



ÁRBOLES GENERADORES CON SUCESIONES DE GRADOS DADAS

Tesis que presenta
Maria Elena Martinez Cuero
Para obtener el grado de
Doctora en Ciencias (Matemáticas)

Director de tesis: **Dr. Eduardo Rivera Campo**

Jurado calificador:

Presidente: Dr. Eduardo Rivera Campo

Secretario: Dra. Rita Esther Zuazua Vega

Vocal: Dra. Mucuy-Kak del Carmen Guevara Aguirre

Vocal: Dr. Joaquín Tey Carrera

Vocal: Dr. Ilan Abraham Goldfeder Ortiz

Ciudad de México a 08 de septiembre de 2023.

Agradecimientos

Quiero dar las gracias a la Universidad Autónoma Metropolitana por siempre mantener las puertas abiertas y brindarme el espacio para estudiar un posgrado y a CONACYT por el respaldo dado.

Gracias a Hilarión, Ofe, Julie y Jorgito, quienes son parte de mi familia. Su apoyo y cariño me han ayudado a mantener la motivación para lograr mis metas y sueños. A mis amigos Julio, Jessica e Ismael por su compañía y apoyo.

Agradezco de manera sincera a las Dras. Rita Esther Zuazua Vega y Mucuy-Kak del Carmen Guevara Aguirre y a los Drs. Joaquín Tey Carrera e Ilan Abraham Goldfeder Ortiz por sus sugerencias invaluable. Su participación como sinodales se refleja de manera clara en el mejoramiento de la tesis.

De manera especial doy las gracias al Dr. Eduardo Rivera Campo por ser mi director de tesis, a quien aprecio, respeto y admiro. Sin su apoyo, la investigación no habría sido posible.

Para Terry, quien en polvo de estrella se volvió.
Para Coronita, Hashito y Rominita.

Índice general

Resumen	6
Introducción	7
1. Preliminares	9
1.1. Definiciones	9
1.2. Operación de Adopción	11
2. Existencia	13
2.1. Condición necesaria	15
2.2. Condición suficiente	16
2.3. Complejidad	19
3. Gráfica de $\pi(\sigma)$-árboles	25
3.1. Transformación φ	26
3.2. Gráfica de $\pi(\sigma)$ -árboles	26
3.3. Algoritmos	33
3.3.1. Algoritmo Mixto	33
3.3.2. Algoritmo Simple	44
3.4. Cotas	60
Conclusiones	65
Trabajo a futuro	66
Bibliografía	67

Resumen

Un árbol generador T de una gráfica G es una subgráfica de G tal que T es un árbol y el conjunto de vértices del árbol T es igual al conjunto de vértices de G .

En este trabajo damos una condición necesaria y una condición suficiente para que una gráfica simple tenga al menos un árbol generador con sucesión de grados preestablecida. También demostramos que el problema de decidir si una gráfica tiene un árbol generador con cierta sucesión de grados es NP -completo.

Definimos una transformación φ , basada en la Operación de Adopción dada por Fekete *et al.* [9], que convierte un árbol generador T de una gráfica simple G en cualquier otro árbol generador de G cuya sucesión de grados es una permutación de la sucesión de grados de G .

Dado un árbol T con sucesión de grados σ definimos una gráfica $G_{\pi(\sigma)}$ cuyos vértices son todos los árboles que tienen como sucesión de grados a una permutación de σ y en la cual dos árboles R y S son adyacentes si uno se obtiene del otro usando la transformación φ . Demostramos que para toda sucesión arbórea σ , la gráfica $G_{\pi(\sigma)}$ es conexa y damos dos algoritmos que encuentran una trayectoria entre cualquier par de vértices de una gráfica $G_{\pi(\sigma)}$. Finalmente, usando nuestros algoritmos, acotamos el diámetro de las gráficas $G_{\pi(\sigma)}$.

Introducción

Uno de los resultados básicos de la teoría de las gráficas es la caracterización de las gráficas conexas: una gráfica es conexa si y sólo si contiene un árbol generador.

Existen muchos algoritmos eficientes (Búsqueda en Anchura, Búsqueda en Profundidad, entre otros) que sirven para decidir si una gráfica G contiene o no un árbol generador. Incluso en el caso en que las aristas de G tienen pesos y hay una cota para el peso total del árbol generador buscado (algoritmo de Prim, algoritmo de Kruskal, etc).

Los árboles generadores de una gráfica pueden ser de muy diversas formas: desde trayectorias hasta estrellas sin que la existencia de unos implique la existencia de otros para determinada gráfica. Por ejemplo, una gráfica conexa puede no contener una trayectoria generadora. Otra gráfica podría contener una trayectoria generadora pero no contener un árbol generador cuyos vértices tengan grado 1 o 3. En este trabajo estudiamos condiciones para la existencia y propiedades de árboles generadores con determinadas características.

En el capítulo 1 presentamos algunos conceptos básicos de Teoría de Gráficas que serán utilizados en el resto de la tesis.

En el capítulo 2 damos una condición necesaria y una condición suficiente para que una gráfica tenga un árbol generador cuyos vértices v_1, v_2, \dots, v_n tengan grados d_1, d_2, \dots, d_n para una sucesión de enteros positivos dada.

También en ese capítulo demostramos que el problema de decidir si una gráfica G tiene o no un árbol generador con cierta sucesión de grados es un problema NP -completo.

En el capítulo 3 estudiamos el conjunto \mathcal{T}_σ de árboles generadores de una gráfica completa etiquetada K_n cuyos vértices v_1, v_2, \dots, v_n tienen grados $\pi(d_1), \pi(d_2), \dots, \pi(d_n)$, respectivamente, en donde π es una permutación de la sucesión de los grados $\sigma = d_1, d_2, \dots, d_n$ de algún árbol generador de K_n . Introducimos una transformación φ que aplicada a un árbol T con n vértices produce un árbol $\varphi(T)$ cuya sucesión de grados es una permutación de la sucesión de grados de T . Demostramos que aplicando la transformación φ de manera sucesiva podemos transformar un árbol $T \in \mathcal{T}_\sigma$ en cualquier otro $T' \in \mathcal{T}_\sigma$. Además damos una cota para el número de veces que es necesario aplicar la transformación φ para ir del árbol T al árbol T' .

Capítulo 1

Preliminares

En este capítulo daremos algunos conceptos básicos que se encuentran dentro de la literatura de Teoría de Gráficas [4] y [23]. También definiremos algunos conceptos nuevos que son necesarios para el desarrollo de los capítulos 2 y 3.

1.1. Definiciones

Una *gráfica simple* G consta de una pareja de conjuntos finitos (V, E) , donde V es un conjunto no vacío de elementos llamados *vértices* y E es un conjunto de pares no ordenados de elementos de V llamados *aristas*. En este trabajo llamamos *gráficas* a las gráficas simples. Una gráfica se puede representar en el plano a través de un dibujo, donde sus vértices son puntos y las aristas son arcos de curvas simples cuyos extremos son los vértices correspondientes. En la figura 1 podemos ver la representación de una gráfica G , donde el conjunto de vértices es $V(G) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ y el conjunto de aristas es $E(G) = \{v_1v_2, v_1v_3, v_1v_4, v_2v_4, v_2v_3, v_3v_4, v_5v_6, v_5v_7, v_6v_7\}$.

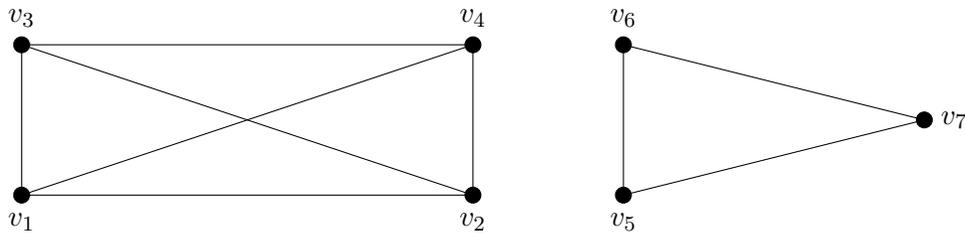


Figura 1: Gráfica simple G

El *grado* $d_G(v)$ de un vértice v en una gráfica simple G es el número de aristas de G que inciden en el vértice v . Si no existe confusión escribiremos $d(v)$ en lugar de $d_G(v)$.

Una *trayectoria* en una gráfica simple G es una sucesión $u = v_1, v_2, \dots, v_k = v$ de vértices distintos de G junto con las aristas $v_i v_{i+1}$ para $i = 1, 2, \dots, k - 1$. Una trayectoria de G es *trayectoria hamiltoniana* si incluye a todos los vértices de G . Un *ciclo* en G es la trayectoria antes descrita junto con la arista vu .

Una gráfica G es *conexa* si para cada par de vértices u, v de G la gráfica tiene una trayectoria con extremos u y v . Observemos que la gráfica de la figura 1 no es conexa, ya que no existe una trayectoria que tenga como extremos a los vértices v_1 y v_7 . En el presente texto solo vamos a trabajar con gráficas conexas, en caso contrario se aclarará la situación.

Un *árbol* T es una gráfica conexa y acíclica. Un *árbol generador* T de una gráfica G es una subgráfica de G tal que T es un árbol y $V(T) = V(G)$.

Una sucesión de enteros positivos d_1, d_2, \dots, d_n es una *sucesión gráfica* si existe una gráfica simple G con vértices v_1, v_2, \dots, v_n tal que $d_G(v_i) = d_i$ para $i = 1, 2, \dots, n$. P. Erdős y T. Gallai [8] demostraron que una sucesión de enteros positivos d_1, d_2, \dots, d_n , con $d_1 \geq d_2 \geq \dots \geq d_n$ es sucesión gráfica si y sólo si $\sum_{i=1}^n d_i$ es par y

$$\sum_{i=1}^k d_i \leq k(k-1) + \sum_{i=k+1}^n \min\{k, d_i\} \text{ para } 1 \leq k \leq n.$$

Una sucesión gráfica d_1, d_2, \dots, d_n es *sucesión arbórea* si existe un árbol con n vértices cuyos grados son d_1, d_2, \dots, d_n . Es bien conocido que $\sigma = d_1, d_2, \dots, d_n$ es una sucesión arbórea sí y solo sí $\sum_{j=1}^n d_j = 2(n-1)$. Ver, por ejemplo el libro de J. A. Bondy y U. S. R. Murty [4], página 27, ejercicio 2.1.10.

Sean T un árbol con $V(T) = \{v_1, v_2, \dots, v_n\}$ y $\sigma = d_1, d_2, \dots, d_n$ una sucesión arbórea. T es un σ -árbol si $d_T(v_i) = d_i$, para $i = 1, 2, \dots, n$. Denotamos por \mathcal{T}_σ al conjunto de todos los $\pi(\sigma)$ -árboles generadores de K_n para alguna permutación $\pi(\sigma) = \pi(d_1), \pi(d_2), \dots, \pi(d_n)$ de σ . En la figura 2 podemos observar al conjunto de todos los $\pi(\sigma)$ -árboles generadores de K_4 considerando $\sigma = 3, 1, 1, 1$.

Sea T un árbol. Si se selecciona un vértice en T como su vértice raíz r , entonces T es un árbol con raíz r . Un *árbol con raíz dirigido* es un árbol con raíz cuyas aristas están orientadas de la raíz hacia afuera.

Considerando la definición anterior, sean T un árbol con raíz r dirigido y $x \neq r$ un vértice de T . Notemos que para x existe un único vértice $p(x)$ adyacente a x en T tal que la arista $p(x)x$ está dirigida de $p(x)$ a x . El vértice $p(x)$ se considera como el *padre* de x en T y el vértice x como un *hijo* de $p(x)$. Con esta orientación de las aristas en T , si el vértice $p(x)$ está en el nivel k del árbol T , entonces el vértice x está en el nivel $k + 1$ en T . Entonces dado que cada vértice en T está en un nivel de T , el vértice raíz r de T va a estar en el nivel 0 de T , los vértices hijos de r en T van a estar en el nivel 1 de T , los vértice hijos de los vértices hijos de r en T van a estar en el nivel 2 de T y así sucesivamente.

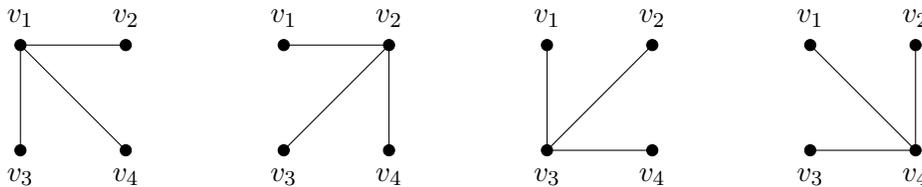


Figura 2: \mathcal{T}_σ

1.2. Operación de Adopción

Dentro de la literatura de Teoría de Gráficas encontramos la *Operación de Adopción* dada por Fekete *et al.* [9] que se define como sigue: Sea T un árbol y sean u, v y x vértices de T tal que x es adyacente a v y x no se encuentra en la trayectoria que va de u a v . Se dice que u *adopta* al vértice x si se reemplaza la arista vx por la arista ux (ver figura 3).

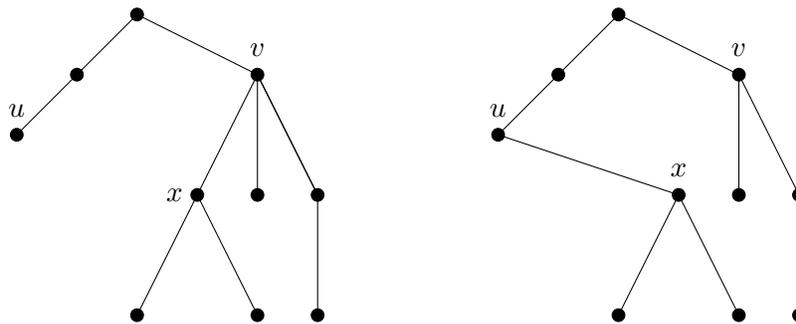


Figura 3: Operación de adopción

En el artículo de Fekete *et al.* [9] se estudió el problema de encontrar un árbol generador T de una gráfica con el menor peso posible y tal que los grados de sus vértices estén acotados por un valor entero. En mi tesis de maestría [16], utilizo la operación de adopción como parte de una heurística, eficiente en términos de tiempo de ejecución, que transforma un árbol generador T con peso mínimo de una gráfica G en un árbol generador T' de G con grados 1 o k y cuyo peso sea a lo más $k(k-1)$ veces el peso de T .

En el capítulo 3, introducimos una variante de la Operación de Adopción que llamaremos *transformación φ* , en la cual el vértice u no solo adopta a un vértice adyacente a v , sino a cada vértice adyacente a v que no se encuentra en la trayectoria que va del vértice u al vértice v . Denotamos la aplicación de la transformación a un árbol T considerando dos vértices u y v en T con $\varphi(T, u, v)$ (ver figura 4).

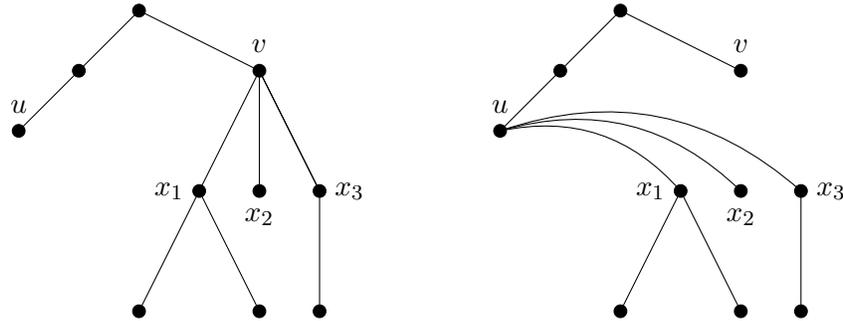


Figura 4: $\varphi(T, u, v)$

Capítulo 2

Existencia

Ø. Ore [20] dio una condición suficiente para que una gráfica G tenga una trayectoria hamiltoniana.

Teorema 1. (*Ø. Ore*). Sea G una gráfica simple con n vértices. Si $d_G(u) + d_G(v) \geq n - 1$ para toda pareja u, v de vértices de G no adyacentes, entonces G tiene una trayectoria hamiltoniana.

El Teorema de Ore ha sido generalizado por diferentes autores y en diferentes direcciones:

Un conjunto de vértices U de una gráfica G es *independiente* si ninguna arista de G tiene ambos extremos en U . En [24], S. Win dio una condición suficiente para la existencia de árboles con grado máximo acotado por un entero r en términos de la suma de grados de conjuntos de r vértices independientes.

Teorema 2. (*S. Win*). Sean G una gráfica con $n \geq 3$ vértices y r un entero con $2 \leq r \leq n - 1$. Si $d_G(u_1) + d_G(u_2) + \dots + d_G(u_r) \geq n - 1$, para todo conjunto de r vértices independientes u_1, u_2, \dots, u_r de G , entonces G tiene un árbol generador con grados acotados por r .

H. Broersma y H. Tuinstra [2] estudiaron el problema de existencia de árboles generadores con un número acotado de vértices con grado 1.

Teorema 3. (*H. Broersma, H. Tuinstra*). Sean G una gráfica con $n \geq 3$ vértices y $2 \leq s \leq n - 1$ un entero. Si $d_G(u) + d_G(v) \geq n - s + 1$ para cualquier par de vértices u, v de G no adyacentes, entonces G tiene un árbol generador con a lo más s vértices con grado 1.

Posteriormente E. Rivera-Campo [21] generalizó el teorema de Broersma y Tuinstra dando una condición suficiente para la existencia de árboles generadores en los que todos los grados y el número de vértices con grado 1 estén acotados.

Teorema 4. (*E. Rivera-Campo*). Sean n, k y d_1, d_2, \dots, d_n enteros con $1 \leq k \leq n-1$ y $2 \leq d_1 \leq d_2 \leq \dots \leq d_n \leq n-1$. Si G es una gráfica k -conexa con vértices w_1, w_2, \dots, w_n tal que $d_G(u) + d_G(v) \geq n-1 - \sum_{j=1}^k (d_j - 2)$ para cualquier par de vértices u, v de G no adyacentes, entonces G tiene un árbol generador con a lo más $2 + \sum_{j=1}^k (d_j - 2)$ vértices con grado 1 y tal que $d_T(w_i) \leq d_i$ para $i = 1, 2, \dots, n$.

El número de independencia $\alpha(G)$ de una gráfica G es el tamaño del conjunto de vértices independiente más grande de G .

V. Chvátal y P. Erdős [5] demostraron que si una gráfica k -conexa G tiene número de independencia $\alpha(G) \leq k+1$, entonces G tiene una trayectoria hamiltoniana. V. Neumann-Lara y E. Rivera-Campo [19] generalizaron este resultado al demostrar que si $\alpha(G) \leq 1 + k(r-1)$ para un entero $r \geq 2$, entonces G tiene un árbol generador con grado máximo $\Delta(G) \leq r$.

Sea $\sigma = d_1, d_2, \dots, d_n$ una sucesión arbórea. Denotamos con $s_1(\sigma)$ y $s_2(\sigma)$ al número de términos de σ iguales a 1 y a 2, respectivamente. H. Enomoto y K. Ozeki [7] presentaron la siguiente conjetura, misma que probaron en el caso $s_1(\sigma) + s_2(\sigma) \leq k+1$.

Conjetura 1. (*H. Enomoto y K. Ozeki*). Sean k un entero positivo, G una gráfica k -conexa con n vértices v_1, v_2, \dots, v_n y $\sigma = d_1, d_2, \dots, d_n$ una sucesión arbórea. Si $\alpha(G) \leq 1 + \min\{\sum_{j=1}^k (d_{i_j} - 1) : i_j \in \{1, 2, \dots, n\}\}$, entonces G tiene un σ -árbol generador.

Una *componente* de una gráfica G es una subgráfica de G conexa máxima. El número de componentes de G se denota por $\omega(G)$.

Una condición necesaria para que una gráfica tenga un ciclo hamiltoniano está dada por el siguiente teorema. Ver, por ejemplo, J. A. Bondy y U. S. R. Murty [4].

Teorema 5. Si una gráfica simple G tiene un ciclo hamiltoniano, entonces $w(G-S) \leq |S|$ para cualquier subconjunto propio no vacío S de vértices de G .

De manera análoga se tiene la siguiente condición necesaria para que una gráfica G tenga una trayectoria hamiltoniana.

Teorema 6. Si una gráfica simple G tiene una trayectoria hamiltoniana, entonces $w(G-S) \leq 1 + |S|$ para cualquier subconjunto propio no vacío S de vértices de G .

