C_R^{tw}| < |C_{NR}^{tw}|$.

DEMOSTRACIÓN. Es fácil notar que $(T^{tw})^{[\#3]}$ tiene dos componentes conexas y son el twin graph de C_R y C_{NR} , respectivamente. Ahora si enraizamos T^{tw} en un vértice relevante, entonces sin contar la raíz, cada vértice de T^{tw} que se encuentre en un nivel múltiplo de tres, N_{3i} , tiene un padre que es único para dicho vértice y por ser twin graph y por el Lema 1.24 no existe otro vértice con el mismo padre. Luego la raíz al ser relevante tiene al menos tres hijos en el nivel N_1 , por lo tanto $|C_R^{tw}| < |C_{NR}^{tw}|$. ■

Ya sabemos que una de las componentes de $T^{[\#3]}$ es un árbol. Para describir la otra componente necesitamos la definición de Descomposición Bipartita Especial (DBE) de un grafo, pero para ello vamos a motivar la definición con un ejemplo. Supongamos que tenemos un árbol T como el de la siguiente figura (enraizado en un vértice relevante) cuyo cubo exacto tiene dos componentes conexas y este árbol no es de la familia *Exception*. Por el Lema 1.28 podemos asumir que este árbol no contiene hojas que compartan padres, ya que dichas hojas tienen el mismo comportamiento en el cubo exacto (misma vecindad). Y vamos a identificar ciertas partes del árbol que tendrán relevancia en el cubo exacto.

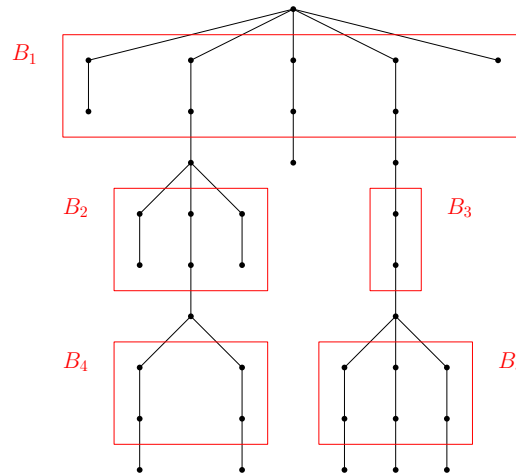


Figura 1.10: Árbol T con segmentos marcados.

Ahora veamos como se ve el cubo exacto de T en la imagen 1.11:

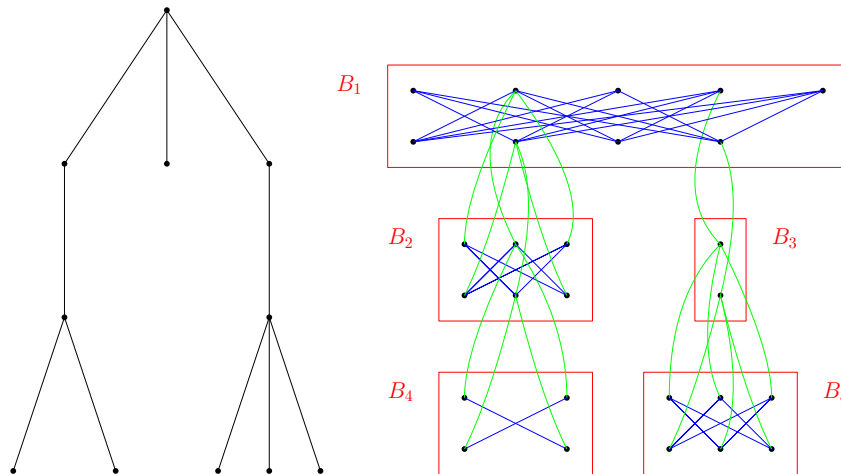


Figura 1.11: Cubo Exacto de T con segmentos marcados.

Si nos fijamos en $T^{[\#3]}$ la componente que no contiene a los vértices relevantes de T cumple que B_1, \dots, B_5 consisten en bipartitos completos menos un matching maximal. Las aristas azules representan aristas dentro del subgrafo B_i . Luego hay una conexión entre cada uno de estos subgrafos que corresponde a las aristas verdes. Estas conexiones cumplen ciertas reglas. Es así como vamos a definir el siguiente concepto:

Descomposición Bipartita Especial

Sea B un grafo bipartito. Diremos que B admite una Descomposición Bipartita Especial (DBE) de largo m si:

Existe una secuencia ordenada de m bipartitos $(B_i)_{i=1}^m$ vértice disjuntos y no vacíos, de la forma $V(B_i) = P_i \cup Q_i$ y $E(B_i) = E(P_i, Q_i)$, tal que $V(B) = \cup_{i=1}^m V(B_i)$ y cumple las siguientes propiedades:

1. Para todo $i \in [m]$ se tiene que $|P_i| \geq |Q_i|$.
2. Para todo $i \in [m]$ se tiene que si $Q_i = \emptyset$ entonces $|P_i| = 1$ y no existe $j > i$ tal que $xy \in E(B)$ con $x \in V(B_i)$ y $y \in V(B_j)$.
3. Para todo $i \in [m]$ si $Q_i \neq \emptyset$ entonces B_i es un grafo que cumple que existe un matching maximal, M_i , en el complemento de B_i tal que B_i es un bipartito completo menos M_i , es decir, $B_i = K_{P_i, Q_i} \setminus M_i$.
4. Para todo $i \in [m]$ con $Q_i \neq \emptyset$ y para toda arista $xy \in M_i$ con $x \in P_i$, $y \in Q_i$ se cumple alguna de las siguientes dos condiciones:
 - (a) Para todo $j > i$ $E(\{x, y\}, B_j) = \emptyset$.
 - (b) Existe un único $j_{xy} > i$ tal que $E(\{x, y\}, B_{j_{xy}}) \neq \emptyset$ y las aristas de este conjunto son $x\hat{x} \in E(B)$ para todo $\hat{x} \in P_{j_{xy}}$ y $y\hat{y} \in E(B)$ para todo $\hat{y} \in Q_{j_{xy}}$.

Además para aristas distintas $x_1y_1, x_2y_2 \in M_i$ se tiene $j_{x_1y_1} \neq j_{x_2y_2}$

5. Para todo $i \geq 2$ existe un único $j < i$ tal que existe una arista $\hat{y} \in M_j$ tal que para todo $x \in P_i$ y para todo $y \in Q_i$ se tiene que $x\hat{x} \in E(B)$ y $y\hat{y} \in E(B)$.

A cada bipartito B_i lo llamaremos *segmento* de la descomposición, es decir hay m segmentos.

Ahora dada una DBE $(B_i)_{i=1}^m$ de un grafo B , vamos a definir el grafo \widehat{T} asociado a esta DBE. Los vértices de \widehat{T} son los segmentos B_i y entre dos vértices B_i y B_j hay una arista si y solamente si existe en B una arista entre algún vértice de B_i y algún vértice de B_j .

Lema 1.30 *Sea $(B_i)_{i=1}^m$ la DBE de B . Entonces el grafo \widehat{T} asociado a dicha descomposición es un árbol.*

DEMOSTRACIÓN. Como es una DBE la condición 5 nos asegura que cada segmento B_i con $i \geq 2$ está conectado a un segmento B_j con $j < i$. Esto implica que todo segmento tiene un camino hacia el segmento B_1 , por lo tanto \widehat{T} es conexo. Para ver que no tiene ciclos lo haremos por contradicción. Supongamos que tiene un ciclo, eso implica que debe existir un vértice B_i que está conectado a dos vértices B_j y B_k con $j < i$ y $k < i$ pero eso contradice la condición 5 de la existencia de un único segmento. Por lo tanto no tiene ciclos. En conclusión \widehat{T} es árbol. ■

Mostraremos que si nuestro árbol no pertenece a la familia *Excepcion*, entonces el twin graph de la componente que no contiene los relevantes tiene una DBE.

Una definición que será necesaria, para un grafo G y un vértice v vamos a definir el siguiente conjunto $N_G^2(v) := \{u \mid d_G(u, v) \in \{1, 2\}\}$, es decir los vértices que están a distancia uno o dos de v en G .

Lema 1.31 *El twin graph de C_{NR} , es decir C_{NR}^{tw} , tiene una DBE.*

DEMOSTRACIÓN. Se sabe que T tiene un vértice relevante, r , por lo que enraizamos T en dicho vértice y usaremos los niveles asociados. Por cada vértice v en C_R , denotaremos por $l(v) \in \mathbb{N}$ al nivel al cual pertenece v en T (Notar que $l(r) = 0$).

Usando el algoritmo *BFS* en T recorreremos el árbol y enumeraremos (ordenaremos) los vértices v que se encuentran en los niveles múltiplos de tres ($v \in C_R$) y además cumplan que tengan al menos un hijo. Al final tendremos una secuencia $(v_i)_{i=1}^m$ de vértices, donde $v_1 = r$. Vamos a ver que por cada uno de estos vértices v_i cumple que sus hijos y nietos forman un segmento de una DBE de C_{NR}^{tw} . En otras palabras para cada $i \in [m]$ definimos el segmento bipartito, B_i inducido en C_{NR}^{tw} por:

$$V(B_i) = P_i \cup Q_i := (N_T^2(v_i) \cap N_{l(v_i)+1} \cap C_{NR}^{tw}) \cup (N_T^2(v_i) \cap N_{l(v_i)+2} \cap C_{NR}^{tw})$$

Veamos que $(B_i)_{i=1}^m$ es una DBE:

1. Notar que por la definición de C_{NR}^{tw} se tiene que cada vértice en P_i tiene a lo más un hijo en C_{NR}^{tw} . Por lo tanto $|P_i| \geq |Q_i|$.
2. Es claro que si $Q_i = \emptyset$ significa que los elementos de P_i son hojas en T y por la construcción de C_{NR}^{tw} solo puede haber un vértice, es decir $|P_i| = 1$. Además dicho vértice no está conectado a ningún segmento B_j con $j > i$ ya que es una hoja en T .
3. Veamos ahora el matching. Notar que el matching esta dado por $M_i = \{xy \mid x \in P_i \text{ y se tiene que } y \text{ es hijo de } x \text{ en } T\}$. Es fácil notar que el segmento B_i es un bipartito completo menos el matching M_i .
4. Sea $i \in [m]$ y $Q_i \neq \emptyset$. Sea $xy \in M_i$ con $x \in P_i$ y $y \in Q_i$. Si y no tiene nietos en T , entonces notar que $\forall j > i$ se tiene $E(\{x, y\}, B_j) = \emptyset$. En el caso que y tenga nietos en T , por el Lema 1.24 se tiene que y tiene un único hijo en T , es decir que $\exists! v_j$ con $j > i$, y se tiene que $x\hat{x} \in E(B) \forall \hat{x} \in P_j$ y $y\hat{y} \in E(B) \forall \hat{y} \in Q_j$. Además notar que los elementos en Q_i no pueden tener el mismo hijo por lo tanto se cumple la condición 4 de la definición de DBE.
5. Notar que para todo $i \geq 2$ se tiene que el bisabuelo de v_i es algún v_j con $j < i$ y cumple que $\exists \hat{x}\hat{y} \in M_j$ $x\hat{x} \in E(B)$, $\forall x \in P_i$ y $y\hat{y} \in E(B) \forall y \in Q_i$. Además es el único $j < i$ que lo cumple.

■

Ahora considerando $(B_i)_{i=1}^m$ una DBE del twin graph de C_{NR} , para cada segmento B_i , con $V(B_i) = P_i \cup Q_i$ y $E(B_i) = E(P_i, Q_i)$, vamos a definir el conjunto de twins en Q_i ,

denotado por $\mathcal{T}_{C_{NR}}^Q(B_i)$, como el conjunto de vértices en Q_i tal que cada uno de esos vértices representan twins (más de una hoja) en C_{NR} , es decir $\mathcal{T}_{C_{NR}}^Q(B_i) = \{[u] \in V(B_i) : |[u]| \geq 2\}$. De manera análoga se define el conjunto de twins en P_i , denotado por $\mathcal{T}_{C_{NR}}^P(B_i)$, donde se ven los twins en P_i . Luego el conjunto de twins en B_i se define como la unión de los dos conjuntos anteriores $\mathcal{T}_{C_{NR}}(B_i) := \mathcal{T}_{C_{NR}}^P(B_i) \cup \mathcal{T}_{C_{NR}}^Q(B_i)$. Finalmente el conjunto de twins de la DBE, se define como la unión de los conjuntos de twins de todos los B_i , es decir $\mathcal{T}_{C_{NR}} = \cup_{i=1}^m \mathcal{T}_{C_{NR}}(B_i)$.

Estos conjuntos son importantes porque dichos vértices representan hojas en los árboles.

