

95
201



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE CIENCIAS

RESULTADOS RECIENTES SOBRE EL ANCHO DE BANDA EN TEORIA DE GRAFICAS

T E S I S
Que para obtener el título de
A C T U A R I A
p r e s e n t a
VERONICA ROMERO GARCIA



Director de Tesis: M. en C. VIRGINIA ABRIN BATULE

México, D. F.

1997

TESIS CON
FALLA DE ORIGEN



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

PAGINACION VARIA

COMPLETA LA INFORMACION



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

M. en C. Virginia Abrín Batule
Jefe de la División de Estudios Profesionales de la
Facultad de Ciencias
P r e s e n t e

Comunicamos a usted que hemos revisado el trabajo de Tesis:

RESULTADOS RECIENTES SOBRE EL ANCHO DE BANDA EN TEORIA DE
GRAFICAS

realizado por VERONICA ROMERO GARCIA

con número de cuenta 0922043-6 , pasante de la carrera de ACTUARIA

Dicho trabajo cuenta con nuestro voto aprobatorio.

Atentamente

Director de Tesis	M. EN C. VIRGINIA ABRIN DATULE
Propietario	DR. ZEFERINO PARADA GARCIA
Propietario	DR. FERNANDO BRAMBILA PAZ
Suplente	M. EN E. ARTURO LORENZO VALDES
Suplente	ACT. MAURICIO AGUILAR GONZALEZ

Virginia Abrin Batule
Parada G.
Fernando
Mauricio Aguilar Gonzalez

Consejo Departamental de Matemáticas


M. EN A.P. MA. DEL PILAR ALONSO REYES

Agradecimientos

*A mis padres, gracias
por todo su amor y su apoyo...
Los quiero mucho.*

A mis hermanos, con mucho cariño.

*A Eric, por el amor y la alegría con que
has llenado mi vida.*

*A todos mis amigos y profesores
por su motivación y ejemplo para alcanzar
una de mis metas.*

*A los miembros del jurado,
sin su apoyo no hubiera sido posible
la culminación de este trabajo.*

*Deseo expresar mi más profundo agradecimiento
a la maestra Virginia Abrín, muchas gracias por la
paciencia y el apoyo incondicional que me brindaste
en todo momento.*

Resultados Recientes Sobre el Ancho de Banda
en Teoría de Gráficas

Verónica Romero García

Director de Tesis
M. en C. Virginia Abrín Batule

Contenido

1	Gráficas y Complejidad Algorítmica	1
1.1	Teoría de las Gráficas	1
1.2	Complejidad Algorítmica	12
1.2.1	Problemas, Algoritmos y Complejidad.	12
1.2.2	Definición del orden de una función	15
1.2.3	Definición alterna del orden de una función	16
1.2.4	Complejidad Asintótica	17
1.2.5	Introducción a estructuras de datos	19
1.3	Introducción a NP-Completes	21
1.3.1	Definición formal de algoritmo	21
1.3.2	Las clases \mathcal{P} y \mathcal{NP}	22
2	Ancho de Banda de una Gráfica	31
2.1	Introducción	31
2.2	El Ancho de Banda y su Relación con Matrices.	35
2.3	El Problema del Ancho de Banda es NP-Completo.	38
2.4	El Ancho de Banda y otros Parámetros.	41
2.4.1	Cotas sobre el ancho de banda en gráficas conexas	43
2.4.2	El Ancho de Banda y el Tamaño de una Gráfica.	44
2.5	Ancho de banda de gráficas especiales	45
2.5.1	Ancho de banda de producto de gráficas	46
3	Ancho de Banda y Operaciones Locales	49
3.1	Introducción y Notación.	49
3.2	Cotas superiores e inferiores.	51

Lista de Figuras

1.1	Representación geométrica de una gráfica	2
1.2	Gráfica que ilustra la definición de adyacencia.	3
1.3	Una gráfica y su complemento.	4
1.4	Dos diferentes representaciones de $K_{3,3}$	5
1.5	Gráfica completa con 5 vértices K_5	6
1.6	Gráficas 3 – regulares, también llamadas cúbicas.	6
1.7	Una gráfica y dos de sus subgráficas.	7
1.8	Una gráfica para ilustrar la definición de camino.	7
1.9	Ciclo con 7 vértices C_7 y trayectoria de 5 vértices P_5	8
1.10	Ejemplo de un árbol	9
1.11	Las excentricidades de los vértices de un árbol.	11
1.12	Representación geométrica de una digráfica.	11
1.13	Una instancia del problema del agente viajero y un recorrido de longitud 27, el cuál es el mínimo posible en este caso.	13
1.14	Comparación de funciones de complejidad de tiempo polinomial y exponencial.	18
1.15	Efectos del mejoramiento de la tecnología sobre algunos algoritmos de tiempo polinomial y exponencial.	19
1.16	Representación gráfica de G	20
1.17	Representación por listas de adyacencia de la gráfica G	21
1.18	Una primera vista de las clases P y NP.	22
1.19	Representación gráfica de una máquina de Turing	23
1.20	Cinta antes de iniciar la computación	24
1.21	La clase de problemas NP-Completo.	26
2.1	Gráfica conexa con 8 vértices.	32
2.2	Al definir g en G $B_g(G) = 6$	33
2.3	Al definir f sobre los vértices de G , $B_f(G) = B(G)$	33
2.4	Conversión de una numeración f en una numeración propia f'	34

2.5	P_8^3	35
2.6	Numeración óptima de $K_{4,3}$ que proporciona un ancho de banda igual a 4.	46
2.7	Ejemplo de un producto de trayectorias: $P_3 \times P_3$	46
2.8	Composición de las gráficas G_1 y G_2 de la figura 2.7.	47
3.1	Una cota sobre el aumento del ancho de banda bajo adición de aristas.	54
3.2	G_n	59
3.3	$(G_n)_m^{u,v}$	60
3.4	$H_n = K_{2n+1,2}$	61
3.5	$(H_n)_m^{u,v} = K_{2n+1,1}$	62
3.6	G_n definida para probar cotas en subdivisión de aristas.	65
3.7	$B((G_n)_s^{(u,v)}) = \lfloor \frac{1}{4}(3B(G) - 1) \rfloor$	65
3.8	G_n definida para probar cotas en contracción de aristas.	66
3.9	$B((G_n)_c^{(u,v)}) = \lfloor \frac{1}{2}(3B(G) - 1) \rfloor$	67
3.10	Gráfica tal que bajo contracción de una arista el ancho de banda de la gráfica resultante disminuye en una unidad	68
3.11	Ejemplo de C_{2n+1}^2 con $n = 3$	69
3.12	Ejemplo de una gráfica tal que bajo subdivisión de aristas el ancho de banda se incrementa en 1.	69

Introducción

En el capítulo 1, se da una breve introducción a la Teoría de las Gráficas y se presentan conceptos básicos de la Teoría de la Complejidad Algorítmica, comenzando con la definición formal de un algoritmo utilizando máquinas de Turing y continuando con la descripción de las clases de problemas \mathcal{P} y \mathcal{NP} , finalmente concluimos este capítulo presentando un procedimiento para demostrar cuándo un problema es $NP - Completo$.

En el capítulo 2, se estudian teoremas relacionados con el ancho de banda de una gráfica, el cual lo podemos definir de la siguiente forma:

Dada una gráfica G con al menos una arista, sea $V(G)$ el conjunto de vértices de G y $E(G)$ el conjunto de aristas. Definimos una numeración f para los vértices de G como una función inyectiva de $V(G)$ a los enteros positivos.

Sea

$$B_f = \max_{(u,v) \in E(G)} \{|f(u) - f(v)|\}$$

y

$$B(G) = \min_f B_f$$

A B_f se le denomina el ancho de banda de G relativo a f y a $B(G)$ se le denomina el ancho de banda de G .

Quizá la más importante aplicación de la noción del ancho de banda aparece en conexión con matrices ralas. Dada una matriz rala de tamaño $n \times n$, $A = (a_{ij})$, sea G la gráfica cuyo conjunto de vértices es $V = \{v_1, \dots, v_n\}$ donde v_i es adyacente a v_j para $i \neq j$ si y solo si $a_{ij} \neq 0$ o $a_{ji} \neq 0$. Entonces $B(G) \leq k$ si y solo si existe una matriz de permutación P tal que todos los elementos distintos de cero de $P^T A P$ se encuentran sobre la diagonal principal o sobre alguna de las k primeras superdiagonales o las k primeras subdiagonales.

En cálculos que envuelven matrices ralas de gran tamaño es deseable tener una matriz almacenada de tal forma, que todos los elementos distintos

de cero se encuentren tan cerca de la diagonal como sea posible; ya que de esta forma la matriz puede manipularse, en la mayoría de los casos, con un considerable ahorro de tiempo y espacio de almacenamiento.

Finalmente, en el capítulo 3 encontramos que mientras el problema de encontrar el ancho de banda de una gráfica es NP -Completo existen algunas clases de gráficas para las cuales es posible encontrar su ancho de banda en tiempo polinomial y en particular podemos ver que si realizamos cierto tipo de operaciones locales en gráficas podemos encontrar cotas superiores para el ancho de banda de la gráfica resultante en términos del ancho de banda de la gráfica original y en algunos casos podemos encontrar la reenumeración óptima para estas gráficas.

Capítulo 1

Gráficas y Complejidad Algorítmica

En este capítulo introduciremos el lenguaje básico de la teoría de las gráficas y de la teoría de la complejidad algorítmica. Las definiciones y conceptos esbozados aquí servirán posteriormente como material de apoyo para los siguientes capítulos.

§1.1 Teoría de las Gráficas

Esta sección presenta la terminología básica y algunos resultados elementales de la teoría de gráficas.

Una gráfica $G = (V, E)$ consiste de un conjunto finito no vacío $V = \{v_1, v_2, \dots, v_p\}$ cuyos elementos¹ son llamados *vértices* junto con un conjunto no ordenado $E = \{e_1, e_2, \dots, e_q\}$ cuyos elementos son subconjuntos de V de tamaño 2 denominados *aristas*, esto es, $e_k = (i, j)$, donde $i, j \in V$, y se llaman *los extremos* de la arista e_k . Denotaremos el número de vértices en una gráfica por $p = |V|$ y el número de aristas por $q = |E|$.

¹La siguiente es una lista de sinónimos que son utilizados en la literatura: punto-línea, vértice-arista, nodo-arco, 0-simplejo-1-simplejo.

Ejemplo 1.1.1

$$\begin{aligned}
 G &= (V(G), E(G)) && \text{en donde} \\
 V(G) &= \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9\} \\
 E(G) &= \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\} \\
 e_1 &= (v_1, v_7), & e_2 &= (v_1, v_5), & e_3 &= (v_4, v_5), & e_4 &= (v_4, v_7), \\
 e_5 &= (v_6, v_8), & e_6 &= (v_2, v_8), & e_7 &= (v_2, v_3), & e_8 &= (v_8, v_3).
 \end{aligned}$$

La representación geométrica de una gráfica tiene lugar como un conjunto de puntos (*vértices*) en el espacio los cuales están interconectados por un conjunto de líneas (*aristas*). Los puntos o vértices están colocados arbitrariamente en el plano, y una línea conecta a la pareja de puntos i y j si y solo si $e = (i, j) \in E$.

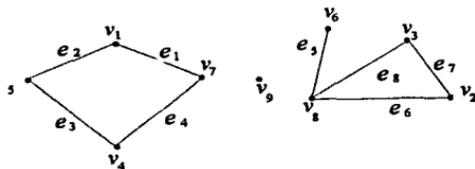


Figura 1.1: Representación geométrica de una gráfica

Si una arista e tiene a v como extremo, entonces decimos que e es *incidente* con v . Si $(u, v) \in E$ entonces u se dice ser *adyacente* a v . De forma semejante se tiene que dos aristas son *adyacentes* si tienen un extremo en común. En la figura 1.2 se ilustra esta definición mediante la gráfica G , los vértices u y v son *adyacentes* pero u y w no lo son; mientras que las aristas x y y son *incidentes* pero no lo son x y z .

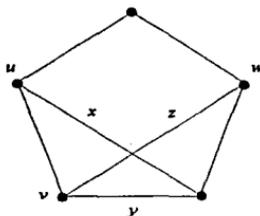


Figura 1.2: Gráfica que ilustra la definición de adyacencia.

El *grado* de un vértice v , que se denota $d(v)$, es el número de aristas incidentes con v y también lo podemos definir como el número de vértices adyacentes con v . Un vértice para el cual $d(v) = 0$ es llamado un vértice *aislado*. El siguiente teorema se refiere a los vértices de una gráfica.

Teorema 1.1.1 *El número de vértices de grado impar en una gráfica es par.*

Demostración:

Como cada arista tiene dos extremos entonces

$$2|E| = \sum_{v \in V} d(v)$$

Sean V_1 y V_2 tales que:

$V_1 = \{v_i \in V | d(v_i) = 2k\}$ (vértices de grado par)

$V_2 = \{v_j \in V | d(v_j) = 2k + 1\}$ (vértices de grado impar), $k \in \mathbb{N}$

De aquí que $V = V_1 \cup V_2$ y

$$2|E| = \sum_{v \in V} d(v) = \sum_{v_i \in V_1} d(v_i) + \sum_{v_j \in V_2} d(v_j)$$

para alguna $n \in \mathbb{N}$

$$2n = 2|E| - \sum_{v_i \in V_1} d(v_i) = \sum_{v_j \in V_2} d(v_j)$$

por lo tanto el número de vértices de grado impar es par ■

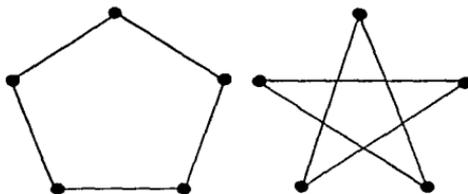


Figura 1.3: Una gráfica y su complemento.

El complemento de una gráfica $G = (V, E)$ es la gráfica $G^c = (V, E^c)$, en donde $E^c = \{e : e \notin E\}$, es decir, dos vértices son adyacentes en G^c si y solo si estos no son adyacentes en G .

Dos gráficas G y H son *isomorfas* (se escribe $G \cong H$) si existe una correspondencia uno a uno entre los vértices de G_1 y los vértices de G_2 que preserva adyacencias. Es decir, G y H se dicen *isomorfas* si existen biyecciones $\theta : V(G) \rightarrow V(H)$ y $\phi : E(G) \rightarrow E(H)$ tal que $e = (u, v) \in E(G)$ si y solo si $\phi(e) = (\theta(u), \theta(v))$; al par de funciones (θ, ϕ) se le llama un isomorfismo entre G y H . Para mostrar que dos gráficas son isomorfas, basta con indicar un isomorfismo entre ellas. Por ejemplo, las gráficas de la figura 1.4 son isomorfas (ambas representa a $K_{3,3}$); el par de funciones (θ, ϕ) definidas por

$$\theta(v_1) = u_1, \quad \theta(v_2) = u_2, \quad \theta(v_3) = u_3, \quad \theta(v_4) = u_4, \quad \theta(v_5) = u_5, \quad \theta(v_6) = u_6$$

y

$$\begin{aligned} \phi(e_1) &= a, & \phi(e_2) &= b, & \phi(e_3) &= c, \\ \phi(e_4) &= d, & \phi(e_5) &= e, & \phi(e_6) &= f, \\ \phi(e_7) &= g, & \phi(e_8) &= h, & \phi(e_9) &= i \end{aligned}$$

es un isomorfismo entre las gráficas

A continuación introduciremos algunas clases especiales de gráficas.

Una gráfica $G = (V, E)$ en donde todo par de vértices distintos son adyacentes se llama *completa*. La gráfica completa con n vértices se denota por K_n . La figura 1.5 muestra una gráfica completa con 5 vértices K_5 .

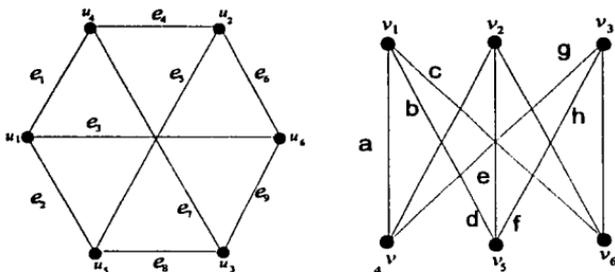
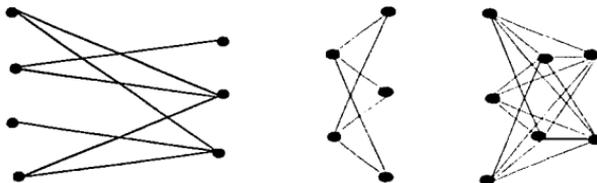


Figura 1.4: Dos diferentes representaciones de $K_{3,3}$.

En una gráfica *regular* todos los vértices tienen el mismo grado, si este es k entonces la gráfica es llamada k -regular. Nótese que K_n es $(n-1)$ -regular. La figura 1.6 muestra dos ejemplos de gráficas regulares, nótese que la primera de ellas es la gráfica completa con 4 vértices.

Una gráfica $G = (V, E)$ es llamada *bipartita* si existe una partición de V en dos conjuntos V_1 y V_2 ajenos, no vacíos y cuya unión es V , tal que cada arista de G tiene un extremo en V_1 y el otro en V_2 . Si cada vértice de V_1 está conectado a cada vértice de V_2 entonces G se dice que es una gráfica *bipartita completa*. En este caso denotamos la gráfica por $K_{n,m}$ donde $|V_1| = n$ y $|V_2| = m$. Podemos hacer una generalización de la definición de gráficas bipartitas a gráficas k -partitas donde k es un entero mayor que 2.



Gráfica bipartita. Ejemplos de gráficas k -partitas completas.

Una gráfica H es una *subgráfica* de G (se escribe $H \subseteq G$) si $V(H) \subseteq V(G)$

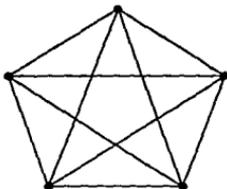


Figura 1.5: Gráfica completa con 5 vértices K_5

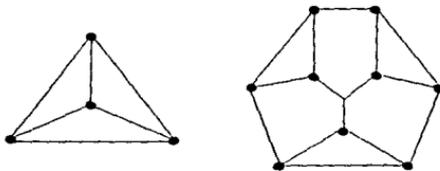


Figura 1.6: Gráficas 3 – regulares, también llamadas cúbicas.

y $E(H) \subseteq E(G)$, es decir H es una gráfica que tiene todos sus vértices y aristas en G . Si H es una subgráfica de G entonces G es llamada una *supergráfica* de H . Cuando $H \subseteq G$ pero $H \neq G$, se dice que H es una subgráfica propia de G . Una subgráfica *generadora* es una subgráfica que contiene todos los vértices de G . Una subgráfica de G *inducida* por un subconjunto de sus vértices, $V' \subset V$, es la gráfica consistente de V' y las aristas de G que tengan ambos extremos en V' . Se denota como $\langle V' \rangle$. En la figura 1.7, G_2 es subgráfica generadora pero G_1 no lo es; G_2 es una subgráfica inducida pero G_1 no.