Sean $n \geq 2$ un entero y $\sigma = d_1, d_2, \dots, d_n$ una sucesión arbórea. A continuación damos una condición necesaria y una condición suficiente para que una gráfica G con n vértices v_1, v_2, \dots, v_n tenga un árbol generador T con $d_T(v_i) = d_i$ para $i = 1, 2, \dots, n$. Nuestros teoremas 7 y 8 generalizan los teoremas 6 y 1, respectivamente.

2.1. Condición necesaria

El siguiente resultado generaliza el teorema 6 para dar una condición necesaria para la existencia de un árbol generador con sucesión de grados σ .

Teorema 7. *Sean G un gráfica simple con n vértices y S un subconjunto de vértices de G . Si G tiene un árbol generador con grado máximo r , entonces $w(G-S) \leq 1+(r-1)|S|$.*

Demostración. Sea T un árbol generador con grado máximo r . La prueba la realizaremos demostrando las siguientes desigualdades:

$$w(G-S) \leq w(T-S) \leq 1+(r-1)|S|.$$

Dado que T es un árbol generador de G , entonces $T-S$ es una subgráfica de $G-S$, donde $V(T-S) = V(G-S)$ y $E(T-S) \subseteq E(G-S)$.

Observemos que si $E(G-S) \setminus E(T-S) = \emptyset$, entonces $G-S = T-S$. Por lo tanto $w(G-S) = w(T-S)$.

Suponiendo que $E(G-S) \setminus E(T-S) \neq \emptyset$, notemos que $G-S$ se obtiene de $T-S$, añadiendo una a una las aristas de $G-S$ que no están en $T-S$. Observemos que en cada paso, el número de componentes se mantiene o se reduce en una unidad, dependiendo de si la arista añadida tiene sus vértices extremos en una misma componente o en componentes distintas. Por lo tanto, $w(G-S) \leq w(T-S)$.

La segunda desigualdad se va a demostrar por inducción sobre el número de vértices del conjunto S .

Si $S = \{v\}$, entonces $w(T-v) = d_T(v) \leq r = 1+(r-1)$. Supongamos que $|S| \geq 2$. Sea P la trayectoria más larga en T , tal que sus vértices extremos pertenecen al conjunto S . Sea a uno de los vértices extremos de P . Notemos que en $T-\{a\}$ una de sus componentes C contiene todos los vértices de $S' = S-\{a\}$. Entonces, por hipótesis de inducción $w(C-S') \leq 1+(r'-1)|S'|$. Dado que $d_C(x) \leq r' \leq d_T(x) \leq r$ para todo x en S' , entonces

$$w(T-S) = (d_T(a) - 1) + w(C-S') \leq (d_T(a) - 1) + 1 + (r' - 1)|S'| \leq$$

$$(r - 1) + 1 + (r - 1)|S'| = 1 + (r - 1)(|S'| + 1) = 1 + (r - 1)|S|.$$

Por lo tanto,

$$w(T-S) \leq 1+(r-1)|S|.$$

□

2.2. Condición suficiente

A continuación presentamos una condición suficiente para garantizar la existencia de árboles generadores con sucesión de grados σ . Nuestro resultado [17] generaliza el teorema de Ore.

Teorema 8. (*M. E. Martínez-Cuero, E. Rivera-Campo*) Sean $r \geq 2$ un entero y G una gráfica etiquetada con conjunto de vértices $V(G) = \{v_1, v_2, \dots, v_n\}$ con $n \geq r + 1$. Si $d(u) + d(v) \geq \frac{(2r-3)n-(2r-5)}{r-1}$ para cada par u, v en $V(G)$ de vértices no adyacentes, entonces G tiene un árbol generador T con una sucesión de grados σ para cada sucesión arbórea $\sigma = d_1, d_2, \dots, d_n$ con $1 \leq d_i \leq r$ con $i = 1, 2, \dots, n$.

Para cada entero positivo k , sean $X_k = \{x_1, x_2, \dots, x_k\}$, $Y_k = \{y_1, y_2, \dots, y_k\}$ y $Z_{k,r} = \{z_1, z_2, \dots, z_{2k(r-2)+2}\}$ conjuntos de vértices disjuntos por pares y sea $G_{k,r}$ la gráfica completa con el conjunto de vértices $X_k \cup Y_k \cup Z_{k,r}$ con todas las arista $x_i y_i$, $1 \leq i, j \leq k$, removidas. Para el caso $k = 2$ y $r = 3$ (ver figura 5).

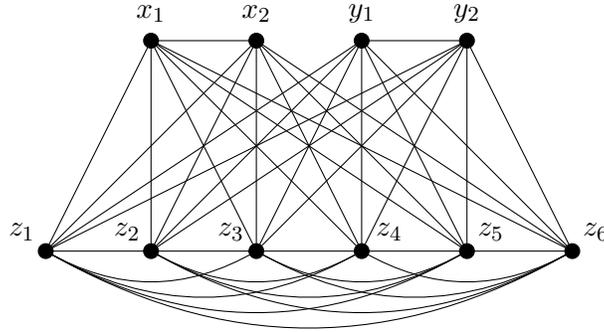


Figura 5: Gráfica $G_{2,3}$

Afirmamos que la gráfica $G_{k,r}$ no tiene un árbol generador T tal que $d_T(x_i) = d_T(y_i) = r$, $i = 1, 2, \dots, k$ y $d_T(z_j) = 1$, $j = 1, 2, \dots, 2k(r-2) + 2$; porque si T es tal árbol, entonces $T - Z_{k,r}$ sería un árbol generador de la subgráfica $G[X_k \cup Y_k]$ de $G_{k,r}$, inducido por el conjunto $X_k \cup Y_k$, lo cual no es posible ya que $G[X_k \cup Y_k]$ no es conexas. Por otro lado, si u y v no son vértices adyacentes en $G_{k,r}$, sin pérdida de generalidad asumimos que $u \in X_k$ y $v \in Y_k$. Por otro lado, dado que:

$$n = |G_{k,r}| = |X_k| + |Y_k| + |Z_{k,r}| = 2k(r-1) + 2, \text{ entonces } k = \frac{n-2}{2(r-1)} \text{ y}$$

$$d_{G_{k,r}}(u) + d_{G_{k,r}}(v) = d_{X_k}(u) + d_{Z_{k,r}}(u) + d_{Y_k}(v) + d_{Z_{k,r}}(v) = 2((k-1) + 2k(r-2) + 2).$$

Entonces, $d_{G_{k,r}}(u) + d_{G_{k,r}}(v) = \frac{(2r-3)n-(2r-4)}{r-1}$. Esto muestra que la condición de la suma de los grados en el teorema 8 es justa.

A continuación se demuestra el teorema 8.

Demostración del teorema 8. Suponemos que el resultado es falso. Entonces para cierto entero $n \geq 4$ y cierta sucesión arbórea $\sigma = d_1, d_2, \dots, d_n$ con $1 \leq d_i \leq r$, $i = 1, 2, \dots, n$, existe un contraejemplo. Sea G una gráfica etiquetada con el conjunto de vértices $V(G) = \{w_1, w_2, \dots, w_n\}$ tal que G no tiene un árbol generador con la sucesión de grados σ y tal que $d(u) + d(v) \geq \frac{(2r-3)n-(2r-5)}{r-1}$ para cada par u, v de vértices no adyacentes en G . Elegimos a G con el mayor número posible de aristas sin dejar de ser un contraejemplo con el mismo número de vértices. Observemos que G no puede ser completa ya que no sería un contraejemplo.

Sean u, v vértices no adyacentes de G . Por la elección de G , la gráfica $G + uv$ tiene un árbol generador T con la sucesión de grados σ . Por lo tanto G contiene un bosque generador $F = T - uv$ con exactamente dos componentes T_u y T_v con $u \in V(T_u)$ y $v \in V(T_v)$ tales que $d_F(u) = d_i - 1$, $d_F(v) = d_j - 1$ y $d_F(w_r) = d_r$ para cada vértice w_r diferente a los vértices u y v , donde i y j son $u = w_i$ y $v = w_j$.

Si consideramos u como el vértice raíz de T_u y a v como el vértice raíz de T_v , orientamos las aristas de F de tal forma que T_u and T_v se conviertan en árboles dirigidos \vec{T}_u y \vec{T}_v de su raíz hacia afuera, respectivamente. Para cada vértice w diferente a los vértices u y v , sea w^- el único vértice de G tal que la arista w^-w es orientada de w^- a w en \vec{F} . Sea:

$$A_u = \{y^- \in V(T_u) : uy \in E(G)\} \text{ y } B_u = \{x \in V(T_u) : vx \in E(G)\},$$

$$A_v = \{y^- \in V(T_v) : vy \in E(G)\} \text{ y } B_v = \{x \in V(T_v) : ux \in E(G)\}.$$

Notemos que $|A_u \cap B_u| = 0$, ya que de lo contrario existe $z^- \in A_u \cap B_u$ tal que al hacer $(F - z^-z) + \{uz, vz^-\}$ se obtiene un árbol generador de G con sucesión de grados σ , lo cual es una contradicción (ver figura 6). Análogamente $|A_v \cap B_v| = 0$ y por lo tanto

$$|A_u| + |B_u| = |A_u \cup B_u| \leq n_u \text{ y } |A_v| + |B_v| = |A_v \cup B_v| \leq n_v,$$

donde n_u y n_v son el número de vértices de T_u y T_v , respectivamente.

Para cada vértice x de G , abusando de la notación, denotamos $d_u(x)$ y $d_v(x)$ el número de vértices en T_u y T_v , respectivamente, los cuales son adyacentes a x en G .

Claramente

$$|B_u| = d_u(v) \text{ y } |B_v| = d_v(u).$$

Puesto que $d_F(w_r) = d_r \leq r$ y que por cada vértice y^- que pertenece al conjunto A_u hay al menos una arista uy en $E(G)$ y a lo más $r - 1$ aristas uy en $E(G)$, tenemos $|A_u| \leq d_u(u) \leq |A_u|(r - 1)$. Por lo cual

$$|A_u| \geq \frac{d_u(u)}{r - 1}.$$

Análogamente

$$|A_v| \geq \frac{d_v(v)}{r - 1}.$$

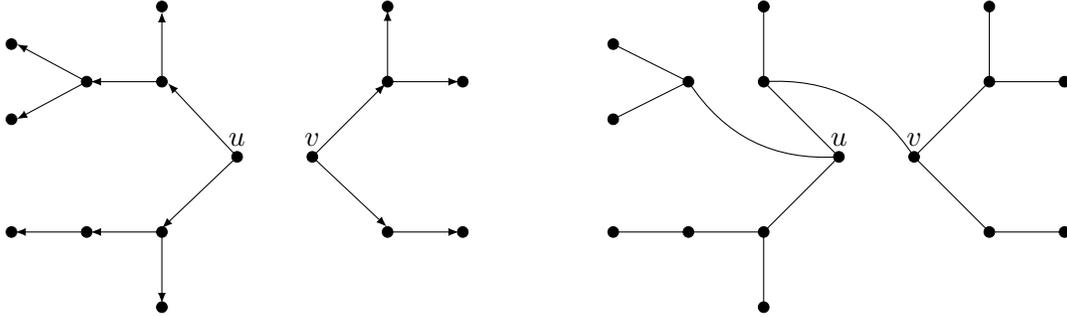


Figura 6: Bosque F con $z^- \in A_u \cap B_u$ (izquierda). Árbol $(F - z^-z) + \{uz, vz^-\}$ (derecha).

Dado que el ex-grado de cada vértice \vec{F} es a lo más $r - 1$, entonces

$$\frac{d_u(u)}{r - 1} + d_u(v) \leq |A_u| + |B_u| \leq n_u \text{ y } \frac{d_v(v)}{r - 1} + d_v(u) \leq |A_v| + |B_v| \leq n_v.$$

Por lo tanto

$$d_u(u) + (r - 1)d_u(v) \leq (r - 1)n_u \text{ y } d_v(v) + (r - 1)d_v(u) \leq (r - 1)n_v.$$

Dado que u y v no son adyacentes a sí mismos en G , $d_u(u) \leq n_u - 1$ y $d_v(v) \leq n_v - 1$. Por lo tanto

$$(r - 2)d_u(u) \leq (r - 2)(n_u - 1) \text{ y } (r - 2)d_v(v) \leq (r - 2)(n_v - 1)$$

Entonces, al sumar $d_u(u) + (r - 1)d_u(v) \leq (r - 1)n_u$ y $(r - 2)d_u(u) \leq (r - 2)(n_u - 1)$, obtenemos:

$$(r - 1)d_u(u) + (r - 1)d_u(v) \leq (2r - 3)n_u - (r - 2)$$

Por otro lado, al sumar $d_v(v) + (r-1)d_v(u) \leq (r-1)n_v$ y $(r-2)d_v(v) \leq (r-2)(n_v-1)$, obtenemos:

$$(r-1)d_v(v) + (r-1)d_v(u) \leq (2r-3)n_v - (r-2)$$

Entonces, al sumar las últimas dos desigualdades obtenemos:

$$(r-1)d_u(u) + (r-1)d_v(u) + (r-1)d_u(v) + (r-1)d_v(v) \leq n_u(2r-3) + n_v(2r-3) - (r-2) - (r-2)$$

$$(r-1)(d_u(u) + d_v(u)) + (r-1)(d_u(v) + d_v(v)) \leq (2r-3)(n_u + n_v) - 2(r-2)$$

$$(r-1)d(u) + (r-1)d(v) \leq (2r-3)n - (2r-4)$$

$$(r-1)(d(u) + d(v)) \leq (2r-3)n - (2r-4)$$

$$d(u) + d(v) \leq \frac{(2r-3)n - (2r-4)}{r-1}.$$

Lo cual no es posible dado que $d(u) + d(v) \geq \frac{(2r-3)n - (2r-5)}{r-1}$.

□

2.3. Complejidad

Un *problema de decisión* es un problema cuya respuesta es sí o no. Ejemplos de estos problemas son, entre muchos otros, el de decidir si una gráfica es conexa y el de decidir si una gráfica tiene una trayectoria hamiltoniana.

Un problema de decisión pertenece a la clase P si existe un polinomio $f(n)$ y un algoritmo que resuelve este problema para cualquier instancia de tamaño n en un número de pasos acotado por $f(n)$. Por otro lado, un problema de decisión está en la clase NP si existe un algoritmo que, cuando la respuesta es positiva, comprueba en tiempo polinomial que la solución propuesta es correcta. Existen problemas de decisión que pertenecen a la clase NP y que no se sabe si pertenecen o no a la clase P . Un ejemplo es el problema de decidir si una gráfica tiene una trayectoria hamiltoniana.

Finalmente, un problema de decisión es *NP-completo* si está en la clase NP y cualquier otro problema en la clase NP puede reducirse a este en tiempo polinomial.

En esta sección demostramos que el problema de decidir si una gráfica tiene un σ -árbol generador es NP -completo.

Teorema 9. *Sean $n \geq 4$ un número entero y $\sigma = d_1, d_2, \dots, d_n$ una sucesión de grados arbórea. El problema de decidir si una gráfica etiquetada G con conjunto de vértices $V(G) = \{w_1, w_2, \dots, w_n\}$ tiene un σ -árbol generador es NP -completo.*

Para demostrar que el teorema anterior es cierto, utilizaremos la siguiente estrategia usual:

- (I) Mostrar que el problema de decidir si G tiene un σ -árbol generador está en la clase NP .
- (II) Bajo el conocimiento de que el problema de decidir si una gráfica G^* tiene una trayectoria hamiltoniana entre dos vértices dados pertenece a la clase NP -completa, mostrar que este problema se reduce de manera polinomial al problema de decidir si cierta gráfica G tiene un σ -árbol para cierta sucesión arbórea σ .

A continuación demostramos el punto (I).

Demostración. Sea G una gráfica etiquetada cuyo conjunto de vertices es $V(G) = \{w_1, w_2, \dots, w_n\}$ y sean $\sigma = d_1, d_2, \dots, d_n$ una sucesión arbórea y T una propuesta como solución al problema de decidir si G tiene un σ -árbol generador está en la clase NP .

Para demostrar este punto, daremos un algoritmo con complejidad polinomial, basado en el algoritmo de Búsqueda en Profundidad (DFS), al cual llamamos **DFS-MODIFICADO**.

El algoritmo DFS-MODIFICADO determina en un tiempo polinomial si T es un σ -árbol generador de G . Al igual que en el algoritmo DFS, a partir de un vértice arbitrario v se visita uno de sus vecinos y este proceso se vuelve a repetir recursivamente con cada vecino y se va formando una trayectoria que va creciendo conforme se van visitando vértices. Cuando ya no es posible que la trayectoria siga creciendo, entonces se repite el proceso con cada uno de los vértices vecinos no visitados del último vértice en la trayectoria visitado. El proceso continua de esta manera hasta que se termina de visitar a todos los vértices que pueden ser alcanzados desde v . DFS-MODIFICADO se detiene si en algún momento del proceso se visita un vértice por segunda vez, ya que esto implica que T tiene al menos un ciclo y por lo tanto T no es solución.

Si el número total de vértices visitados es $n = |V(G)|$, entonces T es un árbol generador de G . En caso contrario se determina que T no es solución, dado que T es un bosque.

Algoritmo 1: DFS-MODIFICADO

```

entrada:  $T, \sigma$ 
salida : “ $T$  es una solución” o “ $T$  no es solución”
1 para todo vértice  $w \in V(T)$  hacer
2   |  $estado[w] \leftarrow$  “no visitado”;
3   |  $p[w] \leftarrow$  “nulo”;
4 fin
5  $terminado = 0$ ;
6  $esValido \leftarrow$  DFS-VISITADO( $T, w$ );
7 si  $esValido = verdadero$  y  $terminado = n$  entonces
8   | para todo vértice  $w_i \in V(T)$  hacer
9     | si  $d_T(w_i) \neq \sigma[i]$  entonces
10    |   | Responder: “ $T$  no es una solución”;
11    |   | Salir;
12    | fin
13  | fin
14  | Responder: “ $T$  es una solución”;
15 en otro caso
16 | Responder: “ $T$  no es solución”;
17 fin

```

Si resulta que T es un árbol generador de G , se verifica si $d_T(w_i) = d_i$, $1 \leq i \leq n$. En el caso que la igualdad anterior no sea cierta para al menos un vértice w_i en $V(T)$, entonces se determina que T no es solución, de lo contrario se puede decir que T sí es una solución para el problema de decisión.

El algoritmo DFS-MODIFICADO recibe como entradas a T y $\sigma = [d_1, d_2, \dots, d_n]$ y se definen $Adj[w]$ como el campo que contiene a todos los vértices vecinos de w en T y $p[w]$ como el campo que almacena al antecesor de w . En la parte de la inicialización del algoritmo se considera el estado de cada uno de los vértices w de T como “no visitado”, mediante la asignación de “no visitado” a la variable $estado[w]$ y se asigna al antecesor de w $p[w]$ como “nulo” y se establece el contador global $terminado$ (ver las líneas 1 – 5). Posteriormente a la variable $esValido$ se le asigna la salida de la subrutina DFS-VISITADO (ver la línea 7). Si la salida de la subrutina DFS-VISITADO es igual a *verdadero* y el contador $terminado$ es igual a n , entonces para cada vértice de T se examina si $d_T(w_i)$ es distinto a d_i , en caso que ocurra esto para algún vértice w_i se concluye que T no es solución, de lo contrario se determina que T es una solución. Por otro lado, si la salida de la subrutina DFS-VISITADO es igual a *falso*, entonces se concluye que T no es solución (ver las líneas 7 – 17).

A continuación presentamos la subrutina **DFS-VISITADO**.

Algoritmo 2: DFS-VISITADO

```

entrada:  $T, w$ 
salida : falso o verdadero
1  $estado[w] \leftarrow$  “visitado”;
2 para todo vértice  $v \in Adj_T[w] \setminus \{p[w]\}$  hacer
3   si  $estado[v] =$  “no visitado” entonces
4      $p[v] \leftarrow w$ ;
5      $esValido \leftarrow$  DFS-VISITADO( $T, v$ );
6     si  $esValido = falso$  entonces
7       regresar: falso;
8     fin
9   en otro caso
10    Responder: “ $T$  no es solución, ya que tiene al menos un ciclo”;
11    regresar: falso;
12  fin
13 fin
14  $terminado = terminado + 1$ ;
15 regresar: verdadero;

```

La subrutina DFS-VISITADO recibe como entradas a T y sin pérdida de generalidad un vértice de partida w de T . En la parte de la inicialización del algoritmo se le asigna a la variable $estado[w]$ la condición de “visitado” (ver la línea 1).

