UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

FACULTAD DE CIENCIAS QUÍMICAS, FÍSICAS Y MATEMÁTICAS

ESCUELA PROFESIONAL DE MATEMÁTICA

TESIS

ARBOLES DE EXPANSIÓN EN OPTIMIZACION DE CAMINOS

PRESENTADO POR:

Br. PILAR RIVERO CURI

PARA OPTAR AL TÍTULO PROFESIONAL DE:

LICENCIADA EN MATEMÁTICA

ASESORA:

Dra. ANA MARÍA MARLITT COLQUEHUANCA ARIAS

CUSCO – PERÚ

2025

INFORME DE ORIGINALIDAD

(Aprobado por Resolución Nro.CU-303-2020-UNSAAC)

El que suscribe, Asesor del trabajo de investigación/tesis titulada: **"Arboles De Expansión en Optimización de Caminos"** presentado por: **Br. Pilar Rivero Curi** con DNI: **72625639**, para optar al título profesional de **Licenciada en Matemática**. Informo que el trabajo de investigación ha sido sometido a revisión por **2** veces, mediante el Software Antiplagio, conforme al Art. 6° del *Reglamento para Uso de Sistema Antiplagio de la UNSAAC* y de la evaluación de originalidad se tiene un porcentaje de 7%.

Evaluación y acciones del reporte de coincidencia para trabajos de investigación conducentes a grado académico o título profesional, tesis

Porcentaje	**Evaluación y Acciones**	**Marque con una (X)**
Del 1 al 10%	No se considera plagio.	X
Del 11 al 30 %	Devolver al usuario para las correcciones.	
Mayor a 31%	El responsable de la revisión del documento emite un informe al inmediato jerárquico, quien a su vez eleva el informe a la autoridad académica para que tome las acciones correspondientes. Sin perjuicio de las sanciones administrativas que correspondan de acuerdo a Ley.	

Por tanto, en mi condición de asesor, firmo el presente informe en señal de conformidad y **adjunto** la primera hoja del reporte del Sistema Antiplagio.

Cusco, 19 de agosto de 2025

....
Firma

ANA MARIA MARLITT COLQUEHUANCA ARIAS

Nro. de DNI 25001081

ORCID 0000-0003-0939-7799

Se adjunta:

1. Reporte generado por el Sistema Antiplagio.
2. Enlace del Reporte Generado por el Sistema Antiplagio: **oid:** 27259:484759415

PILAR RIVERO CURI

ARBOLES DE EXPANSION EN OPTIMIZACION DE CAMINOS.pdf

Universidad Nacional San Antonio Abad del Cusco

Detalles del documento

Identificador de la entrega

trn:oid:::27259:484759415

Fecha de entrega

19 ago 2025, 9:54 p.m. GMT-5

Fecha de descarga

19 ago 2025, 9:57 p.m. GMT-5

Nombre de archivo

ARBOLES DE EXPANSION EN OPTIMIZACION DE CAMINOS.pdf

Tamaño de archivo

1.4 MB

74 Páginas

15.319 Palabras

75.615 Caracteres

7% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Bibliografía
- Texto citado
- Texto mencionado
- Coincidencias menores (menos de 10 palabras)
- Base de datos de Crossref
- Base de datos de contenido publicado de Crossref

Fuentes principales

5% Fuentes de Internet
0% Publicaciones
5% Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alertas de integridad para revisión

No se han detectado manipulaciones de texto sospechosas.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

DEDICATORIA

A mi mamá, Agustina Curi Montañez, por ser siempre mi constante fuente de inspiración y dedicación.

A mi papá, Hernan Rivero Gonzales, por su confianza en mí. Y a mi hermano, Dario Ciro Rivero Curi, por su amabilidad y apoyo.

Este logro es fruto de su paciencia, comprensión y apoyo en cada etapa de este proceso, les dedico este trabajo con todo mi cariño.

AGRADECIMIENTOS

Expreso mi profundo agradecimiento a la Universidad Nacional de San Antonio Abad del Cusco, por formarme profesionalmente en un excelso ambiente académico.

Manifiesto mi especial gratitud a mi asesora la Dra. Ana María Marlitt Colquehuanca Arias y al Dr. Marco Antonio Herrera Vargas por sus valiosos aportes y recomendaciones, los cuales fueron fundamentales para la culminación de este trabajo.

Agradezco también a los profesores del Departamento Académico de Matemática y Estadifica por sus enseñanzas fundamentales en mi formación académica.

Extiendo mi agradecimiento a mis amigos Salvador de la Cruz, Sra. Gloria Nina, Teresa Aparicio e Hilda Flores, por su constante aliento y motivación.

Finalmente, quiero expresar mi reconocimiento a Brayan Huaman, Edgar Huarca, Jakeline Pauccar y a mis compañeros del código 2016-II, con quienes compartí experiencias y momentos memorables.

ÍNDICE

INDICE DE FIGURAS

RESUMEN

El presente trabajo de investigación se enmarca en el área de la matemática discreta y tiene como propósito profundizar en el estudio teórico de las propiedades matemáticas de caminos en grafos, con énfasis en los árboles de expansión para optimizar caminos. Mediante un enfoque axiomático y rigurosamente formal, se analiza la validez de los algoritmos de Kruskal y Prim, fundamentales en la construcción de árboles de expansión mínimos o máximos, según lo requiera el problema.

La investigación realizada es de tipo básica y de nivel exploratorio-descriptivo, busca demostrar la solidez matemática de estos algoritmos, destacando su relevancia y utilidad en procesos de optimización de rutas y estructuras de conectividad. Con ello, se valida su correcto funcionamiento desde una perspectiva teórica y se amplía la base formal de la teoría de grafos, caminos y árboles de expansión. Los resultados obtenidos aportan una base sólida para futuras investigaciones y aplicaciones, así como un recurso útil para estudiantes e investigadores de matemática y disciplinas afines.

Palabras Clave: grafos, caminos, árboles de expansión, algoritmo de Kruskal, algoritmo de Prim

ABSTRAC

This research project falls within the field of discrete mathematics and aims to deepen the theoretical study of the mathematical properties of paths in graphs, with a particular focus on spanning trees. Through an axiomatic and rigorously formal approach, the validity of Kruskal's and Prim's algorithms is analyzed, both of which are fundamental in the construction of minimum or maximum spanning trees, depending on the requirements of the problem.

The study is basic in nature and follows an exploratory-descriptive level of analysis. It seeks to demonstrate the mathematical soundness of these algorithms, highlighting their relevance and usefulness in the optimization of routes and connectivity structures. In doing so, it validates their functioning from a theoretical perspective and enriches the conceptual framework of graph theory, including paths and spanning trees. The findings provide a solid foundation for future research and applications, as well as a valuable resource for students and researchers in mathematics and related fields.

Keywords: *graph theory, paths, spanning trees, Kruskal algorithm, Prim algorithm*

INTRODUCCION

La presente investigación aborda el estudio de los árboles de expansión en grafos, enfocados en la optimización de caminos, un problema recurrente en áreas como redes de comunicación, diseño de circuitos y transporte. La investigación se centra exclusivamente en grafos ponderados no dirigidos, los cuales permiten modelar conexiones entre nodos con pesos asociados sin una orientación específica. Este tipo de grafos resulta esencial para la construcción de árboles de expansión mínimos que conecten todos los vértices sin formar ciclos con el menor costo posible.

Se excluyen de este estudio los grafos dirigidos y no ponderados, ya que no cumplen con las condiciones necesarias para aplicar los algoritmos seleccionados. Los grafos dirigidos introducen relaciones asimétricas entre los nodos, mientras que los no ponderados impiden evaluar comparativamente los pesos de las conexiones.

La estructura del presente trabajo se organiza de la siguiente manera: el Capítulo 1 presenta el planteamiento del problema, la formulación de objetivos y justificación de la investigación. En el Capítulo II se desarrolla el marco teórico, incluyendo los antecedentes y las bases teóricas de grafos. El Capítulo III expone la metodología utilizada en el estudio. En el Capítulo IV se presentan y analizan los resultados, abordando la teoría de caminos, árboless de expansión y los algoritmos de Kruskal y Prim, junto con sus respectivas pruebas de validez. Al término del estudio se presentan la discusión de resultados, las conclusiones y las recomendaciones de la investigación.

CAPITULO I: PLANTEAMIENTO DEL PROBLEMA

1.1 SITUACIÓN PROBLEMÁTICA

Actualmente, se opera en un entorno empresarial altamente competitivo esto obliga a organizaciones y empresas a implementar estrategias orientadas a mejorar su eficiencia y capacidad de adaptación. Debido a la complejidad y naturaleza dinámica de algunas de sus redes se requieren soluciones eficaces para mejorar su planificación y toma de decisiones.

Este panorama impulsa la constante búsqueda de soluciones innovadoras que permitan la minimización de recorridos y al mismo tiempo mejoren en la eficiencia operativa.

En este contexto, el uso de grafos resalta como una herramienta eficaz y practica para abordar estos problemas reales, ya que un grafo puede generar una considerable variedad de árboles que conectan todos los vértices de una red.

Al optimizar este proceso, es posible determinar el árbol de expansión optimo del grafo, esto significa encontrar en función del problema su árbol de expansión mínimo o máximo.

1.2. FORMULACIÓN DEL PROBLEMA

1.2.1. PROBLEMA GENERAL

¿Sera posible optimizar caminos utilizando árboles de expansión?

1.2.2. PROBLEMAS ESPECÍFICOS

a) ¿De qué manera se pueden establecer los conceptos fundamentales de la teoría de caminos, conexidad y árboles de expansión?
b) ¿Es posible demostrar la validez del algoritmo de Kruskal para obtener árboles de expansión óptimos?
c) ¿Se puede verificar la validez del algoritmo de Prim para obtener árboles de expansión óptimos?

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Optimizar caminos utilizando árboles de expansión

1.3.2. OBJETIVOS ESPECÍFICOS

a) Establecer los conceptos fundamentales de la teoría de caminos, conexidad y árboles de expansión.
b) Demostrar la validez del algoritmo de Kruskal para obtener árboles de expansión óptimos.
c) Verificar la validez del algoritmo de Prim para obtener árboles de expansión óptimos.

1.4. JUSTIFICACIÓN E IMPORTANCIA

La aplicación de la teoría de grafos, en particular de los árboles de expansión, es una herramienta muy eficaz para la modelación y la optimización de caminos.

Se busca profundizar en una comprensión axiomática y rigurosa de los fundamentos matemáticos que sustentan la teoría de grafos, caminos y árboles de expansión ya que, aunque existe bibliografía y trabajos estos se encuentran mayormente enfocados en investigaciones de carácter aplicado o computacional, lo que evidencia una escasa atención al desarrollo de sus propiedades matemáticas subyacentes.

En este contexto, se aspira ofrecer una perspectiva clara, precisa y sólidamente fundamentada sobre el estudio de caminos y árboles de expansión óptimos, sustentada en demostraciones formales y generalizaciones que contribuyan a enriquecer el marco conceptual y analítico de la teoría de grafos.

Lo que permita una mejor comprensión del tema lo que resultará de utilidad para estudiantes de la carrera de Matemática y otras disciplinas afines interesadas en la optimización de rutas y caminos. En conclusión, los resultados obtenidos en este estudio construirán una base valiosa para futuras investigaciones y aplicaciones en proyectos reales de optimización.

CAPITULO II: MARCO TEÓRICO

2.1. ANTECEDENTES

2.1.1. ANTECEDENTES INTERNACIONALES.

Onofre (2024) en su tesis de licenciatura, *Modelo de redireccionamiento vial mediante algoritmo de colonia de hormigas y teoría de grafos en la transitabilidad vial de extranca Rio Seco 2022 - Bolivia*, tuvo como propósito realizar un análisis de datos y proponer un modelo de redireccionamiento vial basado en el análisis de datos, con el fin de optimizar la toma de decisiones respecto a la redistribución del flujo vehicular en el sector de Río Seco – Ex Tranca. A partir de simulaciones realizadas con dicho modelo, se identificaron beneficios proponiéndose su aplicación en otros sectores urbanos con problemáticas similares.

Este estudio aporta un enfoque aplicado de la teoría de grafos para mejorar la transitabilidad vial, se relaciona con mi trabajo en la medida en que ambos utilizan estructuras de grafos para optimizar rutas. Aunque el enfoque de Onofre se basa en algoritmos de colonia de hormigas, el objetivo común es la optimización de caminos, lo que fortalece la relevancia de emplear grafos en problemas de transporte.

Velasquez & Reyes (2023) en su tesis de licenciatura, *Teoría de grafos y sus aplicaciones en modelos de rutas óptimas* de la *Universidad Nacional Autónoma De Nicaragua*, estudio algoritmos de la teoría de grafos para diseñar un árbol de expansión de peso mínimo entre puestos de salud de León y el Hospital principal del municipio de León de Nicaragua y formulo una propuesta de rutas más cortas entre estas.

Este trabajo es directamente relevante para mi investigación, ya que emplea árboles de expansión de peso mínimo, uno de los ejes centrales de mi estudio. Su aplicación en el ámbito de la salud muestra la versatilidad del modelo, respaldando la utilidad de los árboles de expansión en la optimización de rutas reales.

Nogueira (2017) en su tesis de maestría, *Grafos e problemas de caminos* de la *Universidad Federal de Vicosa de Brasil*, implemento clases didácticas a estudiantes de una escuela de Belo Horizonte mediante los algoritmos Dijkstra, Prim, Kruskal y Floyd con el lenguaje de programación C, concluyendo con la posibilidad de inclusión de la teoría de grafos en la educación secundaria.

Este estudio contribuye desde el punto de vista educativo al demostrar cómo los algoritmos clásicos de grafos pueden ser comprendidos y utilizados por estudiantes. En relación con mi trabajo, desarrolla algoritmos como Prim y Kruskal, los cuales también son empleados en árboles de expansión, lo que valida su aplicabilidad tanto académica como práctica.

Patiño & Guillermo (2013) en su tesis de maestría, *La enseñanza de la teoría de grafos como estrategia para desarrollar procesos de matematización* de la *Universidad Sergio Arboleda de Colombia*, analizó el impacto de implementar la teoría de grafos para la mejora de habilidades matemáticas. Para ello, aplicó una prueba de resolución de problemas a dos grupos: uno conformado por estudiantes del programa de talentos de la Universidad Sergio Arboleda que recibieron formación en teoría de grafos, y otro compuesto por alumnos de un colegio reconocido que no contaban con dicho conocimiento. El estudio permitió evaluar cómo el aprendizaje de esta teoría influye en los procesos de matematización, evidenciando diferencias significativas entre ambos grupos.

Aunque este estudio se enfoca en el aspecto pedagógico, su aporte radica en mostrar cómo la teoría de grafos puede fortalecer el pensamiento lógico y matemático. Esto guarda relación indirecta con mi trabajo, al evidenciar que el dominio de grafos (incluidos los árboles de expansión) fomenta la capacidad de resolver problemas complejos, como los relacionados con rutas óptimas.

Piedra & Paternostro (2009) en su tesis de licenciatura, *Aplicaciones de la teoría de grafos en la informática de Bogotá Colombia*, tuvo como objetivo evidenciar el uso de la teoría de grafos en áreas como redes de comunicación, la gestión de información y la secuenciación de programas. Para ello abordó la representación y diseño de estructuras de redes mediante grafos, resaltando la relevancia de los grafos dirigidos en la ejecución de programas informáticos y su utilidad en la estructuración de bases de datos, a través de ejemplos ilustrativos con el propósito de introducir conceptos fundamentales y aplicaciones de la teoría de grafos en el ámbito de la computación.