Las siguientes definiciones juegan un papel importante en teoría de las gráficas. Una sucesión alternada de vértices y aristas $v_0, e_0, v_1, e_1, v_2, \dots, e_{k-1}, v_k$, $k \geq 1$, es llamada un *camino*. Una secuencia como la anterior en ocasiones se denota también por $v_0, v_1, v_2, \dots, v_{k-1}, v_k$, $k \geq 1$ y es llamada un $v_0 - v_k$

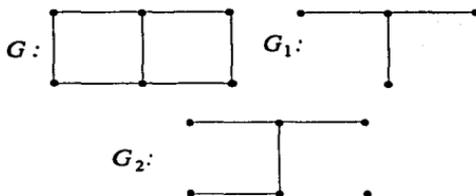


Figura 1.7: Una gráfica y dos de sus subgráficas.

camino si $(v_{i-1}, v_i) \in E(G)$ para $i = 1, \dots, k$. El nodo v_0 se denomina *origen*, el nodo v_k *destino* y los nodos $\{v_1, v_2, \dots, v_{k-1}\}$ nodos intermedios. También podemos representar un camino como una secuencia de aristas e_1, e_2, \dots, e_k , donde $e_i = (v_{i-1}, v_i) \in E$ para $i = 1, \dots, k$. La *longitud* de un camino $v_0, v_1, v_2, \dots, v_k$ o e_1, e_2, \dots, e_k es k , el número de aristas que contiene. Un camino es llamado una *paseo* si todas las aristas son distintas. Análogamente un camino es llamado *trayectoria* si no repite vértices. Un $v_0 - v_k$ camino es *cerrado* si $v_0 = v_k$. Llamamos *ciclo* a un paseo cerrado en donde los únicos vértices que se repiten son el inicial y el final. Una gráfica se dice *acíclica* si no contiene ciclos.

En la gráfica G de la figura 1.8, $v_1, v_2, v_5, v_2, v_3, v_5$ es un camino el cual no es una paseo; $v_1, v_2, v_5, v_4, v_2, v_3$ es un paseo el cual no es una trayectoria; v_1, v_2, v_5, v_4 es una trayectoria y v_4, v_2, v_3, v_5, v_4 es un ciclo.

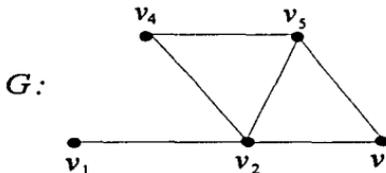


Figura 1.8: Una gráfica para ilustrar la definición de camino.

Se denota por C_n a la gráfica que es un ciclo con n vértices y por P_n a una trayectoria con n vértices.

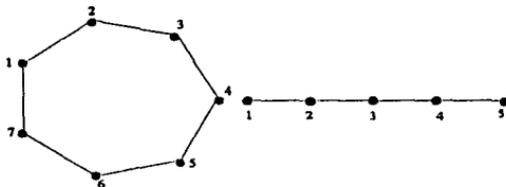


Figura 1.9: Ciclo con 7 vértices C_7 y trayectoria de 5 vértices P_5 .

Dos vértices v_i y v_j se dice que están conectados si existe una trayectoria de v_i a v_j . Por convención cada vértice está conectado con sí mismo.

La conexidad es una relación de equivalencia sobre el conjunto de vértices de una gráfica. De aquí que induzca una partición de dicho conjunto en subconjuntos V_1, V_2, \dots, V_k , en donde un par de vértices están relacionados si y solo si están en el mismo subconjunto. Las subgráficas inducidas por los subconjuntos de vértices V_1, V_2, \dots, V_k son llamadas las *componentes conexas* de la gráfica. Una gráfica es *conexa* si tiene una sola componente, en otro caso se dice que es *disconexa*.

Una subgráfica *generadora* de una gráfica conexa G es una subgráfica de G que se obtiene eliminando solo aristas de forma tal que cualquier par de vértices queden conectados, es decir se eliminan ciclos.

Sea H una componente conexa de G . Si $H - v$ es desconexa, entonces v se denomina *punto de corte*. Si H no contiene puntos de corte entonces decimos que H es un *bloque*, algunas veces también se le llama una gráfica o componente *2-conexa*. Si H contiene una arista e tal que $H - e$ no es conexa, entonces e se dice que es una *arista de corte* o *puente*. A los puntos que no son de grado 1, ni tampoco son de corte se les llama de *articulación*.

Una gráfica con uno o más puntos de corte es llamada una gráfica *separable*. Esto se refiere al hecho de que los bloques de una gráfica separable pueden identificarse desconectando la gráfica por cada punto de corte en turno de tal forma que cada componente de la nueva gráfica contiene una copia del punto de articulación. Es claro que cualquier gráfica es el resultado

de la unión de sus bloques.

Una clase importante de gráficas son los árboles, en donde un árbol es una gráfica conexa y acíclica. Un bosque es una gráfica cuyas componentes son árboles.

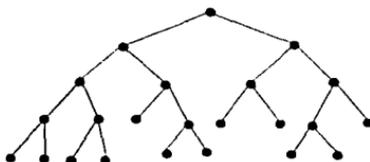


Figura 1.10: Ejemplo de un árbol

Proposición 1.1.1 Sea $G = (V, E)$ una gráfica con p vértices. Las siguientes proposiciones son equivalentes.

1. G es un árbol.
2. Existe una única trayectoria entre cualquier par de vértices en G .
3. G es conexa y $p = q + 1$.
4. G es acíclica y $p = q + 1$.
5. G es acíclica y si cualquier par de vértices no adyacentes de G son unidos por una arista e , entonces $G + e$ tiene exactamente un ciclo.
6. G es conexa, no se trata de K_p para $p \geq 3$ y si cualquier par de vértices no adyacentes de G son unidos por una arista e , entonces $G + e$ tiene exactamente un ciclo.
7. G no es ni $K_3 \cup K_1$ ni $K_3 \cup K_2$, $p = q + 1$ y si cualquier par de vértices no adyacentes de G son unidos por una arista e , entonces $G + e$ tiene exactamente un ciclo.

Los árboles son gráficas que tienen la propiedad de ser las gráficas conexas mínimas con respecto al número de aristas. Para una demostración de 1.1.1 vease [9] Harary.

La distancia $d(u, v)$ entre dos puntos u y v en G es la longitud de una trayectoria de longitud mínima que los une; en otro caso $d(u, v) = \infty$. Si G es conexa, la distancia es una métrica, es decir para todos los puntos $u, v, w \in V(G)$ se cumple que

1. $d(u, v) \geq 0$, con $d(u, v) = 0$ si y solo si $u = v$.
2. $d(u, v) = d(v, u)$
3. $d(u, v) + d(v, w) \geq d(u, w)$.

La excentricidad $e(v)$ de un vértice v se define como la distancia máxima que existe entre él y cualquier otro vértice de G . Esto se puede expresar:

$$e(v) = \max_{u \in V \setminus \{v\}} \{d(u, v)\}$$

El diámetro de una gráfica G corresponde a la excentricidad máxima calculada sobre el conjunto de vértices de G es decir:

$$D(G) = \max_{v \in V} \{e(v)\}$$

Análogamente definimos el radio de una gráfica G como la excentricidad mínima sobre el conjunto de vértices de G , y lo denotamos como:

$$r(G) = \min_{v \in V} \{e(v)\}$$

Un punto v se dice punto central si $e(v) = r(G)$ y el centro de G es el conjunto de todos sus puntos centrales.

Decimos que un nodo es periférico si su excentricidad es igual al diámetro de la gráfica; de aquí que la periferia de una gráfica G sea el conjunto de nodos periféricos:

$$\text{Periferia}(G) = \{v \in V \mid e(v) = D(G)\}$$

En la gráfica de la figura 1.11 se muestra la excentricidad de cada punto. Esta gráfica se trata de un árbol con diámetro 7, radio 4, y el centro consiste

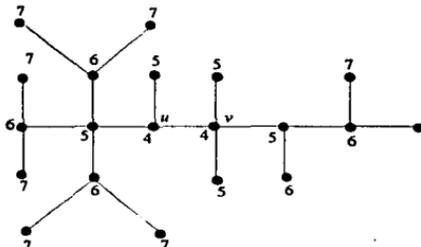


Figura 1.11: Las excentricidades de los vértices de un árbol.

de los dos vértices u y v con la mínima excentricidad que es 4 y todos los vértices de excentricidad 7 forman la periferia de la gráfica.

Una *gráfica dirigida* o *digráfica* $\mathcal{D} = (V, \mathcal{A})$ consiste de un conjunto no vacío $V = \{v_1, v_2, \dots, v_n\}$ cuyos elementos se llaman vértices y un conjunto $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$ cuyos elementos son subconjuntos ordenados de V de tamaño 2 llamados *arcos*. Para el arco (v_i, v_j) , se entiende que la dirección va del primer vértice v_i al segundo vértice v_j . En una digráfica, (v_i, v_j) y (v_j, v_i) son elementos distintos. Geométricamente representamos una digráfica con flechas o arcos para indicar el orden. En la figura 1.12 podemos observar una digráfica.

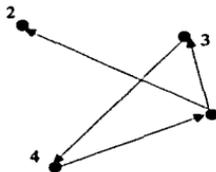


Figura 1.12: Representación geométrica de una digráfica.

Para completar las definiciones necesarias de Teoría de Gráficas en este

trabajo, introduciremos el concepto de gráficas con peso. En algunas aplicaciones es natural asignar un número a cada arista de una gráfica. Para cualquier arista e el número al cual nos referimos se escribe como $w(e)$ y es llamado el peso de la arista. Naturalmente la gráfica en cuestión es llamada gráfica con peso. El peso de una gráfica es igual a la suma de los pesos de sus aristas.

§1.2 Complejidad Algorítmica

1.2.1 Problemas, Algoritmos y Complejidad.

Para nuestros propósitos un *problema* será una pregunta general para ser contestada. Usualmente esta pregunta posee algunos *parámetros* o variables libres, cuyos valores no se especifican. Un problema se describe por medio de:

- (1) una descripción general de todos sus parámetros y
- (2) un enunciado de cuales propiedades de la respuesta o solución son requeridas para satisfacer el problema.

Una *instancia* de un problema se obtiene especificando valores particulares para todos los parámetros del problema.

Ejemplo 1.2.1 Como un ejemplo, consideremos el clásico "problema del agente viajero." Los parámetros de este problema consisten de un conjunto finito $C = \{c_1, c_2, \dots, c_m\}$ de "ciudades" y para cada par de ciudades (c_i, c_j) en C la "distancia" $d(c_i, c_j)$ entre ellas. Una solución es un ordenamiento $\langle c_{\pi(1)}, c_{\pi(2)}, \dots, c_{\pi(m)} \rangle$ que minimice el recorrido entre las ciudades dadas. Es decir tendremos que calcular:

$$\min \left\{ \left(\sum_{i=1}^{m-1} d(c_{\pi(i)}, c_{\pi(i+1)}) \right) + d(c_{\pi(m)}, c_{\pi(1)}) \right\}$$

Esta expresión representa la longitud de un recorrido que comienza en $c_{\pi(1)}$, visita cada una de las ciudades y después regresa a $c_{\pi(1)}$ directamente de la última ciudad $c_{\pi(m)}$.

Una instancia del problema del agente viajero, ilustrada en la figura 1.13, esta representada por $C = \{c_1, c_2, c_3, c_4\}$, $d(c_1, c_2) = 10$, $d(c_1, c_3) = 5$, $d(c_1, c_4) = 9$, $d(c_2, c_3) = 6$, $d(c_2, c_4) = 9$ y $d(c_3, c_4) = 3$. El ordenamiento $\langle c_1, c_2, c_4, c_3 \rangle$ es una solución para esta instancia y corresponde al recorrido de mínima longitud posible : 27.

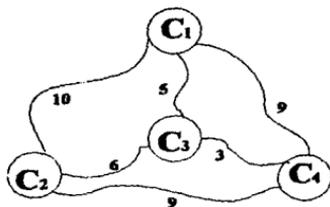


Figura 1.13: Una instancia del problema del agente viajero y un recorrido de longitud 27, el cuál es el mínimo posible en este caso.

Los algoritmos son en general pasos de procedimientos que se siguen para resolver un problema. Concretamente podemos suponerlos como programas escritos en algún lenguaje, procesados en una computadora.

Se dice que un algoritmo resuelve un problema Π si éste puede ser aplicado a cualquier instancia I de Π y se garantiza que siempre se produce una solución para la instancia I . En este caso el término "solución" es entendido estrictamente en el sentido introducido con anterioridad, de modo que, en particular, un algoritmo no resuelve el problema del agente viajero a menos que siempre construya un ordenamiento que proporcione un recorrido de longitud mínima.

En general estamos interesados en encontrar el algoritmo más "eficiente" para resolver un problema. En el sentido más amplio, la noción de eficiencia relaciona todos los recursos de cómputo necesarios para ejecutar un algoritmo. Sin embargo por algoritmo "más eficiente" entenderemos de aquí en adelante el más rápido. Como los requerimientos de tiempo son frecuentemente un factor dominante ya sea que un algoritmo específico sea lo bastante eficiente para usarse en la práctica o no, a partir de este momento nos concentraremos primordialmente en este recurso en particular al analizar la eficiencia de un algoritmo.

Los requerimientos de tiempo de un algoritmo son expresados por conveniencia en términos de una variable simple, el "tamaño" de una instancia del problema, el cual refleja la cantidad de datos de entrada necesarios para describir la instancia. Esto es conveniente ya que esperamos que la dificul-

tad relativa a cada instancia de un problema varíe aproximadamente con su tamaño. Frecuentemente el tamaño de las instancias de un problema son medidas de manera informal. Para el problema del agente viajero por ejemplo, el número de ciudades es comúnmente usado como tamaño de problema. Sin embargo una instancia que involucre m ciudades del problema incluye una colección de $m(m-1)/2$ números para definir la distancia entre las ciudades y el tamaño de estos números también contribuye en la cantidad de datos de entrada. Por lo tanto, si vamos a tratar los requerimientos de tiempo precisamente de manera matemática, tenemos que tener cuidado al definir el tamaño de las instancias de tal forma que todos estos factores sean tomados en cuenta.

Para hacer esto observemos que la descripción de una instancia del problema que proporcionamos como entrada para la computadora puede ser vista como una simple cadena de símbolos seleccionados de un alfabeto finito. Aunque existen muchas maneras diferentes para describir un problema, podemos suponer que una en particular ha sido elegida de antemano y que cada problema tiene asociado con ésta un esquema codificador fijo, el cual transforma las instancias del problema en las cadenas que las describen. La longitud de entrada para una instancia I de un problema Π se define como el número de símbolos en la descripción de I obtenidos del esquema codificador para Π . Este número es la longitud de entrada y es usado como la medida formal del tamaño de la instancia.

La función de tiempo-complejidad para un algoritmo expresa, sus requerimientos de tiempo dando, para cada posible longitud de entrada, la mayor cantidad de tiempo necesario para el algoritmo para resolver una instancia del problema de tal tamaño. Por supuesto esta función no está bien definida hasta que uno fija el esquema codificador que se usará para determinar la longitud de entrada y la computadora o el modelo de computadora que se usará para determinar el tiempo de ejecución.

La *complejidad* de un algoritmo es simplemente el número de pasos computacionales (sumas, multiplicaciones, comparaciones, etc.) que se requieren para transformar la entrada de datos en el resultado de un cómputo. Generalmente ésta es una función de la cantidad de datos de entrada, comúnmente llamada el *tamaño del problema*. Para algoritmos en Teoría de las Gráficas el tamaño del problema es determinado por una o quizá ambas variables, $|V| = p$ y $|E| = q$.

Para un problema de tamaño s , denotamos la complejidad de un algoritmo A de gráficas por $C_A(s)$, podemos omitir el subíndice A cuando no existen

ambigüedades. $C_A(s)$ puede variar significativamente si un algoritmo A es aplicado a gráficas estructuralmente diferentes, las cuales no obstante son del mismo tamaño. Necesitamos por consiguiente ser más específicos en nuestra definición. En este trabajo, $C_A(s)$ significará la complejidad en el peor caso. Es decir, es el número máximo sobre todos los tamaños de entrada s , de pasos computacionales requeridos para la ejecución de un algoritmo A . El *tiempo-complejidad esperado* es el *promedio* sobre todas las entradas de tamaño s , del número de pasos computacionales requeridos.

Las complejidades de dos algoritmos para el mismo problema en general difieren. Sean A_1 y A_2 tales algoritmos y suponemos que $C_{A_1}(n) = \frac{1}{2}n^2$ y que $C_{A_2}(n) = 5n$. Entonces decimos que A_2 es más rápido que A_1 para todos los problemas de tamaño $n > 10$. De hecho cualesquiera que sean los coeficientes de n^2 y n en estas expresiones, con la restricción de que tienen que ser positivos, A_2 podría ser más rápido que A_1 para todo n mayor que algún valor n_0 . Esto se debe a que el crecimiento asintótico de la función cuando el tamaño del problema tiende a infinito es más rápido en el caso de n^2 que el de n . La complejidad de A_2 se dice que es de menor orden que la de A_1 .

1.2.2 Definición del orden de una función

Definición 1.2.1 Dadas dos funciones positivas F y G cuyo dominio sean los números naturales, decimos que el orden de F es menor o igual al orden de G si existen k y n_0 constantes positivas tales que:

$$F(n) \leq K \cdot G(n) \quad \forall \quad n > n_0.$$

Si el orden de F es menor o igual que el orden de G entonces escribimos $F = O(G)$ o decimos que F es $O(G)$. F y G son del mismo orden si se cumple que $F = O(G)$ y que $G = O(F)$. Es conveniente escribir $\theta(G)$ para especificar el conjunto de todas las funciones que son del mismo orden que G . Aunque hemos definimos a $\theta(G)$ como un conjunto, por convención escribimos $F = \theta(G)$ para referirnos al hecho $F \in \theta(G)$. Ilustrando estas definiciones, vemos que $5n$ es $O\left(\frac{1}{2}n^2\right)$ pero que $5n \neq \theta\left(\frac{1}{2}n^2\right)$ porque $\frac{1}{2}n^2$ no es $O(5n)$. Notemos que los términos de menor orden de una función pueden ser ignorados al determinar el orden total. Esto significa, por ejemplo, que el polinomio $(3n^3 + 6n^2 + n + 5)$ es $O(3n^3)$. Es conveniente cuando especificamos el orden de una función hacerlo en términos de la función representativa más simple. Por ejemplo $(3n^3 + 6n^2)$ es $O(n^3)$ y $\left(\frac{1}{2}n^2\right)$ es $O(n^2)$.

1.2.3 Definición alterna del orden de una función

Cuando comparamos dos funciones en términos de su orden, es conveniente tomar la siguiente definición alternativa. Sea $\lim_{n \rightarrow \infty} F(n)/G(n) = L$, vemos que:

(i) Si $L = a$ constante positiva, entonces $F = \theta(G)$.

(ii) Si $L = 0$, entonces F es de orden menor que G .

(iii) Si $L = \infty$, entonces G es de orden menor que F .

Para ilustrar esta definición tomemos los siguientes ejemplos.