Lema 1.32 *Sea $(B_i)_{i=1}^m$ la DBE de C_{NR}^{tw} dada en la demostración del Lema 1.31 y sea \widehat{T} el árbol asociado a dicha DBE. Entonces existe un isomorfismo $\varphi : V(\widehat{T}) \rightarrow V(S) \subseteq V(C_R)$ tal que:*

1. *Si $x \in VC(C_R)$ entonces $x \in \varphi(\widehat{T})$. En otras palabras solamente algunas hojas en C_R no tienen preimagen.*
2. *$\forall i \in [m], |Q_i| \geq |N_{C_R}(\varphi(B_i)) \cap \varphi(\widehat{T})| + \mathbf{1}_{N_{C_R}(\varphi(B_i)) \setminus \varphi(\widehat{T}) \neq \emptyset} + |\mathcal{T}_{C_{NR}}^Q(B_i)| - \mathbf{1}_{i \geq 2}$.*

DEMOSTRACIÓN. Sea $(v_i)_{i=1}^m$ la secuencia de vértices de C_R de la demostración del Lema anterior. Vamos a enraizar en $v_1 = r$. Sea $(B_i)_{i=1}^m$ la DBE del Lema anterior y \widehat{T} el árbol asociado a dicha DBE. El isomorfismo evidente, $\varphi : V(\widehat{T}) \rightarrow V(S) \subseteq V(C_R)$ es el que asocia un vértice $B_i \in V(\widehat{T})$ al vértice v_i , es decir $\varphi(B_i) = v_i$.

Es fácil notar que es un isomorfismo ya que B_i y B_j representan los hijos y nietos de v_i y v_j en T respectivamente, por lo tanto $B_i B_j \in E(\widehat{T}) \iff$ existe un vértice $x \in B_i$ y un vértice $y \in B_j$ tal que $xy \in C_{NR}$. Y esto último es equivalente a que v_i y v_j están a distancia tres en T es decir $v_i v_j \in E(C_R)$.

Ahora veamos que cumplen las dos propiedades:

Veamos la primera propiedad. Por la elección de los vértices v_i se tiene que si $x \notin \varphi(\widehat{T})$ entonces $x \notin \{v_1, \dots, v_m\}$, por lo tanto x no tiene hijos en T y por lo tanto solo está a distancia tres de su bisabuelo, es decir x es hoja en C_R .

Ahora veamos la segunda propiedad. Notar que por cada hijo de $\varphi(B_i) = v_i$ en C_R se tiene un vértice en Q_i y además en Q_i están los vértices que representan twins en C_{NR}^T por lo tanto $|Q_i| \geq |N_{C_R}(v_i)| + |\mathcal{T}_{C_{NR}}^Q(B_i)| - \mathbf{1}_{i \geq 2} = |N_{C_R}(v_i) \cap \varphi(\widehat{T})| + |N_{C_R}(v_i) \setminus \varphi(\widehat{T})| + |\mathcal{T}_{C_{NR}}^Q(B_i)| - \mathbf{1}_{i \geq 2} \geq |N_{C_R}(v_i) \cap \varphi(\widehat{T})| + \mathbf{1}_{N_{C_R}(v_i) \setminus \varphi(\widehat{T}) \neq \emptyset} + |\mathcal{T}_{C_{NR}}^Q(B_i)| - \mathbf{1}_{i \geq 2}$ (el -1 cuando i es mayor o igual que 2 es porque el padre de v_i no cuenta). ■

Finalmente podemos presentar la caracterización.

Teorema 1.33 Sea $G = C_R + C_{NR}$ un grafo con dos componentes conexas, C_R un árbol y C_{NR} un grafo bipartito (que en caso de que también sea un árbol cumple que $|C_R^{tw}| \leq |C_{NR}^{tw}|$) tal que G no es el cubo de algún árbol de la familia Excepcion. Existe un árbol T tal que $T^{[\#3]} = G$ si y solamente si el twin graph de C_{NR} tiene una DBE $(B_i)_{i=1}^m$ de manera que el árbol, \widehat{T} , asociado a dicha DBE cumple que existe un isomorfismo $\varphi : V(\widehat{T}) \rightarrow V(S) \subseteq V(C_R)$ tal que:

1. Si $x \in VC(C_R)$ entonces $x \in \varphi(\widehat{T})$. En otras palabras solamente algunas hojas en C_R no tienen preimagen.
2. $\forall i \in [m], |Q_i| \geq |N_{C_R}(\varphi(B_i)) \cap \varphi(\widehat{T})| + \mathbf{1}_{N_{C_R}(\varphi(B_i)) \setminus \varphi(\widehat{T}) \neq \emptyset} + |\mathcal{T}_{C_{NR}}^Q(B_i)| - \mathbf{1}_{i \geq 2}$.

DEMOSTRACIÓN. Si existe un árbol T tal que $T^{[\#3]} = G$ la existencia de una DBE que cumpla lo pedido se tiene por los Lemas 1.31 y 1.32.

Ahora supongamos que tenemos una DBE $(B_i)_{i=1}^m$ de C_{NR}^{tw} que cumple que existe un isomorfismo que cumple las condiciones del teorema. Para ver que existe un árbol vamos a construir dicho árbol con el siguiente procedimiento:

Vamos a construir un árbol T cuyos vértices en principio serán $V(C_R) \cup V(C_{NR}^{tw})$. Veamos las aristas. Vamos a recorrer cada uno de los segmentos de la DBE en orden de manera que para cada segmento B_i primero agregamos las aristas entre $\varphi(B_i)$ y los vértices en P_i . Luego agregamos las aristas entre P_i y Q_i , las que están dadas por el matching M_i asociado. Luego escogemos un vértice x en Q_i que no represente un conjunto de twins, es decir que no se encuentre en $\mathcal{T}_{C_{NR}}^Q(B_i)$ (ya que dichos vértices no pueden tener hijos en el árbol pues serán hojas del mismo padre). Finalmente tomamos los vértices $\varphi(B_j)$ que eran hijos de $\varphi(B_i)$ (es decir $j > i$) en C_R , cada uno de esos vértices los conectamos con algún vértice Q_i que este libre y sin repetirse entre ellos. Por la condición 2 del isomorfismo se tiene que tenemos suficientes vértices en Q_i para realizar estas acciones.

Finalmente para todo $v \in V(T) \cap V(C_{NR}^{tw}) \cap \mathcal{T}_{C_{NR}}$ vamos a ‘desdoblar v ’ en un proceso que denotaremos por $Desdoblar(T, v)$. Notar que el vértice v representa una clase de equivalencia $v = \{v_1, \dots, v_l\}$. Por lo tanto vamos a eliminar v de V_T y vamos a agregar a V_T los vértices v_1, \dots, v_l como hojas conectadas al que era el padre de v en T (ya que v era hoja en T).

El detalle del algoritmo se muestra en el Procedimiento 2:

Procedimiento 2 Construcción árbol usando DBE.

Input: $C_R, C_{NR}, (B_i)_{i=1}^m$ y el isomorfismo φ

Output: T un árbol tal que $T^{\{\#3\}} = G$

$\hat{T} \leftarrow$ el árbol de la DBE $(B_i)_{i=1}^m$

$V_T \leftarrow V(C_R) \cup V(C_{NR}^{tw})$

$E_T \leftarrow \emptyset$

$T = (V_T, E_T)$

for $B_i \in (B_i)_{i=1}^m$ (respetando el orden) **do**

$E_T \leftarrow E_T \cup \{\varphi(B_i)p_i | p_i \in P_i\}$

$E_T \leftarrow E_T \cup M_i$

$Usados = \mathcal{T}_{C_{NR}}^Q(B_i)$

Sea $x \in Q_i \setminus Usados$ arbitrario

$E_T \leftarrow E_T \cup \{xy | y \in N_{C_R}(\varphi(B_i)) \setminus \varphi(\hat{T})\}$

$Usados \leftarrow Usados \cup \{x\}$

for $\varphi(B_j) \in N_{C_R}(\varphi(B_i)) \cap \varphi(\hat{T})$ con $j > i$ **do**

Sea $z \in Q_i \setminus Usados$ arbitrario

$E_T \leftarrow E_T \cup z\varphi(B_j)$

$Usados \leftarrow Usados \cup \{z\}$

end for

end for

for $v \in \mathcal{T}_{C_{NR}}$ **do**

$Desdoblar(T, v)$

end for

return $T = (V_T, E_T)$

Veamos su correctitud. Es decir que efectivamente retorna un árbol cuyo cubo exacto es G . Primero veamos que T es conexo, para ello notar que en T todos los vértices que eran del segmento B_i están conectados entre si (por el matching M_i) en T y ese segmento esta conectado al vértice $\varphi(B_i)$. Luego gracias a la condición 2 del isomorfismo podemos asegurar que todo los hijos de $\varphi(B_i)$ en C_R (considerando $\varphi(B_1)$ como la raíz) se conectan a un vértice adecuado de Q_i del segmento B_i , y por la condición 1 se tiene que en este proceso todos los vértices terminan en la misma componente conexa y al ‘desdoblar’ los vértices sigue conexo. Para ver que T no contiene ciclos hay que darse cuenta de que los únicos tipos de aristas que agregamos son entre un vértice $\varphi(B_i)$ con un vértice en P_i , luego entre los vértices de P_i con los de Q_i (con un matching por lo tanto no hay ciclos), luego entre vértices de Q_i con vértices $\varphi(B_j) \in N_{C_R}(\varphi(B_i)) \cap \varphi(\hat{T})$ con $j > i$ y vértices en $N_{C_R}(\varphi(B_i)) \setminus \varphi(\hat{T})$ (estos últimos son hojas) y así seguimos el mismo procedimiento con los siguientes segmentos B_k con $k > i$ (Recordar que los segmentos no comparten vértices entre ellos). Por lo tanto la única opción de que hubiera un ciclo es conectar dos vértices distintos de Q_i al mismo vértice pero esto no ocurre ya que un vértice Q_i recibe un único vértice en $N_{C_R}(\varphi(B_i)) \cap \varphi(\hat{T})$ o esta conectado a puras hojas que eran de $N_{C_R}(\varphi(B_i)) \setminus \varphi(\hat{T})$.

Por la construcción se tiene que el cubo de este árbol es isomorfo a G . ■

En las figuras 1.12, 1.13 y 1.14 se observa una ejemplificación del Procedimiento 2.

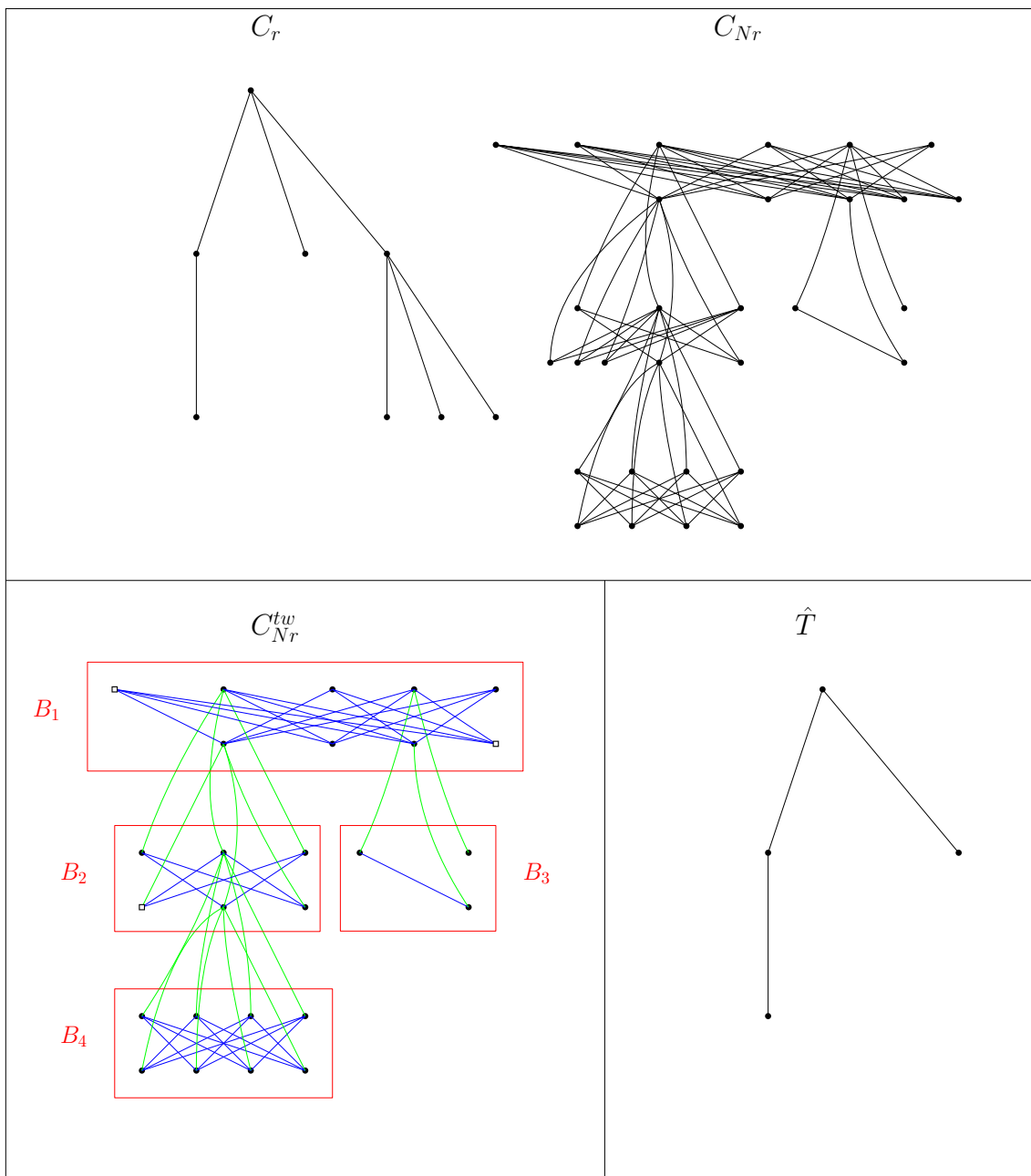


Figura 1.12: C_R , C_{NR} , C_{NR}^{tw} y \hat{T} .