Este trabajo aporta una visión informática del uso de la teoría de grafos, especialmente en la representación de estructuras de red. Aunque no se centra en árboles de expansión, su relación con mi trabajo radica en el uso compartido de los grafos como herramienta estructural clave para la organización y optimización de sistemas, lo cual puede extenderse a caminos y rutas.

2.1.2. ANTECEDENTES NACIONALES.

Páez (2023) en su tesis de licenciatura *Representación de estructuras algebraicas basadas en la Teoría de Grafos*, de la *Universidad Nacional José Faustino Sánchez Carrión* de Huacho, tuvo como objetivo desarrollar una representación de estructuras algebraicas mediante el uso de la teoría de grafos. Como resultado, formuló diversas proposiciones, propiedades y ejemplos que evidencian una relación estrecha entre los grafos y dichas estructuras algebraicas.

Este trabajo aporta una visión teórica de la aplicación de los grafos en el campo del álgebra, demostrando su versatilidad en distintas áreas matemáticas. Aunque no se enfoca directamente en caminos o rutas, su relación con mi tesis radica en el uso conceptual de los grafos lo cual también es base en la construcción de árboles de expansión.

Arias (2017) en su tesis de licenciatura *Aplicación de la teoría de grafos en el diseño de rutas de transporte desde las zonas de producción agrícola hasta la planta de procesamiento*, de la *Pontificia Universidad Católica Del Perú* de Lima, aplicó la teoría de grafos para optimizar el sistema de transporte y reducir los costos logísticos en la recolección y concentración de jalapeños de una empresa agroindustrial que trabaja en la sierra y selva central del país. Dicho estudio identificó un total de 144 rutas que cubrían 5,838 Km y validó la efectividad del método de dos fases utilizando programación lineal entera y el lenguaje AMPL (A Mathematical Programming Language), evidenciando áreas con potencial de mejora.

Este estudio es altamente relevante para mi investigación, ya que aborda la optimización de rutas de transporte mediante grafos. Aunque utiliza programación lineal, la lógica de minimizar distancias o costos se alinea con el objetivo de los árboles de expansión en encontrar caminos óptimos, fortaleciendo la aplicabilidad práctica de los modelos teóricos.

Matias (2018) en su tesis de licenciatura, *Evaluación de dos metodologías de solución al problema de redes de transporte* de la *Universidad Nacional Santiago Antúnez de Mayolo* de Ancash, realizo un diagnóstico del transporte urbano de Huaraz y el estudio de una simulación del transporte urbano concluyendo con la diferenciación a través ventajas y desventajas de los algoritmos de red de flujo de transporte máximo y flujo de coste mínimo.

Este estudio contribuye al análisis comparativo de los algoritmos en redes de transporte, aportando criterios para elegir métodos adecuados según las condiciones del sistema. Aunque se enfoca en redes de flujo y no directamente en árboles de expansión, comparte el interés en la optimización de caminos, complementando el enfoque de mi tesis desde otra perspectiva.

Condori (2016) en su tesis de licenciatura, *Organización espacial de la red de carreteras de la Región Puno mediante la teoría de grafos*, de la *Universidad Nacional Del Altiplano* de Puno, desarrollo fundamentos de la teoría de grafos para estructurar la red vial nacional y regional de la región Puno, haciendo uso de herramientas como la matriz de adyacencia y la accesibilidad topológica.

Esta investigación es un referente para mi tesis, ya que aplica directamente la teoría de grafos en la organización de redes viales. Aunque no se centra en árboles de expansión, su análisis estructural de la red de caminos permite entender el potencial de aplicar algoritmos de optimización como los que utilizo en mi trabajo.

Tocto Inga (2012) en su tesis de licenciatura, *Comparación de los algoritmos Prim y Kruskal* de la *Universidad Nacional De Ingeniería* de Lima, describió que son los algoritmos, características de un buen algoritmo, funciones de complejidad y algunos conceptos base de la teoría de grafos para el desarrollo de los algoritmos Prim y Kruskal, concluyendo con la comparación de complejidad de dichos algoritmos.

Este estudio es fundamental para mi trabajo, ya que Prim y Kruskal son precisamente los algoritmos que permiten construir árboles de expansión de peso optimo. Su comparación técnica aporta una base sólida para elegir el algoritmo más eficiente según las características de la red de caminos a optimizar.

2.1.3. ANTECEDENTES LOCALES.

Chirme (2023) en su tesis de maestría, Descripción por teoría de redes y simulación de rutas de transporte en el casco urbano de la ciudad del Cusco 2023 de la Universidad Nacional de San Antonio Abad del Cusco, simulo un sistema de rutas de transporte en el casco urbano de la ciudad del Cusco utilizando el programa Grafo, optimizando tiempos y costos en movilización dentro del centro urbano de Cusco.

Esta investigación es especialmente relevante para mi tesis, ya que aplica la teoría de redes (basada en grafos) en un contexto real de transporte urbano, enfocándose en la optimización de rutas. Aunque emplea simulación con software, su enfoque en minimizar tiempos y costos coincide directamente con los objetivos de los árboles de expansión en la optimización de caminos, reforzando la utilidad práctica de este enfoque en escenarios urbanos.

2.2. BASES TEÓRICAS

2.1.1. TEORÍA DE GRAFOS

La teoría de grafos es una rama de matemática e informática que se encarga del estudio de relaciones y conexiones entre objetos modeladas a través de estructuras llamadas grafos. Un grafo se encuentra compuesto por un conjunto de vértices y aristas, esta última representa las relaciones entre los vértices. Los grafos pueden ser dirigidos o no dirigidos, dependiendo de si las conexiones entre sus vértices tienen una dirección especifica.

Esta teoría se utiliza para modelar una amplia variedad de problemas en diversos campos, como la informática, ingeniería, redes de comunicación, optimización de rutas, entre otros. Su inicio se remonta con el problema de los puentes de Königsberg, donde la resolución de dicho problema contribuyó con la formulación de varios conceptos fundamentales en esta teoría.
(Biggs & Wilson, 1986)

2.1.2. GRAFO

DEFINICIÓN 2.1.2.1. Sean V(G) y E(G) dos conjuntos no vacíos. Se define el grafo G, denotado por $G = (V(G), E(G))$, al par ordenado formado por $V(G)$ el conjunto de vértices de G y $E(G)$ el conjunto de aristas de G, cada arista conectando dos vértices de $V(G)$. (Diestel, 2000)
Cuando no exista ambigüedad con otros grafos, se simplifica la notación como $G = (V, E)$.

DEFINICIÓN 2.1.2.2. Sean $G = (V, E)$ un grafo y los vértices $x, y \in V$. Se define una arista, denotado por $e = \{x, y\}$, al subconjunto $e = \{x, y\} \subseteq E$ que une o conecta a los vértices x e y.

En grafos no dirigidos, la orden de x e y no altera la arista, de modo que $\{x, y\} = \{y, x\}$.

Por conveniencia una arista también puede representarse solo por xy, además $E \subseteq V \times V$.

(Diestel, 2000)

DEFINICIÓN 2.1.2.3. Sean $G = (V, E)$ un grafo y una arista $e = xy \in E$. Se denominan los extremos de la arista e a los vértices x e y. (Bondy & Murty, 1976)

DEFINICIÓN 2.1.2.4. Sean $G = (V, E)$ un grafo y los vértices $x, y \in V$. Se llama:

a) Vértices adyacentes a x e y, si existe una arista que une dichos vértices, $\exists\ xy \in E$.

b) Aristas adyacentes, a las aristas que comparten un vértice en común, es decir que compartan el extremo de una arista.

c) $e = xy$ una arista incidente en los vértices x e $y \in V$, si $e = xy \in E$.

(Diestel, 2000)

ESCOLIO 2.1.2.1

Es importante advertir que la terminología utilizada en la teoría de grafos no está completamente estandarizada, lo que conlleva a ciertas variaciones en los términos empleados entre distintos textos. Por ejemplo, los vértices también suelen denominarse nodos, puntos, objetos u estados, y las aristas, bordes, ramas, lados, líneas o conexiones. Con el fin de evitar posibles confusiones, en el presente trabajo la notación utilizada serán los términos vértice y arista.

DEFINICIÓN 2.1.2.5. Sean $G = (V, E)$ un grafo y $e \in E$ una arista. Se dice que e es un bucle o lazo, si conecta un vértice consigo mismo, es decir, $e = \{x, x\}$ para algún $x \in V$.(Ruohonen, 2006)

DEFINICIÓN 2.1.2.6. Sean $G = (V, E)$ un grafo y $e \in E$ una arista. Se dice que la arista e es ponderada (o valuada), si se le asocia un número real no negativo, denotado por $w(e)$ el cual representa su peso o valor (por ejemplo, el costo, distancia, o tiempo). (Bondy & Murty, 2008)

DEFINICIÓN 2.1.2.7.

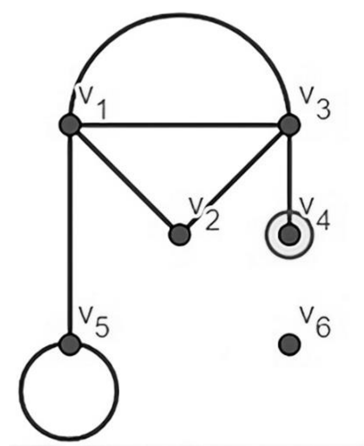
Sean $G = (V, E)$ un grafo y dos o más aristas $f, h \in E$. Se llaman aristas paralelas f y h, sí dichas aristas poseen los mismos vértices extremos, es decir son incidentes en los mismos extremos. (Ruohonen, 2006)

2.1.3. REPRESENTACIÓN GRÁFICA DE UN GRAFO

Sea $G = (V, E)$ un grafo. Cada vértice $v \in V$ será representado por un punto y cada arista $e \in E$ será representado por una curva o línea, como puede apreciarse en la Figura 1.

Figura 1

Representación gráfica de un grafo



La figura representa un grafo $G = (V, E)$ en $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ y con un conjunto de aristas $E = \{\{v_1v_3\}, \{v_1v_3\}, \{v_1v_2\}, \{v_1v_5\}, \{v_2v_3\}, \{v_4v_3\}, \{v_5v_5\}\}$.

Existen otras formas de representar un grafo. Por ejemplo, para ingresarlo a un programa es útil su representación matricial o mediante listas, lo cual abordaremos más adelante. A partir de los términos básicos previamente definidos, podemos enunciar algunas categorías o tipos de grafos.

DEFINICIÓN 2.1.3.1. Sean $G = (V(G), E(G))$ y $H = (V(H), E(H))$ dos grafos. Se denomina:

a) Grafo nulo a G, si no tiene vértices, es decir si $\mathrm{V(G)} = \emptyset$.

b) Grafo trivial (o singleton) a G, si no tiene arista, pero posee un vértice.

c) Grafos idénticos a G y H, denotado por $G = H$, si $V(G) = V(H)$ y $E(G) = E(H)$.

(Bondy & Murty, 1976)

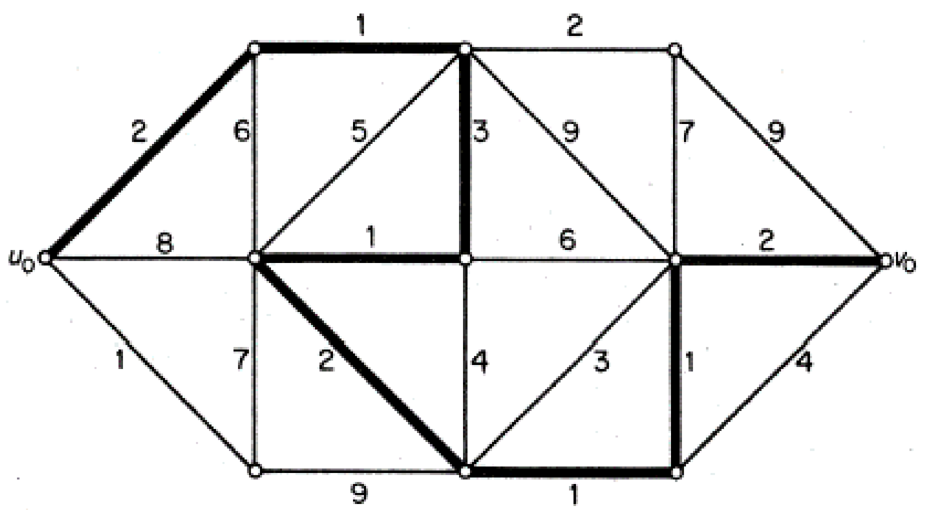
ESCOLIO 2.1.3.1

En adelante cuando mencionemos un grafo daremos por entendido que es un grafo no dirigido y no nulo salvo que se indique lo contrario.

DEFINICIÓN 2.1.3.2. Sea $G = (V, E)$ un grafo. Se dice que G es un grafo ponderado, si todas las aristas $e \in E$ son aristas ponderadas. (Bondy & Murty, 1976)

Figura 2

Representación gráfica de un grafo ponderado



Tomado de *Graph theory with applications* (p. 16), por Bondy, J., & Murty, U. (1976). North Holland.Oxford.

DEFINICIÓN 2.1.3.3. Sea $G = (V, E)$ un grafo. Se dice:

a) Grafo finito a G, si su conjunto de vértices V y su conjunto aristas E son finitos.

b) Grafo planar a G, si admite una representación gráfica en el plano tal que sus aristas se intersecan únicamente en sus vértices extremos.

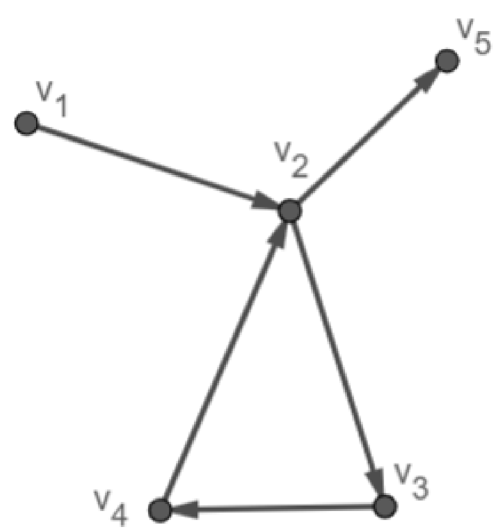
c) Grafo simple a G, si no contiene aristas paralelas ni bucles. (Bondy & Murty, 1976)

DEFINICIÓN 2.1.3.4. Sean $G = (V, E)$ un grafo y E un conjunto finito. Se denomina a G un multígrafo, si cada arista une dos vértices distintos de V (es decir, no existen bucles) y si cualquier par de vértices distintos en V están unidos por un número finito de aristas. Dicho de otro modo, un multígrafo admite aristas múltiples, pero no bucles. (Koh et al., 2007)

DEFINICIÓN 2.1.3.5. Sea un grafo $\mathrm{G} = (\mathrm{V}, \mathrm{E})$. Se dice que G es un grafo dirigido (o dígrafo), si cada arista de E es un par ordenado (x, y) con $x, y \in V$ y $x \neq y$. Donde cada par (x, y) representa una flecha que apunta desde el vértice origen x hacia el vértice de destino y. (Jungnickel, 2008)

Figura 3

Representación gráfica de un grafo dirigido



Tenemos un grafo dirigido $G = (V, E)$ en $V = \{v_1, v_2, v_3, v_4, v_5\}$ y con un conjunto de flechas $E = \{(v_1, v_2), (v_2, v_5), (v_2, v_3), (v_3, v_4), (v_4, v_2)\}$.