Ejemplo 1.2.2 Sean $F(n) = 3n^2 - 4n + 2$ y $G(n) = \frac{1}{2}n^2$.

$$\text{Entonces } L = \lim_{n \rightarrow \infty} F(n)/G(n) = \lim_{n \rightarrow \infty} \frac{3n^2 - 4n + 2}{\frac{1}{2}n^2} = 6,$$

por lo tanto $F = \theta(G)$.

Ejemplo 1.2.3 Sean $F(n) = \log_2 n$ y $G(n) = n$.

$$\text{Entonces } L = \lim_{n \rightarrow \infty} F(n)/G(n) = \lim_{n \rightarrow \infty} \frac{\log_2 n}{n} =$$

$$\lim_{n \rightarrow \infty} \frac{\ln n}{n} \cdot \log_2 e = \lim_{n \rightarrow \infty} \left(\frac{\log_2 e}{n} \right) = 0$$

En este caso podemos utilizar la regla de L'Hôpital ya que se cumple que

$$\lim_{n \rightarrow \infty} F(n) = \lim_{n \rightarrow \infty} G(n) = \infty$$

y dado que las derivadas F' y G' y el límite existen, entonces:

$$\lim_{n \rightarrow \infty} \frac{F(n)}{G(n)} = \lim_{n \rightarrow \infty} \frac{F'(n)}{G'(n)}$$

Como $L = 0$, vemos que $\log_2 n$ es de orden menor que n .

Ejemplo 1.2.4 En este caso sean $F(n) = x^n$ y $G(n) = n^k$, donde x y k son constantes arbitrarias pero fijas, ambas mayores que uno. Definimos $U(n) = F(n)/G(n)$, así que:

$$U(n+1)/U(n) = x(n/(n+1))^k$$

Así para k fija siempre podemos encontrar un valor suficientemente grande de n , digamos n_0 , tal que para $n > n_0$:

$$U(n+1) \simeq x \cdot U(n)$$

por lo tanto para $n \geq n_0$

$$U(n) \simeq x^{n-n_0} U(n_0)$$

$$L = \lim_{n \rightarrow \infty} U(n) = \infty$$

Así que F , función exponencial en n , es de orden mayor que cualquier polinomio en n .

1.2.4 Complejidad Asintótica

El orden de $C_A(s)$ describe el comportamiento asintótico de $C_A(s)$ cuando $s \rightarrow \infty$. Si $C_A(s)$ es $O(F)$, entonces A se dice que es un $O(F)$ -algoritmo. La complejidad asintótica determina esencialmente el tamaño más grande del problema que puede ser manipulado. Si dos algoritmos para el mismo problema son del mismo orden, entonces ninguno funciona significativamente mejor que el otro. Para s suficientemente grande, la diferencia es insignificante comparada con la que sería para dos algoritmos de distinto orden.

Un algoritmo se dice ser de tiempo polinomial para un problema X si $C_A(k)$ es $O(k^p)$ para algún p fijo.

La distinción entre los algoritmos de tiempo polinomial y exponencial tiene particular significado cuando consideramos la solución de problemas de gran tamaño. En la siguiente tabla se ilustran las diferencias en la velocidad de crecimiento entre varias funciones típicas de complejidad de cada tipo, donde las funciones expresan el tiempo de ejecución en términos de microsegundos. Obsérvese la velocidad explosiva de crecimiento para las dos funciones de complejidad exponencial.

Resulta interesante realizar un examen de los efectos del perfeccionamiento de la tecnología de cómputo sobre algoritmos de cierta complejidad. La siguiente tabla muestra cómo el problema de mayor tamaño que puede resolverse en una hora, puede cambiar si se cuenta con una computadora 100 o 1000 veces más rápida que la máquina actual. Obsérvese que con los algoritmos de orden 2^n un incremento del 1000% en la velocidad de cómputo únicamente incrementa en 10 el tamaño del problema mayor que puede resolverse en una hora, mientras que con los algoritmos de orden n^5 el tamaño del problema casi se cuadruplica.

Estas tablas indican algunas de las razones del por qué los algoritmos de tiempo polinomial son generalmente vistos como más deseables que los algoritmos de tiempo exponencial. Este punto, junto con la distinción entre dos tipos de algoritmos, es central para la noción de intratabilidad y también para la teoría de NP-Completes.

Función de Tiempo- Complejidad	Tamaño del problema					
	10	20	30	40	50	60
n	0001 segundo	0002 segundo	0003 segundo	0004 segundo	0005 segundo	0006 segundo
n^2	.001 segundo	.004 segundo	.006 segundo	.016 segundo	.025 segundo	.036 segundo
n^3	.001 segundo	.008 segundo	.027 segundo	.064 segundo	.125 segundo	.216 segundo
n^5	.1 segundo	3.2 segundos	24.3 segundos	1.7 minutos	3.2 minutos	13.0 minutos
2^n	.001 segundo	1.0 segundo	17.9 minutos	12.7 días	35.7 años	366 siglos
3^n	.059 segundo	58 minutos	6.5 años	3835 siglos	2×10^8 siglos	1.3×10^{11} siglos

Figura 1.14: Comparación de funciones de complejidad de tiempo polinomial y exponencial.

La existencia de un algoritmo para un problema no garantiza en términos prácticos que éste pueda resolverse. Se dice que un problema no está bien resuelto hasta que se conozca un algoritmo de tiempo polinomial para resolverlo. Por lo tanto, nos referiremos a un problema como *intratable* si es muy difícil que un algoritmo de tiempo polinomial lo resuelva. Diremos que un algoritmo es eficiente si se trata de cualquier $O(P)$ –algoritmo, donde P es un polinomio en el tamaño del problema. Esta distinción entre algoritmos de tiempo polinomial "eficientes" y algoritmos de tiempo exponencial "ineficientes" admite algunas excepciones cuando el problema de interés tiene cierto tamaño. Podemos ver en el cuadro de la figura 1.14 que el algoritmo de orden 2^n es más rápido que el algoritmo de orden n^5 para $n \leq 20$.

Existen algoritmos de tiempo exponencial que pueden ser bastante útiles en la práctica. Por ejemplo, en Programación Lineal el método Simplex es un algoritmo de orden exponencial, sin embargo en la práctica es bastante rápido, razón por la cual se recurre a él para resolver una gran cantidad de problemas. Otro ejemplo serían los algoritmos tipo "branch-and-bound" (ramificar y acotar) para el "problema de la mochila": han sido tan acertados que muchos lo consideran como un problema "bien resuelto". Desafortunadamente ejemplos como éstos son realmente raros y se conocen algoritmos de

Tamaño del problema más grande que puede resolverse en una hora

Función de Tiempo-Complejidad	Con la presente computadora	Con una computadora 100 veces más rápida	Con una computadora 1000 veces más rápida
n	N_1	$100 N_1$	$1000 N_1$
n^2	N_2	$10 N_2$	$31.6 N_2$
n^3	N_3	$4.64 N_3$	$10 N_3$
n^5	N_4	$2.5 N_4$	$3.98 N_4$
2^n	N_5	$N_5+6.64$	$N_5+9.97$
3^n	N_6	$N_6+4.19$	$N_6+6.29$

Figura 1.15: Efectos del mejoramiento de la tecnología sobre algunos algoritmos de tiempo polinomial y exponencial.

tiempo exponencial para muchos problemas, pero muy pocos de ellos son considerados como útiles en la práctica. Estos resultados sobre los algoritmos de orden exponencial mencionados no ha sido obstáculo para que se continúe investigando sobre la existencia de algoritmos de tiempo polinomial para resolver estos problemas.

El tiempo de complejidad se define como la complejidad en el peor caso, y el hecho de que un algoritmo sea de orden 2^n significa solamente que al menos un problema de tamaño n requiere de tal tiempo de ejecución. La mayoría de los problemas pueden requerir mucho menos tiempo que ese.

1.2.5 Introducción a estructuras de datos

En esta sección introduciremos representaciones elementales de gráficas para propósitos computacionales.

Matrices de Adyacencia y Listas de Adyacencia.

Las estructuras de datos que mostraremos son comúnmente usadas para representar gráficas. En particular veremos que el uso de listas de adyacencia representa una contribución importante para la eficiencia de un algoritmo.

Una *matriz de adyacencia* para la gráfica $G = (V, E)$ es una matriz A de tamaño $n \times n$, tal que:

$$A(i, j) = \begin{cases} 1 & \text{si } (i, j) \in E, \\ 0 & \text{en otro caso.} \end{cases}$$

Si G es una gráfica no dirigida entonces $A(i, j) = A(j, i)$, por lo que si G es una gráfica dirigida entonces A es generalmente asimétrica.

Para una gráfica con n vértices su matriz de adyacencia tiene n^2 entradas de aquí que cualquier algoritmo que requiera de inspeccionar estas entradas será al menos de orden n^2 , perdiéndose así la posibilidad de contar con un algoritmo lineal.

Si G es una gráfica para la cual el cociente q/p es muy pequeño su matriz de adyacencia es mala. Esto significa gran parte de sus entradas son iguales a cero, almacenarla y operar con ella resulta muy costoso, principalmente en tiempo de ejecución, de aquí que debemos contar con otro tipo de representaciones que minimicen espacio en memoria y tiempo de ejecución. Serán las listas de adyacencia o listas ligadas las que nos proporcionarán los requerimientos solicitados.

En la representación de *listas de adyacencia* de una gráfica cada vértice tiene una lista de sus vértices adyacentes asociada.

Ejemplo 1.2.5 Para la siguiente gráfica podemos ver su representación por listas de adyacencia y por matriz de adyacencia:

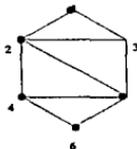


Figura 1.16: Representación gráfica de G .

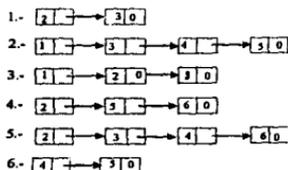


Figura 1.17: Representación por listas de adyacencia de la gráfica G .

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Matriz de adyacencia

§1.3 Introducción a NP-Completes

1.3.1 Definición formal de algoritmo

Hasta este momento hemos hablado de complejidad algorítmica sin tener una definición formal del concepto de algoritmo. Cualquier algoritmo consiste de un conjunto de instrucciones, dicha descripción puede ser incorporada en un programa de computadora. Precizando esta idea y basándonos en la definición de algoritmo de acuerdo a la tesis de Church (Kurt, 1984 [10]) diremos que existe un algoritmo para resolver un problema en particular si y solo si éste puede resolverse por medio de una *máquina de Turing* la cual se detiene para cualquier entrada de datos.

Las máquinas de Turing fueron introducida en 1936 por Alan Turing y proveen un modelo de computación que puede describirse brevemente además de que debido a su simplicidad representa un modelo que facilita la teoría en algoritmos. Otros modelos se usan comúnmente para probar teoremas de

complejidad, algunos de ellos utilizan acceso aleatorio de memoria y por lo tanto resultan ser más realistas. Puede probarse que todos estos modelos son equivalentes con respecto al tiempo de complejidad polinomial, es decir cada uno puede simular un cómputo sobre otro modelo con un costo de tiempo polinomial.

Por otra parte el utilizar polinomios constituye una ventaja algebraica para la teoría de la complejidad ya que representan una clase de funciones cerrada bajo las operaciones de adición, multiplicación y composición. Es decir, un polinomio es el resultado de la adición o multiplicación de dos polinomios y además puede obtenerse un polinomio de otro polinomio. Estas características son útiles desde dos puntos de vista: primero, un algoritmo complejo puede ser subdividido en un número pequeño de componentes cada uno de tiempo polinomial. Si la complejidad del algoritmo original está compuesta por la adición, multiplicación y composición de sus componentes, entonces ésta puede ser de tiempo polinomial. Segundo, es muy frecuente que un problema pueda resolverse transformándolo en otro. Si tenemos algoritmos eficientes tanto para el proceso de transformación como para el segundo problema, entonces tenemos un algoritmo eficiente para el problema inicial.

1.3.2 Las clases \mathcal{P} y \mathcal{NP} .

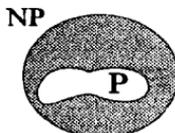


Figura 1.18: Una primera vista de las clases \mathcal{P} y \mathcal{NP} .

Con el fin de definir las clases de problemas \mathcal{P} y \mathcal{NP} primero tenemos que completar la definición formal de un algoritmo por medio de la descripción de la *Máquina de Turing (MT)*.

La máquina de Turing es un modelo simple de computadora universal, la cual tiene solo dos partes: una unidad de control y una unidad de almacenamiento. La unidad de almacenamiento es una sencilla cinta infinita.

La cinta está dividida en celdas que pueden almacenar sólo un carácter de un alfabeto finito cada una. La unidad de control tiene un número finito de estados, ésta revisa o explora la cinta por medio de una sencilla cabeza lectora/escritora.

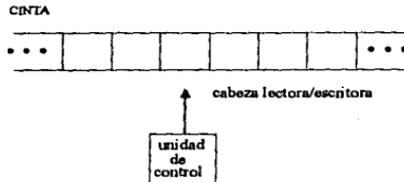


Figura 1.19: Representación gráfica de una máquina de Turing

La computación en una máquina de Turing se realiza mediante la repetición de un ciclo de acciones que envuelven a la cabeza lectora. Las acciones en un ciclo son las siguientes:

La cabeza lectora lee el contenido de una celda, entonces dependiendo de qué es lo que está leyendo y cuál es el estado, reemplaza el contenido de la celda con un nuevo símbolo (este puede ser de hecho el actual símbolo), entra en un nuevo estado (el cual puede ser el actual estado) y mueve una celda a la derecha o a la izquierda antes de empezar el siguiente ciclo. Los cambios ocurridos en un cualquier ciclo se determinan por medio de una quinteta, o instrucción, escrita así:

$$(q_i, t_i, q_f, t_f, m)$$

en donde q_i es el actual estado, q_f el nuevo estado, t_i es el actual símbolo en la celda y t_f el nuevo símbolo. La cabeza lectora puede o no moverse a la derecha o a la izquierda esto depende del valor que tome m , respectivamente, m tiene el valor +1 o el valor -1. Como el conjunto de estados, Q , y el conjunto de símbolos (llamado el alfabeto de la cinta), T , son finitos, cualquier cómputo puede completarse especificando por medio de un conjunto finito de quintetas. Tal conjunto constituye un programa (no secuencial) para la MT.

Cualquier programa lo podemos representar también por medio de una tabla que llamaremos tabla de Turing. La tabla de Turing especifica para

cada estado de la unidad de control y por cada símbolo del alfabeto un conjunto de posibles acciones (en el caso de máquinas no determinísticas) o una sola acción (en el caso de máquinas determinísticas). Una acción consiste de tres partes: cambiar el estado de la unidad de control, imprimir un nuevo símbolo sobre la celda bajo la cabeza lectora/escritora y mover la cabeza a lo más una celda a la derecha o a la izquierda. Por el momento diremos que un algoritmo es determinístico si para cualquier par (q_i, t_i) existe precisamente una quinteta para determinar (q_f, t_f, m) ; el algoritmo entonces se dice que se ejecuta en una *Máquina de Turing determinística* o *MTD*.

Convencionalmente el estado inicial para un programa en una *MT* es q_0 y una celda específica es leída. Si las celdas de la cinta están etiquetadas $\dots, C(-2), C(-1), C(0), C(1), C(2), \dots$, diremos que $C(0)$ será siempre la celda que comenzará a leerse. Los cálculos se detienen cuando la *MT* entra en un estado del conjunto de estados finales, $F \subset Q$. El resultado de la ejecución del algoritmo se encuentra codificado con los símbolos que quedan sobre la cinta o está indicado por el estado particular por que la *MT* se detuvo. El segundo caso es especialmente apropiado para *problemas de decisión* los cuales requieren simplemente un 'sí' o 'no' como respuesta.

Ejemplo 1.3.1 El siguiente es un ejemplo de una *MT*. Supongamos que queremos saber cuándo un entero $M (> 1)$ divide exactamente a un segundo entero N . La figura 1.20 muestra una *MTD* que resuelve este problema de decisión.

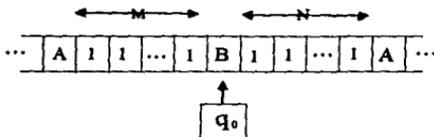


Figura 1.20: Cinta antes de iniciar la computación

Elementos del programa

$$Q = \{q_0, q_1, q_2, q_s, q_N\}, \quad F = \{q_s, q_N\}, \quad T = \{A, B, 0, 1, 2\}$$

Quintetas que representan el programa

$(q_0, A, q_2, A, +1)$	$(q_1, A, q_N, *, *)$	$(q_2, A, q_S, *, *)$
$(q_0, B, q_0, B, -1)$	$(q_1, B, q_1, B, +1)$	$(q_2, B, q_2, B, +1)$
$(q_0, 0, q_0, 0, -1)$	$(q_1, 0, q_1, 0, +1)$	$(q_2, 0, q_2, 0, +1)$
$(q_0, 1, q_1, 2, +1)$	$(q_1, 1, q_0, 0, +1)$	$(q_2, 1, q_0, 1, -1)$
$(q_0, 2, q_0, 2, -1)$	$(q_1, 2, q_1, 2, +1)$	$(q_2, 2, q_2, 1, +1)$

La parte superior del diagrama muestra la cinta al principio de los cálculos. Ambos, M y N , requieren una representación unaria en el algoritmo para trabajar. Los símbolos A y B son usados simplemente como puntuación. Asumimos que la celda $C(0)$ está ocupada por el símbolo B . El estado final q_S indica una respuesta 'sí' mientras que q_N indica 'no'. El conjunto de las 15 quintetas listadas en el diagrama constituye el programa de la MT , el cual opera substrayendo repetidamente M de N . Si al realizar parcialmente una de estas substracciones ninguno de los N queda en la cinta, entonces el estado de la MT es q_1 y la celda que se encuentra a la derecha contiene el símbolo A se examina. La quinteta apropiada entonces requiere que el estado sea q_N . Notemos que en esta quinteta el carácter $*$ indica la irrelevancia de especificar t_f o m . Similarmente, si una de estas substracciones ha sido completada y ninguno de los N permanece en la cinta entonces el estado de la MT es q_2 y la celda de la derecha que contiene el símbolo A se examina. La quinteta asignada causa que entre el estado q_1 .

Por conveniencia, la teoría de la NP - *Completes* ha sido diseñada para aplicarse solamente a problemas de decisión.² Abstractamente, un problema de decisión Π consiste simplemente de un conjunto D_Π de instancias y un subconjunto $Y_\Pi \subseteq D_\Pi$ de instancias que responden positivamente al problema. Esto no es una restricción debido a que podemos convertir cualquier problema en un problema de decisión muy parecido. Por ejemplo, para el problema del agente viajero podemos utilizar el problema de decisión: ¿Contiene la gráfica G un recorrido de longitud menor que k ? En este caso se dice que G es una instancia del problema (del agente viajero).

Podemos ahora definir la clase de problemas llamada \mathcal{P} . La clase \mathcal{P} consiste en los problemas decisión para los cuales existe un algoritmo de tiempo polinomial para resolverlos. En otras palabras, \mathcal{P} contiene todos los problemas de decisión que pueden ser resueltos, con $p(S)$ pasos computacionales, por medio de una Máquina de Turing Determinística MTD la cual se detiene

²Por un problema de decisión entendemos aquel que sólo tiene dos posibles soluciones, la respuesta es "sí" o la respuesta es "no".

para cualquier entrada de datos. Entonces podemos decir que \mathcal{P} consiste en aquellos problemas que pueden resolverse eficientemente.