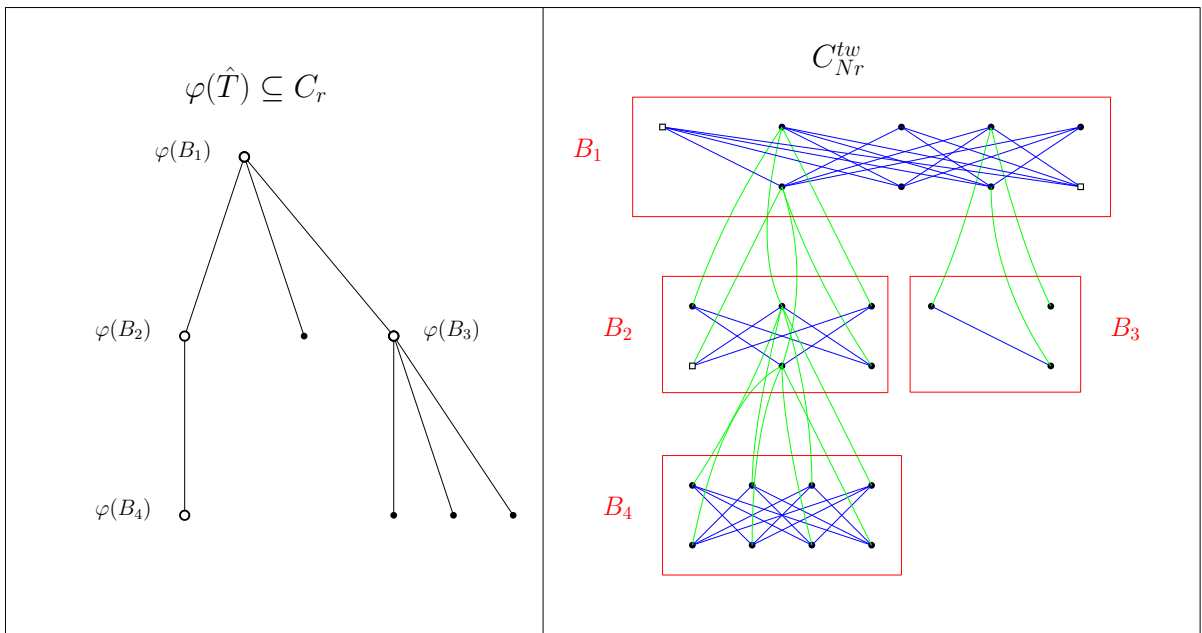


Figura 1.13: Isomorfismo φ .

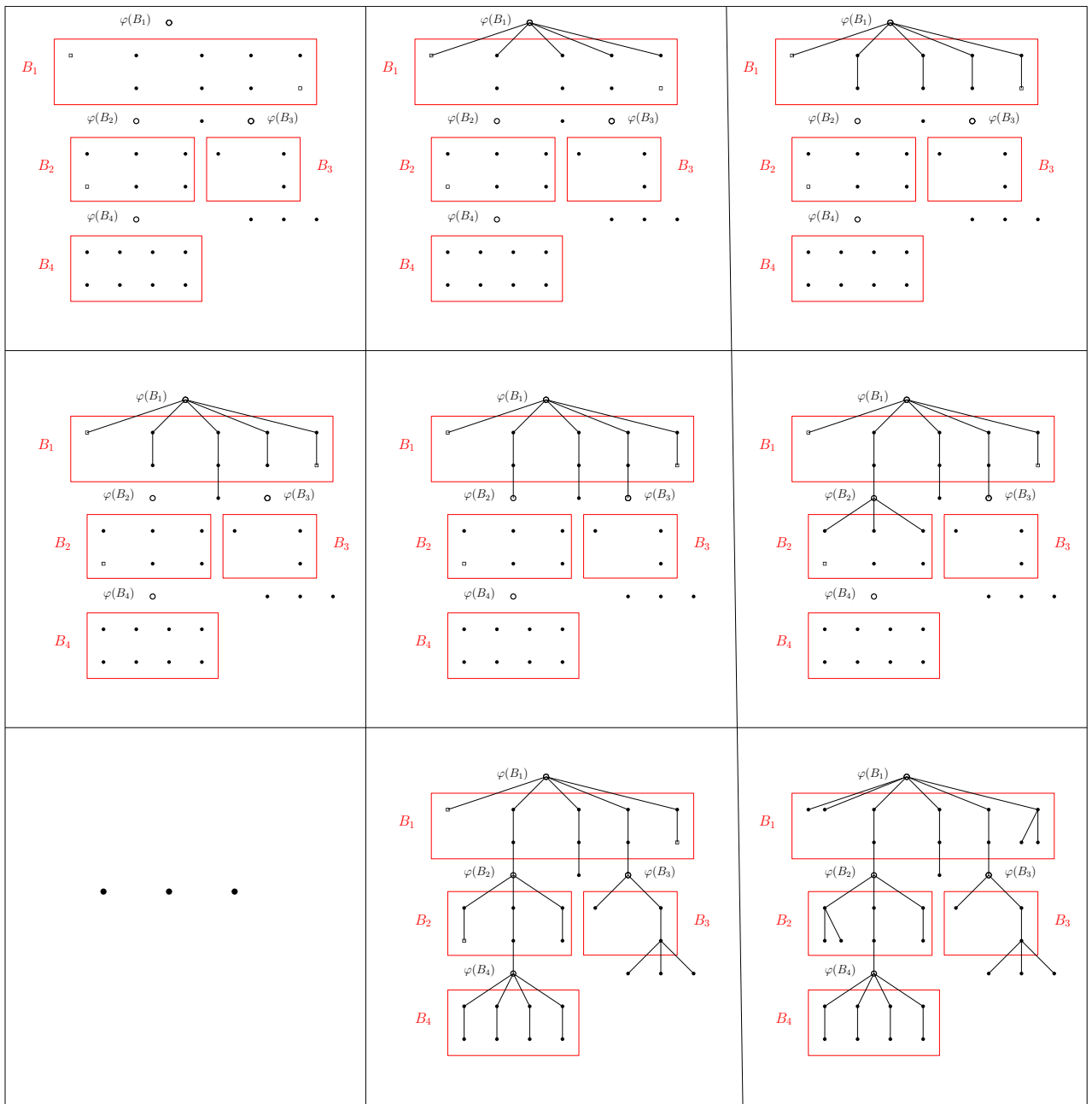


Figura 1.14: Procedimiento 2.

Ahora que tenemos la caracterización, surge la problemática de cómo poder identificar una posible *DBE* que nos permita construir un árbol.

A continuación veamos algunos lemas que nos motivarán a crear un algoritmo que resuelva esta problemática.

El siguiente lema nos caracteriza la estructura que tienen los segmentos de una *DBE*:

Lema 1.34 *Sea $\mathcal{B} = (B_i)_{i=1}^m$ una *DBE* de C_{NR}^{tw} entonces para todo $i \geq 2$ se tiene que para cualquier arista xy del matching M_i asociado a B_i cumple que $C_{NR}^{tw}[N_{C_{NR}^{tw}}(x) \cup N_{C_{NR}^{tw}}(y)] \setminus \cup_{k=1}^{i-1} \mathcal{B}(k)$ es un segmento de \mathcal{B} o es vacío.*

DEMOSTRACIÓN. Directo de la condición 4 de la definición de *DBE* ■

A continuación veremos los siguientes dos lemas que hablan sobre el segmento B_1 de una *DBE*:

Lema 1.35 *Sea $\mathcal{B} = (B_i)_{i=1}^m$ una *DBE* de C_{NR}^{tw} si B_1 tiene al menos 5 vértices, entonces es un grafo maximal que cumple que es un bipartito completo menos un matching maximal.*

DEMOSTRACIÓN. Si no fuera maximal se podría agregar un vértice w tal que B_1 seguiría siendo un bipartito completo menos matching, pero ese vértice pertenece a otro segmento, digamos B_2 . Pero eso implica que w tendría que estar conectado al menos a dos vértices de P_1 o de Q_1 para mantener la condición de ser bipartito completo menos matching maximal. Esto contradice las condiciones para ser *DBE* (condición 4 y 5) ya que todos los vértices de B_2 están conectados a un único par de vértices $x \in P_1$ e $y \in Q_1$ con $xy \in M_1$, es decir w está conectando a lo más a un vértice de P_1 o de Q_1 . ■

Lema 1.36 *Sea $G = C_R + C_{NR}$ un grafo con dos componentes conexas, C_R un árbol y C_{NR} un grafo bipartito (que en caso de que también sea un árbol cumple que $|C_R^{tw}| \leq |C_{NR}^{tw}|$) tal que G no es el cubo de alguien de la familia *Excepcion*. Si existe un árbol T tal que $T^{[3]} = G$ entonces existe una *DBE* de C_{NR}^{tw} con $|B_1| \geq 5$.*

DEMOSTRACIÓN. Consecuencia directa de la demostración del Lema 1.31. ■

Motivados por los resultados anteriores vamos a definir el siguiente procedimiento:

Sea C_{NR} un grafo bipartito, vamos a trabajar con su twin graph C_{NR}^{tw} . La idea es ir armando una *DBE*. Lo primero que se hace es buscar un grafo maximal que sea bipartito completo menos matching con al menos 5 vértices (una consecuencia de la construcción de la demostración del Lema 1.31 nos asegura que se puede armar una *DBE* con B_1 que cumple eso). El primer grafo que encontremos con estas características será B_1 . Esto lo haremos a través del Procedimiento 5, llamado *BuscarB1*, que se explicará más adelante. Luego vamos a identificar los segmentos del siguiente nivel usando el matching M_i asociado a B_1 . Por cada arista del matching M_1 , es decir $xy \in M_1$, vamos a buscar en $\{x, y\} \cap C_{NR}^{tw} \setminus B_1$ un grafo bipartito completo menos matching que contenga a los vértices x e y , y dicho grafo al

quitarle los vértices x e y representará un segmento de la DBE, por lo que lo enumeramos. Luego continuamos realizando este procedimiento con los nuevos segmentos encontrados hasta encontrar todos los segmentos de la DBE, asegurándonos que se cumplan las propiedades para ser DBE. Las condiciones 1 y 2 de la definición de DBE se chequearán con el Procedimiento 4 ,*IdentificarPQ*, que se explicará más abajo.

A continuación vamos a definir un algoritmo que permitirá encontrar un posible candidato para ser el primer segmento de una DBE. Definiremos el algoritmo *BuscarB1* el cual recibe a C_{NR}^{tw} el twin graph de un grafo bipartito y retorna un grafo maximal que cumple que es un bipartito completo menos un matching maximal con al menos 5 vértices.

Primero mencionaremos el esquema general (sin detalles) de un algoritmo que determina si un grafo bipartito B es un bipartito completo menos matching. Dicho algoritmo se llamará 'BCMM'.

Procedimiento 3 BCMM.

Input: B un grafo bipartito

Output: \forall si B es un grafo bipartito completo menos un matching, \mathbb{F} en caso contrario

$(P, Q) \leftarrow$ la bipartición de B (spg: $|P| \geq |Q|$)

if B no es conexo **then**

return \mathbb{F}

end if

if $|\{v \in P : d_B(v) = |Q| - 1\}| = |Q| \wedge |\{v \in P : d_B(v) = |Q|\}| = |P| - |Q|$ **then**

return \mathbb{V}

end if

return \mathbb{F}

A continuación vamos a definir el esquema (sin detalles) de una subrutina que recibe una posible DBE, \mathcal{B} , y retornará verdadero o falso dependiendo de si \mathcal{B} satisface las condiciones 1 y 2 de la definición de ser DBE. Esta subrutina se llamará *IdentificarPQ*. Esta subrutina irá revisando en orden cada uno de los segmentos hasta encontrar algún segmento B_i tal que su bipartición no es equilibrada, es decir una parte tiene más vértices. En caso de encontrar dicho B_i la parte con mayor vértices será P_i mientras que la otra parte será Q_i de la definición de DBE. A partir de B_i podemos propagar el reconocimiento de los P_j y Q_j para los otros segmentos respetando que entre un segmento B_{i_1} y un segmento B_{i_2} solo puede haber aristas entre P_{i_1} con P_{i_2} y entre Q_{i_1} con Q_{i_2} . Luego de haber reconocido todos los P_i y Q_i hay que asegurar que se cumpla la condición 1 y 2, para ello se revisa el cardinal de las partes de cada segmento y se verifica las condiciones.

Procedimiento 4 IdentificarPQ.