Recordemos que las aristas se denotan por $\{x, y\}$ en caso de un grafo no dirigido y por (x, y) en el caso de un grafo dirigido. Cuando las aristas de un grafo no tienen dirección, se denomina grafo no dirigido o simplemente grafo.

DEFINICIÓN 2.1.3.6. Sea $G = (V, E)$ un grafo simple con n vértices.

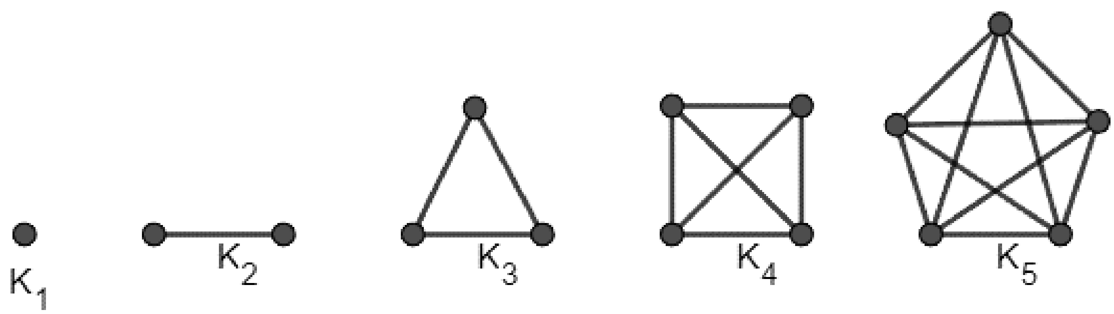
Se denomina a G un grafo completo, denotado por K_n (o grafo n-regular), si contiene todas las aristas posibles entre todos sus vértices. En consecuencia, el número de aristas de K_n :

$$|E| = \binom{n}{2} = \frac{n(n-1)}{2}.$$

(Ruohonen, 2006)

Figura 4

Grafos completos



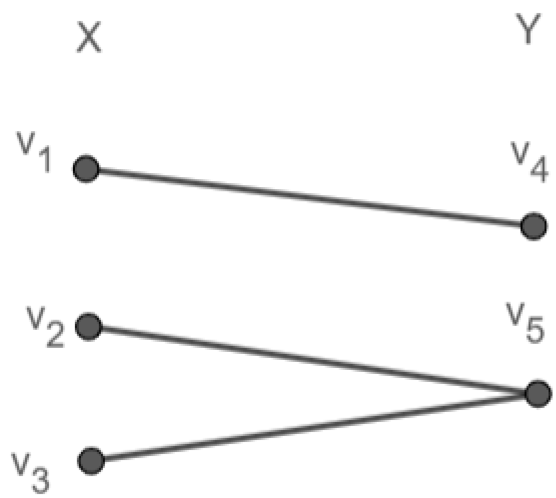
Adaptado de Graph Theory (p. 3), por Ruohonen, K. (2006). Lecture Notes

Observamos los primeros cinco grafos completos K_1, K_2, K_3, K_4 y K_5, notamos que un grafo es completo si todos sus vértices son adyacentes por pares.

DEFINICIÓN 2.1.3.7. Sea $G = (V, E)$ un grafo. Se dice que G es un grafo bipartito, si su conjunto de vértices V se puede expresar como la unión disjunta de dos subconjuntos X e Y tales que cada arista en E conecta un extremo en X y el otro extremo en Y. A esta partición se llama bipartición del grafo G. (Bondy & Murty, 1976)

Figura 5

Representación gráfica de un grafo bipartito

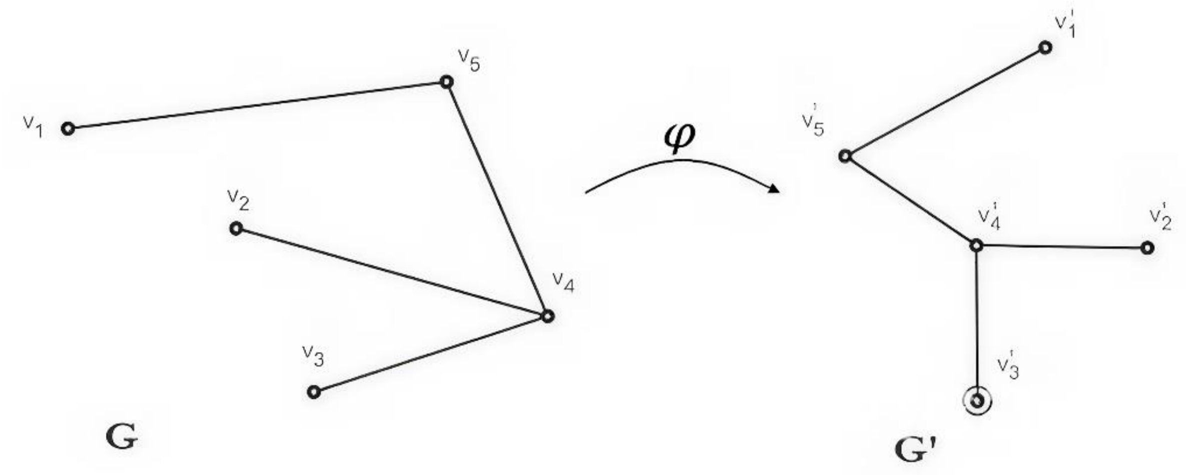


Se observa un grafo no dirigido $G = (V, E)$, $E = \{v_1v_4, v_2v_5, v_3v_5\}, V = \{v_1, v_2, v_3, v_4, v_5\}$ con $X = \{v_1, v_2, v_3\} \wedge Y = \{v_4, v_5\} \subset V, X \neq Y$.

DEFINICIÓN 2.1.3.8. Sean $G = (V, E)$ y $G' = (V', E')$ dos grafos. Se dice que G y G' son isomorfos, denotado $G \cong G'$, sí existe una biyección $\varphi : V \longrightarrow V'$ talque $\forall x, y \in E$ se cumple $xy \in E \Leftrightarrow \varphi(x)\varphi(y) \in E'$, φ se llama un isomorfismo entre G y G'. (Diestel R., 2000)

Figura 6

Representación gráfica de grafos isomorfos



Se observa a los grafos $G = (V, E)$, $G' = (V', E')$ y una biyección $\varphi: V \longrightarrow V'$.

2.1.4. REPRESENTACIÓN MATRICIAL DE UN GRAFO

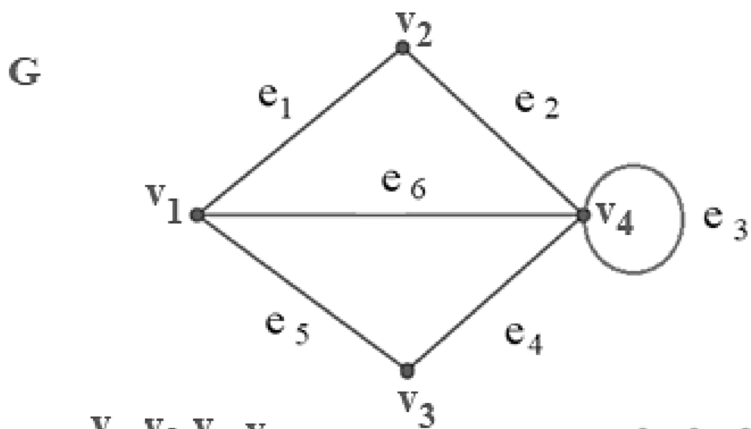
Nos referiremos al número de vértices del grafo por $|V|$ y al número de aristas del grafo por $|E|$

DEFINICIÓN 2.1.4.1. Sean $G = (V, E)$ un grafo, su conjunto de vértices $V = \{v_1, \dots, v_n\}$ y su conjunto de aristas $E = \{e_1, \dots, e_m\}$. Se llama matriz de adyacencia de G, denotado por $A(G) = [a_{ij}]$, a la matriz de orden $|V| \times [V]$ tal que cada entrada a_{ij} representa el número de aristas que conectan el vértice v_i con el vértice v_j. (Bondy & Murty, 1976)

DEFINICIÓN 2.1.4.2. Sean $G = (V, E)$ un grafo, su conjunto de vértices $V = \{v_1, \dots, v_n\}$ y su conjunto de aristas $E = \{e_1, \dots, e_m\}$. Se llama matriz de incidencia de G, denotada por $M(G) = [m_{ij}]$, a la matriz de orden $|V| \times |E|$, talque cada entrada m_{ij} indica el número de veces que el vértice v_i y la arista e_j inciden. (Bondy & Murty, 1976)

Figura 7

Matriz de Adyacencia e Incidencia de G



$$A(G) = \begin{array}{c c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} & \\ \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix} & \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \end{array} \qquad M(G) = \begin{array}{c c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} & \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 2 & 1 & 0 & 1 \end{pmatrix} & \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \end{array}$$

Se observa al grafo $G = (V, E)$ y sus matrices de Adyacencia e Incidencia, representadas por $A(G)$ y $M(G)$ respectivamente.

ESCOLIO 2.1.4.1

Un bucle se cuenta dos veces incidente, es decir como dos aristas incidentes en el mismo vértice.

DEFINICIÓN 2.1.4.3. Sean $G = (V, E)$, $G' = (V', E')$ dos grafos, $U \subseteq V$ un subconjunto de vértices de G y $F \subseteq E$ un subconjunto de aristas de E del grafo completo de G. Se definen las siguientes operaciones entre grafos:

a) La union de los grafos G y G', denotada por $G \cup G'$, es el grafo $(V \cup V', E \cup E')$.

b) La interseccion de los grafos G y G', denotada por $G \cap G'$, es el grafo $(V \cap V', E \cap E')$.

c) Se dice que los grafos G y G' son disjuntos, si $G \cap G' = \emptyset$,

d) El grafo $G - U$ se obtiene eliminando de G todos los vértices en $U \cap V$ junto con las aristas incidentes a estos vértices.

e) Si $U = \{v\}$ es un conjunto unitario, se escribe $G - U = G - v$.

f) El grafo G despues de eliminar el conjunto de aristas F, dado por $G - F = (V, E - F)$.

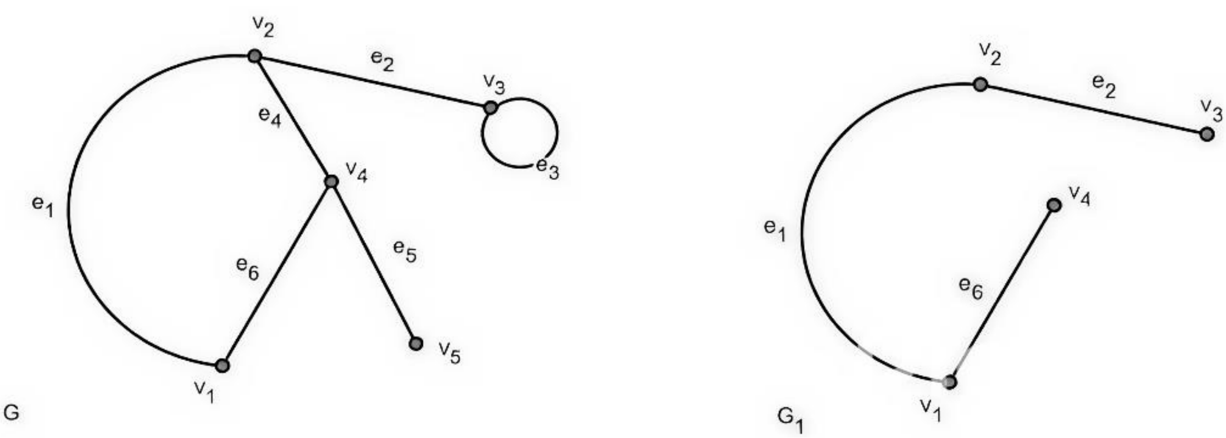
g) El grafo G despues de añadir las aristas de F, esta dado por $G + F := (V, E \cup F)$ donde se asume que $V(G) \subseteq V(F)$. (Diestel R., 2000)

2.1.5. SUBGRAFOS Y COMPLEMENTOS

DEFINICIÓN 2.1.5.1. Sean $G_1 = (V_1, E_1)$ y $G = (V, E)$ dos grafos. Se dice que G_1 es un subgrafo de G, denotado por $G_1 \subseteq G$, sí se cumple que $\emptyset \neq V_1 \subseteq V$ y $E_1 \subseteq E$. (Diestel R., 2000)

Figura 8

Subgrafo de G

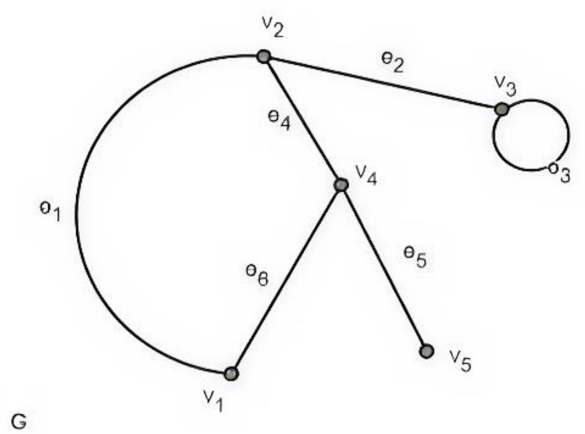


Son los grafos $G_1 = (V_1, E_1)$ y $G = (V, E)$ tal que $G_1 \subseteq G$.

DEFINICIÓN 2.1.5.2. Sean $G = (V, E)$ un grafo y $G_1 = (V_1, E_1)$ un subgrafo de G. Se llama a G_1 un subgrafo expansivo (o generador) de G, si $V_1 = V$; es decir, si contiene todos los vértices de G. (Grimaldi, 2004)

Figura 9

Subgrafo expansivo de G

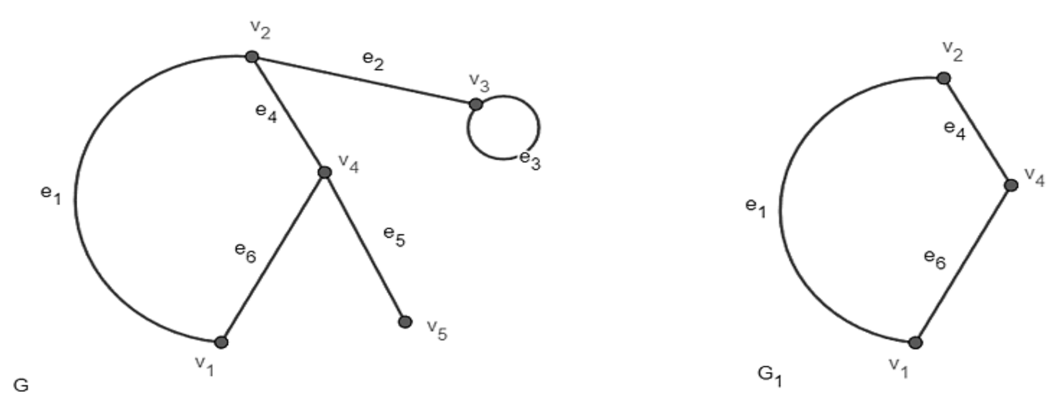


Se observan los grafos $G_1 = (V_1, E_1)$ y $G = (V, E)$, donde G_1 es un subgrafo de G y $V_1 = V$, por lo cual G_1 es un subgrafo expansivo de G.