Posteriormente, se hace un recorrido en profundidad a T para examinar el estado de todos los vértices alcanzables desde el vértice de partida w (ver las líneas 2 – 13). Observemos que en cada llamado de DFS-VISITADO(T, v) el vértice v inicialmente tiene la condición de “no visitado” y luego se le asigna el estado de “visitado” y se determina el antecesor de v e incrementa el contador “terminado” en una unidad (ver la línea 14).

Notemos que si en algún llamado a DFS-VISITADO(T, v) el estado del vértice v es “visitado”, entonces la subrutina regresa “falso” (ver las líneas 9 – 12). De lo contrario regresa “verdadero” (ver la línea 15). □

A continuación demostramos el punto (II).

Demostración. Sean k, m y n enteros positivos tal que $n = k + m - 2$. Dada una sucesión arbórea $\sigma = d_1, d_2, \dots, d_n$ con $d_1 = d_2 = \dots = d_k = 1$ y $d_i \geq 2$ para $k + 1 \leq i \leq n$, para cada gráfica G^* simple etiquetada con conjunto de vértices $V(G^*) = \{v_1, v_2, \dots, v_m\}$, definiremos una gráfica $G = f(G^*)$ y probaremos que satisface que G^* tiene una trayectoria hamiltoniana con extremos v_1 y v_2 si y sólo si G tiene un σ -árbol generador.

Sea $W = \{w_3, w_4, \dots, w_k\}$ un conjunto de vértices ajeno al conjunto de vértices de G^* . La gráfica $G = f(G^*)$ se obtiene de G^* como a continuación se describe:

1. Los vértices v_1 y v_2 de G^* se reetiquetan como w_1 y w_2 respectivamente.
2. Posteriormente, en forma ordenada los vértices v_i , diferentes de v_1 y v_2 en G^* se van renombrando como $w_{k+i-2} = v_i$, con $3 \leq i \leq m$, al instante en que cada uno se hace adyacente a $d_{k+(i-2)} - 2$ vértices de W , siendo w_{k+1} el primero que se hace adyacente a los primeros $d_{k+1} - 2$ elementos que se encuentran en W , luego ya sin considerar estos vértices en W , w_{k+2} se hace adyacente a los siguientes $d_{k+2} - 2$ vértices de W , posteriormente w_{k+3} se hace adyacente a los siguientes $d_{k+3} - 2$ vértices de W , sin considerar los que ya son adyacentes a w_{k+1} y w_{k+2} y esto continúa ocurriendo hasta que w_n se hace adyacente a los últimos $d_{k+m-2} - 2$ vértices de W . Observemos que $\sum_{i=3}^m d_{k+(i-2)} - 2 = (d_{k+1} - 2) + (d_{k+2} - 2) + \dots + (d_{k+(m-2)} - 2) = (d_{k+1} + d_{k+2} + \dots + d_n) - 2(n - k) = (d_1 + d_2 + \dots + d_n) - (d_1 + d_2 + \dots + d_k) - 2(n - k) = 2(n - 1) - k - 2(n - k) = k - 2 = |W|$, ya que de lo contrario no podríamos hacer la descripción anterior.

Notemos que la gráfica G tiene una subgráfica F isomorfa a G^* con $V(F) = \{w_1, w_2, w_{k+1}, \dots, w_n\} = \{v_1, v_2, v_3, \dots, v_m\} = V(G^*)$.

A continuación vamos a demostrar que si la gráfica G^* tiene una trayectoria hamiltoniana con vértice inicial v_1 y vértice final v_2 , entonces la gráfica G tiene un árbol generador T tal que $d_T(w_i) = d_i$ con $1 \leq i \leq n$.

Ya que G^* tiene una trayectoria hamiltoniana $H_{v_1 v_2}$ que inicia en v_1 y termina en v_2 , entonces la subgráfica F de G tiene una trayectoria hamiltoniana $H_{w_1 w_2}$ con vértice inicial w_1 , vértice final w_2 y vértices interiores $w_{k+1}, w_{k+2}, \dots, w_n$. Entonces la subgráfica $T = H_{w_1 w_2} \cup (E(G) \setminus E(F))$ de G es el árbol generador deseado de G .

Ahora vamos a demostrar que si la gráfica G tiene un árbol generador T tal que $d_T(w_i) = d_i$ con $1 \leq i \leq n$, entonces la gráfica G^* tiene una trayectoria hamiltoniana con vértice inicial v_1 y vértice final v_2 .

Sea T el árbol generador de G con las condiciones mencionadas anteriormente. Por la construcción de G , los vértices $\{w_3, w_4, \dots, w_{k-1}, w_k\}$ son vértices de grado uno en T , entonces al eliminar estos vértices en T , obtenemos una subgráfica conexa H de G con $V(H) = \{w_1, w_2, w_{k+1}, w_{k+2}, \dots, w_n\}$ en la que el grado de los vértices $w_{k+1}, w_{k+2}, \dots, w_n$ en H es dos, ya que de no ser así, existe al menos un vértice w_i en H con $k+1 \leq i \leq n$, tal que $d_H(w_i) \geq 3$ o $d_H(w_i) = 1$ y en consecuencia el $d_T(w_i) \geq d_i + 1$ o $d_T(w_i) = d_i - 1$, lo cual, en ambos casos, es una contradicción. Por otro lado, $d_H(w_1) = d_H(w_2) = 1$, ya que si $d_H(w_1) \geq 2$ o $d_H(w_2) \geq 2$, entonces T tendría al menos un ciclo, lo cual es una contradicción. Entonces H es una trayectoria hamiltoniana con vértices inicial w_1 y vértice final w_2 de F . Por lo tanto G^* tiene una trayectoria hamiltoniana que inicia en v_1 y termina en v_2 .

Por lo tanto, G^* con $V(G^*) = \{v_1, v_2, \dots, v_m\}$ tiene una trayectoria hamiltoniana con vértice inicial v_1 y vértice final v_2 si y solo si G con $V(G) = \{w_1, w_2, \dots, w_n\}$ tiene un árbol generador T tal que $d_T(w_i) = d_i$ con $1 \leq i \leq n$.

Finalmente, veamos que al hacer $G = f(G^*)$, $k - 2 = |V(G)| - |V(G^*)|$ es el número de vértices que se le pega a G^* y dado que cada uno de estos vértices nuevos es adyacente a un v_i de G^* con $3 \leq i \leq m$ se usan $k - 2 = |E(G)| - |E(G^*)|$ aristas para hacer esto y el número de vértices que se reetiquetan son m . Por lo cual el número de pasos que realiza la transformación f es $2(k - 2) + m \leq 2n$.

□

Gráfica de $\pi(\sigma)$ -árboles

La *gráfica de árboles* $T(G)$ de una gráfica conexa G es la gráfica cuyo conjunto de vértices es el conjunto de árboles generadores de G en la que dos árboles T y S son adyacentes si uno se obtiene del otro por medio de un intercambio simple de aristas. Es decir que existen aristas t de T y s de S tales que $S = (T - t) + s$.

La gráfica de árboles de cualquier gráfica G es una gráfica conexa. R. L. Cummins [6] demostró que si la gráfica G tiene al menos un ciclo, entonces $T(G)$ tiene un ciclo hamiltoniano. Posteriormente varios autores han estudiado diferentes variantes de la gráfica de árboles, ver por ejemplo Broersma y Li [3], Harary *et al.* [13], Heinrich y Liu [15] y Li *et al.* [18].

Una gráfica D es una *gráfica dirigida* si cada una de sus aristas tienen una orientación. Si la *gráfica subyacente* de D que resulta de ignorar el sentido de las aristas en D es conexa, entonces la gráfica dirigida D se dice que es *débilmente conexa*.

Sea D una gráfica dirigida débilmente conexa. La *gráfica de árboles dirigidos* de D es la gráfica $T_{ci}(D)$ en la que los vértices son todos los árboles generadores de D . Dos árboles T y S son adyacentes en $T_{ci}(D)$ si $S = (T - t) + s$ para ciertas flechas $t \in A(T)$ y $s \in A(S)$ tales que t y s tiene la misma orientación en el único ciclo contenido en $T \cup S$. En [10], A. P. Figueroa Gutiérrez demostró que si D tiene un ciclo dirigido, entonces $T_{ci}(D)$ es conexa. En el mismo trabajo, A. P. Figueroa Gutiérrez generalizó la gráfica de árboles dirigidos a la *gráfica de bases orientadas* de un matroide orientado.

Dada una gráfica G con $n \geq 4$ vértices un *2-switch* en G consiste en quitar dos aristas uv y xy de G y añadir dos aristas ux y yv que no pertenecen al conjunto de aristas de G . Berge [1], Hakimi [12] y Havel [14] demostraron de manera independiente

que dadas dos gráficas F y H con idéntica sucesión de grados existe una sucesión de gráficas G_0, G_1, \dots, G_r con $F = G_0$, $H = G_r$ y tal que para $i = 0, 1, \dots, r - 1$ la gráfica G_{i+1} se obtiene de G_i por medio de un 2-switch. J. Fresán Figueroa [11] demostró que si S y R son dos árboles con idéntica sucesión de grados, entonces se puede escoger la sucesión G_0, G_1, \dots, G_r de tal manera que cada gráfica G_i sea a su vez un árbol.

En este capítulo estudiamos una transformación φ que convierte un árbol T en otro árbol cuya sucesión de grados se obtiene de la sucesión de grados de T al intercambiar el grado de dos vértices. Demostramos que al aplicar sucesivamente la transformación φ siempre es posible ir de un árbol S a cualquier otro árbol R cuya sucesión de grados sea una permutación de la sucesión de grados de S .

3.1. Transformación φ

A continuación presentamos una transformación que llamamos φ , la cual está basada en la Operación de Adopción dada por Fekete *et al.* [9]. Por lo cual, cada vez que se menciona que un vértice u adopta a un vértice y , nos referimos a la manera en que se reemplazan aristas para efectuar la adopción.

Sea T un árbol y sean $u, v \in V(T)$ tales que (i) $d_T(u) = 1$ y $d_T(v) \geq 2$ o (ii) $d_T(u) \geq 2$ y $d_T(v) \geq 2$. En el caso (i) $d_T(u) = 1$ y $d_T(v) \geq 2$, la notación $\varphi(T, u, v)$ significa que el vértice u adopta a cada vértice adyacente a v que no se encuentra en la trayectoria que va de u a v (ver figura 7). En el caso (ii) $d_T(u) \geq 2$ y $d_T(v) \geq 2$, la notación $\varphi(T, u, v)$ significa que el vértice u adopta a cada vértice adyacente a v que no se encuentra en la trayectoria que va de u a v en T y posteriormente v adopta a cada vértice que originalmente era adyacente a u en T , sin considerar al vértice que se encuentra en la trayectoria que va de v a u (ver figura 8). Observemos que en ambos casos los vértices u y v intercambian grados y adyacencias, mientras que los grados y las adyacencias del resto de los vértices se mantienen. Cuando ocurra el caso (i) vamos a decir que se realizó una adopción simple y cuando ocurra el caso (ii) vamos a decir que se realizó una adopción doble.

3.2. Gráfica de $\pi(\sigma)$ -árboles

Sean K_n una gráfica completa etiquetada con conjunto de vértices $V = \{v_1, v_2, \dots, v_n\}$ y $\sigma = d_1, d_2, \dots, d_n$ una sucesión arbórea. Denotamos por Π_σ al conjunto de las permutaciones $\pi(\sigma)$ de σ y por \mathcal{T}_σ al conjunto de todos los $\pi(\sigma)$ -árboles generadores de K_n para alguna permutación $\pi(\sigma)$ de σ . Dada una sucesión arbórea σ , la *gráfica*

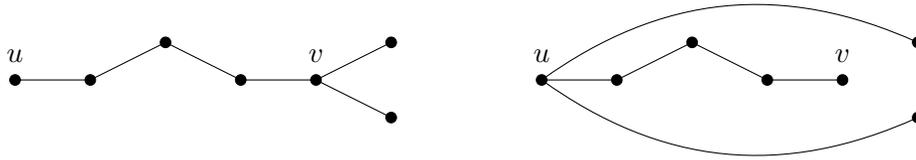


Figura 7: Adopción simple $\varphi(T, u, v)$

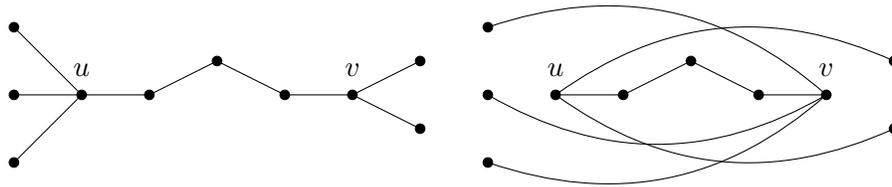


Figura 8: Adopción doble $\varphi(T, u, v)$

de $\pi(\sigma)$ -árboles es una gráfica simple, donde sus vértices son todos los $\pi(\sigma)$ -árboles con $\pi(\sigma) \in \Pi_\sigma$ en la que dos árboles T y T' son adyacentes al aplicar una sola vez la transformación φ , podemos ir de T a T' . Vamos a denotar a la gráfica de $\pi(\sigma)$ -árboles como $G_{\pi(\sigma)}$.

Consideremos a la gráfica completa K_4 con conjunto de vértices $V = \{v_1, v_2, v_3, v_4\}$ y la sucesión arbórea $\sigma = 1, 1, 1, 3$. En este caso $\Pi_\sigma = \{(3, 1, 1, 1), (1, 3, 1, 1), (1, 1, 3, 1), (1, 1, 1, 3)\}$. En la figura 9 podemos observar que la gráfica $G_{\pi(\sigma)}$ es completa.

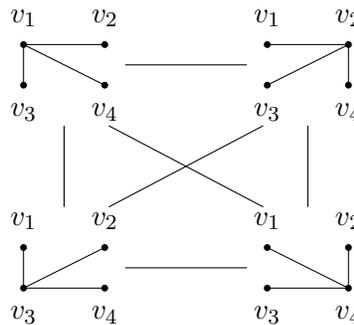


Figura 9: $G_{\pi(\sigma)}$

Por otro lado, si consideramos la misma gráfica K_4 con la sucesión arbórea $\sigma' = 1, 1, 2, 2$, entonces $\Pi_{\sigma'} = \{(1, 1, 2, 2), (1, 2, 1, 2), (1, 2, 2, 1), (2, 1, 1, 2), (2, 1, 2, 1), (2, 2, 1, 1)\}$. En este caso la gráfica $G_{\pi(\sigma')}$ tiene 12 vértices (ver figura 10). Notemos que $G_{\pi(\sigma')}$ no es completa y tiene un ciclo hamiltoniano.

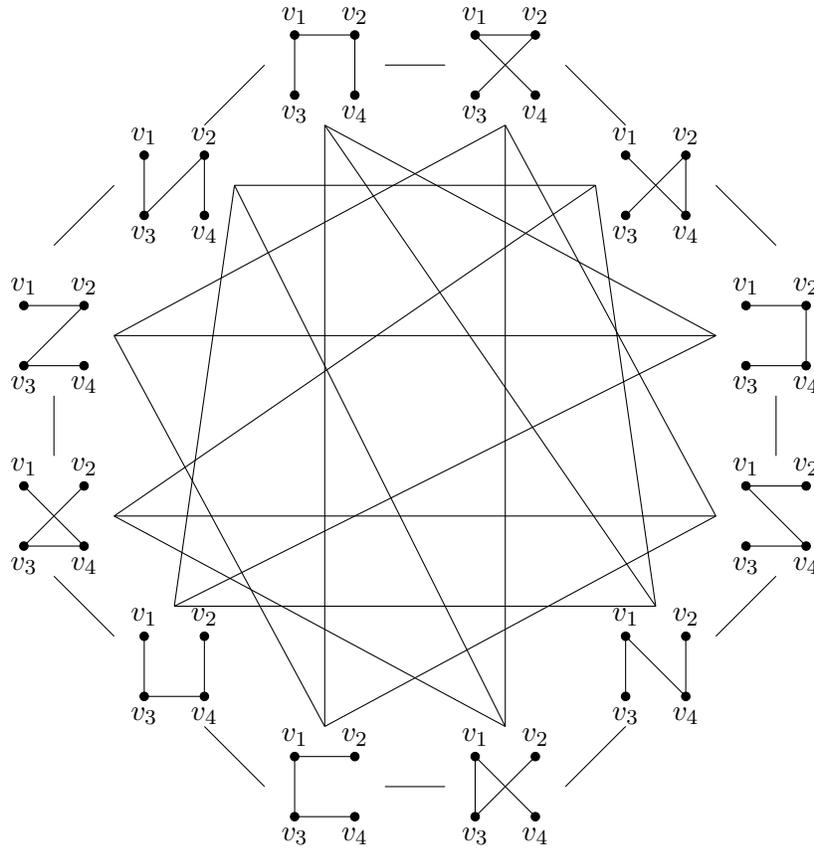


Figura 10: $G_{\pi(\sigma')}$

El principal propósito de esta sección es demostrar que la gráfica $G_{\pi(\sigma)}$ es conexa para toda sucesión arbórea σ (teorema 10) usando el siguiente lema.

Para cada sucesión arbórea $\sigma = d_1, d_2, \dots, d_n$ con $n \geq 3$ y $d_1 \geq d_2 \geq \dots \geq d_n$ definimos un árbol C_σ formado por una trayectoria $P : v_1, v_2, \dots, v_k$ junto con k conjuntos de aristas A_1, A_2, \dots, A_k , pendientes en v_1, v_2, \dots, v_k , respectivamente con $|A_1| = d_1 - 1$, $|A_i| = d_i - 2$ para $i = 2, 3, \dots, k - 1$ y $|A_k| = d_k - 1$, en donde $k = \max\{i : d_i > 1\}$. En la figura 11 podemos ver un ejemplo de un árbol C_σ , donde

$\sigma = 5, 4, 3, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1$, $P : v_1, v_2, v_3, v_4$ y $A_1 = \{v_1v_5, v_1v_6, v_1v_7, v_1v_8\}$, $A_2 = \{v_2v_9, v_2v_{10}\}$, $A_3 = \{v_3v_{11}\}$ y $A_4 = \{v_4v_{12}\}$

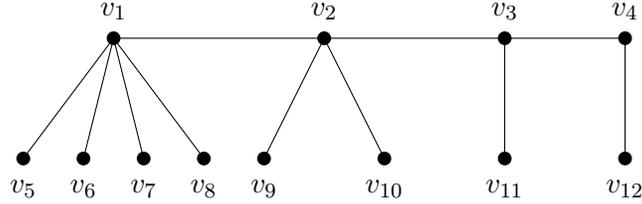


Figura 11: Árbol C_σ , con $\sigma = 5, 4, 3, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1$

Lema 1. Sea T un árbol con $n \geq 3$ vértices v_1, v_2, \dots, v_n y sucesión de grados $\sigma = d_1, d_2, \dots, d_n$ con $d_1 \geq d_2 \geq \dots \geq d_n$. Entonces T se puede convertir en un árbol isomorfo a C_σ al aplicar iteradamente transformaciones $\varphi(T, u, w)$ considerando vértices u y w distintos de v_1 .

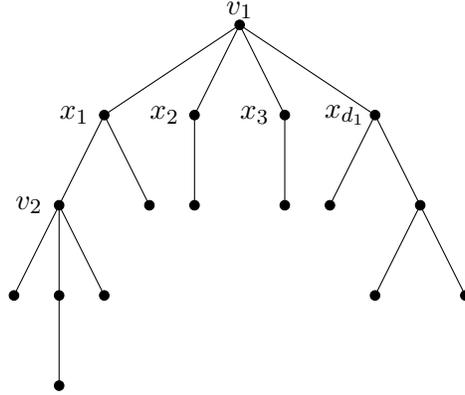
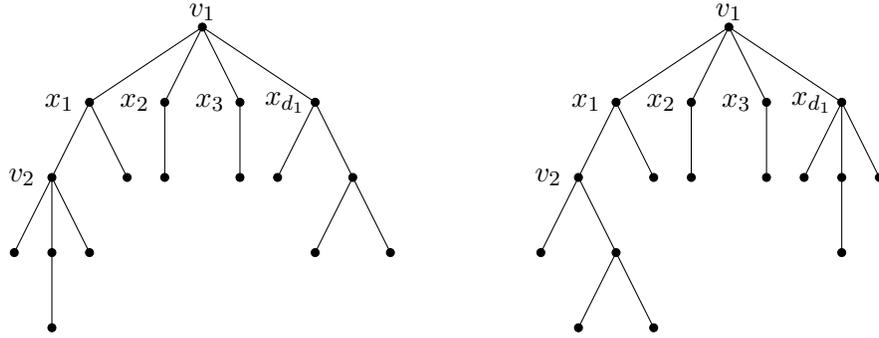
Demostración. Sea $t(\sigma) = \max\{i : d_i \geq 2\}$. Si $t(\sigma) = 1$, entonces T es una estrella por lo que es isomorfo a C_σ . Supongamos que para cierto entero $k \geq 1$ la afirmación es válida si $t(\sigma) \leq k$.