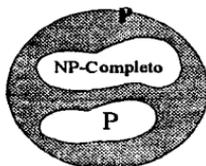


Figura 1.21: La clase de problemas NP-Completo.

Antes de introducir la clase de problemas llamada \mathcal{NP} de manera formal, trataremos de captar la idea intuitivamente. Para este propósito nos referiremos al problema de decisión del agente viajero definido con anterioridad. Como es sabido, no existe algún algoritmo eficiente para el problema del agente viajero. Sin embargo, consideremos la afirmación de que la respuesta a una instancia del problema de decisión fue 'sí' y que se cuenta con un recorrido particular. Es fácil verificar si este recorrido es de longitud menor que k . También sería simple construir un algoritmo de tiempo polinomial para este proceso de verificación. Por supuesto, cualquier conjunto de aristas podrían ser candidatos para tal afirmación, y si fuera posible aplicar el algoritmo de verificación *simultáneamente* a todos estos conjuntos entonces nuestro problema original podría ser resuelto en tiempo polinomial.

Existe una clase de problemas importantes desde el punto de vista práctico, para los cuales no existen algoritmos eficientes, pero algunos de ellos, como el problema del agente viajero, sin embargo se puede *verificar* una posible solución en tiempo polinomial. Este hecho es el que está intrínseco en la definición de la clase de problemas llamada \mathcal{NP} .

Dado un algoritmo de verificación (i.e. una MT), su entrada puede consistir de una instancia del problema y una hipótesis para la misma. En el caso del problema del agente viajero queremos saber si la gráfica G tiene un recorrido de longitud menor que k . Cualquier conjunto de aristas podría servirnos como un punto de partida (hipótesis) y G como una instancia del problema.

A continuación describiremos un algoritmo no determinístico del tipo que usaremos para definir formalmente la clase \mathcal{NP} . Tal algoritmo consistirá de dos etapas. La primer etapa simplemente produce un posible resultado (hipótesis) colocado en las celdas de la cinta $C(-1)$, $C(-2)$, ..., $C(-l)$, y deja la cabeza lectora/escritora sobre la celda $C(0)$ para continuar con la siguiente etapa. Una instancia del problema ocupará presumiblemente las celdas $C(1)$, $C(2)$, ..., $C(n)$. Cualquier cadena de símbolos del alfabeto bastará para dicho resultado. En esta etapa se produce una cadena arbitraria por operar no determinísticamente. El conjunto Q puede contener dos estados especiales q_a y q_b . El estado q_a es un estado de movimiento a la izquierda si la cabeza de la cinta está revisando una celda en blanco, posteriormente se imprime un símbolo arbitrario del alfabeto. Si el nuevo estado es q_a entonces la cabeza de la cinta se mueve hacia la izquierda. En otro caso el nuevo estado es q_b entonces se produce un movimiento hacia la derecha y de aquí en adelante la MT permanece en ese estado, ocasionando que la cabeza de la cinta se mueva siempre a la derecha (sin cambiar el contenido de las celdas) hasta que encuentra a $C(0)$ (el cual puede distinguirse por contener un símbolo particular del alfabeto). En este momento la MT está preparada para entrar en el estado q_0 y en condiciones para la segunda etapa del cómputo. Dicha etapa opera determinísticamente e intenta verificar la hipótesis para la instancia del problema codificado en las celdas $C(1)$ a $C(n)$.

Se dice que una Máquina de Turing no Determinística ($MTND$) resuelve una problema de decisión Π si las siguientes dos condiciones son válidas para todas las instancias I de D_Π :

- (a) Si Π es cierto para I , entonces existen varias hipótesis para I los cuales conducirán a la $MTDM$ a parar en un estado q_s .
- (b) Si Π es falso para I , entonces no existen hipótesis para I los cuales conducirían a la $MTDM$ a parar en un estado q_s .

Hasta este momento hemos definido a las $MTND$ sin hacer referencia a la complejidad. Diremos que la $MTND$ para el problema de decisión D opera en tiempo polinomial si para cada instancia I de D_Π que es cierta, existe una hipótesis que lleva a la etapa de verificación de la $MTND$ a parar en el estado q_s en $p(S)$ pasos computacionales, donde $p(S)$ es un polinomio en S , el tamaño de I .

La clase \mathcal{NP} contiene a todos los problemas que pueden resolverse utilizando una $MTND$ en tiempo polinomial.

La relación entre las clases \mathcal{P} y \mathcal{NP} es fundamental para la teoría de

NP – Completes. Cualquier problema de decisión que puede resolverse mediante un algoritmo determinístico de tiempo polinomial también puede resolverse utilizando un algoritmo no determinístico de tiempo polinomial. Para ver esto, simplemente necesitamos observar que cualquier algoritmo determinístico puede usarse como la etapa de verificación de un algoritmo no determinístico. Si $\Pi \in \mathcal{P}$, y A es cualquier algoritmo determinístico de tiempo polinomial para Π , podemos obtener un algoritmo no determinístico de tiempo polinomial para Π únicamente usando A como la etapa de verificación e ignorando la primer etapa. Por lo tanto, $\Pi \in \mathcal{P}$ implica $\Pi \in \mathcal{NP}$.

Definición 1.3.1 Definimos una transformación polinomial de un problema de decisión Π_1 a un problema de decisión Π_2 ($\Pi_1 \propto \Pi_2$) como una función $f : D_{\Pi_1} \rightarrow D_{\Pi_2}$ que satisface las siguientes condiciones:

1. f se puede calcular mediante un algoritmo de tiempo polinomial; y
2. para todo $I \in D_{\Pi_1}$, $I \in Y_{\Pi_1}$ si y solo si $f(I) \in Y_{\Pi_2}$.

Lema 1.3.1 Si $\Pi_1 \propto \Pi_2$, entonces $\Pi_2 \in \mathcal{P}$ implica $\Pi_1 \in \mathcal{P}$ (y equivalentemente $\Pi_1 \notin \mathcal{P}$ implica $\Pi_2 \notin \mathcal{P}$)³

La relación de transformación polinomial es especialmente de utilidad en la teoría de la NP-Completes debido a que es transitiva:

Lema 1.3.2 Si $\Pi_1 \propto \Pi_2$ y $\Pi_2 \propto \Pi_3$, entonces $\Pi_1 \propto \Pi_3$.

Demostración. Sea $f_1 : D_{\Pi_1} \rightarrow D_{\Pi_2}$ una transformación polinomial de Π_1 en Π_2 y sea $f_2 : D_{\Pi_2} \rightarrow D_{\Pi_3}$ una transformación polinomial de Π_2 en Π_3 . Entonces la función $f : D_{\Pi_1} \rightarrow D_{\Pi_3}$ definida como $f(I) = f_2(f_1(I))$ para toda $I \in \Pi_1$ es la transformación descada de Π_1 en Π_3 . Claramente, $f(I) \in Y_{\Pi_3}$ si y solo si $I \in Y_{\Pi_1}$, y el hecho de que f puede calcularse en tiempo polinomial se sigue de que f_1 y f_2 son transformaciones polinomiales. ■

Definición 1.3.2 Decimos que dos problemas de decisión Π_1 y Π_2 son polinomialmente equivalentes siempre que se cumplan $\Pi_1 \propto \Pi_2$ y $\Pi_2 \propto \Pi_1$.

³Para una demostración ver Garey [6], pp 34-35.

Definición 1.3.3 Un problema de decisión Π es *NP-Completo* si $\Pi \in \mathcal{NP}$ y para todos los problemas de decisión $\Pi' \in \mathcal{NP}$, $\Pi' \leq \Pi$. Por el lema 1.3.1 podemos identificar a los problemas *NP-Completo* como los problemas más difíciles de la clase *NP*.

Lema 1.3.3 Si Π_1 y $\Pi_2 \in \mathcal{NP}$, Π_1 es *NP-Completo* y $\Pi_1 \leq \Pi_2$ entonces Π_2 es *NP-Completo*.

Demostración. Dado que $\Pi_1 \in \mathcal{NP}$, todo lo que tenemos que hacer es mostrar que para cualquier $\Pi' \in \mathcal{NP}$, $\Pi' \leq \Pi_2$. Consideremos cualquier $\Pi' \in \mathcal{NP}$, como Π_1 es *NP-Completo*, por definición tenemos que $\Pi' \leq \Pi_1$. Ahora, por la transitividad de \leq y como por hipótesis sabemos que $\Pi_1 \leq \Pi_2$ entonces esto implica que $\Pi' \leq \Pi_2$. ■

Entonces, podemos pensar en el siguiente proceso para probar que un problema de decisión Π es *NP-Completo*, que consiste en los siguientes cuatro pasos:

- 1.- Mostrar que $\Pi \in \mathcal{NP}$,
- 2.- Seleccionar algún problema *NP-Completo* conocido Π' ,
- 3.- Construir una transformación f de Π' a Π ,
- 4.- Probar que f es una transformación polinomial.

El primer problema *NP-Completo* conocido se trata de un problema de decisión de Lógica Booleana, el cual es referido como el problema de **FACTIBILIDAD (SAT)**:

Sea $U = \{u_1, u_2, \dots, u_m\}$ un conjunto de variables booleanas. Una asignación verdadera para U es una función $t: U \rightarrow \{V, F\}$. Si $t(u) = V$ decimos que u es "verdadera" bajo t ; si $t(u) = F$ decimos que u es "falsa". Si u es una variable en U , entonces u y \bar{u} son literales sobre U . La literal u es verdadera bajo t si y solo si la variable u es verdadera bajo t ; la literal \bar{u} es verdadera si y solo si la variable u es falsa.

Una cláusula sobre U es un conjunto de literales sobre U , tal como $\{u_1, \bar{u}_3, u_8\}$. Este representa la disjunción de estas literales y se satisface por una asignación verdadera si y solo si al menos alguno de sus miembros es verdadero bajo tal asignación. Una colección C de cláusulas sobre U es

satisfactible si y solo si existe alguna asignación verdadera para U que satisface simultáneamente todas las cláusulas en C . Tal asignación verdadera es llamada una *asignación satisfactoria verdadera para C* . El problema de FACTIBILIDAD (*SAT* por sus siglas en inglés) se especifica como sigue:

INSTANCIA: un conjunto U de variables y una colección C de cláusulas sobre U .

PREGUNTA: ¿Existe una asignación satisfactoria verdadera para C ?

Teorema 1.3.1 (*Teorema Fundamental de Cook, 1971*) *El problema de FACTIBILIDAD (SAT) es NP - Completo*⁴

⁴Veáse la demostración en Garey [6], pp 39-44.

Capítulo 2

Ancho de Banda de una Gráfica

§2.1 Introducción

Sea $G = (V, E)$ una gráfica con n vértices.

Definición 2.1.4 Una función uno a uno $f : V \rightarrow \mathbb{N}$ será llamada una numeración o una ordenación de los vértices de G ; si $f(V) = \{1, 2, \dots, n\}$ se dirá una numeración propia de G .

Definición 2.1.5 Equivalentemente, una ordenación de una gráfica puede ser vista como una cadena de vértices y "blancos", donde cada vértice aparece exactamente una vez, por ejemplo b_c_d_a.

La correspondencia entre estas dos definiciones es simple dado que $f(v) = k$ si y solo si v es el k -ésimo de la cadena; de esta forma b_c_d_a corresponde a $f(a) = 7, f(b) = 1, f(c) = 3, f(d) = 5$, donde $V = \{a, b, c, d\}$.

Definición 2.1.6 El ancho de banda de una numeración f para G denotado por $B_f(G)$ es el número:

$$\max \{|f(u) - f(v)| : (u, v) \in E(G)\}$$

y el ancho de banda de G , denotado por $B(G)$, es el número:

$$\min \{B_f(G) : f \text{ es una numeración de } G\}$$

La definición puede restringirse a numeraciones propias de G , tomando en cuenta el siguiente procedimiento para obtener de cualquier numeración f una numeración propia f' tal que $B_{f'}(G) \leq B_f(G)$:

Sea f una numeración de G , $f : V \rightarrow \mathbb{N}$. Podemos construir una secuencia S con los vértices de la gráfica de tal forma que bajo f la secuencia S' de imágenes sea creciente, ahora podemos definir una nueva numeración f' de los vértices de la gráfica, asignando 1 al vértice cuya imagen bajo f sea la menor y así sucesivamente. Podemos afirmar que $f'(v_i) \leq f(v_i) \forall v_i \in V(G)$. Por lo tanto, si $(v_i, v_j) \in V(G)$, $B_{f'}(G) \leq |f'(v_i) - f'(v_j)| \leq |f(v_i) - f(v_j)| \leq B_f(G)$.

Una numeración f tal que $B_f(G) = B(G)$ se dirá una numeración óptima de G .

Ejemplo 2.1.2 La figura 2.1 muestra una gráfica conexa $G = (V, E)$ con 8 vértices.

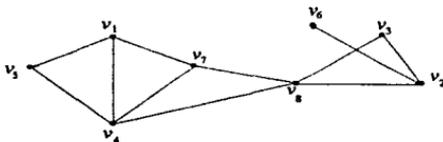


Figura 2.1: Gráfica conexa con 8 vértices.

En dicha gráfica podemos definir una numeración $g : V \rightarrow \mathbb{N}$ tal que $g(v_i) = i$, entonces el ancho de banda relativo a g lo podemos obtener calculando el máximo del siguiente conjunto

$$\left\{ |g(v_i) - g(v_j)| : (v_i, v_j) \in E(G) \right\} = \left\{ |1-5|, |1-7|, |5-4|, |4-7|, |4-1|, |4-8|, |8-7|, |8-2|, |8-3|, |6-2|, |3-2| \right\} = 6, \text{ es decir } B_g(G) = 6.$$

Pero la numeración g no es óptima, ya que podemos encontrar otra numeración $f : V \rightarrow \mathbb{N}$ definida como sigue: $f(v_1) = 7$, $f(v_2) = 2$, $f(v_3) = 3$, $f(v_4) = 6$, $f(v_5) = 8$, $f(v_6) = 1$, $f(v_7) = 5$, $f(v_8) = 4$ de tal forma que $B_f(G) \leq B_g(G)$, ya que $B_f(G) = 2$.

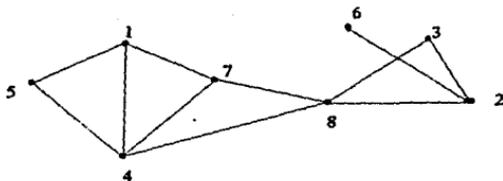
**Ejemplo 2.1.3**

Figura 2.2: Al definir g en G $B_g(G) = 6$

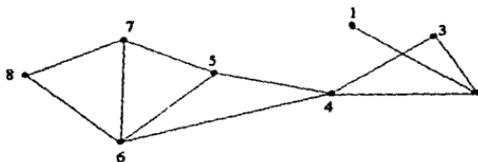


Figura 2.3: Al definir f sobre los vértices de G , $B_f(G) = B(G)$.

Ejemplo 2.1.4 Analicemos el procedimiento descrito para convertir una numeración f en una numeración propia f' . Consideremos la siguiente gráfica:

$$V(G) = \{v_1, v_2, v_3, v_4\}$$

$$E(G) = \{(v_1, v_2), (v_1, v_3), (v_2, v_3), (v_3, v_4)\}$$

Sea $f : V \rightarrow \mathbb{N}$, $f(v_1) = 7$, $f(v_2) = 31$, $f(v_3) = 2$ y $f(v_4) = 18$. En este caso $B_f(G) = 29$. Como $f(v_3) < f(v_1) < f(v_4) < f(v_2)$, definamos f' como $f'(v_3) = 1$, $f'(v_1) = 2$, $f'(v_4) = 3$, $f'(v_2) = 4$ y $B_{f'}(G) = 3$, por lo tanto $B_{f'}(G) \leq B_f(G)$.

A continuación enunciaremos algunos teoremas relacionados con el ancho de banda de una gráfica tratando de dar una breve demostración de los mismos.

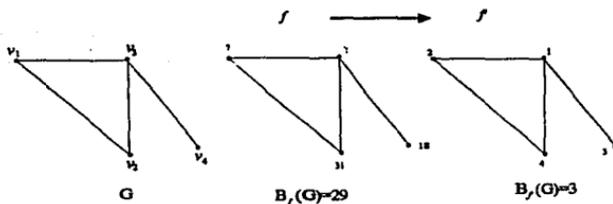


Figura 2.4: Conversión de una numeración f en una numeración propia f' .

Teorema 2.1.2 Si H es una subgráfica de G entonces

$$B(H) \leq B(G)$$

Demostración:

Sea f^* una numeración para G tal que $B_{f^*}(G) = B(G)$, por hipótesis sabemos que $V(H) \subseteq V(G)$ por lo tanto si restringimos la numeración f^* a los vértices de H podemos afirmar que $B(H) \leq B_{f^*}(H)$ ya que $B(H) = \min \{B_f(H) : f \text{ numeración de } H\}$ y por otra parte como $A(H) \subseteq A(G)$ entonces $B_{f^*}(H) \leq B_{f^*}(G) = B(G)$ por lo tanto $B(H) \leq B(G)$. ■

Teorema 2.1.3 Si los componentes de G son G_1, G_2, \dots, G_m , entonces

$$B(G) = \max \{B(G_1), B(G_2), \dots, B(G_m)\}.$$

Demostración:

Sea f cualquier ordenación de G entonces existe otra ordenación f' , con el mismo ancho de banda, en la cual las componentes de G aparecen sin mezclarse como "subcadenas" de f' . Podemos definir $f'(v) = f(v) + Nc(v)$, por ejemplo, donde $c(v)$ es el número de componente que contiene a v y N es suficientemente grande. Dado que el ancho de banda de una gráfica solo involucra vértices adyacentes podemos ver que si u y v son adyacentes se encuentran contenidos en la misma componente, entonces $c(v) = c(u)$ y por lo tanto

$$|f'(v) - f'(u)| = |f(v) + Nc(v) - f(u) - Nc(u)| = |f(v) - f(u)| \blacksquare$$

Definición 2.1.7 Por P_n^l denotamos a la gráfica que se forma por la l -ésima potencia de la trayectoria P_n , es decir, P_n^l tiene n vértices x_1, x_2, \dots, x_n y x_i, x_j son adyacentes en P_n^l si y solo si $0 \leq |i - j| \leq l$.

Teorema 2.1.4 Una gráfica G con n vértices tiene ancho de banda k si y solo si k es el menor entero tal que G puede ser sumergida en P_n^k .

Por ejemplo, cualquier gráfica con 8 vértices y ancho de banda tres podría ser sumergida en P_8^3 ,

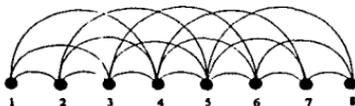


Figura 2.5: P_8^3 .