DEFINICIÓN 2.1.5.3. Sean $G = (V, E)$ y $G_1 = (V_1, E_1)$ dos grafos. Se dice que G_1 es un subgrafo inducido de G, si G_1 es subgrafo de G y contiene todas las aristas $xy \in E$ tales que $x, y \in V_1$. (Diestel R., 2000)

Figura 10

Subgrafo Inducido de G



Se observa que G_1 es un subgrafo inducido por G.

DEFINICIÓN 2.1.5.4. Sea $G = (V, E)$ un grafo. Se llama complemento de G, denotado por $\bar{G}$, al grafo $\bar{G}$, tal que $V(\bar{G}) = V(G)$ y en este grafo dos vértices $x, y \in V$ son adyacentes en $\bar{G}$ sí no son adyacentes en G.

Para cualquier grafo G de n vértices, al superponer G y su complemento $\bar{G}$ en los mismos vértices, se obtiene el grafo completo K_n. En consecuencia, para cualquier grafo G de n vértices se cumple la siguiente relación:

$$|E(G)| + |E(\bar{G})| = |E(K_n)| = \binom{n}{2}$$

Donde:

$|E(G)|$ es el número de aristas del grafo G,

$|E(\bar{G})|$ es el número de aristas del grafo $\bar{G}$ y

$|E(K_n)|$ es el número de aristas del grafo completo K_n.

Es decir, el número de aristas de un grafo G de n vértices más el número de aristas de su complemento $\bar{G}$ es igual al numero de aristas del grafo completo K_n de n vértices.

(Koh, Dong, & Tay, 2007)

2.1.6. ORDEN Y GRADOS DE UN GRAFO

DEFINICIÓN 2.1.6.1. Sea $G = (V, E)$ un grafo. Se definen como el orden del grafo G, denotado por |V|, al número de vértices de G.

Y el tamaño del grafo G, denotado por |E|, al número de aristas de G. (Diestel R., 2000)

DEFINICIÓN 2.1.6.2. Sean $G = (V, E)$ un grafo con $E \neq \emptyset$ y $v \in V$. Se llama el grado de un vértice v, denotado por $\deg(v)$ o $d(v)$, al número de aristas que inciden en v.

Un vértice de grado 1 se llama vértice colgante (o vértice final) y un vértice de grado cero se llama vértice aislado. Además, se definen:

a) El grado mínimo de G, denotado por $\delta(G)$, como

$$\delta(G) = min\{d(v)/v \in V\}.$$

b) El grado máximo de G, denotado por $\Delta(G)$, como

$$\Delta(G) = max\{d(v)/v \in V\}.$$

c) El grado promedio de G, denotado por $d(G)$, como

$$d(G) = \frac{1}{|V|} \sum_{v \in V} d(v).$$

Finalmente obteniendo:

$$\delta(G) \leq d(G) \leq \Delta(G).$$

(Diestel R., 2000)

TEOREMA 2.1.6.1.

Sean $G = (V, E)$ un grafo o multígrafo, con $v \in V$ y $E = \{e_1, e_2, \dots, e_m\}$, donde $m = |E|$ es el número de aristas de G. Se cumple que $\sum_{v \in V} \mathrm{d}(v) = 2|E| = 2m$. Es decir, la suma de todos los grados de los vértices de G será dos veces la cantidad de aristas de G.

DEMOSTRACION

Sea un grafo o multígrafo $G = (V, E)$, con $|E| = m$ (el número de arista de G) y $d(v)$ el grado de un vértice $v \in V$. Cada arista de G conecta a dos vértices.

Entonces al contar el total de grados de todos los vértices cada arista contribuye con dos vértices, así podemos escribir la relación $|E| = \frac{1}{2} \sum_{v \in V} \mathrm{d}(v)$.

Despejando $2|E| = \sum_{v \in V} \mathrm{d}(v)$, como $|E| = m$ tendriamos $2m = 2|E|$.

Por lo tanto, hemos demostrado que $\sum_{v \in V} \mathrm{d}(v) = 2m$. (Bondy & Murty, 2008)

COROLARIO 2.1.6.1. Sea $G = (V, E)$ un grafo o multígrafo. El número de vértices de grado impar de G es par.

DEMOSTRACION

Sea $G = (V, E)$ un grafo con $P \subset V$ el conjunto de vértices pares en G e $I \subset V$ el conjunto de vértices impares de G.

Mostraremos que $|I|$ es par, como $V = PUI$ y por el Teorema 2.1.6.1:

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

$$\sum_{v \in P} d(v) + \sum_{v \in I} d(v) = 2|E|$$

Como $\sum_{v \in P} d(v)$ y $2|E|$ son pares, entonces $\sum_{v \in I} d(v)$ también es par.

Como $d(v)$ es impar para cada v en I, se deduce que $|I|$ debe ser par.

Lo que muestra que el subconjunto de vértices de grado impar de un grafo posee un número par de elementos. (Koh, Dong, & Tay, 2007)

2.3. MARCO CONCEPTUAL

a) **Grafo. -** Sean V(G) y E(G) dos conjuntos no vacíos. Se define el grafo G, denotado por $G = (V(G), E(G))$, al par ordenado formado por $V(G)$ el conjunto de vértices de G y $E(G)$ el conjunto de aristas de G, cada arista conectando dos vértices de $V(G)$.

b) **Arista ponderada. -** Sean $G = (V, E)$ un grafo y $e \in E$ una arista. Se dice que una arista e es ponderada (o valuada), si se le asocia un valor real no negativo, denotado por $w(e)$ el cual representa su peso o valor**.**

c) **Grafo ponderado. -** Sea $G = (V, E)$ un grafo. Se dice que G es un grafo ponderado, si todas las aristas $e \in E$ son aristas ponderadas.

d) **Paseo. -** Sean $G = (V, E)$ un grafo y los vértices $x_0, x_1, \ldots, x_n \in V$. Se llama paseo de x_0 a x_n, denotado por $x_0 - x_n$, a una secuencia finita (sin bucles) de vértices y aristas de G, que comienza en el vértice x_0 y termina en el vértice x_n englobando las n aristas $e_1, \ldots, e_n \in E$.

e) **Camino.** – Sean $G = (V, E)$ un grafo de orden n y un paseo $x_0 - x_n$ en G. Se llama camino, a un paseo en el que ningún vértice se repite, es decir todos sus vértices son distintos entre sí.

f) **Subgrafo.** – Sean $G_1 = (V_1, E_1)$ y $G = (V, E)$ dos grafos. Se dice que G_1 es un subgrafo de G, denotado por $G_1 \subseteq G$, sí se cumple que $\emptyset \neq V_1 \subseteq V$ y $E_1 \subseteq E$.

g) **Subgrafo expansivo.** –Sean $G = (V, E)$ un grafo y $G_1 = (V_1, E_1)$ un subgrafo de G. Se llama a G_1 un subgrafo expansivo de G, si $V_1 = V$ es decir si contiene todos los vértices de G.

h) **Grafo conexo. -** Sea $G = (V, E)$ un grafo no vacío. Se dice que G es conexo o conectado, si para cualesquiera dos vértices $u, v \in V$ están unidos por un camino. Si existe una partición de V en subconjuntos no vacíos $V_1, V_2, \ldots, V_k$, diremos que dos vértices u y v están conectados si y sólo si uv ambos pertenecen al mismo subconjunto V_i para algún i.

i) **Grafo disconexo. -** Sea un grafo no dirigido $G = (V, E)$. Se dice que G es disconexo (o desconectado), si no es conexo, es decir sí G tiene más de un componente conexo.

j) **Árbol. -** Sea $G = (V, E)$ un grafo no dirigido y sin bucles. Se dice que G es un árbol, si es conexo y acíclico (no contiene ciclos).

k) **Árbol de expansión. -** Sean $G = (V, E)$ un grafo conexo no dirigido y T un subgrafo expansivo de G. Se dice que T es un árbol de expansión (o generador) de G, si T es un árbol.

l) **Árbol de expansión máximo. -** Sean $G = (V, E)$ un grafo conexo no dirigido y T un árbol de expansión de G.. Se dice que T es un árbol de expansión máximo (o maximal) de G, si la suma total de los pesos de las aristas es el máximo, es decir, es la mayor entre todos los árboles de expansión de G

m) **Árbol de expansión mínimo. -** Sean $G = (V, E)$ un grafo conexo no dirigido y T un árbol de expansión de G.. Se dice que T es un árbol de expansión mínimo (o minimal) de G, si la suma total de los pesos de sus aristas es el mínimo, es decir, es la menor entre todos los árboles de expansión de G.

n) **Algoritmo de Prim**. - El algoritmo de Prim inicia tomando un vértice aleatorio y en forma secuencial, va incorporando aristas de menor peso que inciden en los vértices ya incluidas con los que aún no están en el árbol parcial que se va construyendo, hasta completar el árbol de expansión mínimo.

o) **Algoritmo de Kruskal**. - El algoritmo de Kruskal construye un árbol de expansión mínimo seleccionando iterativamente aristas de menor peso en orden creciente (en caso de minimización), asegurándose que en cada iteración no se forme ningún ciclo durante la construcción del árbol,

CAPITULO III: METODOLOGÍA

3.1. METODO DE INVESTIGACIÓN

Por su método de investigación el presente trabajo es documental teórico pues se concentra en la recopilación de información en forma documental. (Ñaupas et al., 2018)

3.2. TIPO DE INVESTIGACIÓN

El trabajo realizado corresponde a un tipo de investigación básica o fundamental, dado que se desarrollaron definiciones, teoremas y resultados de caminos en grafos y árboles de expansión.
Se dice que es básica dado que sirve de base a la investigación aplicativa o tecnológica y es fundamental pues es esencial para el desarrollo de la ciencia. (Ñaupas et al., 2018)

3.3. NIVEL DE INVESTIGACIÓN

En la investigación se desarrollaron formalmente la teoría de grafos y árboles de expansión buscando la optimización de caminos mediante árboles de expansión óptima.

Según Ñaupas et al. (2014), es de nivel descriptiva ya que tiene como objetivo recopilar datos e información sobre las características, propiedades, aspectos o dimensiones de los sujetos del estudio. Y es de nivel exploratoria pues se realiza una búsqueda de información, con el propósito de formular problemas e hipótesis para una investigación más profunda de carácter explicativo.

3.4. TECNICA DE INVESTIGACIÓN

Se aplica una técnica de investigación conceptual, dado que esta referida a las técnicas que hacen posible las operaciones racionales, de abstracción, generalización, análisis, síntesis, clasificación o comparación y reglas lógico - formales. (Ñaupas et al., 2018)

3.5. AMBITO DE ESTUDIO

3.5.1. LOCALIZACION

El presente trabajo se realizó en la Escuela Profesional de Matemática de la Universidad Nacional de San Antonio Abad de Cusco desde enero de 2024 a marzo de 2025.

CAPITULO IV: RESULTADOS

4.1. TEORIA DE CAMINOS, CONEXIDAD Y ÁRBOLES DE EXPANSION

Desarrollaremos fundamentos teóricos de la teoría de caminos, que permitirán comprender cómo se estructura el tránsito entre vértices en un grafo. Para posteriormente profundizar en conexidad, indispensable en la existencia de caminos entre todos los vértices de un grafo.

4.1.1. CAMINOS

DEFINICIÓN 4.1.1.1. Sean $G = (V, E)$ un grafo y los vértices $x_0, x_1, \dots, x_n \in V$. Se llama paseo de x_0 a x_n, denotado por $x_0 - x_n$, a una secuencia finita sin bucles de vértices y aristas de G, que comienza en el vértice x_0 y termina en el vértice x_n englobando las n aristas $e_1, \dots, e_n \in E$. Donde cada arista $e_i \in E$ conecta los vértices consecutivos x_{i-1} y x_i, para $i = 1,2, \dots, n$.
Se denomina paseo cerrado, si el paseo comienza y termina en el mismo vértice, es decir $x_0 = x_n$ con $n > 1$.Caso contrario se denomina paseo abierto, cuando $x_0 \neq x_n$.

En un paseo se pueden repetir tanto vértices como aristas. (Grimaldi, 2004)

DEFINICIÓN 4.1.1.2. Sean $G = (V, E)$ un grafo y $P = x - y$ un paseo en G, donde $x, y \in V$ son los vértices de inicio y fin del paseo respectivamente. Se define la longitud del paseo P, denotado por P^m, al número m de aristas que conforman el paseo P. (Diestel R., 2000)

Se permite que $m = 0$ en cuyo caso el paseo se llama trivial, y se representa por $P^0 = K_1$ que corresponde a un grafo de un solo vértice sin ninguna arista.

DEFINICIÓN 4.1.1.3. Sean $G = (V, E)$ un grafo y $P = x_0 \dots x_k$ un paseo en G , con $x_0, x_2, \dots, x_k \in V$ y $0 \leq i \leq j \leq k$. Se definen los siguientes sub paseos de P:

a) El paseo P que inicia normalmente y termina cortado en x_i se denota por:

$$Px_i = x_0 \dots x_i$$

b) El paseo P que inicia desde x_i y termina normalmente se denota por:

$$x_i\,P = x_i \dots x_k$$

c) El paseo P que inicia desde x_i y termina cortado en x_j se denota por:

$$x_i\,P\,x_j = x_i \dots x_j$$

(Diestel R., 2000)

ESCOLIO 4.1.1.1

Para la unión de caminos, se utiliza la notación que vimos en la última definición. Si Px, xQy, yRz y zS son cuatro caminos su unión será nuevamente un camino denotado por: $Px \cup xQy \cup yRz \cup zS$ o simplemente $PxQyRzS$.

DEFINICIÓN 4.1.1.4. Sean $G = (V, E)$ un grafo de orden n y un paseo $x_0 - x_n$ en G. Se definen las siguientes estructuras:

a) Se llama sendero o recorrido, a un paseo en el que ninguna arista se repite en el paseo, es decir todas sus aristas son todas distintas entre sí.

b) Se llama sendero cerrado (o circuito), a un sendero tal que $x_0 = x_n$.

c) Se llama camino, a un paseo en el que ningún vértice se repite, es decir todos sus vértices son distintos entre sí.

d) Se llama ciclo, a un camino cerrado con al menos 3 vértices involucrados, es decir $x_0 = x_n$ y el número de vértices es $n \geq 3$. (Grimaldi, 2004)

ESCOLIO 4.1.1.2.- Un grafo el cual no posee ciclos es llamado un grafo acíclico.

DEFINICIÓN 4.1.1.5. Sean $G = (V, E)$ un grafo y C un ciclo de G.

La longitud de un ciclo C, denotado por C^m, es el número m de aristas que posee dicho ciclo.

(Diestel R., 2000)

Las definiciones anteriormente mencionadas se resumen en la Figura 11:

Figura 11

Paseo, sendero, camino y ciclo

VÉRTICES REPETIDOS	ARISTAS REPETIDAS	ABIERTO	CERRADO	NOMBRE
Si	Si	Si		*PASEO (ABIERTO)*
Si	Si		Si	*PASEO (CERRADO)*
Si	No	Si		*SENDERO*
Si	No		Si	*CIRCUITO*
No	No	Si		*CAMINO*
No	No		Si	*CICLO*

Adaptado de Discrete and combinatorial mathematics (p. 516), por Ralph P. Grimaldi, 2004, Pearson.