Sea T un árbol con sucesión de grados $\sigma = d_1, d_2, \dots, d_n$ con $d_1 \geq d_2 \geq \dots \geq d_n$ tal que $t(\sigma) = k + 1$. Sean x_1, x_2, \dots, x_{d_1} los vértices de T adyacentes a v_1 . Para mayor referencia consideremos a T como un árbol con raíz v_1 de grado d_1 . Sin perder generalidad suponemos que v_2 está en la rama de T que contiene a x_1 (ver figura 12).

Sea T_0 el árbol obtenido de T al aplicar $\varphi(T, v_2, x_{d_1})$. Entonces T_0 es un árbol con raíz v_1 adyacente a vértices x_1, x_2, \dots, x_{d_1} en el que x_{d_1} tiene grado d_2 . En la figura 13 de izquierda a derecha se muestran los árboles T y $T_0 = \varphi(T, v_2, x_{d_1})$.

Sea y_1 un vértice con grado 1 en la rama de T_0 que contiene a x_{d_1} . Al aplicar la transformación $\varphi(T_0, x_1, y_1)$ obtenemos un árbol T_1 con raíz v_1 en el que el vértice x_1 , adyacente a v_1 ya tiene grado 1 (ver figura 14).

Repetimos esta operación en cada vértice x_i con $i = 2, 3, \dots, d_1 - 1$ adyacente a v_1 hasta obtener un árbol T_{d_1-1} con raíz v_1 adyacente a vértices $x_1, x_2, \dots, x_{d_1-1}$ con grado 1 y al vértice x_{d_1} con grado d_2 (ver figuras 15 y 16).

Figura 12: Árbol T Figura 13: De izquierda a derecha se muestra T y $T_0 = \varphi(T, v_2, x_{d_1})$

Si $d_2 = 2$, entonces $d_3 = d_4 = \dots = d_{t(\sigma)} = 2$ lo cual implica que T_{d_1-1} es isomorfo a C_σ (ver figura 17).

En el caso $d_2 \geq 3$, consideramos el árbol H obtenido de T_{d_1-1} al quitar los vértices $v_1, x_1, x_2, \dots, x_{d_1-1}$. H es un árbol con $n - d_1$ vértices cuya raíz x_{d_1} tiene grado $d_2 - 1$ y cuya sucesión de grados σ' es una permutación de $d_2 - 1, d_3, \dots, d_{n-d_1+1}$ con $t(\sigma') = k$. Por hipótesis de inducción H se puede transformar en un árbol isomorfo a $C_{\sigma'}$ al aplicar sucesivamente transformaciones considerando los vértices u y w distintos de x_{d_1} . Afirmamos que si aplicamos la misma sucesión de transformaciones a T_{d_1-1} obtenemos un árbol isomorfo a C_σ . Esto pues si u y w son vértices de H distintos de x_{d_1} , entonces al aplicar $\varphi(T_{d_1-1}, u, w)$ no se modifican las adyacencias de $v_1, x_1, x_2, \dots, x_{d_1-1}$. \square

Teorema 10. *Para cada sucesión arbórea σ la gráfica de árboles $G_{\pi(\sigma)}$ es conexa.*

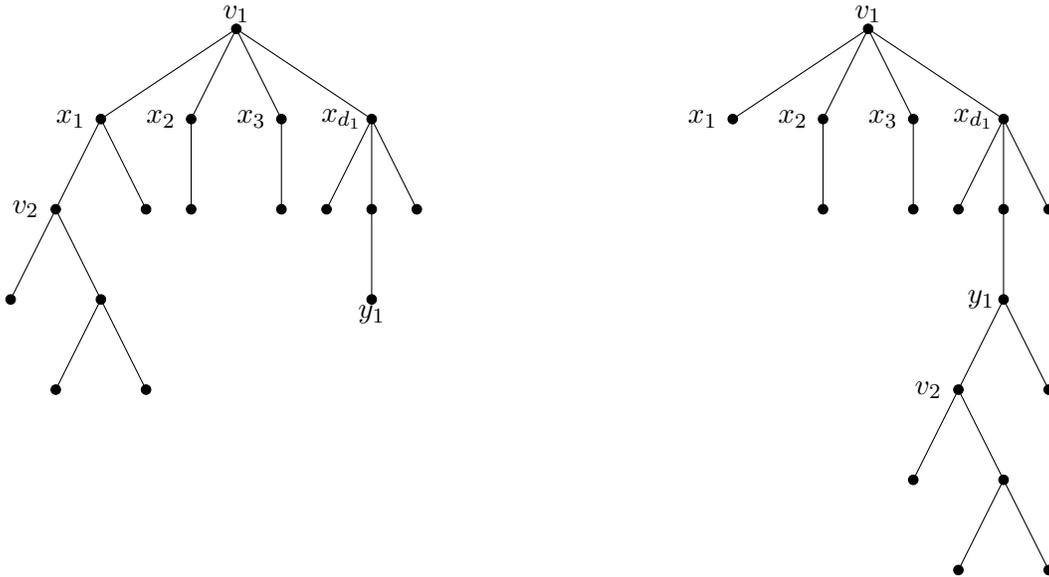


Figura 14: De izquierda a derecha se muestra T_0 y $T_1 = \varphi(T_0, y_1, x_1)$

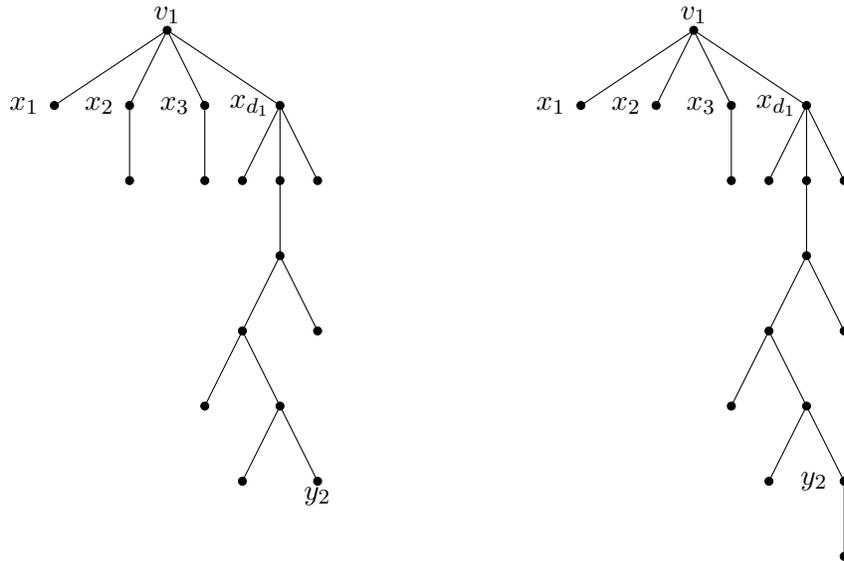


Figura 15: De izquierda a derecha se muestran los árboles T_1 y $T_2 = \varphi(T_1, y_2, x_2)$

Demostración. Sean S y T vértices de $G_{\pi(\sigma)}$. Sin perder generalidad suponemos que la sucesión de grados de S es σ . Denotemos por τ a la sucesión de grados de T .

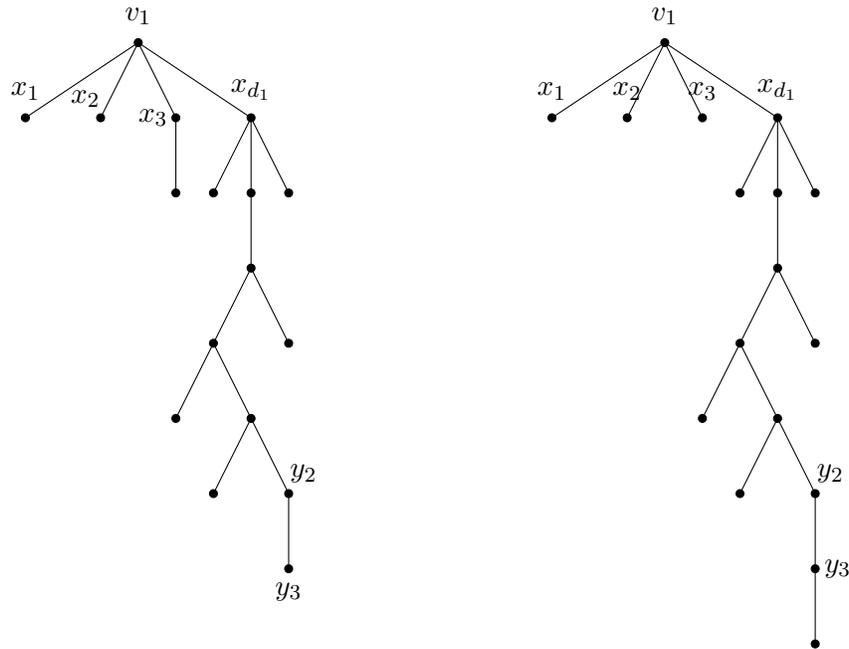


Figura 16: De izquierda a derecha se muestran los árboles T_2 y $T_3 = \varphi(T_2, y_3, x_3)$

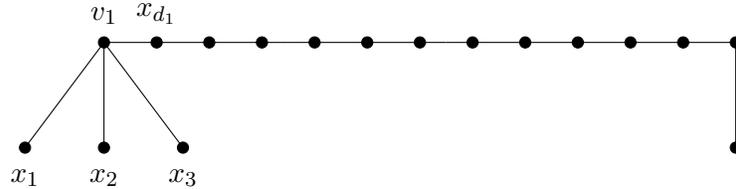


Figura 17: Árbol C_σ

Por el lema 1, S y C_σ están conectados en $G_{\pi(\sigma)}$, análogamente C_τ y T están conectados en $G_{\pi(\sigma)}$. Por otro lado, C_σ y C_τ son isomorfos pues τ es una permutación de σ , por lo tanto C_τ se obtiene de C_σ por medio de intercambios sucesivos de parejas de vértices adyacentes. Cada uno de esos intercambios corresponde a una aplicación de la transformación φ , por lo tanto C_σ y C_τ están conectados en $G_{\pi(\sigma)}$ y por transitividad, S y T también están conectados en $G_{\pi(\sigma)}$.

□

3.3. Algoritmos

En esta sección describiremos dos algoritmos que llamaremos *Mixto* y *Simple* que utilizan de manera sucesiva la transformación φ . El algoritmo Mixto utiliza tanto la adopción simple como la adopción doble y el algoritmo Simple utiliza solamente la adopción simple.

Sean T y T' dos árboles en \mathcal{T}_σ . Paso a paso, los algoritmos van modificando el árbol T cada vez que la transformación φ opera en T . Primero (en caso necesario) iguala el grado de un vértice x de T con el grado del vértice x en T' y después (también caso necesario) hace que la vecindad del vértice x en T sea la misma vecindad que tiene el vértice x en T' . Para asegurar que al final el árbol T se transforme en el árbol deseado T' , las aristas que inciden en el vértice x quedan fijas al final de cada iteración y no son modificadas posteriormente. Sin pérdida de generalidad, se selecciona un vértice r' con grado mayor o igual que dos en T' y se consideran a T' y a T como árboles dirigidos con raíz r' . Notemos que el vértice raíz r' de T y T' se encuentra en el nivel 0 del árbol T y T' .

Sea λ el número total de niveles que tiene el árbol T' y sea x un vértice en T . Vamos a considerar $H_T(x)$ y $H_{T'}(x)$ como los conjuntos de vértices hijos de x en T y T' , respectivamente.

3.3.1. Algoritmo Mixto

Algoritmo 3: Algoritmo Mixto

```

entrada:  $T, T', r'$ 
salida :  $T$  transformado en  $T'$ 
1 para  $N = 0, 1, \dots, \lambda - 1$  hacer
2   para todo  $x$  en el nivel  $N$  del árbol  $T'$  hacer
3     si  $d_{T'}(x) \geq 2$  entonces
4       M-Corregir Grado ( $T, T', r', x$ )
5       M-Corregir Hijos ( $T, T', r', x$ )
6     fin
7   fin
8 fin

```

El procedimiento del algoritmo Mixto es el siguiente: se recorren los niveles del árbol T' a lo ancho, empezando en el nivel 0 hasta el nivel $\lambda - 1$, en búsqueda de vértices x cuyo grado sea mayor que o igual a dos. Por cada vértice x ubicado en T' , se llama

primero a la rutina M-Corregir Grado con la finalidad de que $d_T(x) = d_{T'}(x)$ y después se llama a la rutina M-Corregir Hijos con el propósito de que el conjunto de vértices hijos de x en T coincida con el conjunto de vértices hijos de x en T' (ver las líneas 1 – 8). En la figura 18 tenemos el ejemplo de un árbol T con raíz r' , que con ayuda del algoritmo Mixto puede transformarse en el árbol T' con raíz r' . Mientras que en la figura 19 vemos el árbol T en cierto momento del proceso de transformarse en T' . Notemos que en el árbol T ya operaron las rutinas M-Corregir Grado y M-Corregir Hijos, dado que los grados y conjuntos de los vértices hijos de los vértices r' y v_2 en T son los mismos que tienen los vértices r' y v_2 en el árbol T' . También observemos que el vértice v_4 en T es el próximo vértice en el cual las rutinas del algoritmo Mixto van a operar.

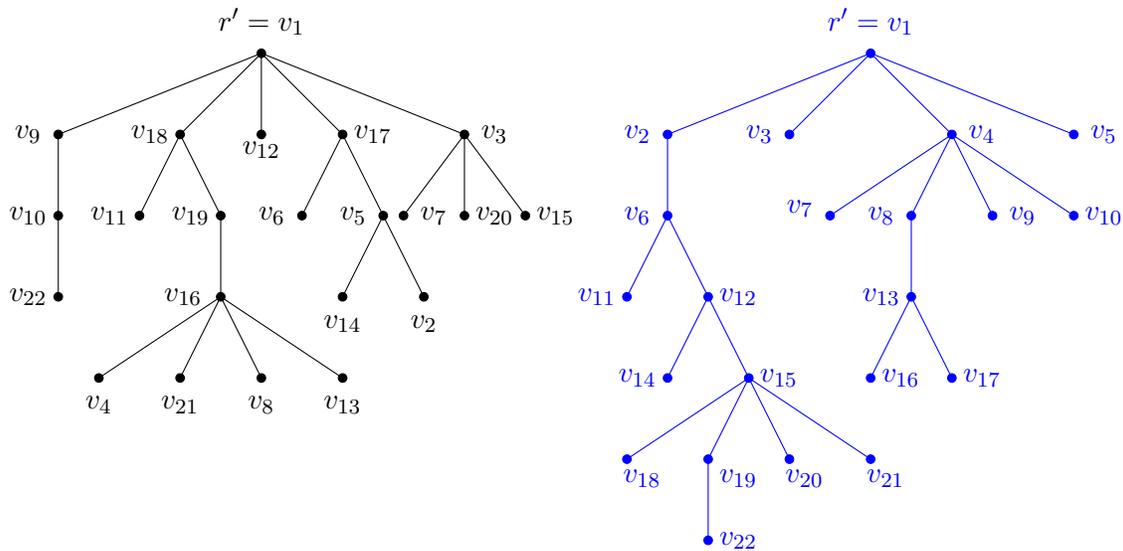


Figura 18: De izquierda a derecha se muestran los árboles T y T' , ambos con raíz $r' = v_1$

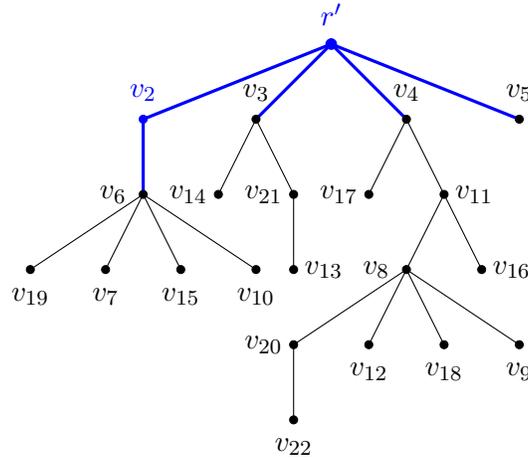


Figura 19: Árbol T con raíz r' en cierto momento del proceso de transformarse en T' , donde $d_T(r') = d_{T'}(r')$, $H_T(r') = H_{T'}(r')$, $d_T(v_2) = d_{T'}(v_2)$ y $H_T(v_2) = H_{T'}(v_2)$

A continuación presentamos la rutina M-Corregir Grado.

Algoritmo 4: M-Corregir Grado

entrada: T, T', r', x
salida : T

- 1 **si** $d_T(x) \neq d_{T'}(x)$ **entonces**
- 2 **si** $x = r'$ **entonces**
- 3 Elegir $y \in V(T)$ tal que $d_T(y) = d_{T'}(r')$
- 4 $T = \varphi(T, r', y)$
- 5 **en otro caso**
- 6 Elegir $y \in V(T)$ tal que $d_T(y) = d_{T'}(x)$ y $d_T(y) \neq d_{T'}(y)$
- 7 **si** y es descendiente de x **entonces**
- 8 Elegir $w \in V(T)$ tal que $d_T(w) = 1$ y que no sea descendiente
- 9 de x
- 10 $T = \varphi(T, w, p(y))$
- 11 $T = \varphi(T, x, y)$
- 12 **en otro caso**
- 13 $T = \varphi(T, x, y)$
- 14 **fin**
- 15 **fin**
- 16 **fin**

El procedimiento de la rutina M-Corregir Grado es el siguiente: si $d_T(x) \neq d_{T'}(x)$, entonces ocurren dos casos: (i) si $x = r'$, se selecciona un vértice y en T tal que $d_T(y) = d_{T'}(r')$ y se hace $T = \varphi(T, r', y)$ para que $d_T(r') = d_{T'}(r')$ (ver las líneas 2 – 4) y (ii) si $x \neq r'$, se busca un vértice y en T tal que $d_T(y) = d_{T'}(x)$ y $d_T(y) \neq d_{T'}(y)$; si y es descendiente de x , entonces se busca un vértice w en T tal que no sea descendiente de x y que $d_T(w) = 1$ (ver las líneas 7 – 9 y la figura 20), luego se hace $T = \varphi(T, w, p(y))$ para que el vértice y deje de ser descendiente de x (ver la línea 10 y las figuras 21 y 22) y enseguida se hace $T = \varphi(T, x, y)$ para que $d_T(x) = d_{T'}(x)$ (ver la línea 11 y las figuras 23 y 24). Ahora bien, si y no es descendiente de x , simplemente se realiza $T = \varphi(T, x, y)$ para que $d_T(x) = d_{T'}(x)$ (ver las líneas 12 – 14 y las figuras 23 y 24).

