RECREANDO RUTAS DE REPARTO CON MST'S

USANDO EL ALGORITMO KRUSKAL

DIANA LAURA REYES YOUSHIMATZ
ID: 173391

Planteamiento

Un empresa que abastece de material de cómputo y papelería a instituciones, empresas y franquicias debe llenar pedidos globales que se realizan una vez por mes. Sin embargo, la planificación de rutas resulta una labor extensa que demanda recursos humanos y de tiempo. Este problema a pesar de presentarse en varias empresas, la mayoría de ellas no llevan a cabo ningún procedimiento estandarizado de rutas óptimas, para atender este problema.

Se presenta el problema con un grafo (figura 1), donde los nodos son franquicias y las aristas ponderadas el tiempo que toma recorrer las rutas entre franquicias. La empresa ha proporcionado las distancias en tiempo entre franquicias y rutas posibles, la cual estima en promedio 180 minutos únicamente en tiempo de transporte.

Propuesta de Solución

A partir del grafo construido, se elabora un árbol de expansión mínima que permita realizar todas las entregas en el menor tiempo posible usando del algoritmo de Kruskal, en el cual la prioridad es el tiempo.

En este caso establecemos la empresa proveedora como el primer nodo, dado que es el lugar donde siempre empiezan las rutas de repartición. Podemos observar que las sucursales que están en los extremos de la ciudad cuentan con uno o dos vértices, mientras que las demás sucursales de la ciudad cuentan con varios nodos adyacentes.

Grafo de posibles rutas

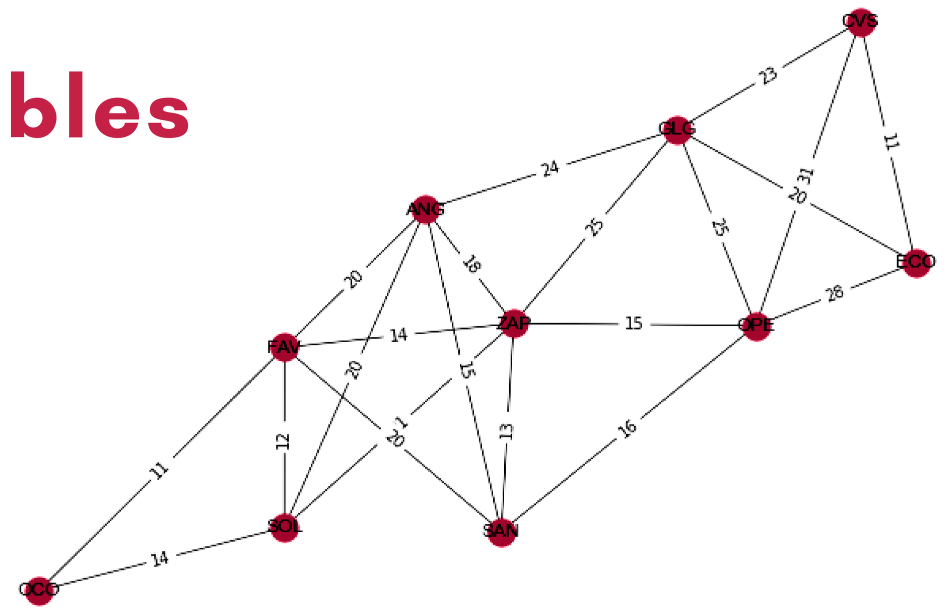

figura 1

MST's y Algoritmo Kruskal

El árbol de expansión con peso mínimo (MST en inglés) es el subconjunto de aristas de un grafo que conecta todos los vértices con el mínimo costo total. El algoritmo de Kruskal es un algoritmo que encuentra el árbol de expansión mínima de un grafo no dirigido y conexo.

El algoritmo comienza ordenando todas las aristas del grafo por peso y luego se agrega al MST sucesivamente la arista de menor peso que no forma un ciclo con las aristas ya seleccionadas en el árbol, donde se utiliza una estructura de datos de conjuntos disjuntos y la técnica de "unión por rango" para unir conjuntos y encontrar si dos vértices pertenecen al mismo conjunto

Correctitud

En cada iteración del ciclo el algoritmo agrega una arista a la solución y une dos conjuntos en uno solo. Se asegura de que la arista seleccionada no forme un ciclo. Después de n-1 iteraciones, donde n es el número de nodos del grafo, el algoritmo habrá construido un árbol de expansión de peso mínimo que abarca todos los nodos.