TEOREMA 4.1.1.1. Sea $G = (V, E)$ un grafo, con $a, b \in V$ y $a \neq b$. Si existe un sendero en G de a a b, entonces hay un camino (en G) que une a los vértices a y b.

DEMOSTRACION

Supongamos que hay un sendero de a hasta b, entonces no se repite en el sendero ninguna arista.

Seleccionamos un sendero de longitud más corta, digamos:

$$\{a, x_1\}, \{x_1, x_2\}, \dots, \{x_n, b\}.$$

Si este sendero no es un camino, tenemos la situación:

$\{a, x_1\}, \{x_1, x_2\}, \dots, \{x_{k-1}, x_k\}, \{x_k, x_{k+1}\}, \dots, \{x_{m-1}, x_m\}, \{x_m, x_{m+1}\}, \dots, \{x_n, b\}$

donde $k < m$ y $x_k = x_m$.

Posiblemente con $k = 0$ y $a = x_0 = x_m$ o $m = n + 1$ y $x_k = b = x_{n+1}$.

Pero entonces tenemos una contradicción porque

$\{a, x_1\}, \{x_1, x_2\}, \dots, \{x_{k-1}, x_k\}, \{x_m, x_{m+1}\}, \dots, \{x_n, b\}$ es el sendero más corto de a a b.

(Grimaldi, 2004)

TEOREMA 4.1.1.2. Sea $G = (V, E)$ un grafo, entonces G contiene un camino de longitud $\delta(G)$ y un ciclo de longitud al menos $\delta(G) + 1$ (siempre que $\delta(G) \geq 2$).

DEMOSTRACION

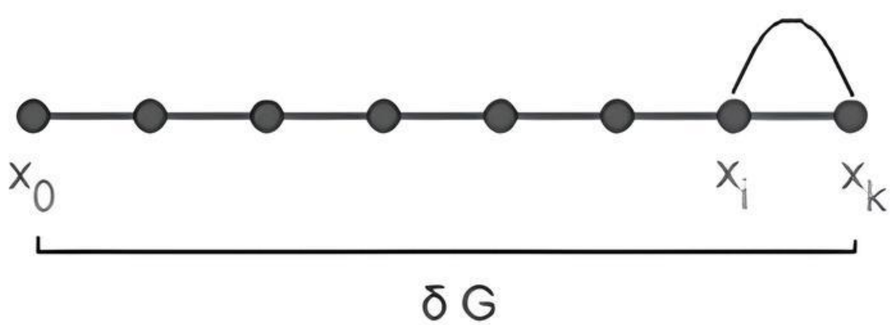
Sea $x_0, \dots, x_k$ el camino más largo en G, entonces todos los vértices de G se encuentran en este camino (Ver Figura 12).

Por tanto $k \ \geq |E| \geq \delta(G)$ (siempre que $\delta(G) \geq 2$).

Sea $i < k$ el mínimo valor talque $x_i x_k \in E(G)$, entonces $x_0, \dots, x_k \cup x_k x_i$ es un ciclo de longitud al menos $\delta(G) + 1$. (Diestel, 2000)

Figura 12

El camino más largo $x_0, \dots, x_k$ *y el ciclo* C: $x_0, \dots, x_k \cup x_k x_i$



Adaptado de Graph theory (p. 8), por Diestel, 2000, Springer.

4.1.2. CONEXIDAD

Ahora desarrollaremos los conceptos fundamentales de conexidad, una propiedad esencial para el estudio de estructuras como los árboles. La conexidad permite establecer la existencia de un camino para cualquier par de vértices en un grafo, lo que constituye un requisito fundamental para abordar con rigurosidad árboles y árboles de expansión.

DEFINICIÓN 4.1.2.1. Sea un grafo $G = (V, E)$. Se dice que G es conexo o conectado, si para cualesquiera dos vértices $u, v \in V$ están unidos por un camino.

Si existe una partición de V en subconjuntos no vacíos $V_1, V_2, \dots, V_k$, diremos que dos vértices u y v están conectados si y sólo si uv ambos pertenecen al mismo subconjunto V_i para algún i.

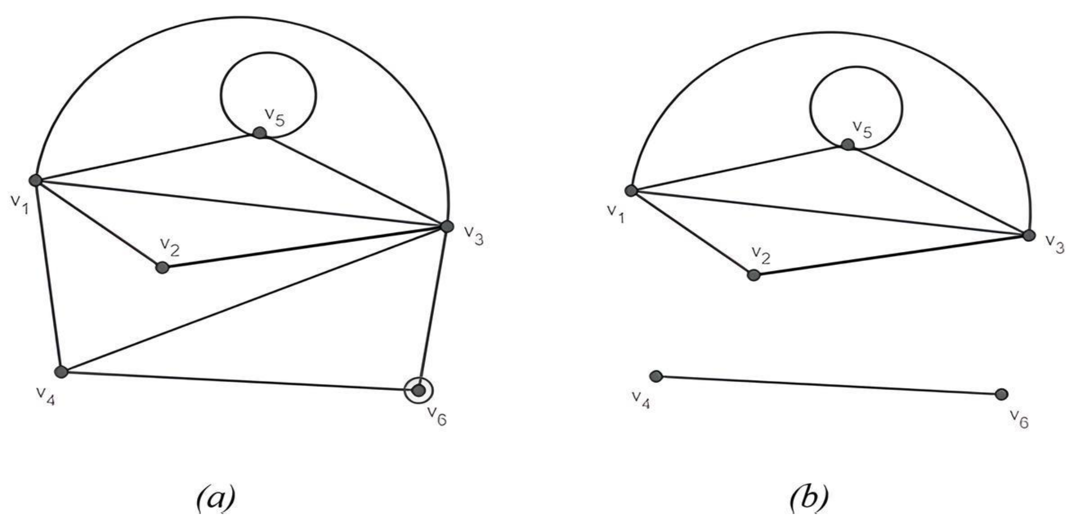
Se denominan componentes conexas de G, a los sub grafos $G[V_1], G[V_2], \dots, G[V_k]$ conexos.

El número de componentes conexas de G se denota por $k(G)$. Si G tiene exactamente un componente, $k(G) = 1$, diremos que G es conexo. (Bondy & Murty, 2008)

DEFINICIÓN 4.1.2.2. Sea un grafo no dirigido $G = (V, E)$. Se dice que G es disconexo (o desconectado), si no es conexo, es decir sí G tiene más de un componente conexo.

Figura 13

Grafo Conexo y Grafo Disconexo



Observamos en: *(a)* un grafo conexo y *(b)* un grafo disconexo con dos componentes conexas.

ESCOLIO 4.1.2.1

Toda componente conexa de un grafo G es por definición un sub grafo conexo de G.

TEOREMA 4.2.1.1.

Sea $G = (V, E)$ un grafo. Si G es disconexo, entonces su complemento $\bar{G}$ es conexo.

DEMOSTRACION

Supongamos que G es disconexo, para demostrar que su complemento $\bar{G}$ es conexo, mostraremos que para cada dos vértices u, v en $\bar{G}$ están unidos por un camino en $\bar{G}$.

Por lo tanto, sean $u, v \in V(\bar{G})$ y recordemos que $V(\bar{G}) = V(G)$.

Si los vértices u, v están en componentes diferentes en G, entonces u y v están unidos por una arista en $\bar{G}$ (por definición de complemento de G).

Si u y v están en una misma componente de G, sea w cualquier vértice en otro componente de G, entonces uwv es un camino $u - v$ en $\bar{G}$. Lo que completa la prueba.

(Koh, Dong, & Tay, 2007)

4.1.3. ARBOL

Desarrollaremos continuación definiciones, características y algunos resultados clave de árboles para posteriormente abordar con mayor profundidad los árboles de expansión.

DEFINICIÓN 4.1.3.1. Sea $G = (V, E)$ un grafo no dirigido y sin bucles. Se dice que G es un árbol, si es conexo y acíclico (no contiene ciclos). (Grimaldi, 2004)

En adelante, para enfatizar la estructura del grafo, cuando hablemos de un árbol lo denotaremos como T en lugar de G.

DEFINICIÓN 4.1.3.2. Sea $T = (V, E)$ un árbol. Se llama hoja a todo vértice de grado 1 en el árbol T, es son aquellos vértices que tienen exactamente una arista incidente. Además, todo árbol tiene al menos dos hojas y, si se elimina una hoja del árbol el subgrafo resultante también es un árbol.

(Diestel, 2000)

TEOREMA 4.1.3.1. (UNICIDAD DE CAMINO)

Sean $T = (V, E)$ un árbol y a, $b \in V$ dos vértices distintos. Entonces hay un camino único $a - b$ que conecta estos vértices en T.

DEMOSTRACION

Dado que T es un árbol, sabemos que T es conexo y hay al menos un camino $a - b$ en T.

Si hubiera más caminos, entonces a partir de dos de esos algunos de los bordes formarían un ciclo.

Como T no tiene ciclos, se descarta la posibilidad de que existan múltiples caminos y garantiza que existe exactamente un camino entre a y b. (Grimaldi, 2004)

TEOREMA 4.1.3.2.

Sea $T = (V, E)$ un árbol, entonces se cumple que $|V| = |E| + 1$. Donde $|V|$ denota el número de vértices de T y $|E|$ es el número de aristas de T.

DEMOSTRACION:

La prueba se obtiene aplicando el principio de inducción matemática.

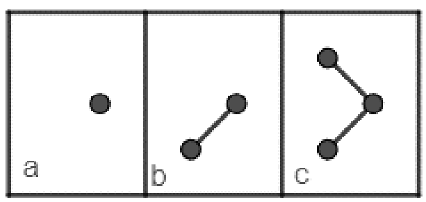
Si $|E| = 0$ y $|V| = |E| + 1 = 1 \Rightarrow$ se tiene un vértice aislado, ver Figura 14(a).

Si $|E| = 1$ y $|V| = |E| + 1 = 2 \Rightarrow$ se tiene un árbol que consta de 2 vértices, ver Figura 14(b).

Si $|E| = 2$ y $|V| = |E| + 1 = 3 \Rightarrow$ se tiene un árbol que consta de 3 vértices, ver Figura 14 (c).

Figura 14

Árbol de 1,2 y 3 vértices



Adaptado de *Discrete and combinatorial mathematics* (p. 583), por Ralph P. Grimaldi, 2004, Pearson.

Supongamos ahora que el teorema es cierto para todo árbol que contenga como máximo k aristas, donde $k > 0$, entonces:

$|E| = k$ y $|V| = |E| + 1 \Rightarrow$ el árbol consta de $k + 1$ vértices.

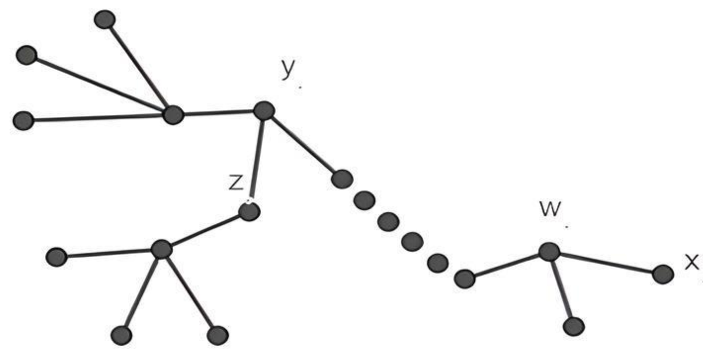
Consideremos gráficamente un árbol $T = (V, E)$ como el de la Figura 15, con $|E| = k + 1$. Si la arista con puntos finales yz se elimina de T, se obtiene dos subárboles:

$T_1 = (V_1, E_1)$ y $T_2 = (V_2, E_2)$, donde $|V| = |V_1| + |V_2|$ y $|E| = |E_1| + |E_2| + 1$.

Uno de estos subárboles podría consistir en un solo vértice si, por ejemplo, si se eliminara la arista con los vértices finales wx.

Figura 15

Árbol $T = (V, E)$ *con* $|E| = k + 1$



Adaptado de *Discrete and combinatorial mathematics* (p. 583), por Ralph P. Grimaldi, 2004, Pearson. La parte punteada indica la parte del árbol no aparece en la figura.

Como $0 \leq |E_1| \leq k$ y $0 \leq |E_2| \leq k$, deducimos por hipótesis de inducción que $|V_i| = |E_i| + 1$, para $i = 1,2$.

En consecuencia:

$$|V| = |V_1| + |V_2|$$

$$|V| = (|\mathrm{E}_1| + 1) + (|\mathrm{E}_2| + 1)$$

$$|V| = (|\mathrm{E}_1| + |\mathrm{E}_2| + 1) + 1$$

$$|V| = |\mathrm{E}| + 1$$

lo cual termina la prueba de inducción. (Grimaldi, 2004)

LEMA 4.1.3.1. (LEMA DEL VÉRTICE FINAL)

Sea $T = (V, E)$ un árbol con $|V| \geq 2$, entonces T tiene al menos dos vértices finales (o colgantes), es decir, T tiene al menos dos vértices de grado 1.

DEMOSTRACION

Sea un árbol $T = (V, E)$ con $|V| = n \geq 2$.

Por el Teorema 2.2.8.2 en todo árbol se tiene $|V| = |E| + 1$, por tanto $|E| = n - 1$.

Por el Teorema 2.2.6.1 tenemos $2|E| = \sum_{v \in V} d(v)$ y por el resultado anterior tendríamos

$$\sum_{v \in V} d(v) = 2(n-1).$$

Como T es un árbol, entonces es conexo y por tanto tenemos $d(v) \geq 1$ para todo $v \in V$.

Si T posee k vértices colgantes, entonces cada uno de los otros $n - k$ vértices son de grado al menos 2 y tenemos:

$$2(n-1) = 2|E|$$

$$2(n-1) = \sum_{v \in V} d(v) \geq k + 2(n-k)$$

De esto vemos que:

$$2(n-1) \geq k + 2(n-k)$$

$$(2n-2) \geq (k + 2n - 2k)$$

$$-2 \geq -k$$

$$k \geq 2$$

Siendo $k \geq 2$ el número de vértices colgantes del árbol T. (Grimaldi, 2004)

TEOREMA 4.1.3.3.

Sea $G = (V, E)$ un grafo sin bucles. Las siguientes afirmaciones son equivalentes:

a) G es un árbol.

b) G es conexo, y la eliminación de cualquier arista de G desconecta G en dos subgrafos que son árboles.

c) G no contiene ciclos y satisface $|V| = |E| + 1$.

d) G es conexo y satisface $|V| = |E| + 1$.

e) G no contiene ciclos y para cualesquiera $a, b \in V$ con $\{a, b\} \notin E$, el grafo resultante de añadir a G la arista $\{a, b\}$ contiene exactamente un ciclo.

DEMOSTRACION

Probaremos $(a) \Rightarrow (b) \Rightarrow (c) \Rightarrow (d) \Rightarrow (e) \Rightarrow (a)$

$[(\boldsymbol{a}) \Rightarrow (\boldsymbol{b})]$**:** Si G es un árbol, entonces G es conexo.

Sea cualquier arista $e = \{a, b\} \in E$, supongamos que $G - e$ es conexo, entonces habría al menos dos caminos en G desde a hasta b, pero esto contradice el Teorema 4.1.3.1. (Unicidad de camino): Sean $a \neq b \Rightarrow \exists!$ camino $a - b$). Por lo tanto, $G - e$ es disconexo y los vértices de $G - e$ pueden dividirse en dos subconjuntos:

1) El vértice a y aquellos vértices a los que se puede llegar desde a por un camino en $G - e$, es decir: $\{a\} \cup \{x_i : a - x_i \text{ es un camino en } G - e\}$;y
2) El vértice b y aquellos vértices a los que se puede llegar desde b por un camino en $G - e$, es decir: $\{b\} \cup \{y_i : b - y_i \text{ es un camino en } G - e\}$.

