

Centro de enseñanza técnica industrial

CETI

Tema: Árbol de Máximo y Mínimo coste Kruskal.



Alumno: Isaac Carrillo Angulo

Reg. 21110358 grupo.6E1

Prof. Mauricio Alejandro Cabrera Arellano

Fecha: 24 de octubre de 2023

Parte Teórica

¿Qué es?

El algoritmo de Kruskal es un algoritmo de la teoría de grafos para encontrar un árbol recubridor mínimo en un grafo conexo y ponderado. Es decir, busca un subconjunto de aristas que, formando un árbol, incluyen todos los vértices y donde el valor de la suma de todas las aristas del árbol es el mínimo. Si el grafo no es conexo, entonces busca un bosque expandido mínimo (un árbol expandido mínimo para cada componente conexa). Este algoritmo toma su nombre de Joseph Kruskal, quien lo publicó por primera vez en 1956. Otros algoritmos que sirven para hallar el árbol de expansión mínima o árbol recubridor mínimo son el algoritmo de Prim, el algoritmo del borrador inverso y el algoritmo de Boruvka.

El algoritmo de Kruskal es un ejemplo de algoritmo voraz que funciona de la siguiente manera:

- se crea un bosque B (un conjunto de árboles), donde cada vértice del grafo es un árbol separado
- se crea un conjunto C que contenga a todas las aristas del grafo
- mientras C es no vacío
- eliminar una arista de peso mínimo de C
- si esa arista conecta dos árboles diferentes se añade al bosque, combinando los dos árboles en un solo árbol
- en caso contrario, se desecha la arista
- Al acabar el algoritmo, el bosque tiene un solo componente, el cual forma un árbol de expansión mínimo del grafo.
- En un árbol de expansión mínimo se cumple:
- la cantidad de aristas del árbol es la cantidad de nodos menos uno

¿Para qué sirve?

El objetivo del algoritmo de Kruskal es construir un árbol (subgrafo sin ciclos) formado por arcos sucesivamente seleccionados de mínimo peso a partir de un grafo con pesos en los arcos.

Un árbol (spanning tree) de un grafo es un subgrafo que contiene todos sus vértices o nodos. Un grafo puede tener múltiples árboles. Por ejemplo, un grafo completo de cuatro nodos (todos relacionados con todos) tendría 16 árboles.

La aplicación típica de este problema es el diseño de redes telefónicas. Una empresa con diferentes oficinas, trata de trazar líneas de teléfono para conectarlas unas con otras. La compañía telefónica le ofrece esta interconexión, pero ofrece tarifas diferentes o costes por conectar cada par de oficinas. Cómo conectar entonces las oficinas al mínimo coste total.

La formulación del MST también ha sido aplicada para hallar soluciones en diversas áreas (diseño de redes de transporte, diseño de redes de telecomunicaciones - TV por cable, sistemas distribuidos, interpretación de datos climatológicos, visión artificial - análisis de imágenes - extracción de rasgos de parentesco, análisis de clusters y búsqueda de superestructuras de quasar, plegamiento de proteínas, reconocimiento de células cancerosas, y otros).

¿Cómo se implementa en el mundo?

Otra aplicación menos obvia es que el árbol de coste total mínimo puede ser usado como solución aproximada al problema del viajante de comercio (traveling salesman problem), recuerde que encontrar la solución óptima a este problema es NP-Hard. La manera formal de definir este problema es encontrar la trayectoria más corta para visitar cada punto al menos una vez. Nótese que si se visitan todos los puntos exactamente una vez, lo que se tiene es un tipo especial de árbol. En el ejemplo anterior, 12 de los 16 árboles son trayectorias de este tipo. Si se tiene una trayectoria que visita algunos vértices o nodos más de una vez, siempre se puede soltar algunos nodos del árbol. En general el peso del árbol total mínimo es menor que el del viajante de comercio, debido a que su minimización se realiza sobre un conjunto estrictamente mayor. Existen diferentes algoritmos y maneras de usar el árbol de coste total mínimo para encontrar la solución al problema del viajante de comercio (con resultados cercanos al óptimo).

¿Cómo lo implementarías en tu vida?

A pesar de que estos algoritmos son conceptos técnicos, su lógica puede aplicarse en situaciones de la vida diaria para tomar decisiones eficientes y efectivas:

Planificación de Proyectos:

Árbol de Expansión Mínima (MST): Puedes aplicar el algoritmo de Kruskal para determinar las tareas críticas y minimizar el tiempo total de finalización del proyecto.

Árbol de Expansión Máxima: En proyectos de expansión o crecimiento

¿Cómo lo implementarías en tu trabajo o tu trabajo de ensueño?

Redes Eléctricas:

En la planificación de redes eléctricas, Kruskal puede ser utilizado para encontrar la conexión más eficiente de estaciones de energía, subestaciones y usuarios, minimizando pérdidas de energía y costos de construcción

Redes de Computadoras:

En la planificación de redes de computadoras, el algoritmo de Kruskal puede utilizarse para encontrar la infraestructura de red más eficiente y económica para interconectar computadoras o dispositivos.

Redes de Telecomunicaciones:

Para la expansión de redes de telecomunicaciones, Kruskal puede ayudar a minimizar los costos de cableado al encontrar la disposición más eficiente de torres de señal o estaciones de repetición