Input: $\mathcal{B} = (B_i)_{i=1}^m$ una secuencia de bipartitos vértice-disjuntos, cada uno de ellos un bipartito completo menos un matching maximal

Output: \forall si \mathcal{B} cumple las condiciones 1 y 2 de la definición de DBE, \mathbb{F} en caso contrario $indice \leftarrow \emptyset$

for $i \in [m]$ **do**

$(A_1, A_2) \leftarrow$ la bipartición de B_i (se puede encontrar con un BFS)

if $|A_1| > |A_2|$ **then**

$(P, Q) \leftarrow (A_1, A_2)$

$B_i = (A_1, A_2) =: (P_i, Q_i)$ (identificamos quien es P_i y Q_i en dicho segmento)

$indice \leftarrow i$

break

end if

if $|A_2| > |A_1|$ **then**

$(P, Q) \leftarrow (A_2, A_1)$

$B_i = (A_2, A_1) =: (P_i, Q_i)$ (identificamos quien es P_i y Q_i en dicho segmento)

$indice \leftarrow i$

break

end if

end for

if $indice = \emptyset$ **then**

return \forall (ya que todos los segmentos tienen el mismo tamaño)

end if

A partir de B_{indice} propagamos el reconocimiento de los (P_j, Q_j) para el resto de los B_j

for $i \in [m]$ **do**

if $|P_i| < |Q_i|$ **then**

return \mathbb{F}

end if

end for

for $i \in [m]$ **do**

if $|Q_i| = 0 \wedge [|P_i| > 1 \vee (\exists j > i \text{ tal que } E(P_i, B_j) \neq \emptyset)]$ **then**

return \mathbb{F}

end if

end for

return \forall

Ahora vamos a definir el esquema general del algoritmo *BucarB1*. La idea es buscar un matching de no aristas con exactamente 2 aristas, y luego revisar si los vértices involucrados cumplen con ser un bipartito completo menos un matching. En caso de serlo vemos si podemos agregar vértices hasta hacerlo un grafo maximal que cumple *BCMM*. Si el conjunto tiene al menos 5 vértices el algoritmo retorna dicho conjunto. En caso de que no, se sigue iterando en búsqueda de otro conjunto.

Procedimiento 5 *BuscarB1*.

Input: C_{NR}^{tw} el twin graph de un grafo bipartito (conexo)

Output: B_1 un grafo maximal tal que es un bipartito completo menos un matching maximal tal que tenga al menos 5 vértices, \emptyset en caso contrario

$(A_1, A_2) \leftarrow$ la bipartición de C_{NR}^{tw}

$M \leftarrow \{a_1a_2 | a_1 \in A_1 \wedge a_2 \in A_2 \wedge a_1a_2 \notin E(C_{NR}^{tw})\}$

$B_1 \leftarrow \emptyset$

for $a_1a_2, b_1b_2 \in M$ con a_1, a_2, b_1, b_2 todos distintos **do**

if $C_{NR}^{tw}[\{a_1, a_2, b_1, b_2\}] \cup \{a_1a_2, b_1b_2\}$ es un bipartito completo **then**

$B_1 \leftarrow C_{NR}^{tw}[\{a_1, a_2, b_1, b_2\}]$

 Agrego a B_1 todos los vértices (y aristas) tal que B_1 es un grafo maximal que es un bipartito completo menos un matching maximal

if $|B_1| \geq 5$ **then**

return B_1

end if

end if

end for

return \emptyset

Con los algoritmos anteriores podemos definir el Procedimiento 6 llamado Búsqueda DBE:

Procedimiento 6 Búsqueda DBE.

Input: C_{NR}^{tw} el twin graph de un grafo bipartito (conexo)

Output: $(B_i)_{i=1}^m$ una DBE de C_{NR}^{tw} con $|B_1| \geq 5$ si es que existe, \emptyset en caso contrario

```
 $\mathcal{B} \leftarrow ()$ 
 $i \leftarrow 1$ 
 $j \leftarrow 2$ 
 $B_1 \leftarrow \text{BuscarB1}(C_{NR}^{tw})$ 
 $\mathcal{B}(1) \leftarrow B_1$ 
while  $\mathcal{B}(i) \neq \emptyset$  do
   $M_i \leftarrow$  el matching de no aristas asociado a  $\mathcal{B}(i)$ 
  for  $xy \in M_i$  do
     $B_j \leftarrow C_{NR}^{tw}[N_{C_{NR}^{tw}}(x) \cup N_{C_{NR}^{tw}}(y)] \setminus \cup_{k=1}^{i-1} \mathcal{B}(k)$ 
    if  $(\text{BCMM}(B_j) = \mathbb{F} \wedge B_j \neq \emptyset) \vee E(B_j, \cup_{k=1}^{j-1} \mathcal{B}(k) \setminus \{x, y\}) \neq \emptyset \vee V(B_j) \cap V(\cup_{k=1}^{j-1} \mathcal{B}(k)) \neq \emptyset$  then
      return  $\emptyset$ 
    end if
    if  $B_j \neq \emptyset$  then
       $\mathcal{B}(j) \leftarrow B_j$ 
       $j \leftarrow j + 1$ 
    end if
  end for
   $i \leftarrow i + 1$ 
end while
if  $V(\mathcal{B}) \neq V(C_{NR}^{tw})$  then
  return  $\emptyset$ 
end if
if  $\text{IdentificarPQ}(\mathcal{B}) = \mathbb{F}$  then
  return  $\emptyset$ 
end if
return  $\mathcal{B}$ 
```

Lema 1.37 El procedimiento 6, 'Búsqueda DBE', cuando retorna una secuencia no vacía entonces dicha secuencia es una DBE de C_{NR}^{tw} .

DEMOSTRACIÓN. Sea $\mathcal{B} = (B_i)_{i=1}^m \neq \emptyset$ la secuencia retornada por el procedimiento 6. Veamos que es una DBE de C_{NR}^{tw} . Por la construcción de los segmentos B_i se tiene que son segmentos vértices disjuntos y no vacíos que cumplen $V(\mathcal{B}) = V(C_{NR}^{tw})$. Ahora revisemos que se satisfacen las 5 condiciones:

1. Se tiene porque $\text{IdentificarPQ}(\mathcal{B}) = \mathbb{V}$.
2. Se tiene porque $\text{IdentificarPQ}(\mathcal{B}) = \mathbb{V}$.
3. Se tiene porque el Procedimiento 5, *BuscarB1*, el cual retorna el segmento B_1 que es un bipartito completo menos matching maximal, luego para el resto de los $j \geq 2$ el procedimiento 3, BCMM, asegura que cada uno de los segmentos sea un bipartito completo menos matching maximal.

4. Sea $i \in [m]$ con $Q_i \neq \emptyset$, sea $xy \in M_i$ con $x \in P_i$ y $y \in Q_i$. Si $C_{NR}^{tw}[N_{C_{NR}^{tw}}(x) \cup N_{C_{NR}^{tw}}(y)] \setminus \cup_{k=1}^{i-1} \mathcal{B}(k) = \emptyset$ entonces se tiene que $E(\{x, y\}, B_j) = \emptyset$ para todo $j > i$ ya que el posible B_j asociado a x, y se definen como el subgrafo $C_{NR}^{tw}[N_{C_{NR}^{tw}}(x) \cup N_{C_{NR}^{tw}}(y)] \setminus \cup_{k=1}^{i-1} \mathcal{B}(k)$. En el caso que dicho subgrafo sea distinto de vacío al pasar por el la parte del if $(BCMM(B_j) = \mathbb{F} \wedge B_j \neq \emptyset) \vee E(B_j, \cup_{k=1}^{j-1} \mathcal{B}(k) \setminus \{x, y\}) \neq \emptyset \vee V(B_j) \cap V(\cup_{k=1}^{j-1} \mathcal{B}(k)) \neq \emptyset$, nos aseguramos que es el único $j > i$ que cumple las condiciones del punto 4.b además de asegurarnos que dichos j 's son distintos entre las no aristas del matching M_i .
5. Dado las elecciones de los segmentos y que $V(\mathcal{B}) = V(C_{NR}^{tw})$ se tiene que para todo $i \geq 2$ existe un $j < i$ tal que $\exists \hat{x}\hat{y} \in M_j$ $\hat{x}\hat{x} \in E(B) \forall x \in P_i$ y $\hat{y}\hat{y} \in E(B) \forall y \in Q_i$. Por argumentos similares al punto anterior se tiene la unicidad ya que el algoritmo retorna segmentos vertices disjuntos y por cada arista del matching de las no aristas se tiene que se escoge un segmento que no tenga vértices anteriormente usados.

■

Lema 1.38 *Sea T tal que no pertenece a la familia Excepcion tal que $T^{[\#3]} = C_R + C_{NR}$. Entonces todo grafo maximal B_1 en C_{NR}^{tw} que cumple que es un bipartito completo menos un matching maximal con al menos 5 vértices cumple que existe un vértice relevante, r , en T , tal que $V(B_1) = \{u \in C_{NR}^{tw} | d_T(u, r) \in \{1, 2\}\}$.*

DEMOSTRACIÓN. Es fácil notar que si tenemos un vértice relevante r en T entonces los vértices $N_T^2(r) := \{u | d_T(u, r) \in \{1, 2\}\}$ cumple que forman un grafo maximal B_1 en C_{NR}^{tw} que cumple que es un bipartito completo menos un matching maximal con al menos 5 vértices. Y dada la maximalidad es fácil ver que este tipo de grafos en C_{NR}^{tw} solo surgen a partir de $N_T^2(r)$ de algún vértice relevante. ■

Teorema 1.39 *Sea $G = C_R + C_{NR}$ un grafo con dos componentes conexas, C_R un árbol y C_{NR} un grafo bipartito (que en caso de que también sea un árbol cumple que $|C_R^{tw}| \leq |C_{NR}^{tw}|$) tal que G no es el cubo de algún árbol que pertenece a la familia Excepcion. Existe un árbol T tal que $T^{[\#3]} = G$ si y solamente si el procedimiento 6, 'Búsqueda DBE', aplicado al twin graph de C_{NR} retorna una DBE $(B_i)_{i=1}^m$ de manera que el árbol, \hat{T} , asociado a dicha DBE cumple que existe un isomorfismo $\varphi : V(\hat{T}) \rightarrow V(S) \subseteq V(C_R)$ tal que:*

1. Si $x \in VC(C_R)$ entonces $x \in \varphi(\hat{T})$. En otras palabras solamente algunas hojas en C_R no tienen preimagen.
2. $|Q_1| \geq |N_{C_R}(\varphi(B_1)) \cap \varphi(\hat{T})| + \mathbf{1}_{N_{C_R}(\varphi(B_1)) \setminus \varphi(\hat{T}) \neq \emptyset} + |\mathcal{T}_{C_{NR}}^Q(B_1)|$.
3. $\forall i \geq 2, |Q_i| \geq |N_{C_R}(\varphi(B_i)) \cap \varphi(\hat{T})| + \mathbf{1}_{N_{C_R}(\varphi(B_i)) \setminus \varphi(\hat{T}) \neq \emptyset} + |\mathcal{T}_{C_{NR}}^Q(B_i)| - 1$.

DEMOSTRACIÓN. La implicancia que toma por hipótesis que el procedimiento retorna una DBE donde el isomorfismo cumple las propiedades, se tiene directamente por el Teorema 1.33.

Ahora supongamos que existe un árbol T tal que su cubo exacto es G . Dado que T contiene al menos un vértice relevante y por el Lema 1.38 y el Lema 1.35 sabemos que el

procedimiento *BuscarB1* encontrará un segmento B_1 . Luego por el lema 1.34 y el Lema 1.37 nos damos cuenta que la construcción entregada por el procedimiento 6 es la misma que se puede entregar usando la demostración del Lema 1.31, ya que sigue una metodología de BFS para rearmar los segmentos. Y por el Lema 1.32 entonces tenemos que se retorna una DBE de C_{NR}^{tw} tal que existe un isomorfismo como el de la caracterización que estamos buscando. ■

Esbozo de un Algoritmo de Decisión

Finalmente podemos esbozar el siguiente algoritmo de reconocimiento que recibe como input un grafo $G = C_R + C_{NR}$ con dos componentes conexas, y decide si existe un árbol T tal que $T^{\{\#3\}}$.

1. **Inicialización:** determinar si G tiene exactamente dos componentes conexas C_R, C_{NR} , con C_R un árbol y C_{NR} un grafo bipartito (que en caso de que también sea un árbol cumple que $|C_R^{tw}| \leq |C_{NR}^{tw}|$) tal que G no es el cubo de algún árbol de la familia *Excepcion*. Si es el cubo de algún árbol de la familia *Excepcion* retorno dicho árbol, pero si no cumple lo anterior retorno que no hay dicho árbol. Y armar el twin graph de C_{NR} .
2. **Encontrar una DBE de C_{NR}^{tw} :** En esta etapa se determina si existe una DBE de C_{NR}^{tw} y se construye el árbol \hat{T} asociado en caso de que sí.
3. **Buscar isomorfismos:** Usar el procedimiento de Matula [10] para encontrar todos los posibles isomorfismos de \hat{T} a un subgrafo de C_R .
4. **Determinar el isomorfismo que cumpla las condiciones:** Determinar cual de los isomorfismos encontrados en el paso anterior cumple las condiciones del Teorema 1.33.
5. **Rearmar el árbol** usar el Procedimiento 2 para encontrar el árbol.

Si alguno de los pasos anteriores falla entonces retornar que no existe dicho árbol.

1.3.3. Cuando $T^{\{\#3\}}$ tiene 1 componente conexa

Por el Lema 1.19 en su versión de vértices relevantes, se tiene que T contiene al menos dos vértices relevantes u y v tal que están a distancia que no es múltiplo de 3.