Estos dos componentes conexos son árboles pues si existiera un bucle o ciclo en cualquiera de estos también estaría en G, lo cual no sucede.

$[(\boldsymbol{b}) \Rightarrow (\boldsymbol{c})]$**:** G es conexo

A continuación, se realizará la demostración por contradicción

Si G contiene un ciclo y sea $e = \{a, b\}$ una arista de dicho ciclo.

Tendríamos que $G - e$ es conexo, lo que contradice la hipótesis del inciso (b).

Entonces G no contiene ciclos, y dado que G es un grafo conexo sin bucles, luego G es un árbol.

Como G es un árbol, por el Teorema 4.1.2 se deduce que $|V| = |E| + 1$.

$[(c) \Rightarrow (d)]$: G no contiene ciclos y $|V| = |E| + 1$.

Supongamos que G es disconexo, talque $k\ (G) = r$ y $G_1, G_2, \dots, G_r$ las componentes conexas de G.

Para 1$\leq i \leq r$, seleccionamos un vértice $v_i \in G_i$ y agregue las $r - 1$ aristas

$\{v_1, v_2\}, \{v_2, v_3\}, \dots, \{v_{r-1}, v_r\}$ a G para formar el grafo $G' = (V, E')$, que es un árbol.

Como G' es un árbol, por el Teorema 4.1.2 tenemos que $|V| = |E'| + 1$,

además $|E'| = |E| + (r - 1)$.

Pero del inciso (c), $|V| = |E| + 1$, entonces se tiene $|E| = |E'|$ y $r - 1 = 0$. Con $r = 1$, obtenemos que $k\ (G) = 1$ por lo que se deduce que G es conexo.

$[(\boldsymbol{d}) \Rightarrow (\boldsymbol{e})]$**:** G es conexo, no tiene ciclos y $|V| = |E| + 1$.

Como G es conexo para todo a, $b \in V$ existe un camino $a - b$ en G.

Si $\{a, b\} \notin E$ y agregamos a G la arista $\{a, b\}$ junto con el camino mencionado esto creara un ciclo.

$[(\boldsymbol{e}) \Rightarrow (\boldsymbol{a})]$**:** G no contiene ciclos, y si $a, b \in V$ con $\{a, b\} \notin E$, entonces el grafo obtenido añadiendo la arista $\{a, b\}$ a G tiene precisamente un ciclo. Lo unico que nos faltaria probar es que G sea conexo.

Si a, $b \in V$ son dos vertices conectados por una arista $\{a, b\} \notin E$, entonces $G \cup \{a, b\}$ contiene un ciclo y al suprimir la arista $\{a, b\}$ de este ciclo obtenemos un camino en G de entre a y b, por tanto, es conexo. Finalmente G es conexo y no contiene ciclos por ende es un árbol.

(Grimaldi, 2004)

ESCOLIO 4.1.3.1

Se conoce a (b) como grafo minimal y a (e) como grafo maximal sin ciclos

4.1.4. ÁRBOLES DE EXPANSIÓN

Un árbol de expansión T de un grafo conexo G sin bucles, es un subgrafo de G que contiene todos los vértices del grafo original y también es un árbol. En este contexto, se considera que las aristas del grafo están ponderadas positivamente asignados de manera arbitraria. El peso de un árbol de expansión T se define como la suma de los pesos de todas sus aristas, y se denota $w(T)$.

Una vez establecidos estos conceptos base, se introducirán los conceptos de árboles de expansión, destacando su utilidad en la optimización de caminos dentro de grafos conexos. Para finalmente presentar el análisis y la demostración de la validez de los algoritmos de Kruskal y Prim verificando su capacidad para generar soluciones óptimas.

DEFINICIÓN 4.1.4.1. Sean $G = (V, E)$ un grafo conexo no dirigido y T un subgrafo expansivo de G. Se llama a T un árbol de expansión (o generador) de G, si T es un árbol. (Grimaldi, 2004)

DEFINICIÓN 4.1.4.2. Sean $G = (V, E)$ un grafo conexo no dirigido ponderado y T un árbol de expansión de G.

a) Se llama a T un árbol de expansión máximo (o maximal) de G, si la suma total de los pesos de las aristas es el máximo, es decir, es la mayor entre todos los árboles de expansión de G.

b) Se llama a T un árbol de expansión mínimo (o minimal) de G, si la suma total de los pesos de sus aristas es el mínimo, es decir, es la menor entre todos los árboles de expansión de G.

(Tocto Inga, 2012)

DEFINICIÓN 4.1.4.3. Sea $G = (V, E)$ un grafo. Se dice que G es un bosque, si G es disconexo y acíclico, es decir, si cada una de sus componentes conexas es un árbol. (Diestel, 2000)

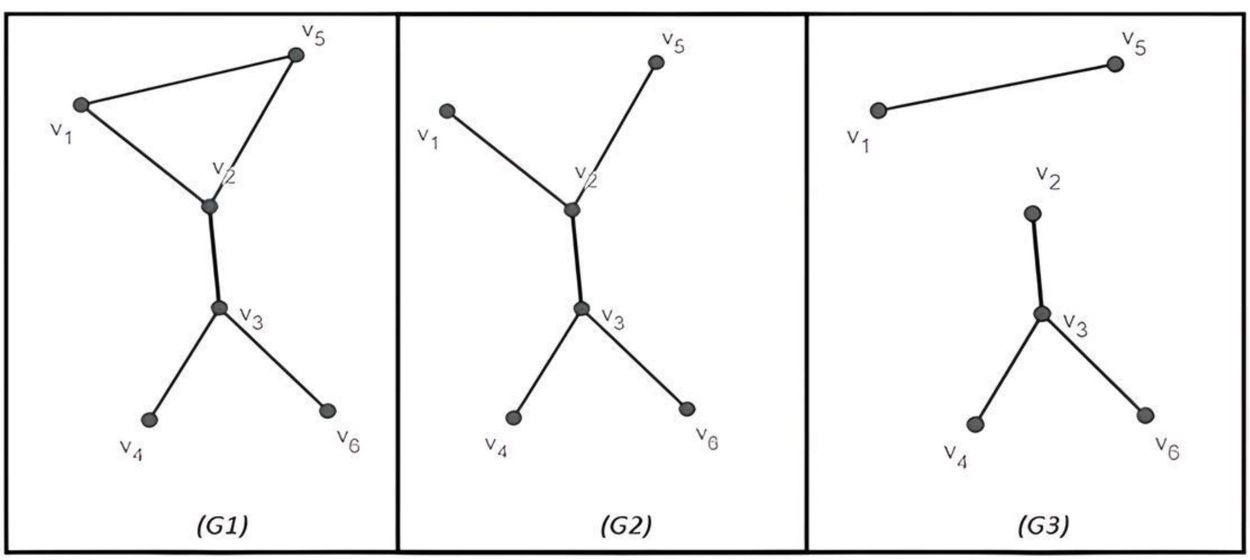
DEFINICIÓN 4.1.4.4. Sea $G = (V, E)$ un grafo disconexo. Se dice que G un bosque de expansión, si G es un subgrafo de expansión y además un bosque. (Grimaldi, 2004)

De la Figura 16 podemos identificar:

1. El grafo G1 no es un árbol pues contiene el ciclo $\{v_1v_5, v_5v_2, v_2v_1\}$.
2. El grafo G2 es un árbol del grafo G1 además G2 es un subgrafo de G1
3. En el grafo G2 los vértices v_1, v_5, v_4, v_6 de grado 1, son hojas del árbol G2.
4. El grafo G3 es no conexo, por lo que no puede ser un árbol. Pero cada componente de G3 es conexo y un árbol, por tanto, llamamos a G3 un bosque.
5. G2 es subgrafo de G1 y contiene todos sus vértices⇒G2 es un subgrafo expansivo de G1.
6. G2 es un árbol y subgrafo expansivo de G1. Por ello G2 es un árbol de expansión para G1.
7. El grafo G3 proporciona un bosque expansivo para el grafo G1.

Figura 16

Árbol y Bosque



Adaptado de *Discrete and combinatorial mathematics* (p. 581), por Ralph P. Grimaldi, 2004, Pearson. En la figura observamos los grafos $G1, G2$ y $G3$

ESCOLIO 4.1.4.1

Un árbol de expansión proporciona la conectividad mínima para un grafo, es decir es un marco esquelético mínimo que une todos los vértices de un grafo. En algunos textos se puede encontrar denotado al árbol de expansión minimal como árbol generador de mínimo coste y al árbol de expansión maximal como árbol generador de máximo coste.

TEOREMA 4.1.4.1. Sea $G = (V, E)$ un grafo, entonces G es conexo si y sólo si G tiene un árbol de expansión de G.

DEMOSTRACION

⇐) Si G tiene un árbol de expansión T, entonces para cada par $a, b \in V$ con $a \neq b$ y por el Teorema 4.1.3.1. (Unicidad de camino) existe un camino único entre a y b, y esto implica que G es conexo.

⇒) Si G es conexo y G no es un árbol.

Eliminemos todos los bucles de G y si el subgrafo resultante G_1 no es un árbol, entonces G_1 debe contener un ciclo C_1.

Retiremos una arista $e_1 \in C_1$ y generamos un subgrafo $G_2 = G_1 - e_1$.

Si G_2 no contiene ciclos, entonces G_2 es un árbol de expansión para G, pues G_2 contiene todos los vértices de G, no tiene bucles y está conexo.

Si G_2 contiene digamos un ciclo C_2, retiramos una arista $e_2 \in C_2$ y generamos el subgrafo $G_3 = G_2 - e_2 = G_1 - \{e_1, e_2\}$. Y una vez más si G_3 no contiene ciclos, diremos que tenemos un árbol de expansión para G. Caso contrario, continuaremos el procedimiento un número finito de veces adicionales hasta llegar a un subgrafo de expansión de G que sea conexo y no contenga ciclos (es decir un árbol de expansión para G). (Grimaldi, 2004)

ESCOLIO 4.1.4.2

Contabilizar el número de árboles generadores o de expansión que existen en un grafo, especialmente en un grafo completo K_n, puede resultar tedioso debido a la gran cantidad de posibles combinaciones de aristas. Afortunadamente, existe una fórmula sutil y elegante que permite calcular dicho propósito, esta fórmula la Formula de Cayley que dice**:** El número de árboles generadores que se pueden formar con n vértices etiquetados es $n^{(n-2)}$. (Cayley, 1897).

4.1.5. ARISTAS DE CORTE

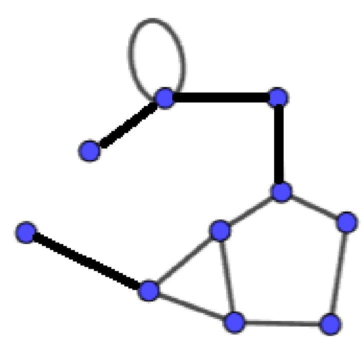
DEFINICIÓN 4.1.5.1. Sean $G = (V, E)$ un grafo y $e \in E$ una arista de G. Se dice que e es una arista de corte de G, si al eliminar e de G el número de componentes conexas aumenta, es decir:

$$k(G - e) > k(G)$$

donde $k(G)$ denota el numero de componentes conexas de G. (Bondy & Murty, 1976)

Figura 17

Aristas de corte de un grafo



En el grafo observamos cuatro aristas de corte en negritas.

TEOREMA 4.1.5.1. Sean $G = (V, E)$ un grafo y $e \in E$ una arista de G. Una arista e es una arista de corte de G si y sólo si e no está contenida en ningún ciclo de G.

DEMOSTRACION

⇒) Realizaremos la demostración por contradicción.

Si e es una arista de corte de G tenemos que $k(G-e) > k(G)$ y existen vértices u y v de G que estén conectados en G pero no en $G-e$.

Es decir, existe algún camino $P = u - v$ en G que necesariamente atraviesa e.

Supongamos que $e = \{x, y\}$ además que x precede a y en P.

En $G - e$:

a) u está conectado a x por una sección de P.

b) y está conectado a v por una sección de P.

Si e estuviera en un ciclo C, x e y estarían conectados en $G-e$ por el camino $C-e$ y por tanto u y v estarían conectados en $G-e$, lo cual es una contradicción.

Entonces e pertenece a ningún ciclo.

⇐) A continuación, se realizará la demostración, por el contrario.

Supongamos que $e = \{x, y\}$ no es una arista de corte de G, entonces $k(G-e) = k(G)$.

Dado que hay un camino $x - y$ en G, x e y están en el mismo componente de G, se deduce que x e y están en el mismo componente de $G - e$.

Por lo tanto, hay un camino $P = x - y$ en $G - e$, pero entonces e está en el ciclo $P + e$ de G.

(Bondy & Murty, 1976)

TEOREMA 4.1.5.2. Sea $G = (V, E)$ un grafo conexo. El grafo G es un árbol si y sólo si cada arista de G es una arista de corte.

DEMOSTRACION:

⇒) Sea G un árbol y sea e una arista de G. Dado que G es acíclico, e no está contenido en ningún ciclo de G y por el teorema anterior, es una arista de corte de G.

$\Leftarrow$) Realizaremos la demostración, por el contrario.

Supongamos que G es conexo, pero no es un árbol. Entonces G contiene un ciclo C, por el anterior Teorema 4.1.5.1 ninguna arista de C puede ser una arista de corte de G. (Bondy & Murty, 1976)

COROLARIO 4.1.5.1. Sea $G = (V, E)$ un grafo conexo. Todo grafo conexo $G = (V, E)$ contiene un árbol de expansión.

DEMOSTRACION:

Sean un grado $G = (V, E)$ conexo, T un subgrafo de expansión conexo de G y e una arista de T. Por definición de conexidad, $k(T) = 1$ y $k(T - e) > 1$ para cada arista e de T.

Se deduce que cada arista de T es una arista de corte y, por el anterior teorema 4.3.2 y al ser G conexo, entonces T es un árbol.

Consecuentemente como T es un subgrafo de expansión de G y un árbol, entonces T es un árbol de expansión del grafo G. (Bondy & Murty, 1976)

TEOREMA 4.1.5.3. Sean un grafo conexo $G = (V, E)$, T un árbol de expansión de G y $e \in E$ una arista de G que no pertenece a T. Entonces el grafo $T + e$ (obtenido al agregar la arista e a T) contiene un único ciclo.

DEMOSTRACION

Sea T un árbol de expansión y por tanto acíclico, dado que T es acíclico cada ciclo de $T + e$ contiene e. Además, C es un ciclo de $T + e$ si y sólo si $C - e$ es un camino en T que conecta los extremos de e.

Según el Teorema 4.1.3.1. (Unicidad de camino) dos vértices de un árbol están conectados por un único camino, entonces T posee un camino único y por lo tanto $T + e$ contiene un ciclo único. (Bondy & Murty, 1976)

4.1.6. GENERACIÓN DE ÁRBOLES DE EXPANSIÓN

La optimización de árboles de expansión consiste en encontrar aquel árbol de expansión que satisfaga criterios específicos del problema que se estudie (maximizar o minimizar).