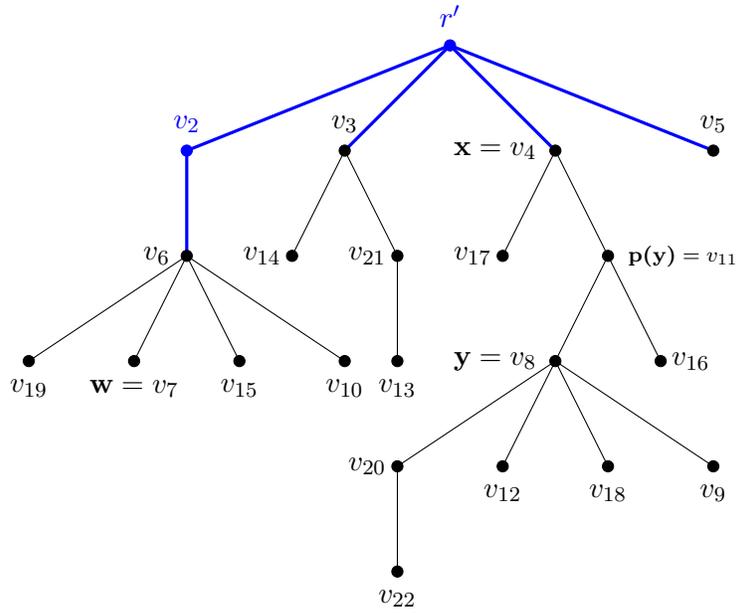


Figura 20: Caso (ii) de la rutina M-Corregir Grado, donde el vértice $y = v_8$ es descendiente de $x = v_4$ y en consecuencia se selecciona un vértice $w = v_7$ en T tal que no es descendiente del vértice $x = v_4$ y $d_T(w) = d_T(v_7) = 1$

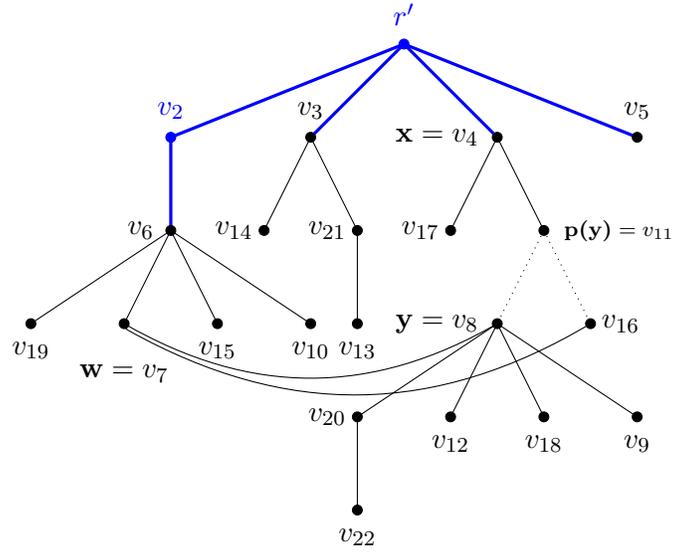


Figura 21: $T = \varphi(T, w, p(y)) = \varphi(T, v_7, v_{11})$

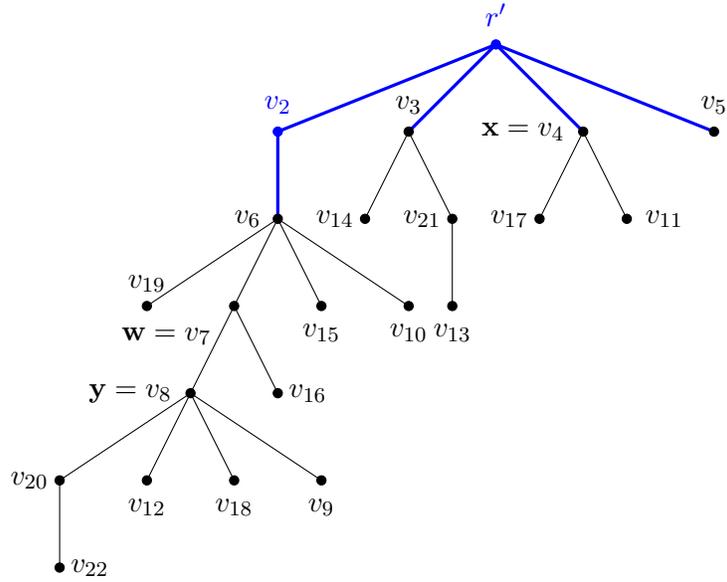


Figura 22: Árbol T con raíz r' , donde el vértice $y = v_8$ no es descendiente del vértice $x = v_4$

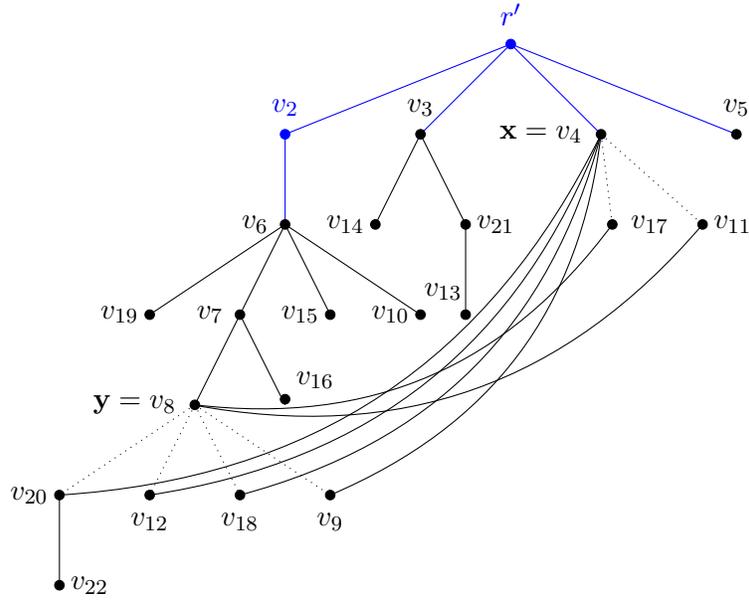


Figura 23: $T = \varphi(T, x, y) = \varphi(T, v_4, v_8)$

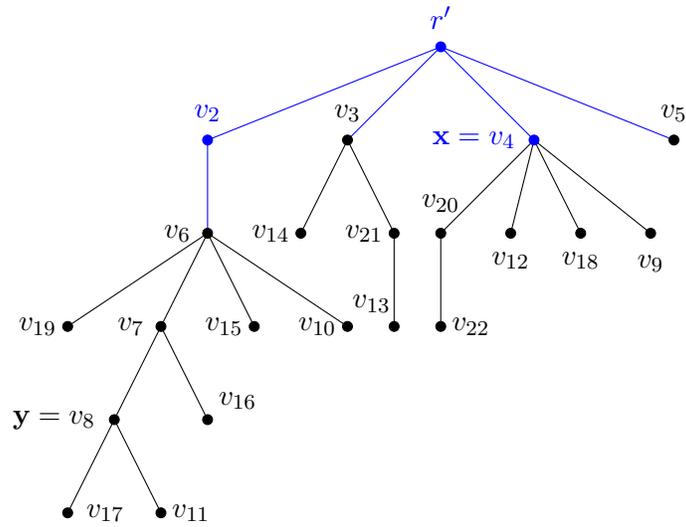


Figura 24: T con raíz r' , donde $d_T(v_4) = d_{T'}(v_4)$

A continuación presentamos la rutina M-Corregir Hijos.

Algoritmo 5: M-Corregir Hijos

entrada: T, T', r', x
salida : T

- 1 $INC = H_T(x) \setminus H_{T'}(x)$
- 2 $CAND = H_{T'}(x) \setminus H_T(x)$
- 3 **mientras** $CAND \neq \emptyset$ **hacer**
- 4 Elegir $v \in INC$ y $z \in CAND$
- 5 **si** z es descendiente de v **entonces**
- 6 $T = \varphi(T, z, v)$
- 7 **en otro caso**
- 8 Elegir $w \in V(T)$ tal que $d_T(w) = 1$ y sea descendiente de v
- 9 $T = \varphi(T, w, p(z))$
- 10 $T = \varphi(T, z, v)$
- 11 **fin**
- 12 $INC = INC \setminus \{v\}$
- 13 $CAND = CAND \setminus \{z\}$
- 14 **fin**

El procedimiento de la rutina M-Corregir Hijos es el siguiente: se genera el conjunto de vértices INC como resultado de la diferencia del conjunto $H_T(x)$ y el conjunto $H_{T'}(x)$, donde los elementos del conjunto INC son los vértices hijos del vértice x en T que son distintos a los vértices hijos del vértice x en T' (ver la línea 1). Después se genera el conjunto CAND como resultado de la diferencia del conjunto $H_{T'}(x)$ y el conjunto $H_T(x)$, donde los elementos del conjunto CAND son vértices que no son hijos de x en T , pero sí vértices hijos de x en T' (ver la línea 2). En la figura 25 podemos observar un ejemplo de la generación de los conjuntos INC y CAND.

Después se ejecuta el bucle *mientras*, que repetirá una serie de instrucciones en tanto el conjunto $CAND \neq \emptyset$ (ver las líneas 3 – 11). Cuando se entra al bucle se elige un vértice $v \in INC$ y un vértice $z \in CAND$. Después de elegir al vértice z ocurren dos casos: (i) si el vértice z es descendiente de v en T , entonces se realiza $T = \varphi(T, z, v)$, para que el vértice z sea hijo de x en lugar del vértice v (ver figura 29) y (ii) si el vértice z no es descendiente de v en T , entonces se busca un vértice w en T tal que sea descendiente del vértice v y $d_T(w) = 1$ (ver la línea 8 y figura 26), luego se hace $T = \varphi(T, w, p(z))$ para que el vértice z sea descendiente del vértice v (ver figuras 27 y 28) y después se hace $T = \varphi(T, z, v)$ para que el vértice z se convierta en hijo de x en

lugar del vértice v en T (ver las líneas 7 – 11 y figura 29). Finalmente se actualizan los conjuntos INC y CAN (ver la línea 13).

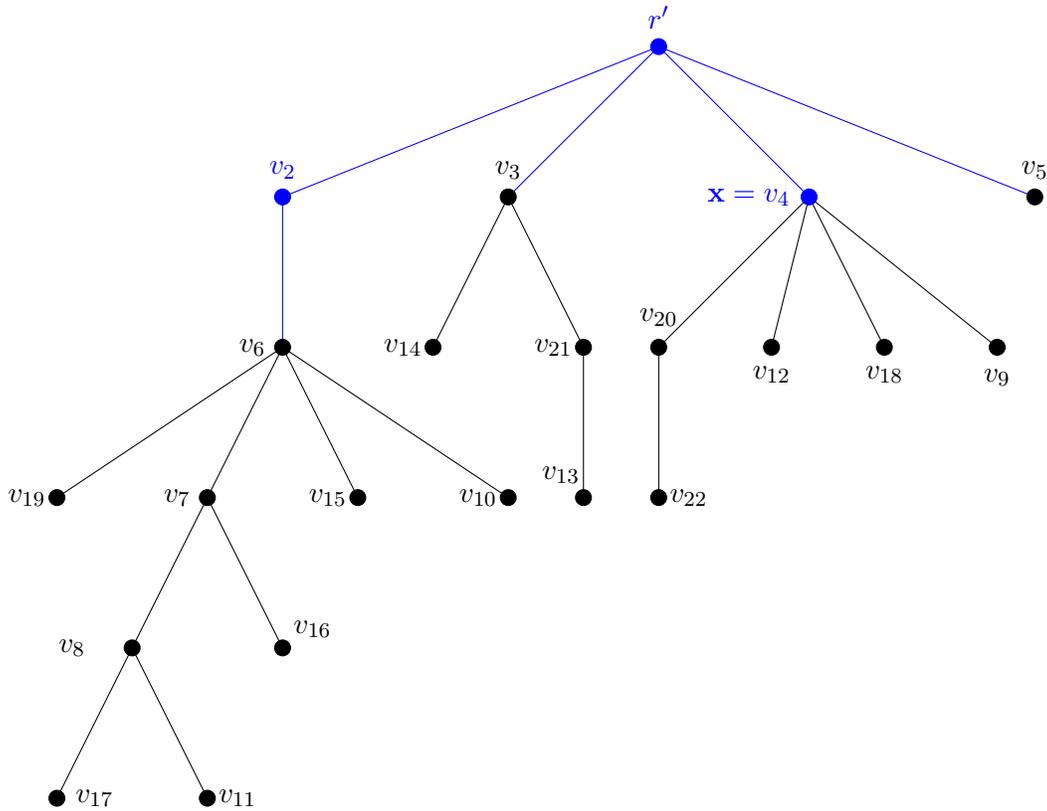


Figura 25: T con raíz r' , donde $\text{INC} = H_T(v_4) \setminus H_{T'}(v_4) = \{v_{20}, v_{12}, v_{18}, v_9\} \setminus \{v_7, v_8, v_9, v_{10}\} = \{v_{20}, v_{12}, v_{18}\}$ y $\text{CAND} = H_{T'}(v_4) \setminus H_T(v_4) = \{v_7, v_8, v_9, v_{10}\} \setminus \{v_{20}, v_{12}, v_{18}, v_9\} = \{v_7, v_8, v_{10}\}$

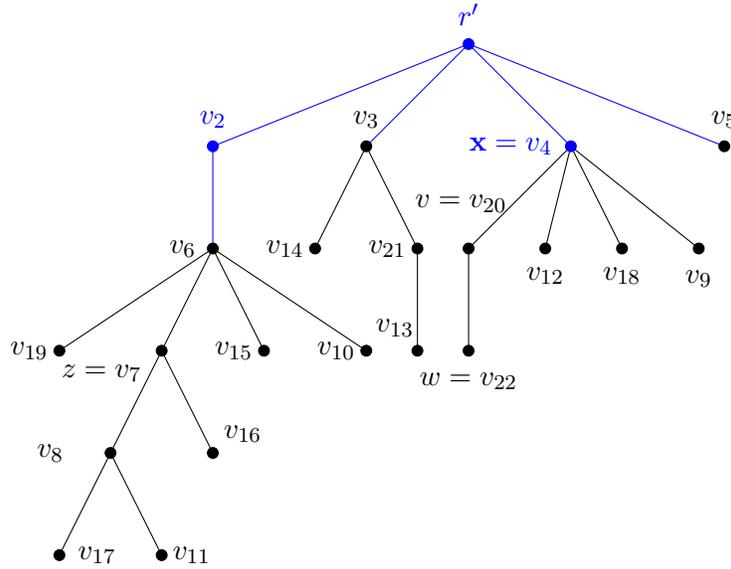


Figura 26: Caso (ii) de la rutina M-Corregir Hijo, cuando el vértice $z = v_7$ no es descendiente del vértice $v = v_{20}$ y en consecuencia se selecciona un vértice $w = v_{22}$ en T tal que sea descendiente del vértice $v = v_{20}$ y $d_T(w) = d_T(v_{22}) = 1$

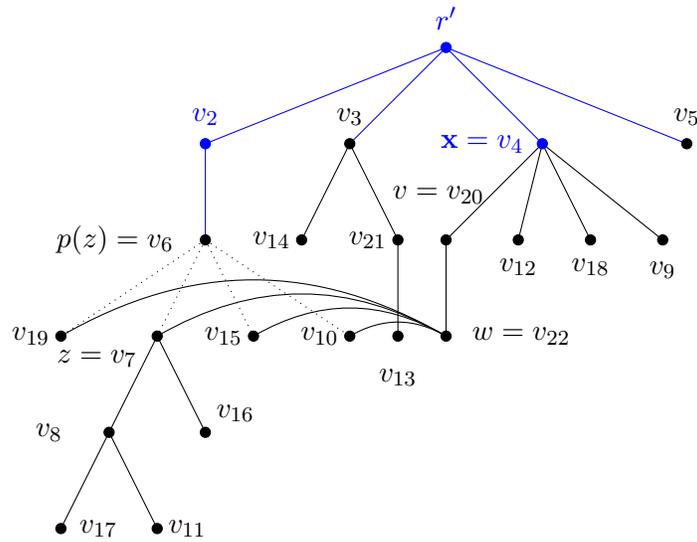


Figura 27: $T = \varphi(T, w, p(z))$

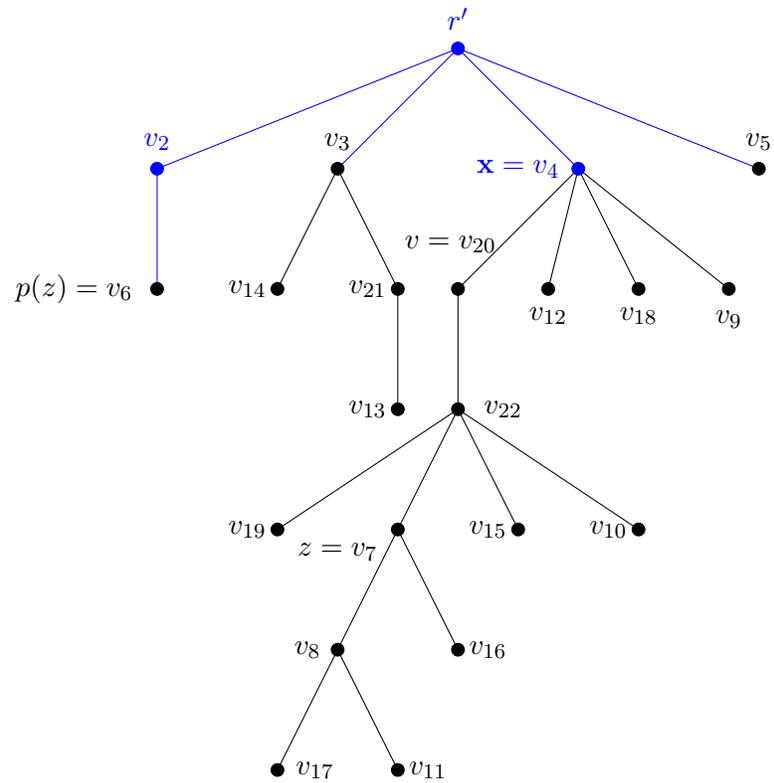


Figura 28: Árbol T con raíz r' , donde el vértice $z = v_7$ es descendiente del vértice $v = v_{20}$

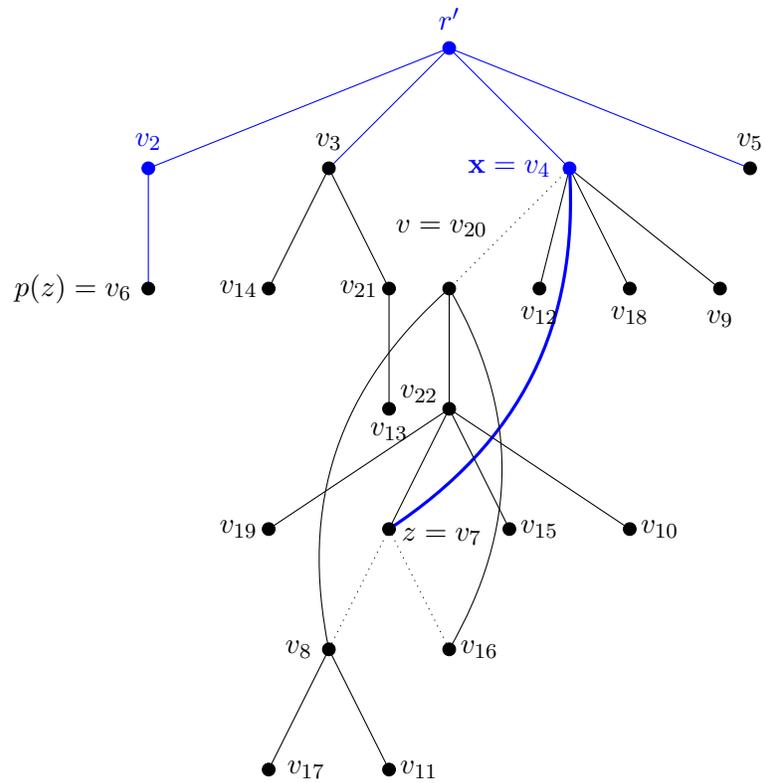


Figura 29: $T = \varphi(T, z, v)$ con raíz r' , donde el vértice $z = v_7$ es un vértice hijo del vértice $x = v_4$

3.3.2. Algoritmo Simple

En esta subsección presentamos el pseudocódigo del algoritmo Simple.

Algoritmo 6: Algoritmo Simple

entrada: T, T', r'
salida : T transformado en T'

```

1 para  $N = 0, 1, 2, \dots, \lambda - 1$  hacer
2   para todo  $x$  en el nivel  $N$  del árbol  $T'$  hacer
3     si  $d_{T'}(x) \geq 2$  entonces
4       S-Corregir Grado( $(T, T', r', x)$ )
5       S-Corregir Hijos ( $T, T', r', x$ )
6     fin
7   fin
8 fin

```

Al igual que en el algoritmo anterior, el procedimiento del algoritmo Simple también consiste en recorrer por niveles y a lo ancho el árbol T' , empezando en el nivel 0 hasta el nivel $\lambda - 1$, en búsqueda de vértices x cuyo grado sea mayor o igual que dos en T' . Por cada vértice x encontrado en T' se llama primero a la rutina S-Corregir Grado para hacer, en caso necesario, que $d_T(x) = d_{T'}(x)$ y luego se invoca a la rutina S-Corregir Hijos para hacer, en caso necesario, que el conjunto de vértices hijos de x en T sea igual al conjunto de vértices hijos de x en T' .

Tomando en cuenta los árboles T y T' con raíz r' de la figura 18, vamos a considerar de nuevo a T como un ejemplo de un árbol que puede transformarse en el árbol T' , pero esta vez con ayuda del algoritmo Simple. Por otra parte, para ejemplificar algunos pasos de las rutinas del algoritmo Simple vamos a partir del árbol T de la figura 19. Recordemos que este árbol, se encuentra en cierto momento del proceso de transformarse en T' . Notemos que por el funcionamiento del algoritmo Simple, las rutinas S-Corregir Grado y S-Corregir Hijos van a operar en el vértice v_4 del árbol T .

Ahora damos paso al pseudocódigo de la rutina S-Corregir Grado.