Este caso queda abierto, pero se espera que al igual que los casos anteriores, es posible encontrar un algoritmo de reconocimiento de tiempo polinomial

Capítulo 2

Número subcromático de potencias de grafos

El objetivo de este capítulo es estudiar el número subcromático de potencias de grafos, buscando mejorar la cota de dicho número para algunas familias de grafos poco densas. Se mostrarán resultados que implican por ejemplo una cota de 95 para cubos de grafos planares, una cota de 6 para los cuadrados de grafos outerplanar, una cota de 10 para los cubos de grafos outerplanar y una cota de 180 para grafos K_5 -minor free.

Los resultados de esta sección 2 se esperan publicar, junto con otros resultados, en un artículo que está en progreso [4].

Como ya se mencionó en la introducción el *número subcromático* de un grafo G es el menor número de colores necesarios para colorear los vértices del grafo, de manera que cada clase de color induce una unión disjunta de cliques. Se denota por $\chi_s(G)$.

Ahora vamos a introducir algunos parámetros de coloreo.

2.1. Números generalizados de Coloreo

De ahora en adelante todos los *ordenamientos* que aparezcan serán ordenamientos lineales de vértices. El conjunto de todos los ordenamientos para un grafo G se denotará por $\mathcal{O}(G)$. Dado un ordenamiento σ denotaremos por $u <_\sigma v$ para indicar que u es menor que v según el ordenamiento σ .

En la introducción se mencionó que en [7] se presentan dos parámetros importantes, el *número de l -coloreo*, col_l , y el *número de l -coloreo débil*, $wcol_l$, que son generalizaciones del número de coloreo clásico. A continuación veremos sus definiciones.

Dado un grafo G , dos vértices $u, v \in V(G)$, dos parámetros $l, k \in \mathbb{N} \cup \{\infty\}$ y un ordenamiento $\sigma \in \mathcal{O}(G)$.

Número de l -coloreo

Decimos que u es l -alcanzable por v si existe un v - u camino de largo a lo más l donde u es el menor de los vértices del camino y todos los vértices internos del camino son mayores (con respecto a σ) que v . En palabras técnicas existe un camino $P = z_0 z_1 \dots z_s$ con $z_0 = v$, $z_s = u$ y $s \leq l$ tal que $u <_\sigma v$ y para todo $i \in [s - 1]$ $v <_\sigma z_i$.

Definiremos el conjunto de los vértices l -alcanzables por v como:

$$Reach_l[G, \sigma, v] := \{v\} \cup \{u \in V(G) \mid u \text{ es } l\text{-alcanzable por } v\}$$

Y con este conjunto podemos definir el *número de l -coloreo con respecto a σ* de la siguiente manera:

$$col_l(G, \sigma) := \max_{v \in V(G)} |Reach_l[G, \sigma, v]|$$

Y finalmente el *número de l -coloreo* de G se define como

$$col_l(G) := \min_{\sigma \in \mathcal{O}(G)} col_l(G, \sigma)$$

Por otra parte podemos relajar un poco la definición, lo que nos lleva al siguiente concepto:

Número de l -coloreo débil

Decimos que u es *débilmente l -alcanzable* por v si existe un v - u camino de largo a lo más l donde u es el menor vértice del camino. En palabras técnicas existe un camino $P = z_0 z_1 \dots z_s$ con $z_0 = v$, $z_s = u$ y $s \leq l$ tal que para todo $i \in [s]$ $u <_\sigma z_{i-1}$.

Definiremos el conjunto de los vértices débilmente l -alcanzables por v como:

$$WReach_l[G, \sigma, v] := \{v\} \cup \{u \in V(G) \mid u \text{ es débilmente } l\text{-alcanzable por } v\}$$

Luego definimos el *número de l -coloreo débil con respecto a σ* de la siguiente manera:

$$wcol_l(G, \sigma) := \max_{v \in V(G)} |WReach_l[G, \sigma, v]|$$

Y finalmente el *número de l -coloreo débil* de G se define como:

$$wcol_l(G) := \min_{\sigma \in \mathcal{O}(G)} wcol_l(G, \sigma)$$

Estos parámetros se pueden generalizar en un solo concepto que vamos a introducir ahora, al cual llamaremos *número de k -hop l -coloreo*, y se denotará por $gcol_{k,l}$.

Número de k -hop l -coloreo

Decimos que u es k -hop l -alcanzable por v si existe un v - u camino de largo a lo más l donde u es el menor vértice del camino, y además se cumple que $|\{j \in [s] | x_j <_\sigma x_{i-1} \forall i \in [j]\}| \leq k$. En palabras técnicas existe un camino $P = z_0 z_1 \dots z_s$ con $z_0 = v$, $z_s = u$ y $s \leq l$ tal que para todo $i \in [s]$ $u <_\sigma z_{i-1}$ y $|\{j \in [s] | z_j <_\sigma z_{i-1} \forall i \in [j]\}| \leq k$.

Definiremos el conjunto de los vértices k -hop l -alcanzables por v como:

$$GReach_{k,l}[G, \sigma, v] := \{v\} \cup \{u \in V(G) | u \text{ es } k\text{-hop } l\text{-alcanzable por } v\}$$

Luego definimos el número de k -hop l -coloreo con respecto a σ de la siguiente manera:

$$gcol_{k,l}(G, \sigma) := \max_{v \in V(G)} |GReach_{k,l}[G, \sigma, v]|$$

Y finalmente el número de k -hop l -coloreo de G se define como:

$$gcol_{k,l}(G) := \min_{\sigma \in \mathcal{O}(G)} gcol_{k,l}(G, \sigma)$$

Dado este nuevo parámetro hay que notar lo siguiente:

Lema 2.1 Para todo grafo G y entero positivo l se tiene que:

1. $col_l(G) = gcol_{1,l}(G)$ (y más específico se tiene $Reach_l[G, \sigma, v] = GReach_{1,l}[G, \sigma, v]$).
2. $wcol_l(G) = gcol_{l,l}(G)$ (y más específico se tiene $WReach_l[G, \sigma, v] = GReach_{l,l}[G, \sigma, v]$).

DEMOSTRACIÓN. Directo de las definiciones. ■

Y para terminar vamos a introducir una última variante de los números de coloreo generalizados que será útil para los resultados que después se presentarán con respecto al número subcromático:

Número de l -coloreo semidébil

Definimos $SemiReach_l(G, \sigma, v)$ como el conjunto de vértices u tal que existe un u - v camino $P = z_0, \dots, z_s$ con $z_0 = u$, $z_s = v$ (Notar que el orden del camino es al revés con respecto a las definiciones de antes), de largo $s \leq l$ tal que u es el menor de los vértices del camino y además $v \leq_\sigma z_i$ para $\lceil \frac{1}{2}l \rceil \leq i \leq s$. Vamos a considerar al igual que antes que $v \in SemiReach_l(G, \sigma, v)$. Luego definimos $swcol_l(G, \sigma)$ como $swcol_l(G, \sigma) := \max_{v \in V} |SemiReach_l(G, \sigma, v)|$ y definimos el número de l -coloreo semidébil $swcol_l(G)$ como el $swcol_l(G, \sigma)$ más pequeño para σ un ordenamiento lineal de los vértices.

Lema 2.2 Para todo grafo G y entero positivo l se tiene que

$$swcol_l(G) \leq gcol_{\lceil \frac{1}{2}l \rceil, l}(G).$$

DEMOSTRACIÓN. Directo de las definiciones. ■

2.2. Acotamiento del número subcromático de potencias de grafos usando el número de coloreo semidébil

Vamos a definir el concepto de clustering.

Un clustering de un grafo H es una partición, \mathcal{X} , de $V(H)$ tal que cada elemento de la partición es un conjunto de vértices que induce un clique en H .

Dado un clustering \mathcal{X} de H , el grafo cociente H/\mathcal{X} es el grafo cuyos vértices son los elementos de \mathcal{X} y entre dos elementos $A, B \in \mathcal{X}$ hay una arista si y solamente si existe $u \in A$ y $v \in B$ tal que uv es una arista de H .

En [11] los autores logran acotar el número subcromático de la potencia de un grafo usando el número de coloreo débil y la noción de clustering. El objetivo de esta sección es buscar un refinamiento de dichas cotas usando el número de coloreo semidébil definido en la sección pasada.

Para ello primero vamos a probar nuestra versión del Teorema 7 de [11] (siguiendo un esquema similar):

Teorema 2.3 *Sea G un grafo, $\sigma \in \mathcal{O}(G)$ y $d \geq 2$ un entero. Entonces existe un clustering \mathcal{X} de G^d que satisface:*

$$\text{col}(G^d/\mathcal{X}) \leq \text{swcol}_{2d}(G, \sigma).$$

Más aún, si d es impar entonces se tiene que

$$\text{col}(G^d/\mathcal{X}) \leq \text{swcol}_{2d-1}(G, \sigma).$$

DEMOSTRACIÓN. Para $u \in V(G)$ definimos $m(u)$ como el menor vértice con respecto a σ que está en $N^{\lfloor \frac{d}{2} \rfloor}[u]$.

Consideremos la relación de equivalencia \sim en $V(G)$ dada por $u \sim v$ si y solamente si $m(u) = m(v)$. Definimos el clustering, \mathcal{X} , de G^d donde sus elementos son las clases de equivalencia de \sim .

Veamos que \mathcal{X} es efectivamente un clustering de G^d . Para ello necesitamos chequear que los elementos de \mathcal{X} son conjuntos de vértices que inducen un clique en G^d es decir hay que probar que $uv \in E(G^d)$ cuando $u \sim v$. Esto se tiene porque

$$d_G(u, v) \leq d_G(u, m(u)) + d_G(m(u), v) = d_G(u, m(u)) + d_G(m(v), v) \leq \left\lfloor \frac{d}{2} \right\rfloor + \left\lfloor \frac{d}{2} \right\rfloor \leq d$$

Para cada $A \in V(G^d/\mathcal{X})$, tomamos algún vértice $u \in A$ y definimos $m(A) := m(u)$. Por la definición de la relación de equivalencia \sim la elección del vértice $u \in A$ es arbitraria.

Ahora definimos en G^d/\mathcal{X} el ordenamiento τ dado por $A <_\tau B$ si y solamente si $m(A) <_\sigma m(B)$. Notar que $m(A) \neq m(B)$ cuando $A \neq B$, por lo tanto τ es en efecto un ordenamiento lineal. Para un conjunto $X \subseteq V(G^d/\mathcal{X})$ definimos $m(X) = \{m(A) | A \in X\}$. En efecto, para cada tal X , se tiene que $|m(X)| = |X|$.

Vamos a demostrar un resultado más fuerte, veamos que $col(G^d/\mathcal{X}, \tau) \leq swcol_{2d}(G, \sigma)$. Dado que $|m(Reach[G^d/\mathcal{X}, \tau, A])| = |Reach[G^d/\mathcal{X}, \tau, A]|$, es suficiente mostrar que para todo $A \in V(G^d/\mathcal{X})$ se tiene que

$$m(Reach[G^d/\mathcal{X}, \tau, A]) \subseteq SemiReach_{2d}[G, \sigma, m(A)]$$

Sea $B \in Reach[G^d/\mathcal{X}, \tau, A]$. Si $A = B$, entonces $m(B) = m(A) \in SemiReach_{2d}[G, \sigma, m(A)]$. Por lo tanto podemos asumir que $B <_\tau A$ y que $BA \in E(G^d/\mathcal{X})$. Notar que $m(B) <_\sigma m(A)$ porque $B <_\tau A$. Dado que $BA \in E(G^d/\mathcal{X})$, entonces existe $b \in V(B)$, $a \in V(A)$ tal que $ba \in V(G^d)$. Entonces existe un b, a -camino $Q = v_0v_1\dots v_q$ en G con $v_0 = b$, $v_q = a$ y $1 \leq q \leq d$.

Por la definición de $m(B)$ existe un $m(B), b$ -camino $P = u_0u_1\dots u_p$ en G con $u_0 = m(B)$, $u_p = b$ y $1 \leq p \leq \lfloor \frac{d}{2} \rfloor$. De forma análoga, existe un $a, m(A)$ -camino $R = w_0w_1\dots w_r$ con $w_0 = a$, $w_r = m(A)$ y $1 \leq r \leq \lfloor \frac{d}{2} \rfloor$.

Veamos que el $m(B), m(A)$ -walk PQR contiene un camino que satisface que $m(B) \in SemiReach_{2d}[G, \sigma, m(A)]$. Dado que el walk tiene largo a lo más $2d$ (a lo más $2d - 1$ si d es impar), cualquier $m(B), m(A)$ -camino contenido en el walk también tendrá a lo más ese largo. Más aún podemos tomar alguno de esos caminos, digamos $T = T_1T_2T_3$, donde T_1 es un camino en P , T_2 es un camino en Q y T_3 un camino en R . Sea $t_0\dots t_s$ la secuencia de vértices de T , con $t_0 = m(B)$, $t_s = m(A)$. Notar que para cada $i \in [\lfloor \frac{d}{2} \rfloor]$ se tiene que $m(b) = m(B) \leq_\sigma v_i$ y como $v_{q-i} \in N^{\lfloor \frac{d}{2} \rfloor}[a]$ se tiene que $m(A) \leq_\sigma v_{q-i}$. Más aún, para todo $i \in [r]$ ($1 \leq r \leq \lfloor \frac{d}{2} \rfloor$) se tiene que $m(B) \leq_\sigma u_i$ y para todo $i \in [p]$ ($1 \leq p \leq \lfloor \frac{d}{2} \rfloor$), como $w_{i-1} \in N^{\lfloor \frac{d}{2} \rfloor}[a]$, se tiene que $m(a) = m(A) \leq_\sigma w_{i-1}$. Dado que $m(B) <_\sigma m(A)$, todos los vértices $t \in T$ cumplen que $m(B) \leq_\sigma t$. Si el largo de T es a lo más $d - 1$ entonces ya se tiene que $m(B) \in SemiReach_{2d}[G, \sigma, m(A)]$, que es lo que se buscaba.