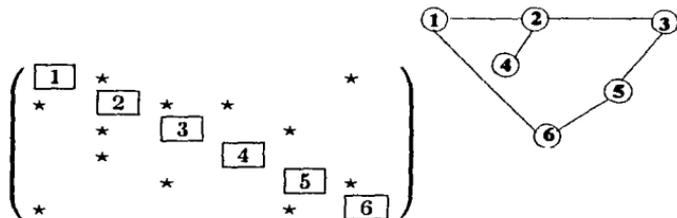
§2.2 El Ancho de Banda y su Relación con Matrices.

Definición 2.2.8 Sea A una matriz simétrica de orden n , se puede definir el ancho de banda de la matriz A como sigue:

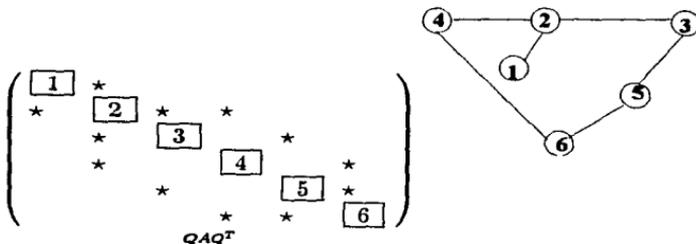
$$B(A) = \max_{\substack{j < i \\ 1 \leq i \leq n}} |i - j|$$

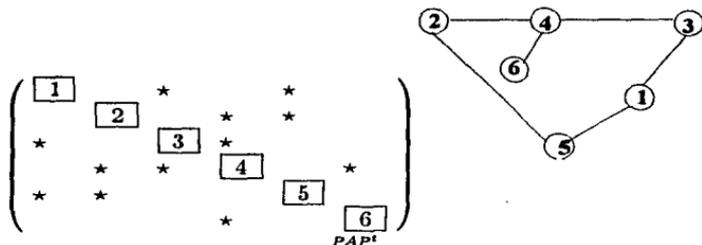
Dada una matriz rara de tamaño $n \times n$, $A = (a_{ij})$, sea G la gráfica cuyo conjunto de vértices es $V = \{v_1, \dots, v_n\}$ donde v_i es adyacente a v_j para $i \neq j$ si y solo si $a_{ij} \neq 0$ o $a_{ji} \neq 0$. La siguiente figura ilustra la estructura de una

matriz con su respectiva gráfica asociada. Denotamos el i -ésimo elemento de la diagonal de la matriz con \boxed{i} para enfatizar su correspondencia con el nodo i de la gráfica. Los elementos distintos de cero fuera de la diagonal están representados por "*".



Para cualquier matriz de permutación $P \neq I$ (de orden n), las gráficas no ordenadas (es decir que no tienen definida alguna ordenación o numeración) de A y PAP^T son las mismas pero las numeraciones asociadas a ellas son diferentes. Por lo tanto, la gráfica sin ordenar de A representa la estructura de A sin sugerir alguna numeración en particular, y representa la clase de equivalencia de las matrices PAP^T , donde P es cualquier matriz de permutación. Encontrar una "buena" permutación para A implica encontrar una buena numeración para los vértices de su gráfica correspondiente. La siguiente figura ilustra estos puntos, dadas dos diferentes permutaciones P y Q de A .





Entonces $B(G) \leq k$ si y solo si existe una matriz de permutación P tal que todos los elementos de $P^T A P$ se encuentran sobre la diagonal principal o sobre alguna de las k primeras superdiagonales o las k primeras subdiagonales.

El problema de minimizar el ancho de banda para matrices es muy importante para algunas aplicaciones de ingeniería tales como ingeniería de estructuras así como en dinámica de fluidos y análisis de redes. Usualmente, las matrices que aparecen en estas aplicaciones son ralas (es decir con un número considerable de elementos iguales a cero) y para resolver el correspondiente sistema de ecuaciones es deseable encontrar una matriz de permutación que produzca un ancho de banda mínimo. El método de eliminación de Gauss requiere $O(n^3)$ operaciones para resolver un sistema de $n \times n$ variables, sin embargo si puede encontrarse una matriz de permutación que coloque todas las entradas distintas de cero sobre las primeras b subdiagonales (superdiagonales), el número de pasos para obtener la solución se reduce en un orden de $O(nb^2)$. Un algoritmo cuyo propósito sea reducir b tanto como sea posible (para una matriz o para su gráfica asociada) es llamado un algoritmo de reducción del ancho de banda. Tales algoritmos, por los resultados que veremos en la siguiente sección, son vistos como aproximaciones en el sentido que no siempre garantizan descubrir el ancho de banda real de una matriz. En el libro de George A. y Joseph Liu [7] se describen y analizan diversos algoritmos que aproximan la solución en tiempo razonable.

§2.3 El Problema del Ancho de Banda es NP-Completo.

Para una matriz A de coeficientes reales y simétrica el problema del ancho de banda consiste en encontrar una permutación simétrica A' de A de tal forma que el valor máximo de

$$|i - j|$$

tomado sobre todas las entradas $a'_{i,j}$ distintas de cero sea mínimo.

Al igual que otros problemas de optimización en combinatoria tales como el problema del agente viajero y el problema del ciclo máximo, el problema de minimizar el ancho de banda de una matriz puede transformarse en un problema de decisión, esto es, en una pregunta que sólo puede ser contestada por un "sí" o por un "no"

Dada la relación que existe entre matrices y gráficas señalada anteriormente, podemos definir en términos formales el problema de minimizar el ancho de banda como sigue:

INSTANCIA: una gráfica $G = (V, E)$ y un entero positivo $k \leq |V|$.

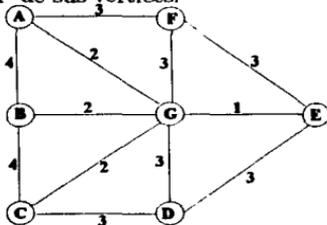
PREGUNTA: ¿Existe una numeración para los vértices de G con ancho de banda menor o igual a k ?

Entonces el *Problema de Minimizar el Ancho de Banda (PMAB)* consiste en descubrir, para cada gráfica G y cada entero positivo k , si existe una numeración f para G tal que el ancho de banda de G relativo a f no exceda a k ($B_f(G) \leq k$).

En 1975 Papadimitriou [13] demostró que el problema del ancho de banda para gráficas -y por lo tanto para matrices- es *NP-Completo*. En seguida daremos un esbozo de la demostración siguiendo el procedimiento descrito en el capítulo I para probar cuando un problema pertenece a dicha clase, antes introduciremos algunos conceptos:

El Problema del Arreglo Lineal (*PAL*) es el siguiente: Dada una gráfica G y una función $l : E(G) \rightarrow \mathbb{N}$ (que etiqueta las aristas de G), ¿es posible acomodar los vértices de G en un arreglo lineal con distancias unitarias entre vértices consecutivos, tal que ningún par de vértices adyacentes tenga distancia mayor que la etiqueta de la arista que los une?. De forma equivalente, ¿existe una numeración f para los vértices de G tal que $(i, j) \in E \Rightarrow |f(i) - f(j)| \leq l[(i, j)]$?

Como una ilustración del *PAL* la gráfica en la siguiente figura se presenta un ejemplo de una gráfica con siete vértices y cuya solución es el arreglo: *CBDGEAF* de sus vértices.



Ejemplo del problema del arreglo lineal.

Dado que la respuesta al *PAL* y al *PMAB* puede obtenerse examinando las componentes conexas de la gráfica por separado, podemos asumir de aquí en adelante todas las gráficas conexas.

El problema de factibilidad de expresiones booleanas con exactamente tres distintas literales por cláusula puede abreviarse como *3-SAT*.

Finalmente, el *PAL Restringido* es un *PAL* donde las únicas etiquetas permitidas para las aristas son b y $2b - 1$ para algún entero positivo b . Nótese que el *PMAB* es también un caso especial del *PAL*, con $l[e] = b$ para todo $e \in E$.

Lema 2.3.4 *Los siguientes problemas están en \mathcal{NP} :*

- a) *3 - SAT*
- b) *PAL*
- c) *PAL Restringido, PMAB.*

Demostración.

a) *3 - SAT* es un caso especial del *SAT* el cuál está en \mathcal{NP} (ver por ejemplo [6], teorema 3.1).

- b) Una máquina no determinística de Turing puede probar simultáneamente todas las posibles permutaciones de V en tiempo polinomial.
- c) Estos problemas son casos especiales del *LAP*. ■

Lema 2.3.5 3 – *SAT* es un problema *NP* – Completo. (Ver por ejemplo [6], teorema 3.1).

Teorema 2.3.5 *PAL* es *NP* – Completo.

Para su demostración se construye una gráfica G tal que cada instancia B del problema 3 – *SAT* es factible si y solo si *PAL* de G tiene solución y se prueba que tal construcción se realiza con tiempo polinomial. Papadimitriou [13] teorema 1.

Corolario 2.3.1 *PAL* Restringido es *NP* – Completo.

Lema 2.3.6 Cualquier *PAL* Restringido con $m > 0$ aristas de longitud

$2b - 1$ puede transformarse en tiempo polinomial en un *PAL* Restringido con $m - 1$ aristas de longitud $2b' - 1$, donde b' es el parámetro del nuevo *PAL* Restringido.

Teorema 2.3.6 El *PMAB* es *NP* – Completo.

Demostación. Reduciendo el *PAL* Restringido. Podemos aplicar el Lema 2.3.6, m veces y de esta forma al final obtener una gráfica sin aristas de longitud $2b - 1$. ■

En 1977 Garey, Graham, Johnson y Knuth [5] demostraron que si la gráfica G en la formulación del problema del ancho de banda es un árbol con vértices con grado menor o igual a 3, es también muy difícil de resolver, es decir se trata de un problema *NP* – Completo.

Teorema 2.3.7 El *PMAB* para árboles T con $\Delta(T) = 3$ es *NP* – Completo.

Además propusieron un $O(n)$ algoritmo para descubrir cuando una gráfica tiene ancho de banda igual a 2 en tiempo lineal.

Teorema 2.3.8 El *PMAB* con $k = 2$ tiene solución en tiempo lineal.

§2.4 El Ancho de Banda y otros Parámetros.

Definición 2.4.1 *Un invariante de una gráfica G es un número asociado a G el cual tiene el mismo valor para cualquier gráfica isomorfa a G .*

Así los números p y q son ciertamente invariantes. Un conjunto completo de invariantes determina una gráfica. Por ejemplo, los números p y q constituyen tal conjunto para todas las gráficas con menos de cuatro vértices.

Dado alguna invariante gráfico-teórica, digamos x , es frecuentemente más fácil obtener una cota inferior que una superior para $B(G)$ en términos de $x(G)$. Esto se debe al hecho que x frecuentemente restringe las posibilidades para una numeración óptima de G , de aquí que imponga valores mínimos sobre $B(G)$, sin embargo, x raramente proporciona información acerca de cómo construir una numeración f para la cual $B_f(G)$ sea muy cercano a $B(G)$. En esta sección examinaremos un número considerable de cotas inferiores para $B(G)$ y un número pequeño de cotas superiores, todas ellas envuelven simples y bien conocidas invariantes x .

El primer teorema relaciona el número p de vértices y el número q de aristas y es debido a Chvátal (1970).

Teorema 2.4.1 *Para cualquier gráfica G ,*

$$B(G) \geq p(G) - \frac{2 - [(2p(G) - 1)^2 - 8q(G)]^{1/2}}{2}.$$

Este resultado puede obtenerse analizando G como una subgráfica de P_p^h .

Los siguientes teoremas involucran el ancho de banda con los grados de los vértices de una gráfica.

Teorema 2.4.2 *Para cualquier gráfica G*

$$B(G) \geq \lceil \Delta(G) / 2 \rceil$$

Demostración:

Sea v un vértice de grado máximo de G , entonces el conjunto de las diferencias $|f(v) - f(u)|$ con u adyacente a v tiene al menos $\Delta(G) / 2$ elementos. ■

Teorema 2.4.3 Para cualquier gráfica G con secuencia de grados d_1, d_2, \dots, d_p que satisfaga $d_1 \leq d_2 \leq \dots \leq d_p$,

$$B(G) \geq \max \{d_j - \lfloor (j-1)/2 \rfloor, d_j/2\}$$

Teorema 2.4.4 Para cualquier gráfica G que no contenga triángulos

$$B(G) \geq \lfloor (3\delta(G) - 1)/2 \rfloor.$$

Definición 2.4.2 La conexidad $\kappa = \kappa(G)$ de una gráfica G es el mínimo número de vértices de cuya eliminación resulta una gráfica desconexa o trivial. Así la conexidad de una gráfica desconexa es 0, mientras que la conexidad de una gráfica conexa con un vértice de corte es 1. La gráfica completa K_p no se puede desconectar con cualquier número de vértices, pero la gráfica trivial resulta después de eliminar $p-1$ vértices; por lo tanto, $\kappa(K_p) = p-1$.

Definición 2.4.3 Una buena coloración de una gráfica es una asignación de colores para sus vértices de modo que ningún par de vértices adyacentes tengan el mismo color. El conjunto de todos los vértices con el mismo color es independiente, es decir ninguna pareja de vértices son adyacentes, y es llamado una clase de color. Una n -coloración de una gráfica G toma n colores; de tal modo que parte a V en n clases de color. El número cromático $\chi(G)$ se define como el mínimo n para el cual G tiene una n -coloración. Una gráfica G es n -colorable si $\chi(G) \leq n$ y es n -cromática si $\chi(G) = n$.

Los siguientes resultados ilustran la relación entre el ancho de banda y otros invariantes tales como la conexidad κ y el número cromático χ de una gráfica G .

Teorema 2.4.5 Para cualquier gráfica G , $B(G) \geq \chi(G) - 1$.

Sea f la numeración de G con la que se obtuvo el mínimo ancho de banda, es decir $B_f(G) = B(G)$. La siguiente es una coloración admisible para $V(G)$: Asigna al vértice j el $c[j] - \text{ésimo}$ donde $1 \leq c[j] \leq B(G) + 1$ y $c[j] \equiv f(j) \pmod{B(G) + 1}$ ■

Teorema 2.4.6 Para cualquier gráfica G , $B(G) \geq \kappa(G)$.

Observación 2.4.1 En el primero de estos teoremas podemos observar que $B(K_n) = \chi(G) - 1$ mientras que en el segundo, $B(P_n^m) = \kappa(P_n^m)$ para todo m y $n \geq m + 1$. De aquí que ninguna de las desigualdades pueda mejorarse con respecto a estos invariantes.

2.4.1 Cotas sobre el ancho de banda en gráficas conexas

Lema 2.4.1 Sea G una gráfica conexa y f una numeración óptima para G , entonces

$$|f(u) - f(v)| \leq B(G) d(u, v), \quad \forall u, v \in V(G) \quad (2.1)$$

Demostración:

Como G es una gráfica conexa $\forall u, v \in V(G)$ existe una trayectoria que los conecta. Sea $u, x_1, x_2, \dots, x_k, v$ los vértices de una trayectoria más corta de u a v entonces

$$\begin{aligned} |f(u) - f(v)| &= |f(u) + (f(x_1) - f(x_1)) + \dots + (f(x_k) - f(x_k)) - f(v)| \\ &\leq |f(u) - f(x_1)| + |f(x_1) - f(x_2)| + \dots + |f(x_k) - f(v)| \leq B(G) d(u, v) \end{aligned}$$

ya que tenemos $d(u, v)$ sumandos y cada uno de ellos es menor o igual a $B(G)$ por ser adyacentes. ■

Teorema 2.4.7 Sea G una gráfica conexa y sean $|V(G)| = p$ y $D(G)$ el diámetro de G entonces

$$\lceil (p-1)/D(G) \rceil \leq B(G)$$

Demostración:

De 2.1 podemos afirmar que en particular para los vértices x y y tal que $f(x) = p$ y $f(y) = 1$ se tiene que $|f(x) - f(y)| = |p - 1| \leq B(G) d(x, y)$ y como $d(x, y) \leq D(G) \forall u, v \in V(G)$ entonces

$$|p - 1| \leq B(G) d(x, y) \leq B(G) D(G) \quad \blacksquare$$

Teorema 2.4.8 Si $B(G) \geq (p+1)/2$, entonces, para cualquier vértice u , $B(G-u) \geq B(G) - 1$.

Demostración:

Sea f una numeración óptima de $G - u$. Definimos una numeración g para G como sigue:

$$g(v) = \begin{cases} f(v) + 1 & \text{si } f(v) \geq \lceil p/2 \rceil \\ \lceil p/2 \rceil & \text{si } v = u \\ f(v) & \text{si } f(v) < \lceil p/2 \rceil \end{cases}$$

Si e es una arista de G que no tiene a u como vértice extremo, el valor

asignado por g es a lo más una unidad más grande que el valor otorgado por f . Cualquier arista que tenga a u como extremo tiene un valor asignado por g el cual está acotado por el $\max\{p - \lceil p/2 \rceil, \lceil p/2 \rceil - 1\} \leq p/2$.

Entonces $(p + 1)/2 \leq B(G) \leq B_g(G) \leq \max\{p/2, B(G - u) + 1\}$. De aquí que el máximo es al menos $(p + 1)/2$ y por lo tanto se sigue que $B(G - u) \geq B(G) - 1$. ■

2.4.2 El Ancho de Banda y el Tamaño de una Gráfica.

Los párrafos siguientes muestran resultados sobre el ancho de banda y el número de aristas de una gráfica. Se establecen cotas inferiores para determinar el número de aristas. Dutton, 1989 [4].

Teorema 2.4.1 *Una gráfica G tiene el mínimo número de aristas sobre todas las gráficas con ancho de banda B si y solo si G es $K_{1,2B-1}$ o K_3 con cualquier número de vértices aislados.*

Demostración:

Sea $G = (V, E)$ una gráfica con el mínimo número de aristas sobre todas las gráficas con ancho de banda B . G debe tener solo una componente no trivial y desde ahora asumimos que G es conexa. De aquí $p \leq 2B$. A G puede añadirse cualquier número de vértices aislados sin alterar ni el número de aristas ni el ancho de banda.

Para concluir con la demostración haremos inducción sobre B .

La conclusión es inmediata para $B = 1$ y $B = 2$, las únicas posibles gráficas son $K_{1,1}$, $K_{1,3}$ y K_3 . Ahora sea G una gráfica conexa con ancho de banda $B \geq 3$, con el mínimo número de aristas, y suponemos que no es $K_{1,2B-1}$.

Por otra parte sabemos que $|E(G)| \leq 2B - 1$. Si $p = 2B$, G tiene que ser un árbol y entonces G es $K_{1,2B-1}$ por el teorema 2.5.3. Entonces asumimos que $p \leq 2B - 1$. La minimalidad de G implica que al quitar cualquier arista, y por lo tanto cualquier vértice, se reduce el ancho de banda. Con esto y por el teorema 2.4.8 implica que $B(G - u) = B - 1$ para cualquier vértice u , y entonces $|E(G - u)| \geq 2(B - 1) - 1$ por la hipótesis de inducción. Como $B \geq 3$, existe un vértice u de grado al menos 3 entonces $|E(G)| \geq |E(G - u)| + 3 = 2B$, contradicción.

El regreso es inmediato, ya que como hemos visto cualquier gráfica que no sea ni $K_{1,2B-1}$ ni K_3 tiene más de $2B - 1$ aristas. ■

El siguiente corolario es una consecuencia inmediata del teorema.