Algoritmo 7: S-Corregir Grado

entrada: T', r', T, x
salida : T

- 1 **si** $d_T(x) \neq d_{T'}(x)$ **entonces**
- 2 **si** $x = r'$ y $d_T(r') = 1$ **entonces**
- 3 Elegir $y \in V(T)$ tal que $d_T(y) = d_{T'}(r')$
- 4 **regresar** $T = \varphi(T, r', y)$
- 5 **fin**
- 6 **si** $x = r'$ y $d_T(r') \neq 1$ **entonces**
- 7 Elegir $w \in V(T)$ tal que $d_T(w) = 1$
- 8 $T = \varphi(T, w, r')$
- 9 Elegir $y \in V(T)$ tal que $d_T(y) = d_{T'}(r')$
- 10 **regresar** $T = \varphi(T, r', y)$
- 11 **fin**
- 12 **si** $x \neq r'$ y $d_T(x) = 1$ **entonces**
- 13 Elegir $y \in V(T)$ tal que $d_T(y) = d_{T'}(x)$ y $d_T(y) \neq d_{T'}(y)$
- 14 **regresar** $T = \varphi(T, x, y)$
- 15 **fin**
- 16 **si** $x \neq r'$ y $d_T(x) \neq 1$ **entonces**
- 17 Elegir $w \in V(T)$ tal que $d_T(w) = 1$ y no sea descendiente de x
- 18 $T = \varphi(T, w, x)$
- 19 Elegir $y \in V(T)$ tal que $d_T(y) = d_{T'}(x)$ y $d_T(y) \neq d_{T'}(y)$
- 20 **regresar** $T = \varphi(T, x, y)$
- 21 **fin**
- 22 **fin**
- 23 **regresar** T

El procedimiento de la rutina S-Corregir Grado funciona de la siguiente manera: si $d_T(x) \neq d_{T'}(x)$, entonces el algoritmo considera solo uno de los cuatro posibles casos: (i) si $x = r'$ y $d_T(r') = 1$, entonces se elige un vértice y en T tal que $d_T(y) = d_{T'}(r')$; por consiguiente se hace $T = \varphi(T, r', y)$ para que el $d_T(r') = d_{T'}(r')$ y finalmente se regresa T (ver las líneas 2–5); (ii) si $x = r'$ y $d_T(r') \neq 1$, entonces se busca un vértice w en T tal que $d_T(w) = 1$ y luego se se hace $T = \varphi(T, w, r')$ para que el $d_T(r') = 1$, posteriormente se selecciona un vértice y en T tal que $d_T(y) = d_{T'}(r')$ y se hace $T = \varphi(T, r', y)$ para que el $d_T(r') = d_{T'}(r')$ y finalmente se regresa T (ver las líneas 6 – 11); (iii) si $x \neq r'$ y $d_T(x) = 1$, entonces se selecciona un vértice y en T tal que $d_T(y) = d_{T'}(x)$ y

$d_T(y) \neq d_{T'}(y)$ y después se hace $T = \varphi(T, x, y)$ para que el $d_T(x) = d_{T'}(x)$ y finalmente se regresa T (ver las líneas 12 – 15) y (iv) si $x \neq r'$ y $d_T(x) \neq 1$, entonces se busca un vértice w en T tal que no sea descendiente de x y $d_T(w) = 1$ y enseguida se realiza $T = \varphi(T, w, x)$ para que $d_T(x) = 1$ (ver las líneas 16 – 18 y las figuras 30 y 31), posteriormente se busca un vértice y en T tal que $d_T(y) = d_{T'}(x)$ y $d_T(y) \neq d_{T'}(y)$ y se hace $T = \varphi(T, x, y)$ para que el $d_T(x) = d_{T'}(x)$ y finalmente se regresa T (ver las líneas 19 – 21 y las figuras 32 y 33). Por otro lado, si no se cumple que $d_T(x) \neq d_{T'}(x)$, entonces simplemente se regresa T .

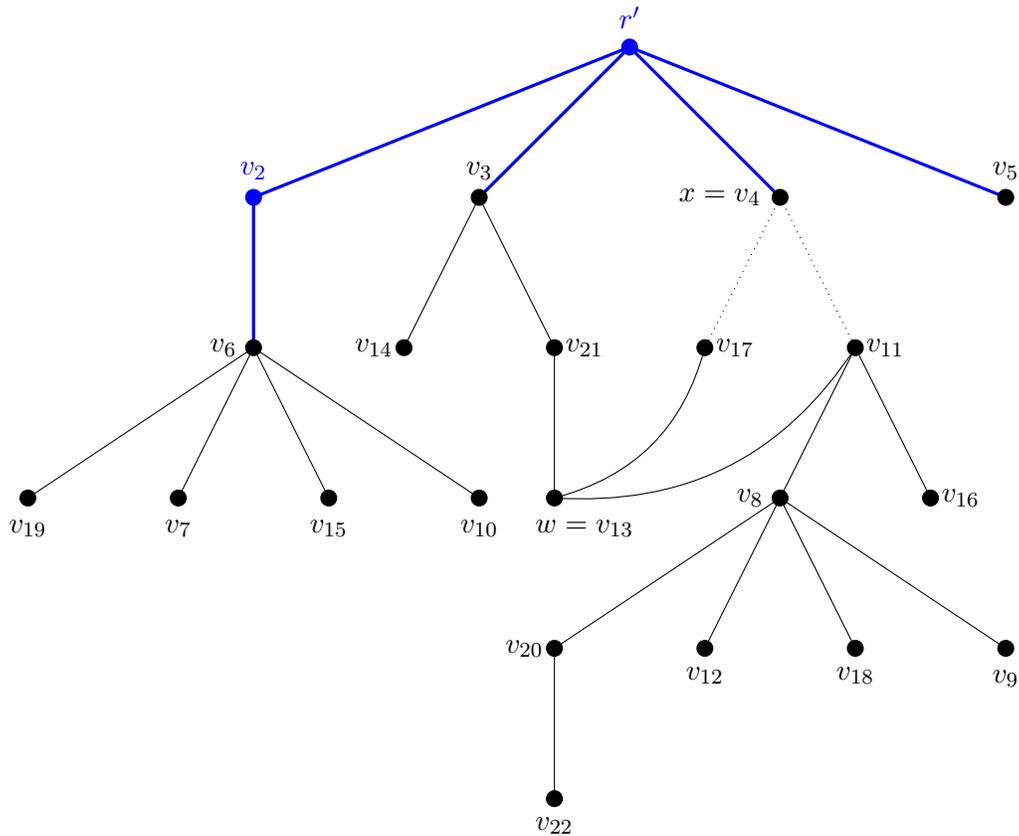


Figura 30: Caso (iv) de la rutina S-Corregir Grado, donde $x = v_4$ y $d_T(x) = 1$, $w = v_{13}$ y $T = \varphi(T, w, x) = \varphi(T, v_{13}, v_4)$

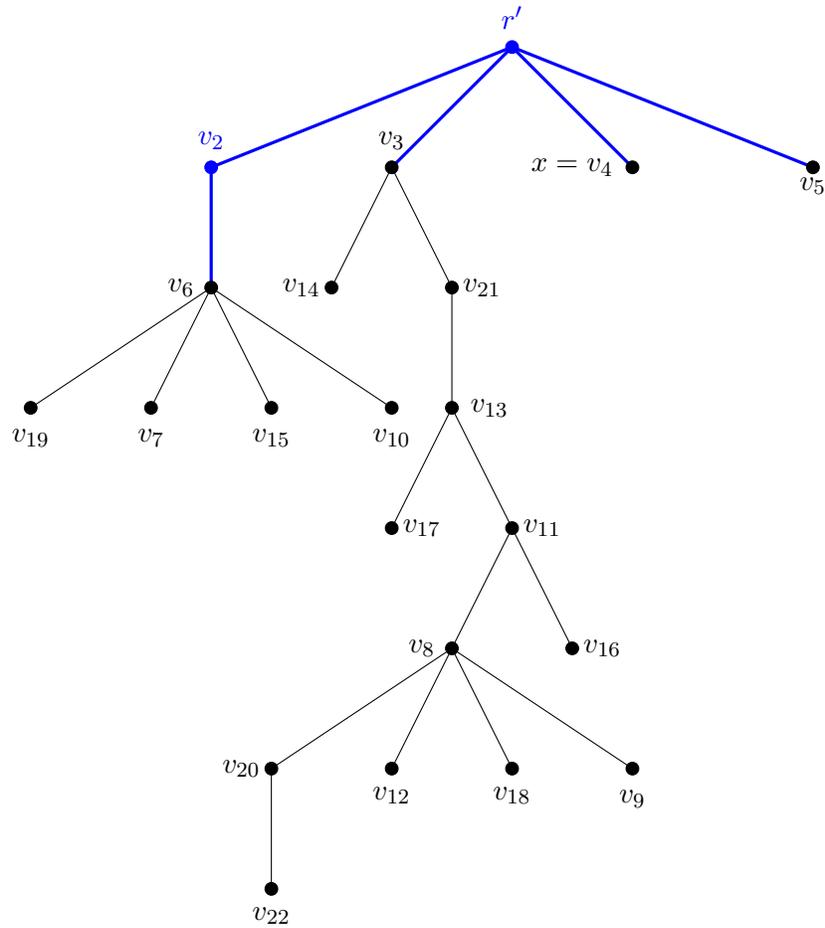


Figura 31: Árbol T con raíz r' , donde $d_T(x) = d_T(v_4) = 1$

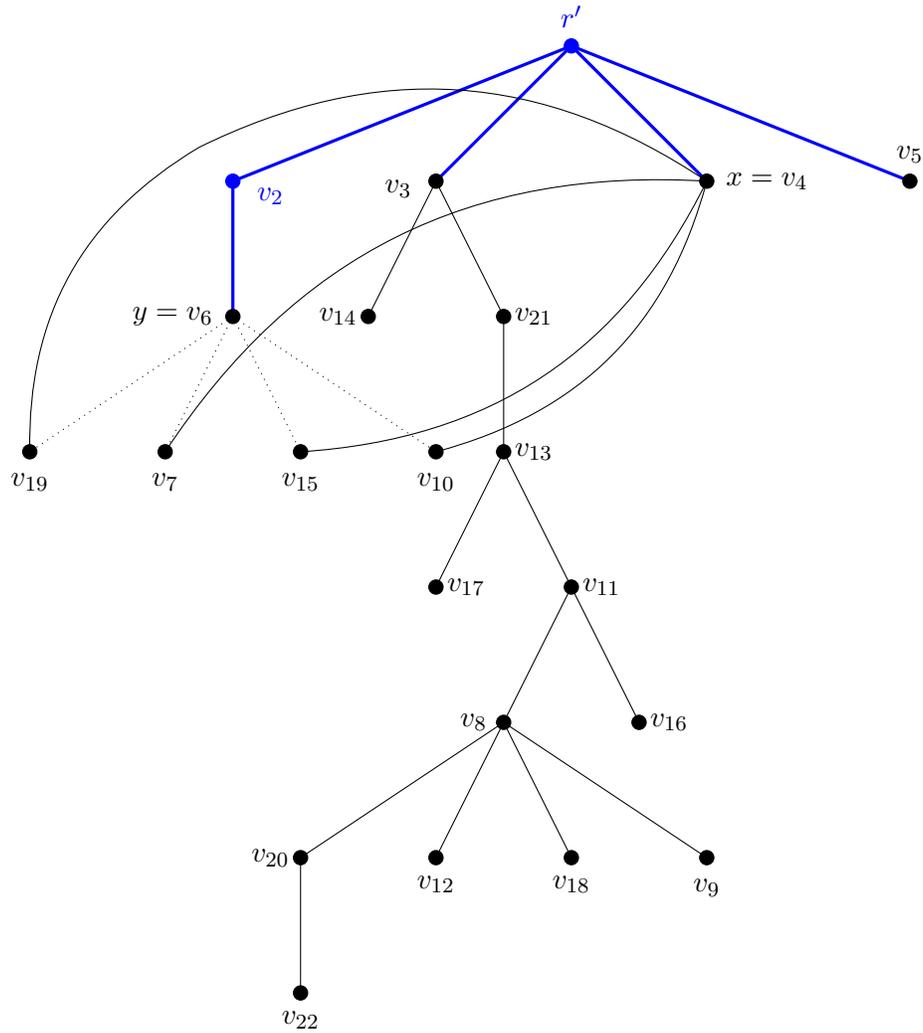


Figura 32: Caso (iv) de la rutina S-Corregir Grado, donde $x = v_4$, $y = v_6$ y $T = \varphi(T, x, y) = \varphi(T, x_4, v_6)$

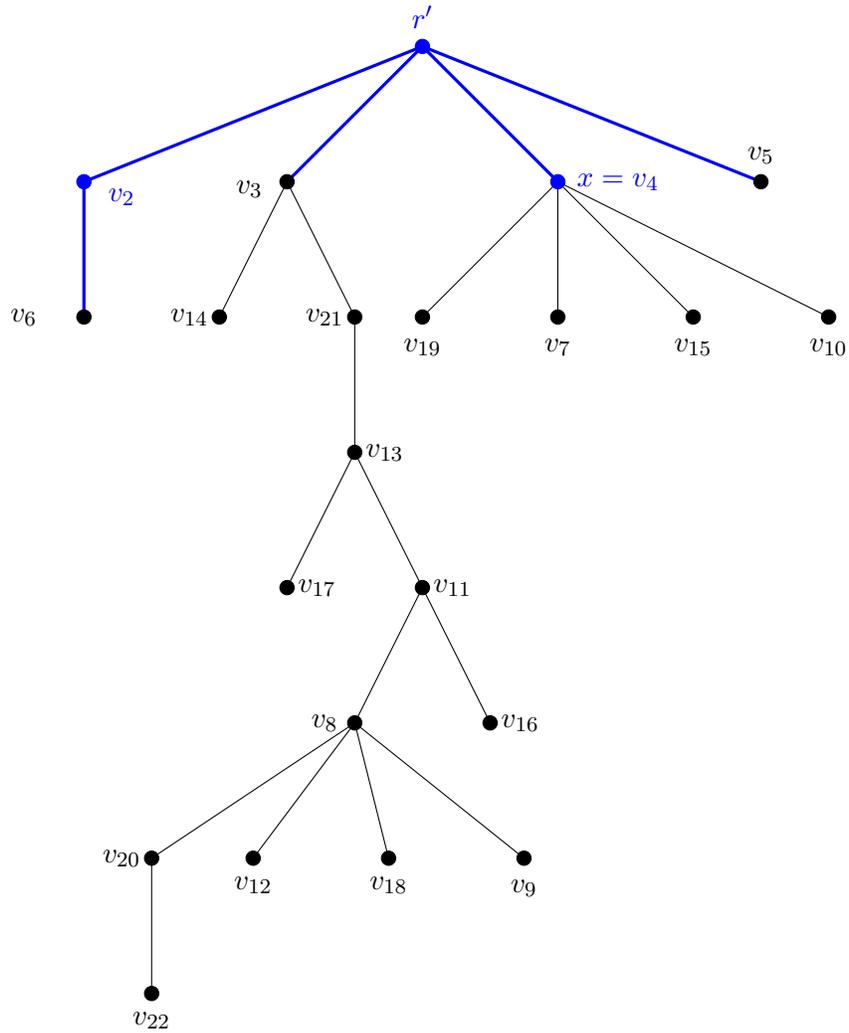


Figura 33: Árbol T con raíz r' , donde $d_T(x) = d_T(v_4) = 5 = d_{T'}(v_4) = d_{T'}(x)$

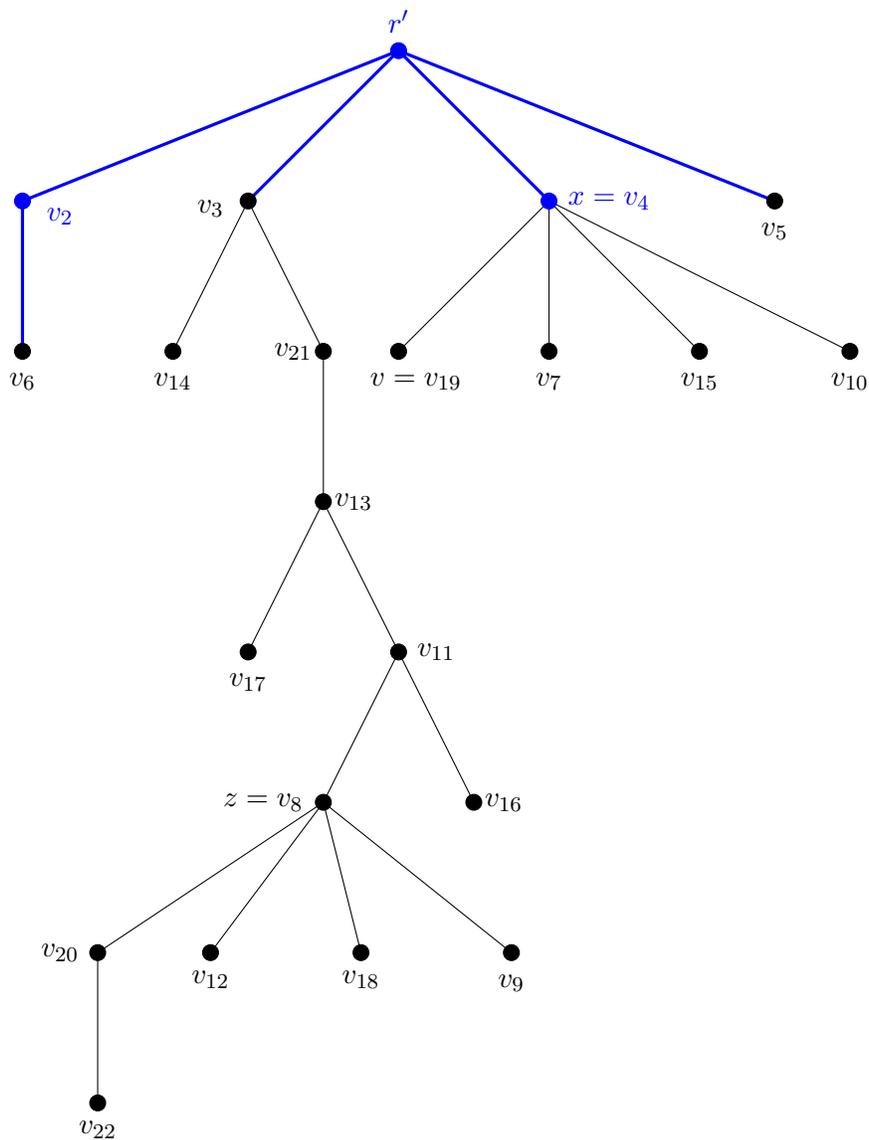


Figura 34: Árbol T con raíz r' , donde $x = v_4$, $\text{INC} = \{v_{19}, v_{15}\}$, $\text{CAND} = \{v_8, v_9\}$, $v = v_{19} \in \text{INC}$ y $z = v_8 \in \text{CAND}$

Finalmente mostramos el pseudocódigo de la rutina S-Corregir Hijos.