En el caso en que el largo s de T es al menos d hay que mostrar que para todo $i \in \{d, \dots, s\}$ se tiene que $m(a) \leq_\sigma t_i$.

Probemos el caso en que d es par, es decir $d = 2h$. Notar que $v_i \in N^h[a]$ para todo $i \in \{h \dots q\}$, lo que implica que para cada tal i se tiene que $m(a) = m(A) \leq_\sigma v_i$. Por nuestra elección de T se tiene que $(P \cup \{v_1, \dots, v_{h-1}\}) \cap T \subseteq \{t_0, \dots, t_{d-1}\}$, esto implica que $\{t_d, \dots, t_s\} \subseteq \{v_h, \dots, v_q\} \cup R$. Y por la observación anterior esto da que para todo $i \in \{d, \dots, s\}$ se tiene que $m(a) \leq_\sigma t_i$.

El caso cuando d es impar, es decir $d = 2h - 1$ (o $2h + 1$), hay que notar que $h \geq \lfloor \frac{d}{2} \rfloor$ y por lo tanto $v_i \in N^h[a]$, lo que implica que $m(a) = m(A) \leq_\sigma v_i$ para todo $i \in \{h, \dots, q\}$. El resto del argumento es análogo finalizando así la demostración. ■

Observación: En el Apéndice A se puede encontrar una versión más general de este Teorema usando $gcol$.

Con este resultado ahora podemos enunciar nuestro acotamiento del número subcromático usando el número de coloreo semidébil.

Teorema 2.4 *Para todo grafo G y cualquier entero fijo $d \in \mathbb{N}$, se tiene que $\chi_s(G^d) \leq \text{swcol}_{2d}(G)$. Más aún si d es impar, se tiene que $\chi_s(G^d) \leq \text{swcol}_{2d-1}(G)$.*

DEMOSTRACIÓN. Sea $l = 2d$ si es que d es par, o $l = 2d - 1$ si es que d es impar. Sea $\sigma \in \mathcal{O}(G)$ tal que $\text{swcol}_l(G, \sigma) = \text{swcol}_l(G)$. El Teorema 2.3 nos garantiza que existe un clustering \mathcal{X} de G^d tal que $\text{col}(G^d/\mathcal{X}) \leq \text{swcol}_l(G)$. Entonces se tiene que $\chi_s(G^d) \leq \chi(G^d/\mathcal{X}) \leq \text{col}(G^d/\mathcal{X}) \leq \text{swcol}_l(G)$. ■

En el Teorema 3.1.1 de [2] se demuestra que para todo grafo planar G se tiene que $\text{swcol}_5(G) \leq 95$. Por lo tanto juntando este resultado con nuestros teoremas se tiene el siguiente resultado:

Teorema 2.5 *Si G es un grafo planar entonces $\chi_s(G^3) \leq 95$.*

DEMOSTRACIÓN. Consecuencia directa de que $\text{swcol}_5(G) \leq 95$ y del Teorema 2.4. ■

2.3. Cotas explícitas de $gcol$ para grafos con treewidth acotado o minors excluidos

En esta sección necesitaremos lo siguiente. De la demostración del Teorema 4.2 de [6] se puede inferir el siguiente resultado:

Lema 2.6 *Sea G un grafo y un ordenamiento $\sigma \in \mathcal{O}(G)$ que cumple que*

$$\max_{v \in V(G)} |\text{Reach}_\infty[G, \sigma, v]| \leq t + 1$$

. Entonces para todo entero positivo k y para todo vértice $v \in V(G)$ se tiene que

$$|\text{WReach}_k[G, \sigma, v]| \leq \binom{t+k}{t}.$$

Antes de ver los resultados obtenidos en esta sección, vamos a introducir los conceptos de k -tree, treewidth y de ordenamiento simplicial.

Un grafo es un k -tree si es un clique de tamaño $k + 1$ o puede ser obtenido de un k -tree más pequeño agregando un vértice que sea adyacente a k vértices que formaban un clique de tamaño k .

El treewidth de un grafo G es el menor entero k tal que G es un subgrafo de un k -tree.

Para un k -tree G , decimos que $\sigma \in \mathcal{O}(G)$ es un ordenamiento simplicial si puede ser obtenido de la siguiente manera: G se puede construir partiendo de un $(k + 1)$ -clique K_0 y los

vértices de K_0 serán los menores en el ordenamiento σ (entre ellos darles un orden arbitrario). Luego para $v \notin K_0$ se tiene que $u <_\sigma v$ si es que u fue añadido al k -tree antes que v .

Lema 2.7 *Sea k un entero positivo, G un k -tree y $\sigma \in \mathcal{O}(G)$ un ordenamiento simplicial. Para todo $v \in V(G)$ se tiene que $\text{Reach}_1[G, \sigma, v] = \text{Reach}_\infty[G, \sigma, v]$.*

DEMOSTRACIÓN. Si v es uno de los $k + 1$ vértices más pequeños con respecto a σ entonces cualquier vértice $u <_\sigma v$ cumple que $u \in \text{Reach}_1[G, \sigma, v]$ como se quería. En otro caso, consideremos C_v la componente de $(G \setminus \text{Reach}_1[G, \sigma, v]) \cup \{v\}$ que contenga a v . Notar que $v \leq_\sigma w$ para todo $w \in C_v$. Si x pertenece a $\text{Reach}_\infty[G, \sigma, v]$ entonces para todo v, x -camino que muestre que x es alcanzable por v , cumple que todos sus vértices internos están en C_v . Pero si dicho camino existe entonces necesariamente se tiene que $x \in \text{Reach}_1[G, \sigma, v]$ ya que era un ordenamiento simplicial, obteniendo así el resultado. ■

En [17] se muestran diversos teoremas que acotan superiormente variantes de los números generalizados de coloreos para diversas familias de grafos. Si replicamos estas técnicas para acotar el número de *gcol* y tomamos en consideración el Lema 2.2 y el Teorema 2.4 podemos encontrar cotas superiores explícitas para el número subcromático de potencias de grafos donde los grafos pertenecen a ciertas familias.

A continuación vamos a ver una cota superior para el caso en que se tenga treewidth acotado. Pero primero necesitaremos un lema.

Lema 2.8 *Sean k, l enteros positivos con $k \leq l$, G un k -tree y $\sigma \in \mathcal{O}(G)$ un ordenamiento simplicial. Para todo $v \in V(G)$ se tiene que*

$$\text{WReach}_k[G, \sigma, v] = \text{GReach}_{k,l}[G, \sigma, v].$$

DEMOSTRACIÓN. Por las definiciones se tiene que $\text{WReach}_k[G, \sigma, v] \subseteq \text{GReach}_{k,l}[G, \sigma, v]$. Veamos la otra inclusión. Sea $u \in \text{GReach}_{k,l}[G, \sigma, v] \setminus \{v\}$. Por la definición existe un u, v -camino $P = w_0 \dots w_s$ con $w_0 = v$, $w_s = u$, $s \leq l$, para todo $i \in [s]$ $u <_\sigma w_{i-1}$ y el conjunto

$$I := \{j \in [s] : w_j \leq_\sigma w_{i-1} \forall i \in [j]\}$$

cumple que $|I| \leq k$.

Notar que el conjunto I es distinto de vacío ya que $w_s = u$ está en el conjunto. Ordenamos los elementos de I de manera creciente, es decir $I = \{j_1, \dots, j_m\}$ donde $j_i \leq j_{i+1}$ para todo $i \in [m - 1]$. Por la definición de I se tiene que $w_{j_{i+1}} <_\sigma w_{j_i}$ para todo $i \in [m - 1]$ y $w_{j_m} = u$.

Veamos que $T = vw_{j_1} \dots w_{j_{m-1}}u$ es un u, v -camino tal que demuestra que $u \in \text{WReach}_k[G, \sigma, v]$. El subcamino $v \dots w_{j_1}$ de P muestra que $w_{j_1} \in \text{Reach}_\infty[G, \sigma, v]$ que por el Lema 2.7 implica que $w_{j_1} \in \text{Reach}_1[G, \sigma, v]$. De forma similar para todo $i \in [m - 1]$ se tiene que $w_{j_i} w_{j_{i+1}} \in E(G)$, porque algún subcamino de P demuestra que $w_{j_{i+1}} \in \text{Reach}_\infty[G, \sigma, w_{j_i}] = \text{Reach}_1[G, \sigma, w_{j_i}]$. Dado que u es el menor vértice en T con respecto a σ , el resultado se tiene. ■

Ahora si vamos a enunciar el Teorema que entrega una cota superior para grafos con treewidth acotado.

Teorema 2.9 Sean k, t, l enteros positivos con $k \leq l$. Para todo grafo G con treewidth a lo más t se tiene que $\text{gcol}_{k,l}(G) \leq \binom{t+k}{t}$.

DEMOSTRACIÓN. Dado que $\text{gcol}_{k,l}(G)$ no puede disminuir si agregamos aristas podemos asumir que G es un k -tree y por lo tanto existe un ordenamiento simplicial σ . Dado que G tiene treewidth a lo más t , el ordenamiento σ satisface que $t + 1 \geq |\text{Reach}_1[G, \sigma, v]|$ que por el Lema 2.7 implica que $t + 1 \geq |\text{Reach}_\infty[G, \sigma, v]|$. Más aún por el Lema 2.8 para todo v y para todo k se tiene que $\text{WReach}_k[G, \sigma, v] = \text{GReach}_{k,l}[G, \sigma, v]$. Y el resultado se tiene finalmente por el Lema 2.6. ■

Una consecuencia directa de esto es lo siguiente

Teorema 2.10 Sea G un grafo con treewidth 2 entonces $\chi_s(G^2) \leq 6$ y $\chi_s(G^3) \leq 10$.

DEMOSTRACIÓN. Usar el Lema 2.2, el Teorema 2.4 y el Teorema 2.9. ■

Recordando que los grafos outerplanar tienen treewidth a lo más 2, se tiene que una consecuencia del Teorema 2.10 es la siguiente:

Teorema 2.11 Sea G un grafo outerplanar entonces $\chi_s(G^2) \leq 6$ y $\chi_s(G^3) \leq 10$.

Finalmente se mostrarán cotas para familia de grafos con clique minors excluidos. Primero necesitamos algunas definiciones y un lema que puede ser deducido de [16].

Una secuencia $\mathcal{H} = \{H_1, \dots, H_r\}$ de subgrafos no vacíos de G es una *descomposición* de G si el conjunto de vértices $V(H_1), \dots, V(H_r)$ es una partición de $V(G)$. Se dice que es *conexa* si todo H_i es conexo.

Sea $H \subseteq G$ un subgrafo conexo de G y sea $f: \mathbb{N} \rightarrow \mathbb{N}$ una función. Se dice que H se f -extiende en G si, para todo $k \in \mathbb{N}$ y $v \in V(G)$, se tiene que $|N^k[v] \cap V(H)| \leq f(k)$.

Se denota por $G[H_{\geq i}]$ el subgrafo de G inducido por $\cup_{j=i}^r V(H_j)$.

Se dice que una descomposición \mathcal{H} es f -flat si cada H_i se f -extiende en $G[H_{\geq i}]$.

Sea C una componente conexa de $G[H_{\geq i+1}]$ con $i \in \{1, \dots, r-1\}$. El número de i -separación de C , $s_i(C)$, es el número s de grafos en $\{H_1, \dots, H_i\}$ tal que están conectados a C . Sea $w_i(\mathcal{H})$ el máximo de los $s_i(C)$ sobre todas las componentes conexas C de $G[H_{\geq i}]$. El ancho de \mathcal{H} se define como $\max_{1 \leq i \leq r-1} w_i(\mathcal{H})$.

En [16] se tiene el siguiente resultado que dice que un grafo con una descomposición conexa de ancho a lo más t cumple que el grafo obtenido al contraer los subgrafos de la descomposición, tiene treewidth a lo más t .

Lema 2.12 Sea G un grafo y sea $\mathcal{H} = \{H_1, \dots, H_r\}$ una descomposición conexa de G de ancho a lo más t . Sea H el grafo obtenido contrayendo cada subgrafo H_i a un solo vértice. Entonces H tiene treewidth a lo más t .