Corolario 2.4.1 Si una gráfica G tiene ancho de banda B , entonces

$$|E(G)| \geq 2B - 1.$$

Una segunda consecuencia del teorema, es el establecimiento, cuando $B \leq p/2$, de valores para $e_{p,B}$, el mínimo número de aristas en cualquier gráfica con p vértices y ancho de banda B , y $e_{p,B}^c$, la cantidad correspondiente para gráficas conexas.

Corolario 2.4.2 Si $B \leq p/2$, entonces $e_{p,B} = 2B - 1$ y $e_{p,B}^c = p - 1$

Demostración:

Del corolario 2.4.1 sabemos que cualquier gráfica con ancho de banda B debe tener al menos $2B - 1$ aristas. La gráfica $K_{1,2B-1}$ más $p - 2B$ vértices aislados tiene exactamente ese número. Claramente $e_{p,B}^c \geq p - 1$ y $K_{1,2B-1}$ con una trayectoria de longitud $p - 2B$ unida a algún vértice extremo muestra la igualdad. ■

§2.5 Ancho de banda de gráficas especiales

Examinaremos en esta sección el ancho de banda de ciertas gráficas como P_n , K_n , $K_{n,m}$, Q_n , árboles; productos de gráficas, tal como el producto cartesiano.

Teorema 2.5.1 $B(P_n) = 1$, $B(C_n) = 2$, $B(K_n) = n - 1$, $B(K_{m,n}) = \lfloor (m - 1) / 2 \rfloor + n$, donde $m \geq n$.

Un procedimiento para numerar los vértices de $K_{m,n}$ de tal forma que $B(K_{m,n}) = \lfloor (m - 1) / 2 \rfloor + n$ es el siguiente: numeramos el conjunto con menos vértices de $K_{m,n}$ con los números del $\lfloor m/2 \rfloor + 1$ a $\lfloor m/2 \rfloor + n$ y el otro conjunto con los números no utilizado de $\{1, 2, \dots, m + n\}$.

Teorema 2.5.2 Sea p el número de vértices de una gráfica k -partita completa y si $n_1 \geq n_2 \geq \dots \geq n_k$ entonces

$$B(K_{n_1, n_2, \dots, n_k}) = p - \left\lfloor \frac{n_1 + 1}{2} \right\rfloor.$$

Teorema 2.5.3 Si G es un árbol, entonces $B(G) \leq p(G) / 2$ y la igualdad se tiene si y solo si G es una estrella y $p(G)$ es par ($G \simeq K_{1, p(G)-1}$).

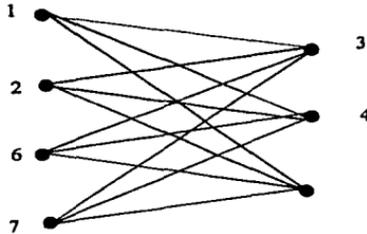


Figura 2.6: Numeración óptima de $K_{4,3}$ que proporciona un ancho de banda igual a 4.

2.5.1 Ancho de banda de producto de gráficas

Definición 2.5.1 Dadas dos gráficas G_1 y G_2 con $V(G_1) = \{u_1, v_1, w_1, \dots\}$ y $V(G_2) = \{u_2, v_2, w_2, \dots\}$ el producto cartesiano de G_1 y G_2 , denotado por $G_1 \times G_2$, es la gráfica cuyo conjunto de vértices es $V(G_1) \times V(G_2)$, dos puntos $u = (u_1, u_2)$ y $v = (v_1, v_2)$ son adyacentes en $G_1 \times G_2$ siempre que $[u_1 = v_1$ y u_2 sea adyacente a $v_2]$ o $[u_2 = v_2$ y u_1 sea adyacente a $v_1]$.

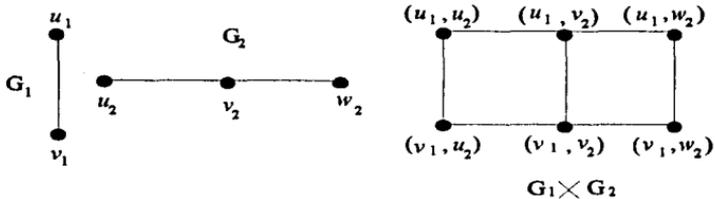


Figura 2.7: Ejemplo de un producto de trayectorias: $P_2 \times P_3$.

Teorema 2.5.4 Para cualesquiera dos gráficas G_1 y G_2 ,

$$B(G_1 \times G_2) \leq \min \{V(G_1) \cdot B(G_1), V(G_2) \cdot B(G_2)\}.$$

Corolario 2.5.1 Para cualesquiera n gráficas G_1, \dots, G_n ,

$$B(G_1 \times \dots \times G_n) \leq \min \left\{ B(G_i) \cdot \prod_{j \neq i} V(G_j) : i = 1, \dots, n \right\}.$$

Teorema 2.5.5 Chvátalová, 1986 [3]. Si $\max \{m, n\} \geq 2$, entonces $B(P_m \times P_n) = \min \{m, n\}$.

Teorema 2.5.6 Chvátalová, 1986 [3]. Si $m \geq 2$ y $k \geq 3$, entonces $B(P_m \times C_k) = \min \{2m, k\}$.

Definición 2.5.2 Dadas dos gráficas G y H con $V(G) = \{u_1, v_1, w_1, \dots\}$ y $V(H) = \{u_2, v_2, w_2, \dots\}$ la composición de G y H , denotado como $G(H)$, es la gráfica con el conjunto de vértices $V(G) \times V(H)$ en donde (u_1, v_1) es adyacente a (u_2, v_2) si u_1 es adyacente a u_2 en G o $u_1 = u_2$ y v_1 es adyacente a v_2 en H .

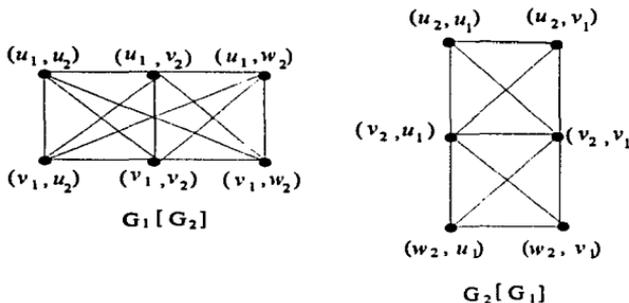


Figura 2.8: Composición de las gráficas G_1 y G_2 de la figura 2.7.

Teorema 2.5.7 Para cualesquiera dos gráficas G y H ,

$$B(G(H)) \leq (B(G) + 1)V(H) - 1$$

La igualdad se alcanza cuando $G = K_n$ y $H = K_m$.

Teorema 2.5.8 Liu 1995 [11]. Sea G una gráfica con m vértices y $n \geq 2$. Si $B(G) \leq \lceil m/2 \rceil - 1$, entonces $B(K_n[G]) = mn - \lceil (m+1)/2 \rceil$.

Demostración:

Sea $V(G) = \{v_1, v_2, \dots, v_m\}$ y $V(K_n) = \{u_1, u_2, \dots, u_n\}$.

Sea $g : V(G) \rightarrow \{1, 2, \dots, m\}$ una numeración propia de G tal que $B_g(G) = B(G)$. Para el conjunto de enteros $\{y_1, y_2, \dots, y_m\}$ suponemos que

$$y_i = \begin{cases} i & \text{si } 1 \leq i \leq \lceil m/2 \rceil, \\ mn - m + i & \text{para } \lceil m/2 \rceil + 1 \leq i \leq m. \end{cases}$$

Entonces $B_f(K_n[G]) \leq mn - \lceil (m+1)/2 \rceil$ para cualquier numeración propia f de $K_n[G]$ tal que $f(u_i, v_j) = y_i$ si y solo si $g(v_j) = i$, puesto que (u_i, v_j) no es adyacente a (u_i, v_k) si $|g(v_j) - g(v_k)| \geq \lceil m/2 \rceil$. Por lo tanto $B(K_n[G]) \leq mn - \lceil (m+1)/2 \rceil$. Por otro lado, dado que $K_n[G] \supseteq K_{n(m)}$, la gráfica n -partita con m vértices en cada conjunto de la partición, se sigue del teorema 2.5.2 que $B(K_n[G]) \geq B(K_{n(m)}) = mn - \lceil (m+1)/2 \rceil$. Así $B(K_n[G]) = mn - \lceil (m+1)/2 \rceil$. ■

Capítulo 3

Ancho de Banda y Operaciones Locales

§3.1 Introducción y Notación.

Las operaciones locales a las que nos referiremos en este capítulo son: refinamiento elemental (subdivisión), contracción y adición de una arista y fusión de dos vértices. En general se mostrará que bajo una operación local, el ancho de banda de una gráfica puede incrementar o decrementar y la razón de cambio no es necesariamente pequeña. Investigaremos la relación entre el ancho de banda de la gráfica original y las gráficas generadas por dichas operaciones.

Definición 3.1.3 Sea G una gráfica y u, v vértices de G . La gráfica $G_a^{(u,v)}$ denota la gráfica obtenida de G al añadir la arista (u, v) , es decir,

$$V(G_a^{(u,v)}) = V(G), \quad E(G_a^{(u,v)}) = E(G) \cup \{(u, v)\}.$$

La gráfica $G_m^{u,v}$ denota la gráfica obtenida de G al fusionar los vértices u y v , es decir,

$$V(G_m^{u,v}) = (V(G) - \{u, v\}) \cup \{z\}, \text{ donde } z \notin V(G),$$

$E(G_m^{u,v}) = (E(G) - \{(x, y) \mid x \in V(G), y \in \{u, v\}\}) \cup \{(x, z) \mid x \notin u, v \text{ y tampoco } (x, u) \in E(G) \text{ o } (x, v) \in E(G)\}.$

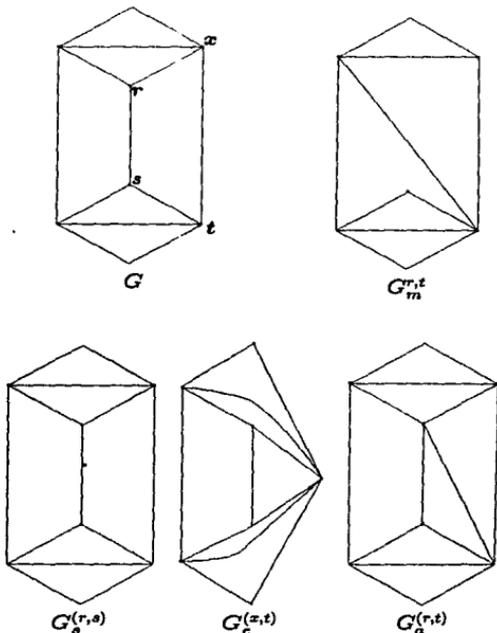
Sea (u, v) una arista de G . La gráfica $G_s^{(u,v)}$ denota la gráfica obtenida de G al subdividir (hacer un refinamiento elemental) la arista (u, v) , es decir,

$$V(G_s^{(u,v)}) = V(G) \cup \{z\}, \text{ donde } z \notin V(G),$$

$$E(G_s^{(u,v)}) = (E(G) \cup \{(u, z), (z, v)\}) - \{(u, v)\}.$$

La gráfica $G_c^{(u,v)}$ denota la gráfica obtenida de G al contraer la arista (u, v) , es decir, es isomorfa a la gráfica $G_m^{u,v}$.

Las operaciones de la Definición 1 están ilustradas en la siguiente figura.



Siempre que nos refiramos al ancho de banda de una gráfica o al valor B_f para alguna numeración f de una gráfica, supondremos que la gráfica en cuestión tiene al menos una arista, siempre que no se especifique otra cosa.

Mientras que el problema de encontrar el ancho de banda de una gráfica es *NP-Completo* (capítulo 2, teorema 2.3.6) existen ciertas clases de gráficas para las cuales el ancho de banda puede determinarse por una fórmula explícita o por un algoritmo de tiempo polinomial. Usando las operaciones anteriores sobre gráficas de las cuáles el ancho de banda ha sido determinado, podemos obtener cotas superiores e inferiores para una gran clase de gráficas.

Usamos $[n]$ para denotar el mínimo entero mayor o igual a n .

En diferentes demostraciones la siguiente cota inferior sobre el ancho de banda de una gráfica será utilizada (Capítulo 2, teorema 2.4.7) :

$$B(G) \geq \lceil (|V(G)| - 1) / D(G) \rceil. \quad (3.1)$$

§3.2 Cotas superiores e inferiores.

La primer operación que vamos a analizar es la adición de aristas, los siguientes lemas (Veáse Wang, 1995 [14]) nos dan información acerca de cuánto es lo máximo que puede incrementarse el ancho de banda de una gráfica al añadir una arista tomando como parámetros el ancho de banda original y el número de vértices de la gráfica.

Lema 3.2.1 Sea $B(G) = b$ y sea $e = (v_l, v_m) \in \bar{E}(G)$ entonces $B(G_a^{(v_l, v_m)}) \leq 2b$.

Demostración:

Sea f una numeración óptima para G , y sea $V(G) = \{v_1, \dots, v_n\}$ numerados de tal forma que $f(v_i) = i$ y sea $e = (v_l, v_m)$ la arista añadida.

La demostración consiste en dar una numeración f' tal que $|f'(x) - f'(y)| \leq 2 |f(x) - f(y)|$ y $|f'(v_l) - f'(v_m)| = 1$.

Sea $r = \lfloor (l + m) / 2 \rfloor$ y definimos $f'(v_r) = 1$ y $f'(v_{r+1}) = 2$. Ahora para cada v_i tal que $|i - r| \leq \min \{r - 1, n - r\}$ definimos

$$f'(v_i) = f'(v_{i+1}) + 2 \quad \text{si} \quad i < r.$$

$$f'(v_i) = f'(v_{i-1}) + 2 \quad \text{si} \quad i > r.$$

Tomando en cuenta que $f'(v_r) = 1$ y $f'(v_{r+1}) = 2$ podemos definir los vértices como

$$f'(v_i) = 2(r - i) + 1 \quad \text{si} \quad i < r.$$

$$f'(v_i) = 2(i - r) \quad \text{si} \quad i > r.$$

Con esto, hemos definido f' para todos los vértices de G excepto para un conjunto de la forma v_1, \dots, v_k o v_{n+1-k}, \dots, v_n dependiendo del signo de $r - \lfloor (n+1)/2 \rfloor$. (Esto depende de quién sea el mínimo entre $r-1$ y $n-r$)

En el primer caso, asignamos

$$f'(v_i) = n + 1 - i \quad \text{para} \quad i \leq k.$$

En el segundo

$$f'(v_i) = i \quad \text{para} \quad i > n - k.$$

La reenumeración de los vértices hacia afuera de v_r logra que

$$|f'(x) - f'(y)| \leq 2 |f(x) - f(y)|.$$

Falta probar que si tomamos cualesquiera dos vértices de $G + e$ no importa el valor que se les asigne con f' siempre $|f'(x) - f'(y)| \leq 2 |f(x) - f(y)|$.

Por ejemplo, si tomamos dos vértices v_i y v_j pueden considerarse los siguientes casos:

1) v_i es tal que $|i - r| \leq \min\{r - 1, n - r\}$ y v_j también cumple que $|j - r| \leq \min\{r - 1, n - r\}$

$$\begin{aligned} \text{a) } i, j < r \text{ entonces } f'(v_i) &= 2(r - i) + 1 \quad \text{y} \quad f'(v_j) = 2(r - j) + 1 \\ |f'(v_i) - f'(v_j)| &= |(2(r - i) + 1) - (2(r - j) + 1)| = |2(j - i)| = 2 |f(v_i) - f(v_j)|. \end{aligned}$$

b) $i < r$ y $j > r$ (o viceversa) entonces $f'(v_i) = 2(r-i) + 2$ y

$$f'(v_j) = 2(j-r)$$

$$|f'(v_i) - f'(v_j)| = |(2(r-i) + 2) - (2(j-r))| = |2(2r - (i+j)) + 2| \leq |2(2j - (i+j)) + 2| = |2(j-i) + 2| \leq 2|i-j| = 2|f(v_i) - f(v_j)|.$$

c) $i, j > r$ entonces $f'(v_i) = 2(i-r)$ y $f'(v_j) = 2(j-r)$

$$|f'(v_i) - f'(v_j)| = |2(i-r) - 2(j-r)| = |2(i-j)| = 2|f(v_i) - f(v_j)|.$$

d) $v_i = v_r$ y $j < r$ entonces $f'(v_r) = 1$ y $f'(v_j) = 2(r-j) + 2$

$$|f'(v_i) - f'(v_j)| = |f'(v_r) - f'(v_j)| = |1 - ((j-r) + 2)| = |r-j-1| \leq |r-j| + 1 \leq 2|r-j| = 2|f(v_i) - f(v_j)|.$$

2) Si $(r - \lfloor (n+1)/2 \rfloor) \geq 0$ significa que el conjunto de vértices que quedó sin reenumerar es de la forma v_1, \dots, v_k y entonces

a) si v_i pertenece a este conjunto y v_j también es obvio que

$$|f'(v_i) - f'(v_j)| \leq 2|f(v_i) - f(v_j)|. \text{ ya que } f'(v_i) = n+1-i \text{ y } f'(v_j) = n+1-j,$$

$$|f'(v_i) - f'(v_j)| = |(n+1-i) - (n+1-j)| = |j-i| \leq 2|i-j| = 2|f(v_i) - f(v_j)|$$

b) si v_i pertenece a este conjunto y v_j no

1) $j < r$, entonces $f'(v_i) = n+1-i$ y $f'(v_j) = 2(r-j) + 2$

$$|f'(v_i) - f'(v_j)| = |(n+1-i) - (2(r-j) + 2)| = |n-r-(i-j)-1| \leq |j-i| + |n-r-1|. \blacksquare$$

La siguiente construcción provee una reenumeración óptima para todos los casos donde el número de vértices de G , $n(G) < 6B(G)$.

Lema 3.2.2 Si $B(G) = b$ y $n(G) \leq 3(b+a) + 1$, donde $a \geq 1$, entonces

$$B(G+e) \leq b+a \text{ para cualquier } e \in E(\bar{G}).$$

Demostación:

Una vez más sea f una numeración óptima para G , y sea $V(G) = \{v_1, \dots, v_n\}$ numerados de tal forma que $f(v_i) = i$, y sea $e = (v_l, v_m)$ con $l < m$ la arista añadida. Si $m - l \leq b + a$ no hay nada que probar, entonces podemos suponer que $m - l > b + a$. Lo que haremos será cambiar la numeración para hacer que los nuevos valores de v_l y v_m difieran por $b + a$. Si escogemos valores de r y s con $l \leq r < s \leq m$ y definimos f' como

$$\begin{aligned} f'(v_l) &= r, & f'(v_m) &= s, \\ f'(v_i) &= i - 1 \text{ para } l < i \leq r, \\ f'(v_i) &= i + 1 \text{ para } s \leq i < m, \\ f'(v_i) &= i = f(v_i) \text{ en otro caso.} \end{aligned}$$

En la siguiente figura puede observarse como se realiza la renumeración

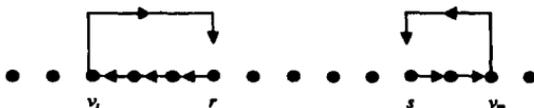


Figura 3.1: Una cota sobre el aumento del ancho de banda bajo adición de aristas.