Si el objetivo es minimizar (el peso total del árbol) entonces se buscará identificar el árbol de expansión mínimo o minimal. Por el contrario, si el objetivo es maximizar se buscará identificar el árbol de expansión máximo o maximal.

A continuación, veremos algoritmos de creación de árboles de expansión los cuales veremos que son similares a los algoritmos de Kruskal y Prim, pero sin ser todavía óptimos.

4.1.6.1. ALGORITMO ÁRBOL GENERADOR O DE EXPANSIÓN

Sea $G = (V, E)$ un grafo, con $|V| = n$ y $|E| = m$. Ordenamos arbitrariamente las aristas de G en una secuencia $(e_1, e_2, \ldots, e_m)$. El algoritmo construye sucesivamente conjuntos de aristas $E_0, E_1, \ldots \subseteq E$, con $E_0 = \emptyset$ donde el conjunto E_i se define como sigue:

$$E_i = \begin{cases} E_{i-1} \cup \{e_i\}, \text{si el grafo } (\mathrm{V}, E_{i-1} \cup \{e_i\}) \text{no tiene ningún ciclo} \\ \quad E_{i-1}, \text{en otro caso} \end{cases}$$

El algoritmo finaliza cuando E_i tiene $n - 1$ aristas o cuando $i = m$. Denotamos por E_t el conjunto de aristas para el cual el algoritmo termina, y el árbol generado será $T = (V, E_t)$. (Matoušek & Nesetril, 2008)

TEOREMA 4.1.6.1 (VALIDEZ DEL ALGORITMO ÁRBOL GENERADOR)

Sea $T = (V, E_t)$ el grafo producido por el algoritmo descrito. Si el algoritmo produce un grafo T con $n-1$ aristas, entonces T es un árbol generador de G. Si T tiene $k < n-1$ aristas, entonces G es un grafo disconexo con $n-k$ componentes.

DEMOSTRACION

Por cómo han sido creados los conjuntos E_i, el grafo T no contiene ningún ciclo.

Si $|E(T)| = n-1$ y no tiene ciclos por el Teorema 4.1.3 (c)

(G no contiene ciclos y $|V| = |E| + 1$) se deduce que T es un árbol.

Además, como $|V(G)| = |V(T)| = n$ entonces T es un árbol generador del grafo G.

Si $|E(T)| = k < n-1 \Rightarrow 1 < n-k$, entonces T es un grafo disconexo por lo que se generara un bosque con $n-k$ componentes.

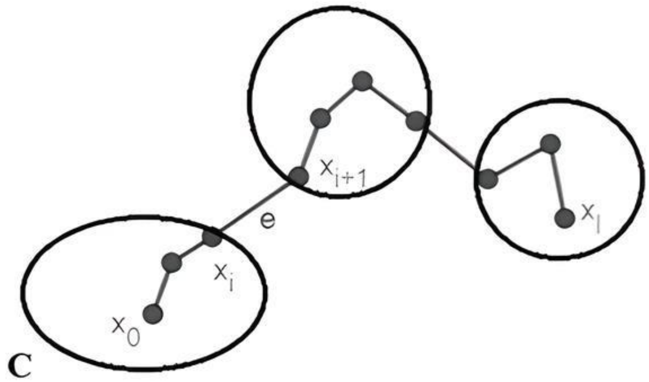
Probaremos que T es un árbol generador de G como T ya es un árbol, solo falta probar que el conjunto de vértices de las componentes del grafo T coinciden con el conjunto de vértices de las componentes del grafo G.

A continuación, se realizará la demostración por contradicción.

Sean x, y dos vértices que están en la misma componente de G, pero en distintas componentes de T. Denotemos por C la componente de T que contiene al vértice x y consideramos un camino $(x = x_0, e_1, x_1, e_2, \ldots, e_l, x_l = y)$ de x a y en G, como en la siguiente figura:

Figura 18

Ilustración de la validez del algoritmo 1



Adaptado de *Invitación a la matemática discreta* (p. 165), por Matoušek; Nesetril, 2008, Reverté.

Sea i el último índice para el que x este contenido en la componente C, con $i < l$ y $x_{i+1} \notin C$. Como la arista $e = \{x_i, x_{i+1}\} \notin T$, entonces podemos deducir que el algoritmo no lo tomo en cuenta porque seguro en algún momento del algoritmo forma un ciclo con algunas aristas ya seleccionas para T. Por lo tanto, el grafo $T + e$ contiene un ciclo, pero esto es imposible pues e conecta dos componentes distintas de T, es decir x e y pertenecen a otra componente de T. Por tanto, llegamos a la contradicción deseada. (Matoušek & Nesetril, 2008)

4.1.6.2. ALGORITMO CREACIÓN DE UN ÁRBOL GENERADOR

Sea $G = (V, E)$ un grafo con $|V| = n$ y $|E| = m$. Construimos sucesivamente subconjuntos de vértices $V_0, V_1, V_2, .., \subseteq V$ y subconjuntos de aristas $E_0, E_1, E_2, .., \subseteq E$ con $E_0 = \phi$ y $V_0 = \{v\}$, donde v es un vértice arbitrario. En cada paso buscamos una arista $e_i = (x_i, y_i) \in E(G)$ tal que:

$$e_i = \begin{cases} x_i \in V_{i-1} \\ y_i \in V - V_{i-1} \end{cases}$$

Y actualizamos los conjuntos:

$$\begin{cases} V_i = V_{i-1} \cup \{y_i\} \\ E_i = E_{i-1} \cup \{e_i\} \end{cases}$$

Y si no existe una arista e_i, el algoritmo termina. Denotamos por E_t el conjunto de aristas en el que el algoritmo se detiene y el árbol construido será $T = (V, E_t)$. (Matoušek & Nesetril, 2008)

TEOREMA 4.1.6.2. (VALIDEZ DEL ALGORITMO CREACIÓN DE ÁRBOL GENERADOR)

Sea $G = (V, E)$ un grafo conexo y T el subgrafo obtenido por el algoritmo descrito. Si el algoritmo termina en un grafo T que contiene n vértices, entonces T es un árbol generador de G.

Caso contrario G es un grafo disconexo y T es un árbol generador de la componente conexa de G que contiene al vértice inicial v.

DEMOSTRACION

El grafo T es un árbol dado que es conexo y al terminar la iteración tiene n vértices y $n - 1$ aristas.

Por el Teorema 4.1.3.3 (d): (G es conexo y $|V| = |E| + 1$ entonces G es un árbol)

Se desprende que T es un árbol, y un subgrafo expansivo pues $V(T) = n = V(G)$.

T es un árbol y un subgrafo expansivo entonces se concluye que es un árbol de expansión de G.

Si T tiene n vértices es un árbol generador, así que supondremos que T tiene $\bar{n} < n$ vértices. Queda por probar que $V(T)$ es el conjunto de vértices de una componente de G. Por contradicción, existen en el grafo G un $x \in V(T)$ y un $y \notin V(T)$ conectados por un camino en G. Como en la demostración del Teorema 4.4.6.1 encontramos en este camino una arista $e = \{x_i, y_i\} \in E(G)$ tal que $x_i \in V_{i-1}$ e $y_i \in V - V_{i-1}$. Así el algoritmo podría haber añadido la arista e y el vértice y_i al árbol, y no habría terminado con el árbol T. Esta contradicción concluye la demostración. (Matoušek & Nesetril, 2008)

4.2 ALGORITMO DE KRUSKAL

Considerando las estructuras de grafos y árboles previamente desarrolladas, ahora presentaremos a continuación técnicas de optimización aplicables tanto a grafos como a multigrafos, ponderados positivos no dirigidos. Entre los métodos más destacados para la optimización de árboles de expansión se encuentra el algoritmo de Kruskal.

El algoritmo de Kruskal construye un árbol de expansión mínimo seleccionando iterativamente aristas de menor peso en orden creciente (en caso de minimización), asegurándose que en cada iteración no se forme ningún ciclo durante la construcción del árbol,

Para este efecto, sea $G = (V, E)$ un grafo conexo, ponderado y sin bucles, donde $|V| = n$, $|E| = m$ y aristas denotadas $e_1, e_2, \ldots, e_m$ tal que $w(e_1) \leq w(e_2) \leq \ldots \leq w(e_m)$.

El algoritmo construye iterativamente una secuencia de subconjuntos de aristas $E_0, E_1, \ldots \subseteq E$, iniciando con $E_0 = \emptyset$. En cada paso i, el conjunto E_i se define como sigue:

$$E_i = \begin{cases} E_{i-1} \cup \{e_i\}, \text{si el grafo } (\mathrm{V}, E_{i-1} \cup \{e_i\}) \text{ no tiene ningún ciclo} \\ E_{i-1}, \text{en otro caso} \end{cases}$$

El algoritmo finaliza cuando E_i tiene $n - 1$ aristas o cuando $i = m$. Denotamos por E_t el conjunto de aristas en el que se detiene el algoritmo. El árbol generador minimal resultante será dado por el subgrafo $T = (V, E_t)$. (Matoušek & Nesetril, 2008)

Si observamos cuidadosamente la ordenación de las aristas se ejecuta según el algoritmo 4.1.6.1 Árbol generador que vimos anteriormente.

4.2.1. TEOREMA DE VALIDEZ DEL ALGORITMO DE KRUSKAL

Sean $G = (V, E)$ un grafo conexo, ponderado y sin bucles y $E^* = \{e_1, e_2, \ldots, e_{n-1}\}$ un subconjunto de aristas de E obtenido mediante la ejecución del algoritmo de Kruskal sobre G. Entonces cualquier árbol de expansión $T^* = (V, E^*)$ construido mediante el algoritmo de Kruskal es un árbol de expansión mínimo.

DEMOSTRACION

Realizaremos la demostración por contradicción, así que supondremos que T^* no es un árbol de expansión mínimo.

Sea un árbol de expansión cualquiera T de G distinto de T^*, talque $w(T) < w(T^*)$ y definimos $f(T)$ el menor índice de i tal que $e_i \notin T$.

Supongamos que T tiene el $f(T)$ máximo posible (entre todos los árboles mejores que T^*, elegimos uno que coincida con el de Kruskal la mayor cantidad de pasos posible)

Sea $f(T) = k$, por su definición se deduce que las aristas $e_1, e_2, \ldots, e_{k-1}$ pertenecen tanto a T y T^*, pero que $e_k \notin T$.

Por el Teorema 4.1.5.3:

(Sean T un árbol de expansión de un grafo conexo $G = (V, E)$ y sea e una arista de G que no está en T, entonces $T + e$ contiene un único ciclo)

$T + e_k$ contiene un único ciclo C.

Dado que $e_k \notin T$, existe una arista e'_k del ciclo C tal que $e'_k \in T$ y $e'_k \notin T^*$ (si no T y T^* serian iguales).

Por el Teorema 4.1.5.1:

(Sean $G = (V, E)$ un grafo y $e \in E$ una arista de G. La arista e es una arista de corte de G si y sólo si e no está contenida en ningún ciclo de G)

Entonces e'_k no sería una arista de corte de $T + e_k$ pues forma parte de un ciclo, luego al eliminar e'_k obtenemos:

$$T' = T + e_k - e'_k$$

un grafo conexo con n vértices y $n - 1$ aristas.

Por el Teorema 4.1.3.3 (d):

(G es conexo y $|V| = |E| + 1 \Leftrightarrow G$ es un arbol)

T' es otro árbol de expansión de G. Consideremos ahora el peso total de T' :

$$w(T') = w(T) + w(e_k) - w(e'_k) \tag{1}$$

Como en el algoritmo de Kruskal e_k fue seleccionada antes que e'_k, significa que:

$$w(e_k) \leq w(e'_k) \tag{2}$$

De (1) y (2) tenemos:

$$w(T') \leq w(T)$$

Así que T' también es un árbol de expansión mínimo (o incluso mejor que T).

Recordemos que $f(T)$ era el primer índice donde T y T^* no coincidían.

En T' se incluyó la arista e_k (que estaba en T^*) y se removió e'_k (que no estaba en T^*), lo que implica:

$$f(T') > k = f(T)$$

Esto contradice la elección de T como el arbol optimo con el valor máximo posible de $f(T)$. Por lo tanto, T no puede tener menor peso que T^* y debe cumplirse que:

$$T = T^*$$

Así, T^* (el árbol generado por el algoritmo Kruskal) da siempre un árbol de expansión mínimo.

(Bondy & Murty, 1976)

4.2.2. RESUMEN DEL ALGORITMO DE KRUSKAL

Sea $G = (V, E)$ un grafo conexo y sin bucles, donde $|V| = n$ y a cada arista $e \in E$ se le asigna un peso $w(e) \in \mathbb{Z}^+$. Para encontrar un árbol de expansión mínimo de G, se siguen los siguientes pasos:

Paso 1: Inicializamos un contador $i = 1$. Seleccionamos una arista $e_1 \in E$ talque $w(e_1)$ sea el peso más pequeño posible del grafo.

Paso 2: Para $1 \leq i \leq n - 2$, si ya se han seleccionado las aristas $e_1, e_2, \dots, e_i$, seleccionar la siguiente arista e_{i+1} de entre las aristas restantes en G de modo que se cumpla:

(a) el peso $w(e_{i+1})$ sea el menor posible y

(b) el subgrafo de G determinado por las aristas $e_1, e_2, \dots, e_i, e_{i+1}$ no forme ciclos.

Paso 3: Incrementar el contador en 1, es decir i por $i + 1$.

a) Si $i = n - 1$, el subgrafo definido por las aristas $e_1, e_2, \dots, e_{n-1}$ será conexo, con n vértices y $n - 1$ aristas, por tanto, un árbol de expansión mínimo para G.

b) Si $i < n - 1$, regresamos al paso (2). (Grimaldi, 2004)

4.3. ALGORITMO DE PRIM

El algoritmo de Prim inicia tomando un vértice aleatorio y en forma secuencial, va incorporando aristas de menor peso que inciden en los vértices ya incluidas con los que aún no están en el árbol parcial que se va construyendo, hasta completar el árbol de expansión mínimo.

En el caso en que el objetivo sea maximizar, ambos algoritmos pueden adaptarse seleccionando las aristas en orden decreciente de peso, construyendo así un árbol de expansión máximo.

Sea $G = (V, E)$ un grafo con $|V| = n$ y $|E| = m$. El algoritmo de Prim construye un árbol de expansión mino de manera recursiva, generando sucesivamente los subconjuntos $V_0, V_1, V_2, .., \subseteq V$ y $E_0, E_1, E_2, .., \subseteq E$. El proceso se inicia con $V_0 = \{v\}$, donde $v \in V$ es un vértice arbitrario y $E_0 = \phi$.

En cada paso, se elige la arista de menor peso $e_i = (x_i, y_i) \in E(G)$ tal que:

$$e_i = \begin{cases} x_i \in V_{i-1} \\ y_i \in V - V_{i-1} \end{cases}$$

Y se actualizan los subconjuntos:

$$\begin{cases} V_i = V_{i-1} \cup \{y_i\} \\ E_i = E_{i-1} \cup \{e_i\} \end{cases}$$

El proceso continua mientras exista una arista que conecte el conjunto actual V_{i-1} con el resto de los vértices. Si no existe tal arista, el algoritmo termina.

Al finalizar, denotamos por E_t el conjunto de aristas en el que el algoritmo se detiene, y se obtiene el grafo $T = (V, E_t)$ el cual es un árbol generador minimal. (Matoušek & Nesetril, 2008)

La selección y ordenación de las aristas del algoritmo Prim se realiza de acuerdo con el procedimiento descrito en el algoritmo 4.1.6.2 Creación de un árbol generador.