Con este resultado podemos enunciar el siguiente lema:

Lema 2.13 *Sea $f: \mathbb{N} \rightarrow \mathbb{N}$ y k, l, t enteros positivos con $k \leq l$. Para todo grafo G que admite una descomposición f -flat conexa de ancho t se tiene que*

$$\text{gcol}_{k,\ell}(G) \leq \binom{t+k}{t} \cdot f(\ell)$$

DEMOSTRACIÓN. La demostración es similar a la de [16, Lemma 3.5]. Sea $\mathcal{H} = (H_1, \dots, H_r)$ una descomposición f -flat conexa de G de ancho t . Sea H el grafo obtenido al contraer los subgrafos H_i en G . Identificamos los vértices de H con los subgrafos H_i . Dado que \mathcal{H} es conexo por el Lema 2.12 se tiene que H tiene treewidth a lo más t . Por el Teorema 2.9 se tiene que $\text{gcol}_{k,\ell}(H) \leq \binom{t+k}{t}$, por lo tanto existe un ordenamiento $\sigma \in \mathcal{O}(H)$ tal que para todo $H_i \in V(H)$ se tiene $|\text{GReach}_{k,\ell}[H, L, H_i]| \leq \binom{t+k}{t}$.

A partir de σ se define el ordenamiento σ' en $V(G)$. Para $u \in H_i$ y $v \in H_j$, con $i \neq j$, se tendrá $u <_{\sigma'} v$ si $H_i <_{\sigma} H_j$. Y para todo $1 \leq i \leq r$ ordenamos los vértices H_i en cualquier orden.

Notar que para todo vértice $v \in H_i$ se satisface que

$$\text{GReach}_{k,\ell}[G, L', v] \subseteq N^\ell[v] \cap \{H_j \mid H_j \in \text{GReach}_{k,\ell}[H, L, H_i]\}.$$

Por lo tanto, se tiene que hay a lo más $\binom{t+k}{t}$ subgrafos entre H_1, \dots, H_r en G que contienen vértices de $\text{GReach}_{k,\ell}[G, L', v]$. Dado que \mathcal{H} es f -flat se sabe que la intersección de cada uno de esos subgrafos con $N^\ell[v]$ es a lo más $f(\ell)$. ■

En [16] se prueba que:

Lema 2.14 *Sea $t \geq 4$ y sea $f: \mathbb{N} \rightarrow \mathbb{N}$ la función $f(\ell) = (t-3)(2\ell+1)$. Para todo grafo K_t -minor free, G , se tiene que existe una descomposición f -flat conexa de G con ancho a lo más $t-2$.*

Ahora podemos enunciar el Teorema.

Teorema 2.15 *Sea k, t, l enteros positivos con $k \leq l$ y $t \geq 4$. Para todo grafo K_t -minor free, G , se tiene que $\text{gcol}_{k,\ell}(G) \leq \binom{t+k-2}{t-2} (t-3)(2l+1)$.*

DEMOSTRACIÓN. Como G es K_t -minor free por el Lema 2.14 se tiene que existe una descomposición f -flat conexa de G con ancho a lo más $t-2$, con $f: \mathbb{N} \rightarrow \mathbb{N}$ la función $f(\ell) = (t-3)(2\ell+1)$. Luego usando el Lema 2.13 tenemos la siguiente cota

$$\text{gcol}_{k,\ell}(G) \leq \binom{t-2+4}{t-2} f(\ell) = \binom{t+k-2}{t-2} (t-3)(2l+1).$$

Demostrando así lo pedido. ■

Los resultados anteriores implican lo siguiente:

Teorema 2.16 *Sea G un grafo K_5 -minor free, entonces $\chi_s(G^2) \leq 180$.*

DEMOSTRACIÓN. Por el Teorema 2.4 se tiene que $\chi_s(G^2) \leq swcol_4(G)$ y por el Lema 2.2 se tiene que $swcol_4(G) \leq gcol_{2,4}(G)$. Y como G es K_5 -minor free, por el Teorema 2.15 se tiene que $gcol_{2,4}(G) \leq \binom{5+2-2}{5-2}(5-3)(2 \cdot 4 + 1) = 180$. Juntando todas las desigualdades se tiene que

$$\chi_s(G^2) \leq swcol_4(G) \leq gcol_{2,4}(G) \leq 180.$$
■

Observación: Usando técnicas más elaboradas es posible mejorar el Teorema 2.15, donde se puede obtener que $gcol_{k,l}(G) \leq \left(\binom{t+k-2}{t-2} - 1\right)(t-3)(2l+1) + (t-3)l + 1$, lo que implica por ejemplo que para G un grafo K_5 -minor free, se tiene que $\chi_s(G^2) \leq 171$. Estos resultados están en progreso en [4].

Conclusión

En la primera parte de esta tesis se logró caracterizar en su totalidad los cuadrados exactos de árboles obteniendo un algoritmo de reconocimiento de tiempo polinomial. Por otra parte se logró caracterizar los cubos exactos de árboles en dos de los tres casos posibles, logrando esbozar un algoritmo, para el caso de dos componentes conexas, que en tiempo polinomial resuelve el problema de reconocimiento. También se encontró una caracterización para los grafos que tienen raíces exactas que son triangle-free.

En la segunda parte se logró acotar el número subcromático de potencias de grafos usando nuevos parámetros de coloreos. Se mejoró a 95 la cota superior del número subcromático de cubos de grafos planares y se lograron resultados para acotar el parámetro *gcol* en grafos con treewidth acotado, lo que implica por ejemplo una cota superior de 6 para el número subcromático de cuadrados de grafos con treewidth 2 y una cota superior de 10 para cubos de grafos con treewidth 2; y esto a su vez implica una cota de 6 y 10 para los cuadrados y cubos de grafos outerplanar, respectivamente. También se logró encontrar una cota superior para el parámetro *gcol* en grafos con clique minors excluidos, lo que implica una cota de 180 para grafos K_5 -minor free.

Problemas Abiertos

Estos estudios abren las puertas a nuevas problemáticas y preguntas interesantes para investigar:

Un problema abierto es buscar caracterizaciones para el resto de las potencias exactas, en particular las potencias pares. Una observación sencilla de demostrar es la siguiente:

Observación: Sea T un árbol y p, q enteros positivos con $p \leq q$, entonces se tiene que $T^{[\#pq]} = (T^{[\#p]})^{[\#q]}$.

Lo que significa que estudiar las potencias exactas pares de árboles se reduce a estudiar las potencias exactas de los clique-tree, ya que $T^{[\#2d]} = (T^{[\#2]})^{[\#d]}$, y ya conocemos la estructura de $T^{[\#2]}$.

Otra idea interesante sería lograr alguna caracterización que sirva para cualquier potencia exacta de árbol.

Por otra parte los algoritmos de decisión presentados en el capítulo 1 resuelven el problema

de identificar si un grafo con dos componentes conexas es el cuadrado exacto o cubo exacto de algún árbol en tiempo polinomial, resolviendo la problemática de si estos problemas se podían o no resolver en tiempo polinomial. Pero probablemente se puedan encontrar algoritmos más eficientes que tengan menor orden de complejidad que el de los algoritmos presentados. Esto ya que dichos algoritmos usan modificaciones del algoritmo de Matula, del cual se conocen mejoras con respecto al tiempo presentadas en el artículo de Shamir y Tsur [15], por lo que debería ser posible encontrar algoritmos más eficientes que resuelvan estos problemas.

Otra problemática abierta viene motivada por lo siguiente: en [3] se demuestra que las raíces cuadradas exactas que son árboles no son únicas, siendo un contraste con respecto a las raíces cuadradas (normales) que son árboles, las cuales sí son únicas bajo isomorfismos. Además es posible encontrar árboles no isomorfos cuyos cubos exactos sí lo son, para ello basta pensar en el Procedimiento 2 el cual se puede modificar levemente para encontrar otro árbol diferente que es raíz cúbica exacta de algún grafo. Por lo que queda como conjetura que para cualquier entero positivo $p \geq 2$ se pueden encontrar dos árboles no isomorfos con diámetro mayor que $2p - 4$ cuyas potencias exactas p -ésimas sí son isomorfas.

En [3] también se presenta la siguiente caracterización de los grafos que tienen raíz cuadrada exacta:

Teorema 3.1: Un grafo G tiene una raíz cuadrada exacta si y solamente si G es igual a $\overline{G}^{[\#2]}$

Por lo tanto el problema abierto es encontrar alguna caracterización similar para el caso de grafos con raíces cúbicas exactas.

Una observación interesante relacionado con ello es la siguiente:

Lema 3.2: Si G es un grafo tal que tiene raíz cúbica exacta entonces se cumple que:

1. $G = \overline{G}^{[\#2]}$
2. $\overline{G}^{[\#3]}$ no tiene aristas

Por lo tanto, se puede concluir que si G tiene raíz cúbica exacta entonces también tiene raíz cuadrada exacta (obviamente son diferentes). En otras palabras la familia de grafos que tienen raíz cúbica exacta está contenida en la familia de grafos que tienen raíz cuadrada exacta.

Otro problema abierto es ver si se pueden reconocer los cubos exactos de grafos outerplanar en tiempo polinomial.

Por otra parte las herramientas presentadas en el capítulo 2, en particular los números generalizados de coloreo tienen un gran potencial para ser usados en diversos problemas de coloreo, como se hizo en el caso de los números subcromáticos de potencias mejorando algunas de las cotas conocidas para estos números. Queda abierto seguir estudiando estos parámetros y sus implicancias tanto en el número cromático como subcromático de potencias y potencias exactas.

Por último, vamos a mencionar una conjetura del autor de esta tesis, que podría ser un resultado más fuerte que el Lema 2.6 en el sentido que esta conjetura implica el lema anterior.

Conjetura 3.3: Sea G un grafo y un ordenamiento $\sigma \in \mathcal{O}(G)$ que cumple que

$$\max_{v \in V(G)} |Reach_\infty[G, \sigma, v]| \leq t + 1$$

. Entonces para todo entero positivo k y para todo v\u00e9rtice $v \in V(G)$ se tiene que

$$|X_k[G, \sigma, v]| \leq \binom{t-1+k}{t-1}$$

donde $X_k[G, \sigma, v] := WReach_k[G, \sigma, v] \setminus WReach_{k-1}[G, \sigma, v]$ y $X_0[G, \sigma, v] = \{v\}$

Lema 3.4: La Conjetura implica el Lema 2.6

DEMOSTRACI\u00d3N. Probaremos el Lema por inducci\u00f3n en k . Para el caso base $k = 1$ se tiene que $|WReach_1[G, \sigma, v]| = |Reach_1[G, \sigma, v]| \leq |Reach_\infty[G, \sigma, v]| \leq t + 1 = \binom{t+1}{t}$. La hip\u00f3tesis inductiva es $|WReach_k[G, \sigma, v]| \leq \binom{t+k}{t}$. Para el paso inductivo notemos que $WReach_{k+1} = WReach_k \cup X_{k+1}$ (y es una uni\u00f3n disjunta. Luego usando la hip\u00f3tesis inductiva y la conjetura se tiene que $|WReach_{k+1}| = |WReach_k| + |X_{k+1}| \leq \binom{t+k}{t} + \binom{t+k}{t-1} = \binom{t+k+1}{t}$ probando as\u00ed el Lema 2.6 \blacksquare

Recordar que el Lema 2.6 es verdad ya que es una consecuencia de los resultados de [6], pero la conjetura podr\u00eda no ser cierta. A\u00fan as\u00ed es interesante mencionar que en caso de ser cierta esta cota es ajustada, es decir existen grafos y ordenamiento de esos grafos en que se alcanza la igualdad.

En la Figura 2.1 se puede presenten tres grafos, para $k = 1, 2$ y 3 , que contienen un ordenamiento tal que cumple la hip\u00f3tesis de la conjetura y alcanzan la cota con igualdad. Dicho ejemplo se puede generalizar a cualquier k .

Generalizando para un k cualquiera se tiene que

$$|X_k[G, \sigma, v]| = \sum_{i_k=1}^t \sum_{i_{k-1}=1}^{i_k} \cdots \sum_{i_1=1}^{i_2} i_1 = \binom{t-1+k}{t-1}.$$

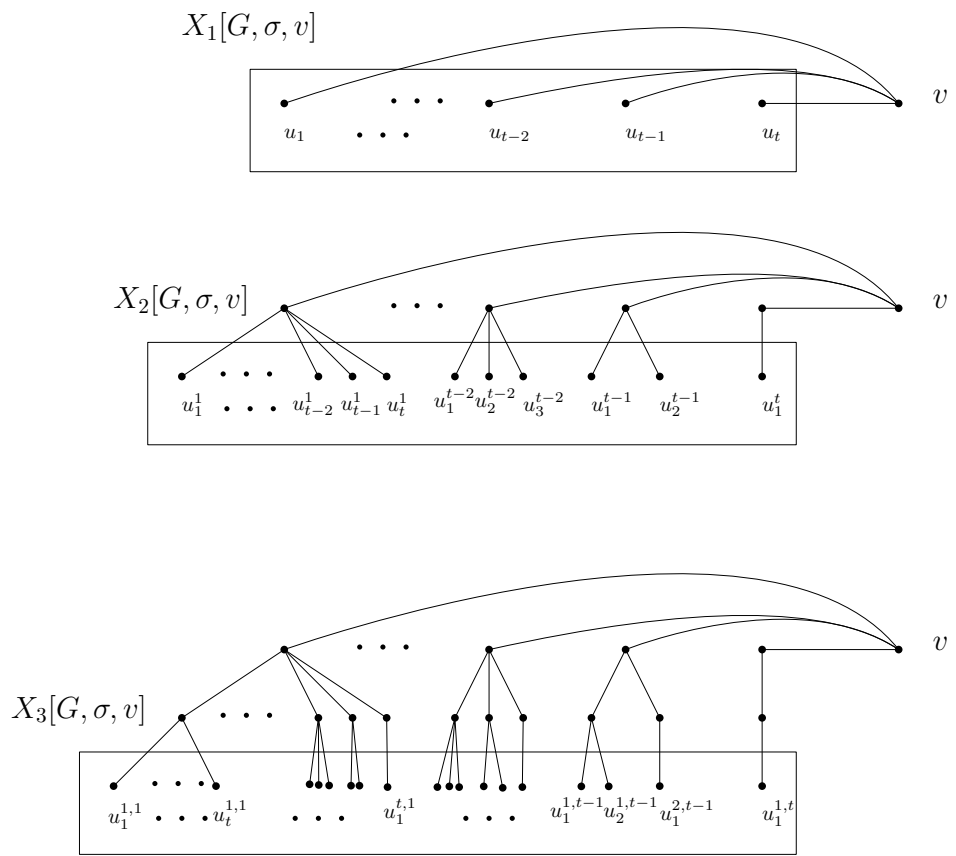


Figura 2.1: Ejemplos para $k = 1$, $k = 2$ y $k = 3$.