Algoritmo 8: S-Corregir Hijos

entrada: T', r', T, x
salida : T

- 1 $INC = H_T(x) \setminus H_{T'}(x)$
- 2 $CAND = H_{T'}(x) \setminus H_T(x)$
- 3 **mientras** $CAND \neq \emptyset$ **hacer**
- 4 Elegir $v \in INC$ y $z \in CAND$
- 5 **si** $d_T(z) = 1$ y z es descendiente de v **entonces**
- 6 **regresar** $T = \varphi(T, z, v)$
- 7 **fin**
- 8 **si** $d_T(z) = 1$ y z no es descendiente de v **entonces**
- 9 Elegir $w \in V(T)$ tal que $d_T(w) = 1$ y sea descendiente de v
- 10 $T = \varphi(T, w, p(z))$
- 11 **regresar** $T = \varphi(T, z, v)$
- 12 **fin**
- 13 **si** $d_T(z) \geq 2$ y z es descendiente de v **entonces**
- 14 Elegir $w \in V(T)$ tal que $d_T(w) = 1$ y sea descendiente de v
- 15 $T = \varphi(T, w, z)$
- 16 **regresar** $T = \varphi(T, z, v)$
- 17 **fin**
- 18 **si** $d_T(z) \geq 2$ y z no es descendiente de v **entonces**
- 19 Elegir $w \in V(T)$ tal que $d_T(w) = 1$ y sea descendiente de v
- 20 $T = \varphi(T, w, p(z))$
- 21 Elegir $u \in V(T)$ tal que $d_T(u) = 1$ y sea descendiente de z
- 22 $T = \varphi(T, u, z)$
- 23 **regresar** $T = \varphi(T, z, v)$
- 24 **fin**
- 25 $INC = INC \setminus \{v\}$
- 26 $CAND = CAND \setminus \{z\}$
- 27 **fin**

El procedimiento de la rutina S-Corregir Hijos es el siguiente: dado el vértice x en el árbol T con raíz r' , se generan los conjuntos INC y CAND. Los elementos del conjunto INC son los vértices que resultan de la diferencia del conjunto $H_T(x)$ y $H_{T'}(x)$, es decir, son los vértices hijos del vértice x en T que no son vértices hijos de x en T' y los elementos del conjunto CAND son los vértices que resultan de la diferencia del conjunto $H_{T'}(x)$ y $H_T(x)$, es decir, son vértices que no son hijos de x en T , pero sí son vértices hijos de x en T' . Posteriormente, se ejecuta el bucle *mientras* que repetirá una serie de pasos mientras el conjunto CAND sea distinto del conjunto vacío. Dentro del bucle se elige un vértice v en el conjunto INC y un vértice z en el conjunto CAND (ver las líneas 1–4 y figura 34). Después, dependiendo del $d_T(z)$ y si z es o no descendiente de v en T el algoritmo considera solo uno de los siguientes cuatro posibles casos: (i) si el $d_T(z) = 1$ y z es descendiente de v , entonces se hace $T = \varphi(T, z, v)$ para que el vértice z sea un vértice hijo del vértice x en lugar del vértice v y finalmente se regresa T (ver las líneas 5–7); (ii) si $d_T(z) = 1$ y z no es descendiente de v , se busca un vértice w en T tal que sea descendiente de v y $d_T(w) = 1$, luego se hace $T = \varphi(T, w, p(z))$ para que el vértice z sea descendiente del vértice v y enseguida se realiza $T = \varphi(T, z, v)$ para que el vértice z sea un vértice hijo de x en lugar del vértice v (ver las líneas 8–12) y finalmente se regresa T ; (iii) si $d_T(z) \neq 2$ y z es descendiente de v , entonces se elige un vértice w en T tal que sea descendiente del vértice v y $d_T(w) = 1$, luego se hace $T = \varphi(T, w, z)$ para que $d_T(z) = 1$ y enseguida se realiza $T = \varphi(T, z, v)$ para que el vértice z se convierta en un vértice hijo de x en lugar de v y finalmente se regresa T (ver las líneas 13–17) y (iv) si $d_T(z) \geq 2$ y z no es descendiente del vértice v , entonces se selecciona un vértice w en T tal que sea descendiente del vértice v y $d_T(w) = 1$, luego se hace $T = \varphi(T, w, p(z))$ para que el vértice z sea descendiente del vértice v (ver las líneas 18–20 y las figuras 35, 36 y 37). Posteriormente se busca un vértice u en T tal que sea descendiente de z y $d_T(u) = 1$ y se realiza $T = \varphi(T, u, z)$ para que $d_T(z) = 1$ (ver las líneas 21 y 22 y las figuras 38 y 39) y se realiza $T = \varphi(T, z, v)$ para que el vértice z sea un vértice hijo de x en lugar v y finalmente se regresa T (ver la línea 23 y las figuras 40 y 41). Por último se actualizan los conjuntos INC y CAND (ver la línea 26).

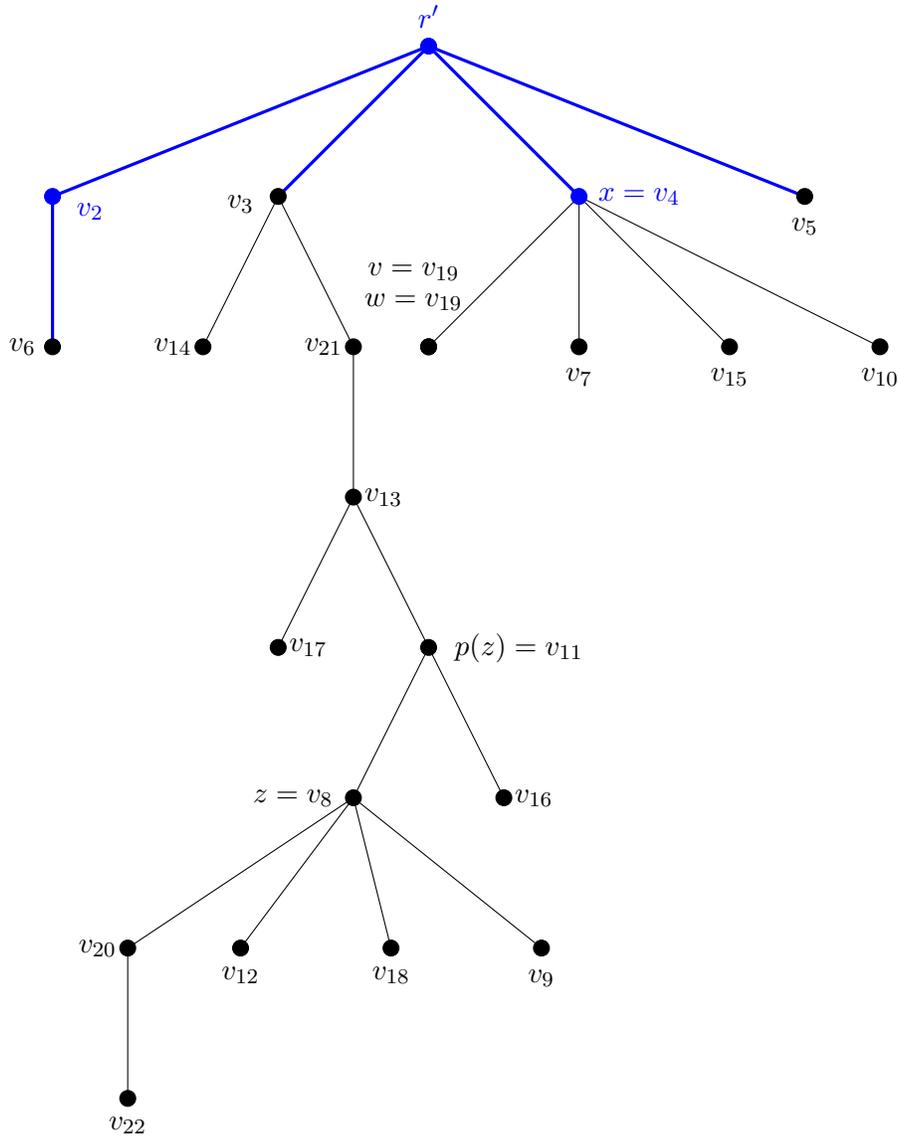


Figura 35: Caso (iv) de la rutina S-Corregir Hijos, donde $z = v_8$, $d_T(v_8) \geq 2$ y no es descendiente del v3rtice $v = x_4$ y en consecuencia se selecciona el v3rtice $w = v_{19}$ en T tal que es descendiente del v3rtice v_8 y $d_T(v_{19}) = 1$

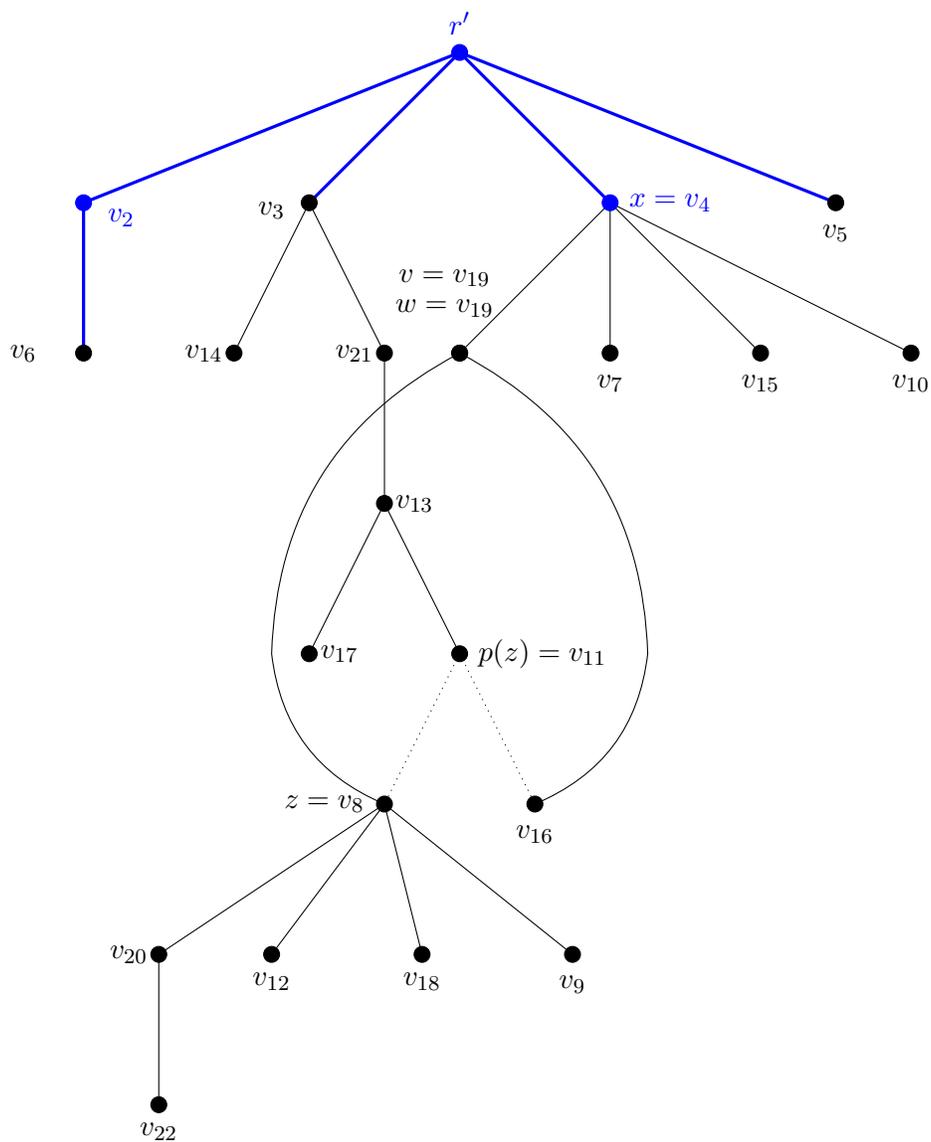


Figura 36: $T = \varphi(T, w, p(z)) = \varphi(T, v_{19}, v_{11})$ con raíz r'

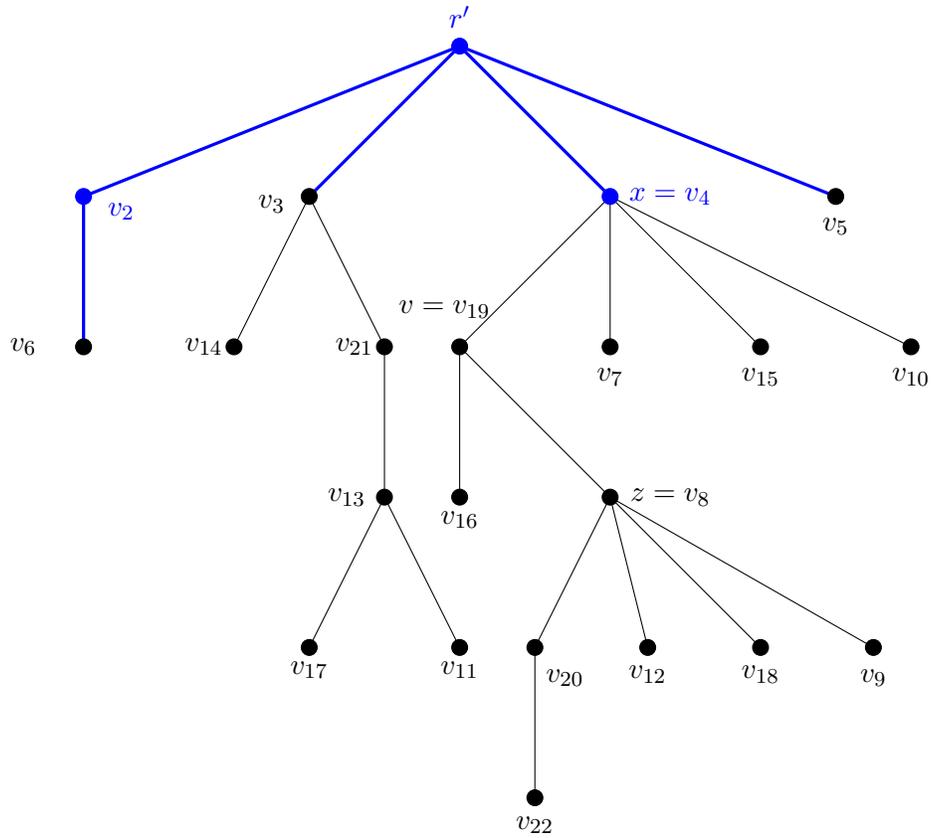


Figura 37: Árbol T con raíz r' , donde el vértice $z = v_8$ es descendiente del vértice v_{19}

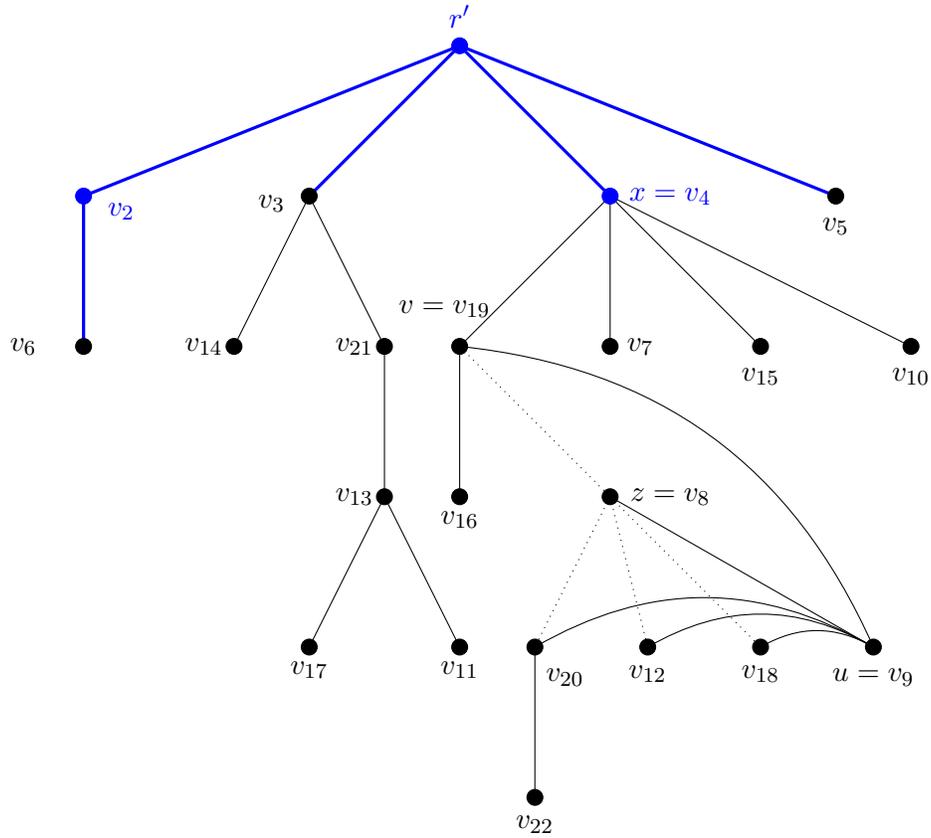


Figura 38: $T = \varphi(T, u, z) = \varphi(T, v_9, v_8)$, con raíz r'

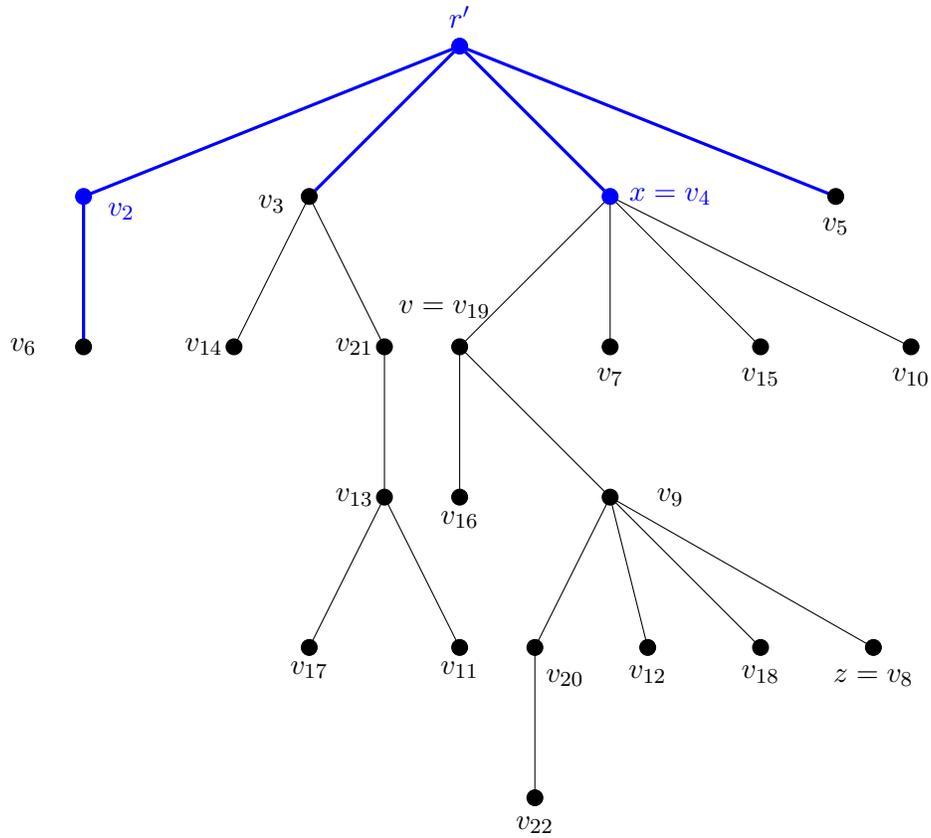


Figura 39: Árbol T con raíz r' , donde $d_T(z) = d_T(v_8) = 1$

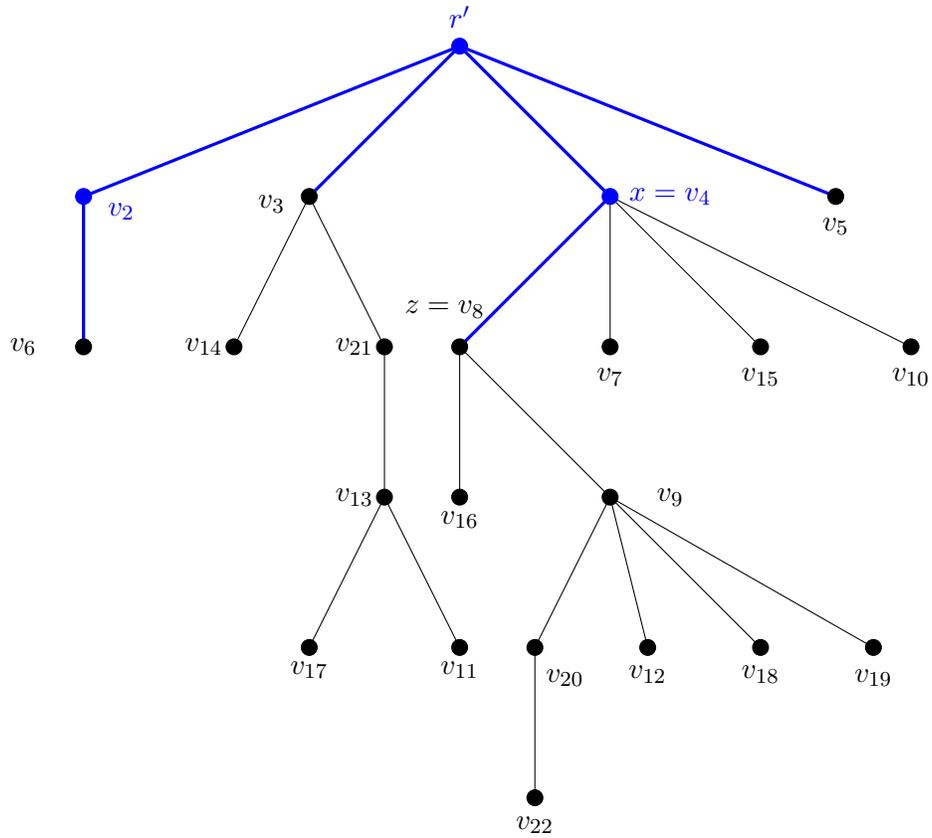


Figura 41: Árbol T con raíz r' , donde el vértice $z = v_8$ es vértice hijo de $x = v_4$

3.4. Cotas

En esta sección damos a conocer resultados sobre cotas superiores referentes al número de llamados que realizan el algoritmo Mixto y el algoritmo Simple a la transformación φ para ir de T a T' .