Queremos elegir r y s de tal forma que $B_{f'}(G + e) \leq b + a$.

Si $r - s = b + a$, entonces cualesquiera dos vértices v_i, v_j diferentes a v_l, v_m (con $l < i \leq r$ y $s \leq j < m$) cuya numeración cambie bajo f' no son adyacentes en G , ya que como $i \leq r$ y $s \leq j$, entonces $s + i \leq r + j$ y por lo tanto $|f(v_j) - f(v_i)| = |j - i| \geq s - r = b + a$.

Ahora, como $a \geq 1$ y los demás vértices aparte de v_l, v_m bajo f' cambian a lo más en 1, necesitamos sólo verificar que la aristas incidentes a v_l y v_m tienen diferencias satisfactorias bajo f' . Para garantizar esto, es suficiente tener

$$r \leq \max\{l + a, 1 + b + a\} \quad \text{y} \quad s \geq \min\{m - a, n - b - a\}.$$

Si $l \geq n - 2(b + a)$, ponemos $r = l$ y $s = l + b + a$, y tenemos $s = l + b + a \geq n - (b + a)$, lo cual es suficiente.

Si $m \leq 1 + 2(b + a)$, entonces simétricamente ponemos $s = m$ y $r = m - b - a$, produciendo $r \leq 1 + b + a$. En los casos restantes, con $l \leq n - 1 - 2(b + a)$ y $m \geq 2 + 2(a + b)$, fijamos $r = 1 + b + a$ y $s = 1 + 2(b + a)$. Entonces la condición $n \leq 1 + 3(b + a)$ produce que $l \leq b + a < r$ y $s \geq n - b - a$, y que todas las condiciones juntas sean suficientes y se satisfagan. ■

Lema 3.2.3 Si $n \geq 6b - 1 - 3k$ con $0 \leq k \leq b - 1$, entonces existe una gráfica G con n vértices y con ancho de banda b y una arista $e \in E(\overline{G})$ tal que $B(G + e) \geq 2b - k$.

Demostración:

Como el ancho de banda de una gráfica es tan grande como el de cualquier subgráfica, podemos suponer que G es la b -ésima potencia de P_n (donde P_n es una trayectoria con n vértices). En otras palabras, dada una numeración óptima f de G tal que $f(v_i) = i \forall i$, entonces $(v_i v_j) \in E(G)$ si y solo si $|i - j| \leq b$.

Sea $e = (v_{1+b}, v_{n-b})$; es suficiente mostrar que $B(G + e) \geq 2b - k$.

Los vértices del conjunto $S = \{v_i : i \leq b + 1 \text{ ó } i \geq n - b - 1\}$ inducen una subgráfica de $G + e$ con diámetro 3. Sea f' una numeración óptima de $G + e$. Si los vértices asignados con los números 1 y n están ambos en S , entonces la subgráfica de $G + e$ inducida por S contiene una arista (u, v) con $|f'(u) - f'(v)| \geq [(n - 1)/3] \geq 2b - k$. Si el vértice x asignado al número 1 o n por f' pertenecen a $G \setminus S$, entonces todos los vértices adyacentes a x tienen números con orientación hacia un lado de $f'(x)$; como cada vértice de $G \setminus S$ tiene $2b$ vértices adyacentes en G , esto implica $B_{f'}(G + e) \geq 2b$. En cualquiera de los dos casos, se tiene que $B(G + e) \geq 2b - k$. ■

Juntos, estos tres lemas producen el siguiente resultado principal:

Teorema 3.2.1 Si $g(b, n)$ denota el valor máximo de $B(G + e)$ cuando G tiene n vértices y ancho de banda b , entonces

$$g(b, n) = \begin{cases} b + 1 & \text{si } n \leq 3b + 4 \\ \lceil (n - 1)/3 \rceil & \text{si } 3b + 5 \leq n \leq 6b - 2 \\ 2b & \text{si } n \geq 6b - 1. \end{cases}$$

Demostración:

Los lemas 3.2.1 y 3.2.3 proveen cotas inferiores y superiores cuando $n \geq 6b - 1$ ($k = 0$). El lema 3.2.2 provee la cota superior cuando $n \leq 3b + 4$ ($a = 1$). Si $3b + 5 \leq n \leq 6b - 1$, existe un único valor entero positivo de a tal que $3(b + a) + -1 \leq n \leq 3(b + a) + 1$. Entonces el lema 3.2.2 establece $g(b, n) \leq b + a = \lceil (n - 1) / 3 \rceil$, y el lema 3.2.3 establece $g(b, n) \geq 2b - (b - a) = b + a$, puesto que $3(b + a) + 1 = 6b - 1 - 3(b - a)$. ■

Ahora investigaremos una cota superior para el ancho de banda de $G_a^{(u,v)}$ en términos de $B(G)$ y la distancia de u y v en G .

Lema 3.2.4 Sea G una gráfica, g una numeración de G , u y v vértices de G , $(u, v) \notin E(G)$. Sea $c = |g(v) - g(u)|$. Existe una numeración f de $G_a^{(u,v)}$ tal que $B_f \leq \lceil \frac{1}{3} \max\{0, c - B_g\} \rceil + B_g$.

Demostración:

Tenemos que probar el lema sólo en el caso en que $c > B_g$. Podemos suponer sin pérdida de generalidad que $g(u) > g(v)$. Consideremos una numeración f de $G_a^{(u,v)}$ definida por:

1. $f(u) = g(u) + \lceil \frac{1}{3}(c - B_g) \rceil$,
2. $f(v) = f(u) + \lceil \frac{1}{3}(c + 2B_g) \rceil$,
3. $f(x) = a_{g(x)}$ para $x \in V(G) - \{u, v\}$, donde a_j es el j -ésimo entero positivo distinto de $f(u)$, $f(v)$.

Como $f(v) - f(u) = \lceil \frac{1}{3}(c + 2B_g) \rceil > B_g$, se sigue que para cualquier arista (x, y) de $G_a^{(u,v)}$ tal que $x, y \notin \{u, v\}$, no podemos tener $g(x) < f(u)$ y $g(y) \geq f(v)$. Por lo tanto, $|f(x) - f(y)| \leq |g(x) - g(y)| + 1 \leq B_g + 1$ para dichas aristas (x, y) . Ya que si por ejemplo $g(x) \geq f(u)$, entonces $f(x) \leq g(x) + 1$.

Sea (u, x) una arista de $G_a^{(u,v)}$, $x \neq v$. Si $g(u) > g(x)$, entonces

$$f(u) - f(x) = g(u) + \lceil \frac{1}{3}(c - B_g) \rceil - a_{g(x)} = g(u) - g(x) + \lceil \frac{1}{3}(c - B_g) \rceil \leq B_g + \lceil \frac{1}{3}(c - B_g) \rceil, \text{ puesto que } (u, x) \text{ es una arista de } G.$$

Si $g(u) < g(x)$, entonces $f(x) - f(u) \leq |g(x) - g(u)| \leq B_g$.

De manera similar podemos probar que $f(v) - f(x) \leq B_g + \lceil \frac{1}{3}(c - B_g) \rceil$ para cualquier arista (v, x) de $G_a^{(u,v)}$, $x \neq u$. Como

$$f(v) - f(u) = \left\lceil \frac{1}{3} (c + 2B_g) \right\rceil = f(v) - f(u) = \left\lceil \frac{1}{3} (c + 3B_g - B_g) \right\rceil = B_g + \left\lceil \frac{1}{3} (c - B_g) \right\rceil \text{ tenemos que } B_f \leq B_g + \left\lceil \frac{1}{3} (c - B_g) \right\rceil. \blacksquare$$

Teorema 3.2.2 Si G es una gráfica conexa y u, v son vértices de G , entonces

$$B(G_a^{(u,v)}) \leq \min \left\{ \left\lceil \frac{1}{3} (d(u, v) + 2) B(G) \right\rceil, 2B(G) \right\}$$

Demostración:

Sea g una numeración óptima para G . Sin pérdida de generalidad podemos suponer que $|g(u) - g(v)| > B(G)$. Claramente se cumple que $|g(u) - g(v)| \leq B(G) D(u, v)$. Por el lema 3.2.4, existe una numeración f de G tal que

$$\begin{aligned} B_f &\leq \left\lceil \frac{1}{3} (|g(u) - g(v)| - B_g) \right\rceil + B_g \\ &\leq \left\lceil \frac{1}{3} (B(G) d(u, v) + 2B(G)) \right\rceil \\ &= \left\lceil \frac{1}{3} (d(u, v) + 2) B(G) \right\rceil. \end{aligned}$$

La desigualdad $B(G_a^{(u,v)}) \leq 2B(G)$ corresponde al lema 3.2.1 y ya ha sido demostrada. \blacksquare

En el siguiente teorema mostraremos que la cota superior sobre el ancho de banda de $G_a^{(u,v)}$ del teorema 3.2.2 es la mejor posible en términos de $B(G)$ y $d(u, v)$.

Teorema 3.2.3 Para cada entero $i = 2, 3, 4$ y para cada entero k existe una gráfica G_k , y vértices u, v en G_k , tal que

- $B(G_k) \geq k$,
- $d(u, v) = i$,
- $B((G_k)_a^{(u,v)}) = \left\lceil \frac{1}{3} (d(u, v) + 2) B(G_k) \right\rceil$

Demostración:

Sea i uno de los enteros 2, 3 o 4. Sea $j = (i + 1)k + i$.

Consideremos las gráficas $G_{k_i} = P_{j(i+2)+1}^j$ y $(G_{k_i})_a^{(u,v)}$, cuando $u = x_{j+1}$, y $v = x_{j(i+1)+1}$. Es claro que $B(G_{k_i}) = j$, $d(u, v) = i$ en G_{k_i} , y $D((G_{k_i})_a^{(u,v)}) = 3$. Por el teorema 3.2.2, $B((G_{k_i})_a^{(u,v)}) \leq \left\lceil \frac{1}{3} (i + 2) j \right\rceil$. Sin embargo, por 3.1,

$$B((G_{k_i})_a^{(u,v)}) \geq [(|V((G_{k_i})_a^{(u,v)})| - 1) / D((G_{k_i})_a^{(u,v)})] = [\frac{1}{3}(i+2)j].$$

Por lo tanto

$$B((G_{k_i})_a^{(u,v)}) = [\frac{1}{3}(i+2)j] = [\frac{1}{3}(d(u,v)+2)B(G_{k_i})]. \blacksquare$$

Ahora investigaremos la relación entre el ancho de banda de $G_m^{u,v}$ y el ancho de banda de G .

Teorema 3.2.4 Si G es una gráfica y u, v son vértices de G , entonces

$$B(G) - 1 \leq B(G_m^{u,v}) \leq 2B(G).$$

Demostración:

Sea z el vértice de $G_m^{u,v}$ el cual no está en G .

(a) Primero probaremos $B(G_m^{u,v}) \leq 2B(G)$.

Sea f una numeración óptima de G . Sin pérdida de generalidad podemos asumir que $f(u) > f(v)$. Sea g_1 un mapeo de $V(G)$ definido de la siguiente manera:

$$g_1(x) = \frac{1}{2} + 2 \left| f(x) - j - \frac{1}{4} \right|, \quad \text{donde } j = \frac{1}{2}(f(u) + f(v)).$$

Podemos verificar que g_1 es un mapeo inyectivo, es decir que $g_1(x) \neq g_1(y)$ para $x \neq y$.

Lo que significa que g_1 es una numeración para G . Además g_1 cumple,

$$g_1(v) - g_1(u) = \frac{1}{2} + 2f(v) + 2j + \frac{1}{2} - \frac{1}{2} - 2f(u) + 2j + \frac{1}{2} = 1 \quad (3.2)$$

Sea g_2 un mapeo de $V(G_m^{u,v})$ definido como sigue:

$$g_2(x) = g_1(x) \quad \text{si } g_1(x) < g_1(u),$$

$$g_2(z) = g_1(u),$$

$$g_2(x) = g_1(x) - 1 \quad \text{si } g_1(x) > g_1(v).$$

Dado que $G_m^{u,v}$ no contiene los vértices u y v y g_1 es una numeración de G que satisface 3.2, g_2 es una numeración de $G_m^{u,v}$.

Sea (x, y) una arista en c la cual también es una arista de G . Entonces

$$\begin{aligned} |g_2(x) - g_2(y)| &\leq |g_1(x) - g_1(y)| = 2 \left| \left| f(x) - j - \frac{1}{4} \right| - \left| f(y) - j - \frac{1}{4} \right| \right| \\ &\leq 2|f(x) - f(y)| \leq 2B(G). \end{aligned}$$

Si (x, z) es una arista en $G_m^{u,v}$, entonces por la definición de g_2

$$\begin{aligned} |g_2(x) - g_2(z)| &= |g_2(x) - g_1(u)| \\ &\leq \min\{|g_1(x) - g_1(u)|, |g_1(x) - g_1(v)|\} \leq 2B(G). \end{aligned}$$

(b) Ahora probaremos la primera desigualdad, $B(G) - 1 \leq B(G_m^{u,v})$.
Sea h una numeración óptima de $G_m^{u,v}$.

Definamos en G una numeración h' de la siguiente forma:

$$h'(x) = h(x) \quad \text{si } h(x) < h(z),$$

$$h'(u) = h(z),$$

$$h'(x) = h(x) + 1 \quad \text{si } h(x) > h(z),$$

$$h'(v) = h(u) + 1.$$

Claramente $B(G) \leq B(G_m^{u,v}) + 1$, y el teorema se cumple. ■

Teorema 3.2.5 Para todo n existe una gráfica G_n , y vértices u, v en G_n tales que $B(G_n) > n$ y $B((G_n)_m^{u,v}) = 2B(G)$.

Demostración:

Sea $j = \lceil \frac{1}{3}n \rceil$. Sea G_n una gráfica tal que $V(G_n) = \{x_1, x_2, \dots, x_{12j+9}\}$, y $E(G_n) = \{(x_i, x_{3j+3}) \mid i \leq 6j+5, i \neq 3j+3\} \cup \{(x_{9j+7}, x_i) \mid i \geq 6j+5, i \neq 9j+7\}$.

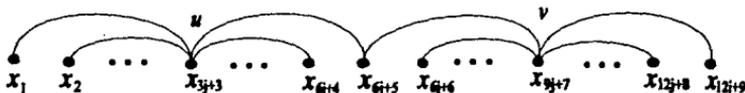


Figura 3.2: G_n

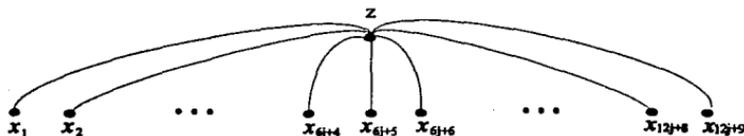
Podemos verificar que $D(G_n) = 4$.

Consideremos una numeración f de G_n definida por $f(x_i) = i$.

Claramente $B_f(G_n) = 3j + 2$.

Usando 3.1, $B(G_n) \geq \lceil (V(G_n) - 1) / D(G_n) \rceil = 3j + 2 > n$.

Sea $u = x_{3j+3}$, $v = x_{9j+7}$.

Figura 3.3: $(G_n)_{m}^{u,v}$

Nuevamente, usando 3.1,

$$\begin{aligned} B((G_n)_{m}^{u,v}) &\geq [(V((G_n)_{m}^{u,v}) - 1) / D((G_n)_{m}^{u,v})] \\ &= [(12j + 8) / 2] = 2B(G_n) \end{aligned}$$

Esto, combinado con el teorema 3.2.4, produce $B((G_n)_{m}^{u,v}) = 2B(G_n)$. ■

Teorema 3.2.6 Para todo n existe una gráfica H_n y vértices u, v en H_n tales que $B(H_n) > n$ y $B((H_n)_{m}^{u,v}) = B(H_n) - 1$.

Demostración:

Sea H_n la gráfica bipartita $K_{2n+1,2}$.

Por 3.2.7 sabemos que $B(K_{2n+1,2}) = n + 2$. Sean u y v los dos vértices de $K_{2n+1,2}$ de grado $2n + 1$.

Observemos que $(H_n)_{m}^{u,v} = K_{2n+1,1}$ y por lo tanto $B((H_n)_{m}^{u,v}) = n + 1 = B(H_n) - 1$. ■

Por lo tanto, las cotas del teorema 3.2.4 no pueden mejorarse.

Definición 3.2.1 Sea G una gráfica que contiene como subgráfica una trayectoria $P_n = (v_0, v_1, \dots, v_n)$, $n \geq 1$, tal que en G , $d(v_1) = d(v_2) = \dots = d(v_{n-1}) = 2$. Sea k un entero, $n > k \geq 0$. Definimos $G_{P_n}^k$ la gráfica que se obtiene de G al eliminar los vértices $v_{k+1}, v_{k+2}, \dots, v_{n-1}$ y todas las aristas incidentes con ellos, y fusionando los vértices v_k y v_n .

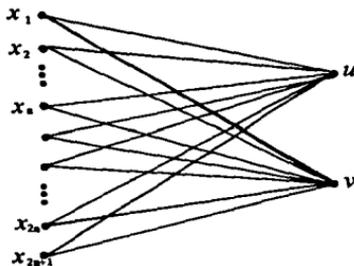


Figura 3.4: $H_n = K_{2n+1,2}$

Ahora investigaremos la relación entre $B(G_{P_n}^k)$ y $B(G)$. De esta relación podremos derivar directamente una cota superior para $B(G_c^{(u,v)})$ y una cota inferior para $B(G_s^{(u,v)})$ en términos de $B(G)$.

Lema 3.2.5 Sea G una gráfica que contiene como subgráfica una trayectoria $P_n = (v_0, v_1, \dots, v_n)$, $n \geq 1$, tal que en G , $d(v_1) = d(v_2) = \dots = d(v_{n-1}) = 2$. Entonces

$$B(G_{P_n}^k) \leq \min \left\{ 2B(G), \left\lceil \frac{(n+2)B(G) - n + k}{k+2} \right\rceil \right\}.$$

Demostración:

Sea f una numeración óptima para G . Consideremos los siguientes dos casos:

(a) $d(v_0) = 1$, $d(v_n) = 1$ respectivamente. En este caso $G_{P_n}^k$ es isomorfa a la subgráfica de G inducida por los vértices $V(G) - \{v_0, v_1, \dots, v_{n-k-1}\}$, $V(G) - \{v_{k+1}, v_{k+2}, \dots, v_n\}$, el lema claramente se satisface.

(b) $d(v_0) > 1$ y $d(v_n) > 1$. Sin pérdida de generalidad podemos suponer que $f(v_0) < f(v_n)$. Sean s, l vértices de G tales que

$$\begin{aligned} f(l) &= \max \{f(u) \mid (v_n, u) \in E(G), u \neq v_{n-1}\}, \\ f(s) &= \min \{f(u) \mid (v_0, u) \in E(G), u \neq v_1\}. \end{aligned}$$

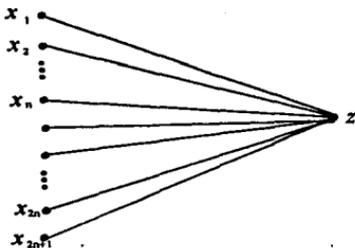


Figura 3.5: $(H_n)_m^{u,v} = K_{2n+1,1}$

Sea z el vértice de $G_{P_n}^k$, el cual no está en G .