Complejidad

- En la primera parte del algoritmo, se ordenan todas las aristas en orden creciente, lo que toma O(E log E) tiempo
- Después se realizan exactamente E operaciones de búsqueda y unión que toman O(log V) tiempo
- En conjunto, la complejidad de Kruskal es O(E log E + E log V), que se simplifica a O(E log E).

Algoritmo de Kruskal

KRUSKAL(G):

1. Inicializar un conjunto vacío F que representará el árbol de expansión mínima (MST).
2. Ordenar todas las aristas del grafo G por peso, en orden ascendente.
3. Para cada arista (u,v) en el conjunto de aristas ordenadas:
   a. Si al agregar la arista (u,v) a F no crea un ciclo, se agrega a F.
   b. De lo contrario, descartarla.
4. Devolver F.

Resultados

Para este problema se decide utilizar el algoritmo de Kruskal, dado que sea conocer un subconjunto del grafo que conecte todos los nodos con un costo mínimo. A diferencia de otros algortimos, como Dijkstra donde se encuentra el camino mínimo entre dos nodos específicos, aquí se desea construir una red que conecte todas las franquicias por lo que la construcción de un MST resulta adecuada para este problema.

En la figura 2 se encuentra el MST obtenido usando el algoritmo de Kruskal. El cual tiene un peso mínimo de 135 min recorriendo todas las franquicias. Podemos observar que la empresa actualmente destina 180 min en promedio para recorrerlas, por lo que el MST representa una propuesta conveniente para la construcción de rutas de reparto con un costo mucho menor.

Árbol de expansión con costo mínimo

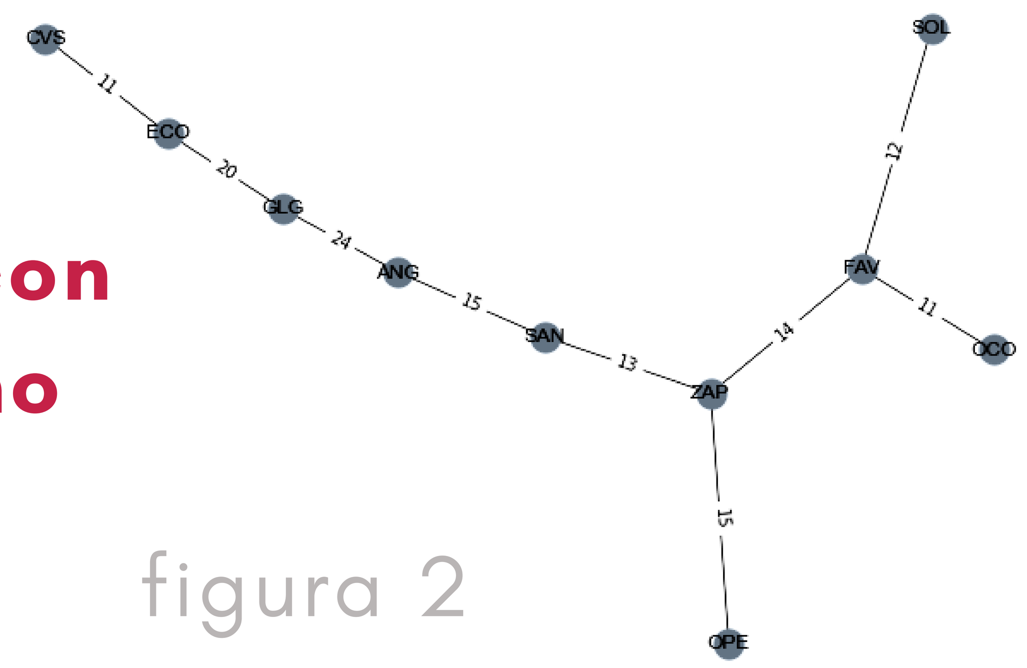

figura 2