DESARROLLO DE UN ALGORITMO EFICIENTE PARA RUTEO PARA VEHÍCULOS ELÉCTRICOS

Daniel Alejandro Mesa Arango
Universidad Eafit
Colombia
damesaa@eafit.edu.co

Kevin Arley Parra Henao
Universidad Eafit
Colombia
kaparrah@eafit.edu.co

Mauricio Toro
Universidad Eafit
Colombia
mtorobe@eafit.edu.co

RESUMEN

Diseñar un algoritmo para encontrar las rutas óptimas para que un conjunto de camiones eléctricos visite un conjunto de clientes. [1]

A corto y largo plazo, está tecnología promete cambiar la forma en qué nos transportarnos, sin embargo, su consumo energético y la necesidad de recargar las baterías empleadas por vehículos eléctricos, hace necesario considerar las rutas que se toman.

Algunos problemas relacionados son: El problema del camino más corto, grafos bipartitos, ciclos impares, coloración, componentes fuertemente conexas y árbol de expansión mínima.

1. INTRODUCCIÓN

El problema a resolver es un Problema de Ruteo de Vehículos (VRP, por sus siglas en inglés, Vehicle Routing Problem) con la modificación de que los vehículos a considerar son eléctricos, por lo que el tiempo de carga y el consumo energético serán factores importantes a tener en cuenta.

El primer problema planteado tipo VPR fue el del agente viajero o TSP (Travelling Salesman Problem) introducido por Flood en 1956. [2]

2. PROBLEMA

La pregunta a resolver es:

Dado una lista de clientes ubicados en un mapa vial bidimensional, un depósito de inicio y fin, y unas restricciones de tiempo y energía ¿cuáles son las rutas para un número libre de camiones eléctricos, para visitar todos los clientes, minimizando el tiempo total, que es la suma del tiempo del recorrido más el tiempo que toman las recargas de batería? [1]

3. TRABAJOS RELACIONADOS

3.1 GRAFOS BIPARTITOS, CICLOS IMPARES Y COLORACIÓN [3]

El problema de verificar si un grafo es bipartito es equivalente al problema de colorabilidad de grafos para dos colores, simplemente son el mismo problema pero con diferente nomenclatura. Con respecto al problema de detección de la existencia de ciclos impares, cualquier grafo con un ciclo de longitud impar es claramente no colorable para dos colores. En vista de que los tres problemas son equivalentes es posible elaborar un solo algoritmo que resuelva los tres problemas.

Para colorear un grafo con dos colores, dado un color asignado a un vértice v, debe colorearse el resto del grafo asignando el otro color a cada vértice adyacente a v. Este proceso es equivalente a alternar los dos colores para los nodos en cada nivel de un árbol DFS, y durante el retorno de las llamadas recursivas verificar si no hay arista que conecte dos nodos del mismo color, ya que tal arista es evidencia de un ciclo de longitud impar.

A continuación ilustraremos esta idea, usando el árbol DFS de la sección de recorrido de grafos. La primera figura muestra el grafo original, la segunda el árbol DFS y la tercera es el grafo coloreado a blanco y gris, usando la heurística descrita anteriormente.

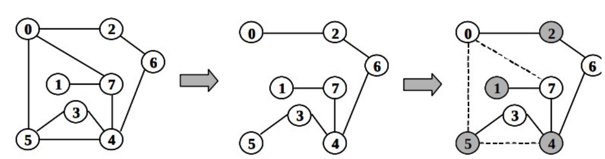

Figura 1

Una vez construido el árbol DFS y coloreado los nodos, es sencillo comprobar que las aristas 5-4 y 0-7 conectan dos nodos del mismo color, lo cual es evidencia de la existencia de un ciclo de longitud impar. De hecho, el ciclo 0-5-3-4-7-0 es un ciclo de longitud impar. Si eliminamos 5-4 y 0-7 el grafo sería colorable y no habría ciclos de longitud impar.

3.2 COMPONENTES FUERTEMENTE CONEXAS

Una componente fuertemente conexa de un dígrafo es un conjunto máximo de nodos en el cual existe un camino que va desde cualquier nodo del conjunto hasta cualquier otro nodo del conjunto. Si el dígrafo conforma una sola componente fuertemente conexa, se dice que el dígrafo es fuertemente conexo. [3]

Una solución para esto seria aplicar el método Kosaraju, que a su vez usa DFS para recorrer el grafo y hallar la solución.

3.3 ÁRBOL DE EXPANSIÓN MÍNIMA [3]

Encontrar la manera menos costosa de conectar todos los puntos, entonces nos enfrentamos al problema de encontrar un árbol de expansión mínima (MST – Minimum Spanning Tree).

Un árbol de expansión de un grafo conexo es un subgrafo que contiene todos los nodos del grafo y no tiene ciclos. El árbol de expansión mínima de un grafo pesado no dirigido es el árbol de expansión cuyo peso (la suma de los pesos de todas sus aristas) no es mayor al de ningún otro árbol de expansión.

El algoritmo de Prim es tal vez el algoritmo de MST más sencillo de implementar y el mejor método para grafos densos. Este algoritmo puede encontrar el MST de cualquier grafo conexo pesado.

3.4 EL CAMINO MAS CORTO

En la Teoría de Grafos, uno de los problemas más conocido es el del camino más corto. El problema consiste en encontrar un camino entre dos vértices (o nodos) de tal manera que la suma de los pesos de las aristas que lo constituyen es mínima. [4]

Una solución sería:

El algoritmo de dijkstra determina la ruta más corta desde un nodo origen hacia los demás nodos para ello es requerido como entrada un grafo cuyas aristas posean pesos.

REFERENCIAS

[1] Restrepo, A., & Toro, M. 2018. Algoritmo para ruteo de vehículos eléctricos (2nd ed.). https://github.com/mauriciotoro/ST0247-Eafit/tree/master/proyecto, March 4, 2018, Medellín.

[2] ROCHA MEDINA, Linda Bibiana; GONZÁLEZ LA ROTTA, Elsa Cristina; ORJUELA CASTRO, Javier Arturo. Una Revisión al Estado del Arte del Problema de Ruteo de Vehículos: Evolución Histórica Y Métodos De Solución. **Ingeniería**, [S.l.], v. 16, n. 2, p. 35-55, dec. 2011. ISSN 2344-8393. Disponible en: <https://revistas.udistrital.edu.co/ojs/index.php/reving/article/view/3832>. Date accessed: 03 mar. 2018. doi:https://doi.org/10.14483/23448393.3832.

[3] Coto, E. 2003. Algoritmos Básicos de Grafos. http://ccg.ciens.ucv.ve/~ernesto/nds/CotoND200302.pdf, March 4, 2018, Caracas.

[4] 3.1. El camino más corto. n.d. E-ducativa.catedu.es. http://e-ducativa.catedu.es/44700165/aula/archivos/repositorio//4500/4724/html/31_el_camino_ms_corto.html, March 4, 2018.

[5] Arias Figueroa, J. 2013. CAMINO MAS CORTO: ALGORITMO DE DIJKSTRA. Algorithms and More. https://jariasf.wordpress.com/2012/03/19/camino-mas-corto-algoritmo-de-dijkstra/, March 4, 2018.

Figura 1: Coto, E. 2003. Algoritmos Básicos de Grafos. http://ccg.ciens.ucv.ve/~ernesto/nds/CotoND200302.pdf, March 4, 2018, Caracas.