Teorema 11. *Sean T y T' árboles con raíz r' en \mathcal{T}_σ . El número de llamados que realiza el algoritmo Mixto a la transformación φ para ir de T a T' es M_φ y satisface que $M_\varphi \leq 4n - 9$, donde n es el orden de los árboles T y T' .*

Antes de dar inicio a la demostración del teorema 11 es importante recordar que cada vez que el algoritmo Mixto llama a φ para que trabaje en el árbol actual se redefine el árbol resultante de nuevo como T .

Demostración. Como hemos mencionado en cada paso el algoritmo Mixto primero hace que el grado de un vértice x en T tenga el mismo grado que tiene en T' y después hace que su conjunto de vértices hijos también sea igual al conjunto de vértices hijos que tiene en T' . El algoritmo Mixto empieza por el vértice raíz r' haciendo que la rutina M-Corregir Grado llame una sola vez a la transformación φ si $d_T(r') \neq d_{T'}(r')$ (ver las líneas 2 – 4). Notemos que si $d_T(r') = d_{T'}(r')$, entonces no se hace ningún llamado a φ .

Luego la rutina M-Corregir Hijos llama un cierto número de veces a φ si el conjunto de vértices hijos del vértice r' en T es distinto al conjunto de vértices hijos de r' en T' .

Afirmación 1. *El número de llamados que hace la rutina M-Corregir Hijos a φ para que el conjunto de vértices hijos de r' en T sea igual al conjunto de vértices hijos de r' en T' es a lo más $2d_T(r')$.*

Notemos que por cada vértice hijo v de r' en T que sea diferente a los vértices hijos de r' en T' la rutina M-Corregir Hijos llama a lo más dos veces a la transformación φ para cambiar al vértice v por un vértice deseado z en T (ver las líneas 4 – 11).

En el caso en que todos los vértices hijos de r' en T sean distintos a los vértices hijos de r' en T' , el número de llamados que realiza la rutina M-Corregir Hijos a φ en T para que el conjunto de vértices hijos de r' en T sea igual al conjunto de vértices hijos de r' en T' es a lo más dos veces el número de vértices hijos de r' en T y como el número de hijos de r' en T es igual $d_T(r')$, entonces el número de llamados que se hace a φ es a lo más $2d_T(r)$.

Por otra parte, para cada uno de los vértices restantes x en T el algoritmo Mixto hace que la rutina M-Corregir Grado llame a lo más dos veces a φ si $d_T(x) \neq d_{T'}(x)$ (ver las líneas 5 – 15). Observemos que si $d_{T'}(x) = d_T(x)$, entonces la rutina M-Corregir Grado no realiza ningún llamado a φ .

Afirmación 2. *El número de llamados que realiza la rutina M-Corregir Hijos a la transformación φ para que el conjunto de vértices hijos de x en T sea igual al conjunto de vértices hijos de x en T' es a lo más $2(d_T(x) - 1)$.*

Observemos que por cada vértice hijo v de x en T que sea diferente a los vértices hijos de x en T' la rutina M-Corregir Hijos llama a lo más dos veces a la transformación φ para cambiar al vértice v por un vértice deseado z en T (ver las líneas 4 – 11).

En el caso en que todos los vértices hijos de x en T sean distintos a los vértices hijos de x en T' , el número de llamados que realiza la rutina M-Corregir Hijos a φ en T para que el conjunto de vértices hijos de x en T sea igual al conjunto de vértices hijos de x en T' es a lo más dos veces el número de vértices hijos de x en T y como el número de vértices hijos de x en T es igual a $d_T(x) - 1$, entonces el número de llamados que se hacen a φ es a lo más $2(d_T(x) - 1)$.

Por otra parte, dado que cualquier árbol tiene al menos dos vértices de grado uno, entonces el número de vértices cuyo grado es mayor que o igual a dos en T' es a lo más el orden de T' menos dos. Entonces, al considerar el funcionamiento del algoritmo Mixto y que el orden de T' es n , las rutinas M-Corregir Grado y M-Corregir Hijos operan a lo más sobre $n - 2$ vértices en T .

En el caso en que las rutinas M-Corregir Grado y M-Corregir Hijos operen sobre $k \leq n - 2$ vértices en T , sean x_1, x_2, \dots, x_k los vértices de grado mayor que o igual a dos en T' etiquetados en el orden en el que opera el algoritmo Mixto para corregir su grado y sus hijos. Asimismo sean $x_{k+1}, x_{k+2}, \dots, x_n$ el resto de los vértices en T' cuyo grado es igual a uno. Observemos que una vez que la rutina M-Corregir Grado hace que el grado del último vértice x_k en T sea igual al grado que tiene en T' el conjunto de vértices hijos de x_k en T coincide con el conjunto de vértices hijos de x_k en T' porque las rutinas M-Corregir Grado y M-Corregir Hijos operaron previamente en los vértices x_1, x_2, \dots, x_{k-1} en T .

Entonces, al considerar todo lo anterior el número de llamados que se hacen a la transformación φ para ir de T a T' es:

$$M_\varphi \leq 1 + 2d_T(x_1) + \sum_{i=2}^{k-1} 2 + 2(d_T(x_i) - 1) + 2$$

Donde el lado derecho de la desigualdad está conformado por tres sumandos. El primer sumando corresponde al número de llamados que se realizan a lo más a la transformación φ para hacer que $d_T(x_1) = d_{T'}(x_1)$ y el conjunto de vértices hijos de x_1 en T tenga los mismos elementos que el conjunto de vértices hijos de x_1 en T' , el segundo sumando corresponde al número de llamados que a lo más se hacen a φ

para que $d_T(x_i) = d_{T'}(x_i)$ y el conjunto de vértices hijos de x_i en T tenga los mismos elementos que el conjunto de vértices hijos de x_i en T' , para cada $2 \leq i \leq k-1$ y el último sumando, que es dos, corresponde al número de llamados a φ que bastarían para hacer que $d_T(x_{k-1}) = d_{T'}(x_{k-1})$ y el conjunto de vértices hijos de x_k en T tenga los mismos elementos que el conjunto de vértices hijos de x_k en T' . De aquí

$$M_\varphi \leq 1 + 2d_T(x_1) + \sum_{i=2}^{k-1} 2d_T(x_i) + 2 = 3 + 2 \sum_{i=1}^{k-1} d_T(x_i).$$

Por otro lado, dado que $k \leq n-2$, $d_T(x_k) \geq 2$ y $d_T(x_{k+1}) = d_T(x_{k+2}) = \dots = d_T(x_n) = 1$, entonces

$$\sum_{i=1}^{k-1} d_T(x_i) = \sum_{i=1}^n d_T(x_i) - \sum_{i=k}^n d_T(x_i) = 2(n-1) - \sum_{i=k}^n d_T(x_i) \leq 2(n-1) - 4 = 2n-6.$$

Por lo cual,

$$3 + 2 \sum_{i=1}^{k-1} d_T(x_i) \leq 3 + 2(2n-6) + 2 = 4n-9$$

$$\therefore M_\varphi \leq 4n-9.$$

□

A continuación damos a conocer el resultado sobre la cota superior concerniente al número de llamados que realiza el algoritmo Simple a la transformación φ para ir de T a T'

Teorema 12. Sean T y T' árboles raíz r' en \mathcal{T}_σ . El número de llamados que realiza el algoritmo Simple a la transformación φ para ir de T a T' es M_φ y satisface que $M_\varphi \leq 5n-10$, donde n es el orden de los arboles T y T' .

Demostración. Dado el funcionamiento del algoritmo Simple, notemos que para todo vértice x encontrado en cierto nivel N de T' tal que $d_{T'}(x) \geq 2$ se llama a la rutina S-Corregir Grado y se compara $d_T(x)$ con $d_{T'}(x)$. En caso que el $d_T(x)$ sea distinto al $d_{T'}(x)$, la rutina S-Corregir Grado llama a φ al menos dos veces con la finalidad de hacer que $d_T(x) = d_{T'}(x)$. Notemos que en el caso que $d_T(x) = d_{T'}(x)$, la rutina S-Corregir Grado no realiza ningún llamado a φ .

Posteriormente se llama a la rutina S-Corregir Hijos donde por cada vértice hijo v de x en T que sea diferente a los vértices hijos de x en T' la rutina S-Corregir Hijos llama a φ a lo más tres veces para intercambiar al vértice v por un vértice deseado z en T .

En el caso que todos los vértices hijos de x en T son diferentes a los vértices hijos de x en T' , el número de llamados que realiza la rutina S-Corregir Hijos a φ para que los vértices hijos de x en T sean iguales a los vértices hijos de x en T' es a lo más tres veces el número de vértices hijos de x en T . Notemos que el número de hijos del vértice raíz r' es $d_T(r')$ y el número de vértices hijos de x es diferente del de r' .

Por otra parte, dado que cualquier árbol tiene al menos dos vértices de grado uno, entonces el número de vértices cuyo grado es mayor que o igual a dos en T' es el orden de T menos dos, es decir, $n - 2$. Por lo cual, los algoritmos S-Corregir Grado y S-Corregir Hijos operan a lo más sobre $n - 2$ vértices en T .

En el caso en el que los algoritmos S-Corregir Grado y S-Corregir Hijos operen sobre $k \leq n - 2$ vértices en T , sean x_1, x_2, \dots, x_k , estos vértices etiquetados en el orden en el que operan las rutinas S-Corregir Grado y S-Corregir hijos. Asimismo sean $x_{k+1}, x_{k+2}, \dots, x_n$ el resto de los vértices de grado uno en T' . Observemos que al operar la rutina S-Corregir Grado sobre el último vértice x_k en T haciendo que $d_T(x_k) = d_{T'}(x_k)$ el conjunto de vértices hijos de x_k en T es igual al conjunto de vértices hijos de x_k en T' debido a que las rutinas S-Corregir Grado y S-Corregir hijos ya operaron en los vértices x_1, x_2, \dots, x_{k-1} en T .

Por lo cual, el número total de llamados que se hace a la transformación φ para ir de T a T' es:

$$M_\varphi \leq 2 + 3d_T(x_1) + \sum_{i=2}^{k-1} (2 + 3(d_T(x_i) - 1)) + 2.$$

Donde el lado derecho de la desigualdad está conformado por tres sumandos. El primer sumando corresponde al número de llamados que se hacen a la transformación φ con la finalidad de que $d_T(x_1) = d_{T'}(x_1)$ y el conjunto de vértices hijos de x_1 en T tenga los mismos elementos que el conjunto de vértices hijos de x_1 en T' , el segundo sumando corresponde al número de llamados que se hacen a φ para que $d_T(x_i) = d_{T'}(x_i)$ y el conjunto de vértices hijos de x_i en T tenga los mismos elementos que el conjunto de vértices hijos de x_i en T' , para cada $2 \leq i \leq k - 1$ y el último sumando que es dos corresponde al número de llamados a φ que bastarían para hacer que $d_T(x_{k-1}) = d_{T'}(x_{k-1})$ y el conjunto de vértices hijos de x_k en T tenga el mismo conjunto de elementos que el conjunto de vértices hijos de x_k en T' . De aquí

$$\begin{aligned}
M_\varphi &\leq (2 + 3d_T(x_1) - 3 + 3) + \sum_{i=2}^{k-1} (2 + 3(d_T(x_i) - 1)) + 2 \\
&= (2 + 3(d_T(x_1) - 1)) + \sum_{i=2}^{k-1} (2 + 3(d_T(x_i) - 1)) + 5 \\
&= \sum_{i=1}^{k-1} 2 + 3(d_T(x_i) - 1) + 5 = \sum_{i=1}^{k-1} 2 + 3 \sum_{i=1}^{k-1} (d_T(x_i) - 1) + 5.
\end{aligned}$$

Por otro lado, dado que:

- (a) $k \leq n - 2$, entonces $k - 1 \leq n - 3$.
- (b) $d_T(x_k) \geq 2$
- (c) $d_T(x_{k+1}) = d_T(x_{k+2}) = \dots = d_T(x_n) = 1$.

Entonces

$$\begin{aligned}
\sum_{i=1}^{k-1} (d_T(x_i) - 1) &= \sum_{i=1}^n (d_T(x_i) - 1) - \sum_{i=k}^n (d_T(x_i) - 1) \\
&= (2(n - 1) - n) - \sum_{i=k}^n (d_T(x_i) - 1) \\
&= (n - 2) - \sum_{i=k}^n (d_T(x_i) - 1) \leq (n - 2) - 1 = n - 3.
\end{aligned}$$

Por lo cual,

$$\sum_{i=1}^{k-1} 2 + 3 \sum_{i=1}^{k-1} (d_T(x_i) - 1) + 5 = 2(n - 3) + 3(n - 3) + 5 = 5n - 10.$$

$$\therefore M_\varphi \leq 5n - 10.$$

□

Conclusiones

En este trabajo damos una condición de tipo Ore en los grados de los vértices de una gráfica G que garantiza la existencia de un árbol generador de G con sucesión de grados predeterminada. Asimismo, demostramos que dada una sucesión arbórea σ , el problema de decidir si una gráfica G tiene un árbol generador con sucesión de grados σ pertenece a la clase de problemas NP -completos.

Por otro lado consideramos el conjunto Π_σ de todos los árboles T cuya sucesión de grados es una permutación de una sucesión arbórea σ y definimos una transformación φ que permite transitar en todo el conjunto Π_σ . Damos dos variantes de un algoritmo que al aplicarlo a dos árboles T y T' en Π_σ , produce una sucesión de árboles S_0, S_1, \dots, S_k en Π_σ tales que $T = S_0$, $T' = S_k$ y que para $i = 1, 2, \dots, k$, T_k se obtiene de T_{k-1} al aplicar la transformación φ .

Trabajo a futuro

Además de trabajar en la conjetura de Enomoto y Ozeki (conjetura 1), pensamos estudiar otras condiciones en la sucesión de grados de una gráfica G que aseguren la existencia de árboles generadores con sucesión de grados específica. Quizás condiciones como la del tipo del teorema de Win (teorema 2) en la que considera la suma de los grados de más de dos vértices de G o como la del teorema de Rivera-Campo (teorema 4) que considera también la conexidad de la gráfica G .

Con respecto a la transformación φ , sería interesante estudiar una generalización, digamos ψ , que transforme una gráfica arbitraria G en otra gráfica G' cuya sucesión de grados sea una permutación de la sucesión de grados de G conservando la propiedad de que usando iteradamente la transformación ψ podamos convertir una gráfica G con sucesión de grados σ en cualquier otra gráfica cuya sucesión de grados sea una permutación de σ . Una posibilidad sería la siguiente: dados dos vértices u y v de G , la transformación ψ intercambiaría cada arista ux de G por una arista xv y cada arista yv de G por una arista yu .

Bibliografía

- [1] Berge, C.: *Graphs and Hypergraphs*, Elsevier, New York, 1973.
- [2] Broersma, H.; Tuinstra, H.: Independence trees and Hamilton cycles, *J. Graph Theory* 29 (1998), 227 – 237.
- [3] Broersma, H.; Li, X.: The connectivity of the leaf-exchange spanning tree graph of a graphs, *Ars Combin.* 43 (1996), 225 – 231.
- [4] Bondy. J. A.; Murty, U. S. R.: *Graph Theory with Applications*, The McMillan Press, 1976.
- [5] Chvátal, V.; Erdős, P.: A note on hamiltonian circuits, *Discrete Math.* 2 (1972), 111 – 113.
- [6] Cummins, R.: Hamilton circuits in tree graphs, *IEEE Trans. Circuit Theory* 13 (1966), 82 – 90.
- [7] Enomoto, H.; Ozeki, K.: The independence number condition for the existence of a spanning f -tree, *J. Graph Theory* 65 (2010), 173 – 184.
- [8] Erdős, P.; Gallai, T.: Graphs with prescribed degrees of vertices (en húngaro), *Mat. Lapok* 11 (1960), 264 – 274.
- [9] Fekete, S. P.; Khuller, S.; Klemmstein, M.; Raghavachari, B.; Young N.: A Network-Flow Technique for Finding Low-Weight Bounded-Degree Spanning Trees, *J. Algorithms* 24 No 2 (1997), 310 – 324.
- [10] Figueroa Gutiérrez, A. P.: La gráfica de bases de un matroide, Tesis de Doctorado, Facultad de Ciencias, Universidad Nacional Autónoma de México, 2009.

- [11] Fresán Figueroa, J. A.: La gráfica de árboles con grados fijos, Tesis de Maestría, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana-I, 2012.
- [12] Hakimi, S. L: On the realizability of a set of integers as degrees of vertices of a graph, *J. Soc. Indust. Appl. Math.* 10 (1962), 496 – 506.
- [13] Harary, F.; Mokken, R. J.; Plantholt, M.: Interpolation theorem for diameters of spanning trees, *IEEE Trans. Circuits and Systems* 30 No 7 (1983), 429 – 432.
- [14] Havel, V.: A remark on the existence of finite graphs (en checo), *Časopis Pěst. Mat* 80 (1955), 477 – 480.
- [15] Heinrich, K.; Liu, G.: A lower bound on the number of spanning trees with k end-vertices, *J. Graph Theory* 12 No 1 (1988), 95 – 100.
- [16] Martínez-Cuero, M. E.: Árboles generadores con peso mínimo y grados acotados, Tesis de Maestría, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana-I, 2015.
- [17] Martínez-Cuero, M. E.; Rivera-Campo, E.: A note on spanning trees with specified degree sequence, *Bol. Soc. Mat. Mex.* 26 (2020), 241 - 245.
- [18] Li, X.; Neumann-Lara, V.; Rivera-Campo, E.: On a tree graph defined by a set of cycles, *Discrete Math.* 271 (2003), 303 – 310.
- [19] Neumann-Lara, V.; Rivera-Campo, E.: Spanning trees with bounded degrees, *Combinatorica* 11 (1991), 55 – 61.
- [20] Ore, Ø.: Note on Hamilton Circuits, *Amer. Math. Mon.* 67 (1960), 55.
- [21] Rivera-Campo, E.: A note on matchings and spanning trees with bounded degrees, *Graphs Combin.* 13 (1997), 159 – 165.
- [22] Rivera-Campo, E.: Spanning trees with small degrees and few leaves, *Appl. Math. Lett.* 25 (2012), 1444 – 1446.
- [23] Wilson, R. J.; Introduction to Graph Theory, Longman Group Ltd., 1972.
- [24] Win, S.: Existenz von Gerüsten Mit vorgeschriebenem Maximalgrad in Graphen, *Abh. Math. Sem. Univ. Hamburg* 43 (1975), 263 – 267.



Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

ACTA DE DISERTACIÓN PÚBLICA

No. 00086

Matrícula: 2162800046

Árboles generadores con sucesiones de grados dadas.

En la Ciudad de México, se presentaron a las 16:30 horas del día 8 del mes de septiembre del año 2023 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

- DRA. RITHA ESTHER ZUAZUA VEGA
- DR. JOAQUIN TEY CARRERA
- DRA. MUCUY KAK DEL CARMEN GUEVARA AGUIRRE
- DR. ILAN ABRAHAM GOLDFEDER ORTIZ
- DR. EDUARDO RIVERA CAMPO



MARIA ELENA MARTINEZ CUERO
ALUMNA

Bajo la Presidencia de la primera y con carácter de Secretario el último, se reunieron a la presentación de la Disertación Pública cuya denominación aparece al margen, para la obtención del grado de:

DOCTORA EN CIENCIAS (MATEMATICAS)

DE: MARIA ELENA MARTINEZ CUERO

y de acuerdo con el artículo 78 fracción IV del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

Aprobar

Acto continuo, la presidenta del jurado comunicó a la interesada el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

REVISÓ

MTRA. ROSALIA SERRANO DE LA PAZ
DIRECTORA DE SISTEMAS ESCOLARES

DIRECTOR DE LA DIVISIÓN DE CBI

Roman Linares Romero
DR. ROMAN LINARES ROMERO

PRESIDENTA

Ritha Esther Zuazua Vega
DRA. RITHA ESTHER ZUAZUA VEGA

VOCAL

Joaquin Tey Carrera
DR. JOAQUIN TEY CARRERA

VOCAL

Mucuy Kak del Carmen Guevara Aguirre
DRA. MUCUY KAK DEL CARMEN GUEVARA AGUIRRE

VOCAL

Ilan Abraham Goldfeder Ortiz
DR. ILAN ABRAHAM GOLDFEDER ORTIZ

SECRETARIO

Eduardo Rivera Campo
DR. EDUARDO RIVERA CAMPO