4.3.1. TEOREMA DE VALIDEZ DEL ALGORITMO DE PRIM

Sean $G = (V, E)$ un grafo conexo, ponderado y sin bucles, entonces el algoritmo de Prim construye un árbol de expansión mínimo para G.

DEMOSTRACION

Sean $T = (V, E_t)$ el árbol de expansión construido por el algoritmo de Prim, y las aristas de E_t numeradas $e_1, e_2, \ldots, e_{n-1}$ de acuerdo con en el orden en que fueron añadidas por el algoritmo de Prim a T.

Realizaremos la demostración por contradicción, así que supondremos que T no es un árbol de expansión mínimo.

Sea T' algún otro árbol de expansión mínimo. Comparemos T y T'.

Definimos $k(T')$ como el mayor índice k tal que las primeras k aristas de T, seleccionadas por Prim $e_1, e_2, \ldots, e_k \in E(T')$ pero $e_{k+1} \notin E(T')$ (esto nos permite medir cuanto coinciden los árboles T y T' antes de empezar a diferir).

De entre todos los árboles de expansión mínimos que cumplen esta condición, seleccionamos aquel con el mayor valor posible de k (es decir el árbol mínimo que coincide más con Prim). Denotemos este árbol por $T^* = (V, E^*)$ y sea $k = k(T^*)$.

Entonces el grafo $T^* + e_{k+1}$ cómo es conexo y ahora posee más de $n - 1$ aristas, entonces contiene un ciclo C. El ciclo necesariamente contiene la arista e_{k+1} ya que $e_{k+1} \notin E^*$ (no estaba en T^*).

Supongamos que $e_{k+1} = \{x, y\}$, con $x \in V_k$ e $y \notin V_k$. Esto ya que Prim elige aristas que conectan el conjunto construido vértices aun no incluidos.

Consideremos el subgrafo parcial de Prim hasta el paso k:

$$T_k = (V_k, E_k) \text{ con } E_k = \{e_1, e_2, \dots, e_k\}$$

Como T^* es un árbol, hay un único camino P entre x y y. Este camino $P = x - y$ junto con la arista e_{k+1} forma el ciclo C dentro del árbol T^*.

En ese camino debe haber al menos una arista $e \in E^*$ que conecta un vértice en V_k con otro que no pertenece a V_k.

Sea esa arista e, obviamente $e \neq e_{k+1}$, y sabemos que $e \in E^*$ y $e_{k+1} \notin E^*$.

Ambas aristas e y e_{k+1} conectan un vértice de V_k con un vértice que no está en V_k.

El algoritmo de Prim selecciona la arista de menor peso posibles en cada paso, entonces:

$$w(e_{k+1}) \leq w(e)$$

Ahora si construimos un nuevo árbol:

$$T' = T^* + e_{k+1} - e$$

Obtenemos un grafo tiene n vértices, $n - 1$ aristas y es conexo

Por el Teorema 4.1.3.3 (d):

(G es conexo y $|V| = |E| + 1 \Leftrightarrow G$ es un arbol)

Tenemos que T' es un árbol de expansión. Veamos el peso total de T' :

$$w(T') = w(T^*) + w(e_{k+1}) - w(e) \leq w(T^*)$$

Esto quiere decir que T'' es también un árbol de expansión mínimo, sin embargo, notamos que T'' coincide con el árbol construido por el algoritmo Prim en más pasos que T^*, es decir:

$$k(T'') > k(T^*)$$

Esto contradice la elección de T^* como el árbol mínimo que más coincidía con Prim.

Esta contradicción prueba que la suposición inicial, que T no era un árbol de expansión mínimo, es falsa. Por tanto, el árbol $T = (V, E_t)$ generado por el algoritmo de Prim es un árbol de expansión mínimo. (Matoušek & Nesetril, 2008)

4.3.2. RESUMEN DEL ALGORITMO DE PRIM

Sea G = (V, E) un grafo conexo, ponderado y sin bucles. Para obtener un árbol óptimo T para G, aplicaremos el siguiente procedimiento:

Paso 1: Inicializamos un contador $i = 1$.

Elegimos un vértice arbitrario $v_1 \in V$ y lo incluimos en el conjunto P. Definiremos el conjunto complementario $N = V - \{v_1\}$ y el conjunto de aristas del árbol $T = \emptyset$,

Paso 2: Para cada i tal que $1 \leq i \leq n - 1$, donde $|V| = n$, se tienen:

$P = (v_1, v_2, \ldots, v_i)$,

$T = (e_1, e_2, \ldots, e_{i-1})$ y

$N = V - P$.

Agregamos a T la arista más corta (de peso mínimo) en G que conecte un vértice $x \in P$ con un vértice $y \in N$. Incluimos el vértice y en P y lo eliminamos de N.

Paso 3: Incrementar el contador i en 1:

Si $i = n$, el subgrafo T de G definido por las aristas $\{e_1, e_2, \ldots, e_{n-1}\}$ es un grafo conexo con n vértices y $n - 1$ aristas y por ende un árbol óptimo para G.

Si $i < n$, regresamos al paso (2).

En caso que el objetivo del problema sea maximizar, el algoritmo puede adaptarse seleccionando en cada paso la arista de mayor peso posible, en orden decreciente de peso, construyendo así un árbol de expansión máximo.

ESCOLIO 4.3.2.1.

Es importante tener en cuenta que el árbol de expansión mínimo obtenido no es único (es decir, puede diferir de otro resultado equivalente), pues pueden existir múltiples árboless diferentes que cumplan con la condición de tener peso total mínimo.

Por otro lado, si se tiene un grafo $G = (V, E)$, con $|V| = n$ y $|E| = a$ aristas, es relvante considerar la eficiencia de los algoritmos de construccion. Segun Tocto Inga (2012), quien analizo la complejidad y comparacion de los algoritmos Kruskal y Prim, se sabe que:

a) Si $a \approx n$, es decir si el grafo es escasamente denso de aristas, entonces nos resultara mas conveniente utilizar el algoritmo de Kruskal.

b) Caso contrario, si $a \approx n^2$, lo que indica un grafo denso de aristas, se es mas conveniente aplicar el algoritmo de Prim.

DISCUSIÓN

Los antecedentes revisados evidencian que muchas investigaciones abordan la teoría de grafos desde un enfoque principalmente aplicativo y computacional. Por ejemplo, Condori (2016), Velásquez & Reyes (2023) y Chirme (2023) utilizaron grafos para representar redes de comunicación y sistemas de carreteras, empleando matrices, algoritmos computacionales y software especializado con el propósito de optimizar rutas. En una línea similar, Piedra & Paternostro (2009) desarrollaron aplicaciones de la teoría de grafos en el ámbito de la informática, destacando su utilidad en las ciencias de la computación. Sin embargo, estos estudios no profundizan en la formulación teórica ni en el desarrollo axiomático de conceptos fundamentales de la teoría de grafos como los caminos, la conexidad o los árboles de expansión. En contraste, el presente trabajo se enfoca en una exposición rigurosa de estas definiciones matemáticas en el contexto de grafos no dirigidos.

Tocto Inga (2012) realizó una comparación general entre los algoritmos de Kruskal y Prim, centrada en sus complejidades computacionales acompañada de definiciones básicas de algoritmos. No obstante, su enfoque no contempla una validación teórica formal del funcionamiento de los algoritmos de Kruskal y Prim desde una perspectiva matemática. El análisis de mi investigación se fundamenta en propiedades esenciales como caminos, conexidad, la generación de árboles y árboless de expansión, aspectos que comúnmente se presentan de manera implícita en estudios de carácter aplicado.

Por otra parte, Nogueira (2017) diseñó sesiones educativas sobre grafos, implementando los algoritmos de Prim, Kruskal y otros en lenguaje C con fines didácticos orientados al aprendizaje de programación en estudiantes de nivel secundario. En contraste, el presente trabajo demuestra la validez teórica de los algoritmos de Kruskal y Prim mediante el análisis formal de su comportamiento en grafos ponderados, considerando las propiedades de corte y la selección de aristas de peso mínimo.

En conjunto, nuestra investigación complementa el panorama de estudios revisados, los cuales priorizan la aplicación práctica y computacional, al enfocarnos en la fundamentación teórica fortalecemos las bases matemáticas que sustentan dichas aplicaciones, ofreciendo un aporte significativo para investigaciones futuras tanto teóricas como aplicadas.

CONCLUSIONES

a) Se establecieron rigurosamente los conceptos fundamentales de la teoría de caminos, conexidad y árboles de expansión.

b) Se demostró la validez del algoritmo de Kruskal para la obtención de árboles de expansión óptimos.

c) Se verifico la validez del algoritmo de Prim para la obtención de árboles de expansión óptimos.

RECOMENDACIONES

a) Demostrar la validez de otros algoritmos de árboles de expansión, como el algoritmo de Boruvka o el de Floyd Warshall, y compararlos con Kruskal y Prim en términos de complejidad y aplicabilidad.

b) Ampliar el estudio hacia otros conceptos fundamentales de la teoría de grafos, como la planaridad, coloración, recorridos de Euler, ciclos de Hamilton o árboles con raíz, para enriquecer el marco teórico a estructuras más generales.

c) Analizar la dualidad entre árboles de expansión mínima y árboles de expansión máxima, estableciendo condiciones teóricas bajo las cuales sus propiedades son análogas. Generalizar resultados a retículos de árboles de expansión.

d) Investigar cómo los árboles de expansión mínima se relacionan con invariantes topológicos (e.g., homología, grupos fundamentales) en grafos interpretados como complejos simpliciales.

e) Se recomienda aplicar los algoritmos de Kruskal y Prim en la resolución de problemas reales vinculados a necesidades sociales, como optimización de rutas de transporte, distribución de servicios básicos o diseño de redes de comunicación, con el fin de evidenciar su utilidad práctica en contextos reales.

BIBLIOGRAFÍA

Arias Rafael, F. (2017). *Aplicación de la teoría de grafos en el diseño de rutas de transporte desde las zonas de producción agrícola hasta la planta de procesamiento.* Pontificia Universidad Católica Del Perú, (Tesis de Maestria).

Biggs, N., Lloyd, E., & Wilson, R. (1986). *Graph Theory.* Oxford University Press.

Bondy, J., & Murty, U. (1976). *Graph theory with applications.* North Holland.Oxford.

Bondy, J., & Murty, U. (2008). *Graph theory.* Springer .

Cayley, A. (1897). *The Collected Mathematical Pappers of Arthur Cayley* (Vol. XIII). Cambridge.

Chirme, H. (2023). Descripción por teoria de redes y simulación de rutas de transporte en el casco urbano de la ciudad del Cusco, 2023. *(Tesis de Maestria).* Universidad Nacional de San Antonio Abad del Cusco, Cusco.

Condori Balcón, D. R. (2016). *Organización espacial de la red de carreteras de la Región Puno mediante la teoría de grafos.* Universidad Nacional Del Altiplano, (Tesis de Licenciatura).

Diestel, R. (2000). *Graph theory* (2 ed.). Springer-Verlag.

Grimaldi, R. (2004). *Discrete and combinatorial mathematics* (5 ed.). Pearson.

Jungnickel, D. (2008). *Graphs, networks and algorithms* (3 ed., Vol. 5). Berlin: Springer.

Koh, M., Dong, M., & Tay, G. (2007). *Introduction to Graph Theory.* World Scientific.

Matias, L. (2018). *Evaluación de metodologías de solución al problema de redes de transporte.* Universidad Nacional Santiago Antunez de Mayolo, (Tesis de Licenciatura).

Matoušek, J., & Nesetril, J. (2008). *Invitación a la matemática discreta.* (Anna Lladó). Reverté. (Trabajo original publicado en 1998).

Nogueira Junior, D. (2017). *Grafos e problemas de caminhos.* Universidade Federal de Vicosa, (Tesis de Maestria).

Ñaupas, H., Mejía, Novoa, & Villagómez. (2014). *Metodología de la investigación cuantitativa-cualitativa y redacción de la tesis* (4ta ed.). Bogotá: Ediciones de la U.

Ñaupas, Valdivia, Palacios, & Romero. (2018). *Metodología de la investigación cuantitativa-cualitativa y redacción de la tesis* (5ta ed.). Ediciones de la U.

Onofre Caza, R. R. (2024). *Modelo de redireccionamiento vial mediante algoritmo de colonia de hormigas y teoría de grafos en la transitabilidad vial de ex-tranca Rio Seco 2022.* Universidad Pública De El Alto, (Tesis de Licenciatura).

Páez Páez, Y. C. (2023). *Representación de estructuras algebraicas basadas en la Teoría de Grafos.* Universidad Nacional José Faustino Sánchez Carrión, (Tesis de Licenciatura).

Patiño Avendaño, B., & Guillermo Charry, O. (2013). *La enseñanza de la teoría de grafos como estrategia para desarrollar procesos de matematización.* Universidad Sergio Arboleda - Colombia, (Tesis de Maestría).

Piedra Hernandez, V. S., & Paternostro Movilla, C. A. (2009). *Aplicaciones de la teoria de grafos en la informática.* Pontificia Universidad Javeriana - Colombia, (Tesis de Licenciatura).

Ruohonen, K. (2006). *GRAPH THEORY.* Lecture Notes.

Tocto Inga, P. (2012). *Comparación de los algorítmos Prim y Kruskal.* Universidad Nacional de Ingeniería, (Tesis de Licenciatura).

Velasquez, P., & Reyes, R. (2023). *Teoría de grafos y sus aplicaciones en modelos de rutas óptimas.* Universidad Nacional Autónoma De Nicaragua, León, (Tesis de Licenciatura).

ANEXO

A) MATRIZ DE CONSISTENCIA

<table>
<tr><td colspan="3">MATRIZ DE CONSISTENCIA</td></tr>
<tr><td colspan="3">TEMA: ÁRBOLES DE EXPANSIÓN EN LA OPTIMIZACIÓN DE CAMINOS</td></tr>
<tr><td>PROBLEMAS</td><td>OBJETIVOS</td><td>METODOLOGIA</td></tr>
<tr><td>PROBLEMA GENERAL</td><td>OBJETIVO GENERAL</td><td rowspan="7">TIPO DE INVESTIGACION
Investigación básica

NIVEL DE INVESTIGACION
Exploratorio descriptivo

METODO DE INVESTIGACION
Documental (Teórico)

TÉCNICA DE INVESTIGACION
Técnica Conceptual</td></tr>
<tr><td>¿Sera posible optimizar caminos utilizando árboles de expansión?</td><td>Optimizar caminos utilizando árboles de expansión</td></tr>
<tr><td>PROBLEMAS ESPECIFICOS</td><td>OBJETIVOS ESPECIFICOS</td></tr>
<tr><td>¿De qué manera se pueden establecer los conceptos fundamentales de la teoría de caminos, conexidad y árboles de expansión?</td><td>Establecer los conceptos fundamentales de la teoría de caminos, conexidad y árboles de expansión.</td></tr>
<tr><td>¿Es posible demostrar la validez del algoritmo de Kruskal para obtener árboles de expansión óptimos?</td><td>Demostrar la validez del algoritmo de Kruskal para obtener árboles de expansión óptimos.</td></tr>
<tr><td>¿Se puede verificar la validez del algoritmo de Prim para obtener árboles de expansión óptimos?</td><td>Verificar la validez del algoritmo de Prim para obtener árboles de expansión óptimos.</td></tr>
</table>