(b1) Suponemos que $f(s) < f(v_0) < \dots < f(v_n) < f(l)$. Sea

$$h = \lceil [(n+2)B(G) - n + k] / (k+2) \rceil.$$

Sea i el entero positivo menor tal que $(f(l) - n + k) - hi \leq f(v_{k-i+1})$.

Sea a_r

el r -ésimo entero positivo para el cual $a_r \notin \{f(l) - n + k - jh \mid 1 \leq j \leq i-1\}$

Definamos la numeración g en $G_{P_n}^k$ de la siguiente manera:

$$g(v) = a_{\text{rango}(f(v))}$$

para $v \in V(G_{P_n}^k) - \{v_{k-i+2}, \dots, v_{k-1}, z\}$,
 donde $\text{rango}(f(v))$ es el rango entre los
 $\{f(x) \mid x \in V(G) - \{v_{k-i+2}, \dots, v_k\}\}$,

$$g(z) = f(l) - n + k + h,$$

$$g(v_{k-i+1}) = f(l) - n + k - hj, \quad 2 \leq j \leq i-1.$$

Intuitivamente g está construida como sigue. Puesto que

$f(l) - f(s) \leq (n+2)B(G)$, y $G_{P_n}^k$ tiene $n-k$ vértices menos que G , la numeración g intenta tender un puente sobre un espacio abierto de tamaño $(n+2)B(G) - n + k$ en una trayectoria de longitud $k+2$. Esto conduce a la definición de h y a la numeración g .

Puesto que $h > B(G)$ tenemos que $|g(u) - g(v)| \leq B(G) + 1$ para cada arista (u, v) en $G_{P_n}^k$, la cual no es incidente con $\{v_{k-i+2}, \dots, v_{k-1}, z\}$. Claramente, $|g(v_{k-1}) - g(z)| = h$ y $|g(v_{k-j+1}) - g(v_{k-j})| = h$ para

$2 \leq j \leq i - 1$. Si (u, z) es una arista de $G_{P_n}^k$, $u \neq v_{k-1}$, entonces

$$\begin{aligned} |g(z) - g(u)| &= |f(l) - n + k + h - a_f(u)| \\ &\leq |f(l) - n + k + h - f(l) + n - k| = h. \end{aligned}$$

Por lo tanto $B(G_{P_n}^k) \leq h$. Ya que $G_{P_n}^k$ es una subgráfica de $G_m^{v_k, v_n}$, $B(G_{P_n}^k) \leq B(G_m^{v_k, v_n}) \leq 2B(G)$ por el teorema 3.2.4.

(b2) Suponemos que la secuencia de enteros $f(s), f(v_0), \dots, f(v_n), f(l)$ no es creciente. Dado que $f(v_0) < f(v_n)$ existe un entero i tal que $f(v_{i-1}) > f(v_i) < f(v_{i+1})$, $0 \leq i \leq n$. Consideremos la gráfica G_1 y la numeración f_1 de G_1 como sigue:

$$G_1 = \begin{cases} G^{(v_i, v_{i-1})} & \text{si } 1 \leq i \leq n, \\ G^{(v_0, v_i)} & \text{si } i = 0. \end{cases}$$

$$f_1(x) = f(x) \quad \text{para } x \in V(G) \cap V(G_1)$$

$$f_1(v'_i) = \begin{cases} f(v_{i-1}) & \text{si } 1 \leq i \leq n; \\ f(v_i) & \text{si } i = 0, \end{cases}$$

donde v'_i es el nuevo vértice obtenido en la operación de contracción de arista. G_1 contiene como su subgráfica la trayectoria $P_{n-1} = (v_0, \dots, v_{i-1}, v'_i, v_{i+1}, \dots, v_n)$, $B_{f_1}(G) \leq B_f(G)$ y $(G_1)_{P_{n-1}}^k$ es isomorfa a $G_{P_n}^k$.

Si $k = n - 1$, entonces G_1 satisface el lema. Si la secuencia de enteros $f_1(s), f_1(v_0), \dots, f_1(v'_i), \dots, f_1(v_n), f_1(l)$ no es creciente y $k < n - 1$, entonces podemos repetir la construcción previa para obtener la gráfica G_2, \dots , etc. De esta forma podemos construir una secuencia de gráficas

$G = G_0, G_1, G_2, \dots, G_j$, $j \leq n - k$ tal que:

- G_r contiene como subgráfica una trayectoria P_{n-r} , $0 \leq r \leq j$ de longitud $n - r$;
- $(G_r)_{P_{n-r}}^k$ es isomorfa a $G_{P_n}^k$, $0 \leq r \leq j$;
- Se cumple que $j = n - k$ o que $f_j(s), f(v_0^j), \dots, f_j(v_{n-j}^j), f_j(l)$ sea una secuencia creciente, donde $P_{n-j} = (v_0^j, \dots, v_{n-j}^j)$.

Por lo tanto, si $j = n - k$, entonces claramente $B(G_{P_n}^k) \leq B(G)$; si no; podemos aplicar la parte (b1) de esta prueba a G_j para verificar el lema. ■

Teorema 3.2.7 Para toda gráfica G y toda arista (u, v) en G ,

$$B(G) + 1 \geq B(G_s^{(u,v)}) \geq \left\lceil \frac{1}{4} (3B(G) - 1) \right\rceil.$$

Demostración:

Sea P_3 una trayectoria en $G_s^{(u,v)}$ consistente de los vértices u, r, v , donde r es el vértice obtenido de la subdivisión de la arista (u, v) de G . Puesto que G puede ser obtenida de $G_s^{(u,v)}$ mediante la contracción de la arista (r, v) de la trayectoria P_3 ,

$$B(G) \leq \min \left\{ 2B(G_s^{(u,v)}), \left\lceil \frac{1}{3} (4B(G_s^{(u,v)}) - 1) \right\rceil \right\}$$

por el lema 3.2.5. Por lo tanto

$$B(G) - \frac{2}{3} \leq \frac{1}{3} (4B(G_s^{(u,v)}) - 1)$$

y

$$\frac{1}{4} (3B(G) - 1) \leq B(G_s^{(u,v)}).$$

Dado que G es isomorfa a la gráfica obtenida de $G_s^{(u,v)}$ mediante la fusión de los vértices u y r , $B(G) \geq B(G_s^{(u,v)}) - 1$ por el teorema 3.2.4. ■

Teorema 3.2.8 Para todo entero $n > 0$ existe una gráfica G_n y una arista (u, v) en G_n tal que $B(G_n) > n$ y $B((G_n)_s^{(u,v)}) = \left\lceil \frac{1}{4} (3B(G) - 1) \right\rceil$.

Demostración:

Sea $j = \left\lceil \frac{1}{4}n \right\rceil$. Sea G_n una gráfica tal que $V(G_n) = \{x_1, x_2, \dots, x_{12j+8}\}$,
y $E(G_n) = \{(x_i, x_{4j+4}) \mid i \leq 6j+4, i \neq 4j+4\} \cup \{(x_{4j+4}, x_{8j+7})\}$

$\cup \{(x_i, x_{8j+7}) \mid i \geq 6j+5, i \neq 8j+7\}$.

Puede verificarse fácilmente que $D(G_n) = 3$.

Consideremos una numeración f de G_n definida por $f(x_i) = i$.

Claramente, $B_f(G_n) = 4j + 3$. Usando 3.1,

$$\begin{aligned} B(G_n) &\geq \left\lceil \frac{V(G_n) - 1}{D(G_n)} \right\rceil \\ &= \left\lceil \frac{12j + 7}{3} \right\rceil = 4j + 3 \end{aligned}$$

Por lo tanto, $B(G_n) = 4j + 3 > n$.

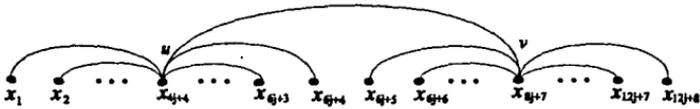


Figura 3.6: G_n , definida para probar cotas en subdivisión de aristas.

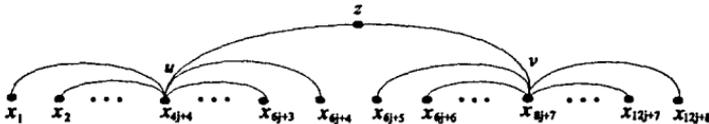


Figura 3.7: $B((G_n)_s^{(u,v)}) = \left\lceil \frac{1}{4} (3B(G) - 1) \right\rceil$

Sea $u = x_{4j+4}$, $v = x_{8j+7}$.

Usando 3.1,

$$\begin{aligned} B((G_n)_s^{(u,v)}) &\geq \left\lceil \frac{V((G_n)_s^{(u,v)}) - 1}{D((G_n)_s^{(u,v)})} \right\rceil \\ &= \left\lceil \frac{12j+6}{4} \right\rceil = 3j+2 \end{aligned}$$

y por el teorema 3.2.7

$$\begin{aligned} B((G_n)_s^{(u,v)}) &\leq \left\lfloor \frac{1}{4} (3B(G) - 1) \right\rfloor \\ &= \left\lfloor \frac{1}{4} (3(4j+3) - 1) \right\rfloor = 3j+2 \end{aligned}$$

$$\therefore B((G_n)_s^{(u,v)}) = \left\lfloor \frac{1}{4} (3B(G) - 1) \right\rfloor. \blacksquare$$

Teorema 3.2.9 Para toda gráfica G y toda arista (u, v) en G ,

$$B(G) - 1 \leq B(G_c^{(u,v)}) \leq \left\lceil \frac{1}{2} (3B(G) - 1) \right\rceil$$

Demostración:

Puesto que $G_c^{(u,v)}$ es obtenida de G por la contracción de la arista (u, v) de la trayectoria $P_1 = (u, v)$, $B(G_c^{(u,v)}) \leq \left\lceil \frac{1}{2} (3B(G) - 1) \right\rceil$ por el lema 3.2.5. Luego $G_c^{(u,v)}$ es isomorfa a la gráfica $G_m^{u,v}$ y por lo tanto $B(G_c^{(u,v)}) \geq B(G) - 1$ por el teorema 3.2.4. ■

Teorema 3.2.10 Para todo entero $n > 0$ existe una gráfica G_n y una arista (u, v) en G_n tal que $B(G_n) > n$ y $B((G_n)_c^{(u,v)}) = \left\lceil \frac{1}{2} (3B(G) - 1) \right\rceil$

Demostración:

Sea $j = \left\lceil \frac{1}{4}n \right\rceil$. Sea G_n una gráfica tal que $V(G_n) = \{x_1, x_2, \dots, x_{6j+4}\}$, y $E(G_n) = \{(x_i, x_{2j+2}) \mid i \leq 3j+1, i \neq 2j+2\} \cup \{(x_{2j+2}, x_{4j+3})\} \cup \{(x_{4j+3}, x_i) \mid i \geq 3j+2, i \neq 4j+3\}$.

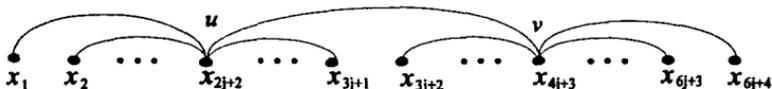


Figura 3.8: G_n , definida para probar cotas en contracción de aristas.

Puede verificarse fácilmente que $D(G_n) = 3$.

Consideremos una numeración f de G_n definida por $f(x_i) = i$. Claramente, $B_f(G_n) = 2j + 3$. Usando 3.1,

$$\begin{aligned} B(G_n) &\geq \left\lceil \frac{V(G_n) - 1}{D(G_n)} \right\rceil \\ &= \left\lceil \frac{6j + 3}{3} \right\rceil = 2j + 1 \end{aligned}$$

Por lo tanto, $B(G_n) = 2j + 1 > n$. Sea $u = x_{2j+2}$, $v = x_{4j+3}$.

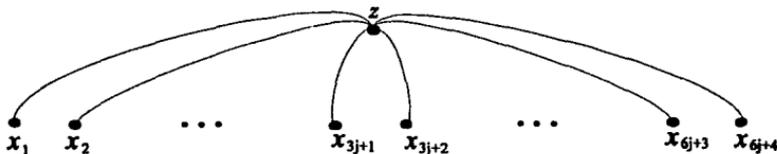


Figura 3.9: $B((G_n)_c^{(u,v)}) = \lceil \frac{1}{2} (3B(G_n) - 1) \rceil$

Por 3.1,

$$\begin{aligned} B((G_n)_c^{(u,v)}) &\geq \left\lceil \frac{V((G_n)_c^{(u,v)}) - 1}{D((G_n)_c^{(u,v)})} \right\rceil \\ &= \left\lceil \frac{6j+2}{2} \right\rceil = 3j+1 \end{aligned}$$

y por el teorema 3.2.9

$$\begin{aligned} B((G_n)_c^{(u,v)}) &\leq \left\lceil \frac{1}{4} (3B(G_n) - 1) \right\rceil \\ &= \left\lceil \frac{1}{2} (3(2j+1) - 1) \right\rceil = 3j+1 \end{aligned}$$

$$\therefore B((G_n)_c^{(u,v)}) = \left\lceil \frac{1}{2} (3B(G_n) - 1) \right\rceil. \blacksquare$$

Observación 3.2.1 Si definimos H'_n como $(H_n)_\alpha^{(u,v)}$, podemos ver que el ancho $B(H'_n) = B(H_n)$; por el teorema 3.2.7 la numeración óptima para H_n asigna los siguientes números a los vértices u y v : $f(u) = \lceil \frac{2n+1}{2} \rceil + 1$ y $f(v) = \lceil \frac{2n+1}{2} \rceil + 2$ entonces la adición de la arista (u, v) no afecta en el ancho de banda; y puesto que $(H_n)_m^{u,v}$ en la prueba del teorema 3.2.6 es isomorfa a $(H'_n)_c^{(u,v)}$, implica que la cota inferior sobre el ancho de banda de $G_c^{(u,v)}$ en el teorema 3.2.9 no puede mejorarse.

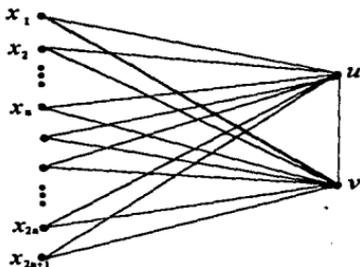


Figura 3.10: Gráfica tal que bajo contracción de una arista el ancho de banda de la gráfica resultante disminuye en una unidad

Observación 3.2.2 Construimos la gráfica C'_{2n+1} de la siguiente manera $C'_{2n} = (C_{2n+1})_a^{(v_{n+1}, v_{n+2})}$ para $n > 2$, entonces por el teorema 3.2.7 el ancho de banda de C'_{2n+1} es 2. Ahora vamos a ver qué ocurre con el ancho de banda de

$$\left(C'_{2n+1} \right)_s^{(v_{n+1}, v_{n+2})}. \text{ Por 3.1, } B \left(\left(C'_{2n+1} \right)_s^{(v_{n+1}, v_{n+2})} \right) \geq \left[\frac{v \left(\left(C'_{2n+1} \right)_s^{(v_{n+1}, v_{n+2})} \right) - 1}{D \left(\left(C'_{2n+1} \right)_s^{(v_{n+1}, v_{n+2})} \right)} \right] =$$

$$\left[\frac{2n+1}{n} \right] = 3 = B(C'_{2n+1}) + 1, \text{ y ahora por el teorema 3.2.7,}$$

$$B \left(\left(C'_{2n+1} \right)_s^{(v_{n+1}, v_{n+2})} \right) \leq B(C'_{2n}) + 1. \text{ Por lo tanto } B \left(\left(C'_{2n+1} \right)_s^{(v_{n+1}, v_{n+2})} \right) =$$

$$B(C'_{2n+1}) + 1$$

Así, los teoremas 3.2.7 y 3.2.9 proporcionan las mejores cotas para el ancho de banda de $G_s^{(u,v)}$ y $G_c^{(u,v)}$ en términos del ancho de banda de G .

ESTA TERCERA NO DEBE
SALIR LA BIBLIOTECA

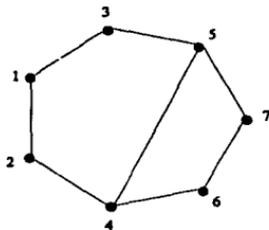


Figura 3.11: Ejemplo de C'_{2n+1} con $n = 3$

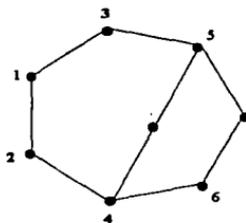


Figura 3.12: Ejemplo de una gráfica tal que bajo subdivisión de aristas el ancho de banda se incrementa en 1.

Bibliografía

1. CHINN, P. Z., CHVÁTALOVÁ J., DEWDNEY A.K., GIBBS N. E. The Bandwidth Problem for Graphs and Matrices —A Survey, *Journal of Graph Theory* Vol. 6 No. 3 (1982), 223-254.
2. CHVÁTALOVÁ J., OPATRY J. The Bandwidth Problem and Operations on Graphs, *Discrete Mathematics*. Vol. 61 (1986), 141-150.
3. CHVÁTALOVÁ J. Optimal Labelling of a Product of Two Paths, *Discrete Mathematics*. Vol. 11 (1975), 249-253.
4. DUTTON R. D., BRIGHAM R. C., On the Size of Graphs of a Given Bandwidth. *Discrete Mathematics*. Vol 76 (1989), 191-195.
5. GAREY M. R., GRAHAM R. L., JOHNSON D. S., KNUTH D. E. Complexity Results for Bandwidth Minimization. *SIAM J. Applied Mathematics*. Vol. 34, No. 3, (1978). 477-495.
6. GAREY M. R., JOHNSON D. S. **Computers and Intractability: A Guide to the Theory of NP-Completeness**. W. H. Freeman and Company ,1979.
7. GEORGE A., LIU J. W. **Computer Solution of Large Sparse Positive Definite System**. Prentice-Hall, Englewood Cliffs, New Jersey, 1981.
8. GIBBONS A. **Algorithmic Graph Theory**, Cambridge University Press 1985.
9. HARARY, F., **Graph Theory**. Addison Wesley, Reading M. A. Segunda Edición 1971.

10. KURT M. **Data Structures and Algorithms 2: Graph Algorithms and NP-Completeness.** Springer Verlag. 1984.
11. LIU J., WILLIAMS K., On Bandwidth and Edgesum for the Composition of Two Graphs. *Discrete Mathematics.* Vol. 143 (1995), 159-166.
12. NEMHAUSER G. L., WOLSEY L.A. **Integer and Combinatorial Optimization.** John Wiley & Sons. 1988.
13. PAPADIMITRIOU Ch. H. The NP-Completeness of the Bandwidth Minimization Problem. *Computing* Vol. 16, (1976) 263-270.
14. WANG J., WEST D. B., YAO B. Maximum Bandwidth Under Edge Addition. *Journal of Graph Theory* Vol. 20 No. 1 (1995), 87-90.