Bibliografía

- [1] Michael O Albertson, Robert E Jamison, Stephen T Hedetniemi, and Stephen C Locke. The subchromatic number of a graph. In *Annals of Discrete Mathematics*, volume 39, pages 33–49. Elsevier, 1989.
- [2] Ahlam Almulhim. *Estimating Low Generalized Coloring Numbers of Planar Graphs*. PhD thesis, Arizona State University, 2020.
- [3] Yandong Bai, Pedro P Cortés, Reza Naserasr, and Daniel A Quiroz. Characterizing and recognizing exact-distance squares of graphs. *arXiv preprint arXiv:2211.02699*, 2022.
- [4] Pedro P Cortés, Pankaj Kumar, Benjamin Moore, Patrice Ossona De Mendez, and Daniel A Quiroz. Subchromatic numbers of powers of graphs with excluded minors. *Manuscript in Preparation*, 2023.
- [5] Reinhard Diestel. *Graph Theory*. Springer Publishing Company, Incorporated, 5th edition, 2017.
- [6] Martin Grohe, Stephan Kreutzer, Roman Rabinovich, Sebastian Siebertz, and Konstantinos Stavropoulos. Coloring and covering nowhere dense graphs. *SIAM Journal on Discrete Mathematics*, 32(4):2467–2481, 2018.
- [7] Hal A Kierstead and Daqing Yang. Orderings on graphs and game coloring number. *Order*, 20:255–264, 2003.
- [8] Lap Chi Lau. Bipartite roots of graphs. *ACM Transactions on Algorithms (TALG)*, 2(2):178–208, 2006.
- [9] Yaw-Ling Lin and Steven S Skiena. Algorithms for square roots of graphs. *SIAM Journal on Discrete Mathematics*, 8(1):99–118, 1995.
- [10] David W Matula. Subtree isomorphism in $O(n^{5/2})$. In *Annals of Discrete Mathematics*, volume 2, pages 91–106. Elsevier, 1978.
- [11] Jaroslav Nešetřil, P Ossona de Mendez, M Pilipczuk, and Xuding Zhu. Clustering powers of sparse graphs. *Electronic Journal of Combinatorics*, 27(4):P4.
- [12] Jaroslav Nešetřil and Patrice Ossona De Mendez. *Sparsity: graphs, structures, and algorithms*, volume 28. Springer Science & Business Media, 2012.
- [13] Keith Paton. An algorithm for the blocks and cutnodes of a graph. *Commun. ACM*, 14(7):468–475, jul 1971.

- [14] Ian C Ross and Frank Harary. The square of a tree. *The Bell System Technical Journal*, 39(3):641–647, 1960.
- [15] Ron Shamir and Dekel Tsur. Faster subtree isomorphism. *Journal of Algorithms*, 33(2):267–280, 1999.
- [16] Jan Van Den Heuvel, Patrice Ossona de Mendez, Daniel Quiroz, Roman Rabinovich, and Sebastian Siebertz. On the generalised colouring numbers of graphs that exclude a fixed minor. *European Journal of Combinatorics*, 66:129–144, 2017.
- [17] Jan van den Heuvel, Hal A Kierstead, and Daniel A Quiroz. Chromatic numbers of exact distance graphs. *Journal of Combinatorial Theory, Series B*, 134:143–163, 2019.

Anexo

Versión con *gcol* del Teorema 2.3

Un poco de notación, recordemos que $u \in G\text{Reach}_{k,l}[G, \sigma, v]$ si existe un camino $P = z_0 z_1 \dots z_s$ con $z_0 = v$, $z_s = u$ y $s \leq l$ tal que para todo $i \in [s]$ $u <_\sigma z_{i-1}$ y $|\{j \in [s] \mid x_j <_L x_{i-1} \forall i \in [j]\}| \leq k$. Dado el camino anterior, a los índices $j \in \{j \in [s] \mid x_j <_L x_{i-1} \forall i \in [j]\}$ los llamaremos *hops*.

Teorema .1 *Sea G un grafo, σ un ordenamiento de los vértices de G y d, r enteros positivos con $d \geq 2$. Entonces existe un clustering \mathcal{X} de G^d que cumple:*

$$\text{wcol}_r(G^d/\mathcal{X}) \leq \text{gcol}_{dr,2dr}(G, \sigma)$$

DEMOSTRACIÓN. Para $u \in V(G)$ definimos $l(u)$ como el menor vértice (con respecto al ordenamiento σ) tal que $l(u) \in W\text{Reach}_{\lfloor \frac{d}{2} \rfloor}[G, \sigma, u]$

Definimos el clustering \mathcal{X} en G^d como sigue: Los elementos de \mathcal{X} son las clases de equivalencia de la siguiente relación

$$u \sim v \iff l(u) = l(v)$$

Veamos que efectivamente es un clustering. Sean $u, v \in V(G^d)$ tal que $u \sim v$, entonces se tiene que

$$d_G(u, v) \leq d_G(u, l(u)) + d_G(l(u), v) = d_G(u, l(u)) + d_G(l(v), v) \leq \left\lfloor \frac{d}{2} \right\rfloor + \left\lfloor \frac{d}{2} \right\rfloor \leq d$$

Por lo tanto $uv \in G^d$.

Para $A \in V(G^d/\mathcal{X})$ definimos $l(A)$ como el vértice $l(u)$ con $u \in V(G)$ que define al elemento A .

Definimos en G^d/\mathcal{X} el orden τ : Sean $A, B \in V(G^d/\mathcal{X})$

$$A <_\tau B \iff l(A) <_\sigma l(B)$$

Vamos a demostrar que $\forall A \in \mathcal{X}$ se tiene que:

$$|WReach_r[G^d/\mathcal{X}, \tau, A]| \leq |GReach_{dr, 2dr}[G, \sigma, l(A)]|$$

Sea $B \in WReach_r[G^d/\mathcal{X}, \tau, A]$. Eso implica que existe un camino $C_0C_1\dots C_p$ en G^d/\mathcal{X} con $C_0 = A$, $C_p = B$, $p \leq r$ y para todo $i \in [0..p]$ se tiene que $C_i \geq_\tau B$. Vamos a definir para cada $i \in [0..p]$ $x_i := l(C_i)$.

Notar que para cada $i \in [0..p-1]$ se tiene que $C_iC_{i+1} \in E(G^d/\mathcal{X})$, por lo que existe $y_i \in C_i$ y $z_i \in C_{i+1}$ tal que $y_iz_i \in E(G^d)$.

Como $y_iz_i \in E(G^d)$ se tiene que en G existe un y_i - z_i camino $Q_i = q_0^i q_1^i \dots q_{s_i}^i$ con $q_0^i = y_i$, $q_{s_i}^i = z_i$ y $s_i \leq d$. Notar que $x_i = l(y_i)$ y $x_{i+1} = l(z_i)$. Además notar que los primeros $\lfloor \frac{d}{2} \rfloor + 1$ vértices en el camino Q_i son mayores o iguales que $l(y_i)$ mientras que los últimos $\lfloor \frac{d}{2} \rfloor + 1$ vértices son mayores o iguales que $l(z_i)$. En resumen $\forall j \in [1.. \lfloor \frac{d}{2} \rfloor]$:

- $q_j^i \geq_\sigma l(y_i) = x_i$
- $q_{s_i-j}^i \geq_\sigma l(z_i) = x_{i+1}$

Como $x_i = l(y_i)$ se tiene que $x_i \in WReach_{\lfloor \frac{d}{2} \rfloor}[G, \sigma, y_i]$. Es decir que en G existe un x_i - y_i camino $P_i = p_0^i p_1^i \dots p_{t_i}^i$ con $p_0^i = x_i$, $p_{t_i}^i = y_i$ y $t_i \leq \lfloor \frac{d}{2} \rfloor$ tal que para todo $j \in [0..t_i]$ se tiene que $p_j^i \geq_\sigma x_i$.

De manera similar como $x_{i+1} = l(z_i)$ se tiene que en G existe un z_i - x_{i+1} camino $R_i = r_0^i r_1^i \dots r_{h_i}^i$ con $r_0^i = z_i$, $r_{h_i}^i = x_{i+1}$ y $h_i \leq \lfloor \frac{d}{2} \rfloor$ tal que para todo $j \in [0..h_i]$ se tiene que $r_j^i \geq_\sigma x_{i+1}$.

Ahora vamos a mostrar que el $l(A)$ - $l(B)$ Walk $W = P_0Q_0R_0P_1Q_1R_1\dots P_{p-1}Q_{p-1}R_{p-1}$ contiene un $l(A)$ - $l(B)$ camino tal que $l(B) \in GReach_{dr, 2dr}[G, \sigma, l(A)]$.

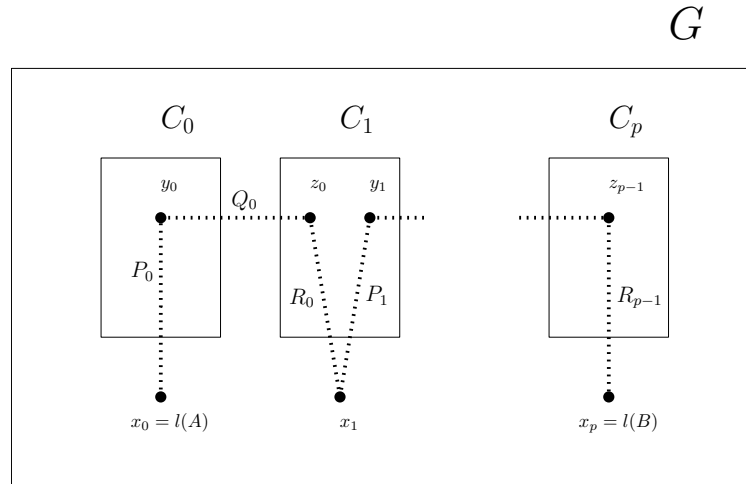


Figura 1: Esquema del Walk

Notar que W contiene un $l(A)$ - $l(B)$ camino F de largo a lo más $(\lfloor \frac{d}{2} \rfloor + d + \lfloor \frac{d}{2} \rfloor)p \leq 2dp \leq 2dr$, lo que nos da la primera condición para ver que $l(B) \in GReach_{dr, 2dr}[G, \sigma, l(A)]$. Veamos las otras dos condiciones.

Sea $i \in [0..p - 1]$. Dado que se tiene que $C_i \geq_\tau B$ entonces se tiene que $l(B) <_\sigma x_i$. Vamos a analizar la parte de F que se encuentra contenido en $P_i Q_i R_i$. Notar que en P_i todos los vértices de dicho camino son mayores que x_i . Luego en Q_i se tiene que en la primera mitad del camino los vértices son mayores que x_i , pero en la segunda mitad son mayores o iguales que x_{i+1} , no a x_i necesariamente, por lo que en Q_i podrían haber a lo más $\lfloor \frac{d}{2} \rfloor$ hops correspondientes al camino F . Luego en R_i se tiene que todos los vértices son mayores que x_{i+1} pero podrían haber tantos hops como vértices por lo que en R_i podrían haber a lo más $\lfloor \frac{d}{2} \rfloor$ hops correspondientes al camino F . De esto podemos concluir que la parte del camino F contenida en $P_i Q_i R_i$ todos los vértices son mayores que alguno de entre x_i o x_{i+1} y que hay a lo más $2 \lfloor \frac{d}{2} \rfloor$ saltos. Esto implica que $l(B)$ es el menor vértice en F y que F tiene a lo más $2 \lfloor \frac{d}{2} \rfloor p \leq dr$ hops. En conclusión $l(B) \in GReach_{dr, 2dr}[G, \sigma, l(A)]$.

Dado que para cada $B \in WReach_r[G^d/\mathcal{X}, \tau, A]$ se tiene un $l(B)$ diferente se puede concluir la desigualdad buscada. ■

Observación: Si d es un número impar entonces la desigualdad del teorema es

$$wcol_r(G^d/\mathcal{X}) \leq gcol_{(d-1)r, (2d-1)r}(G, \sigma).$$