

ALGORITMO PARA UNA SOLUCION APROXIMADA AL RUTEO DE VEHICULOS ELECTRICOS

Kevin Arley Parra Henao

Daniel Alejandro Mesa Arango

Medellín, 17/05/2018

Estructuras de Datos Diseñada

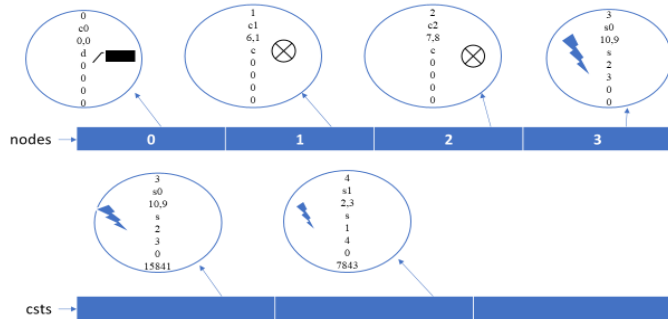
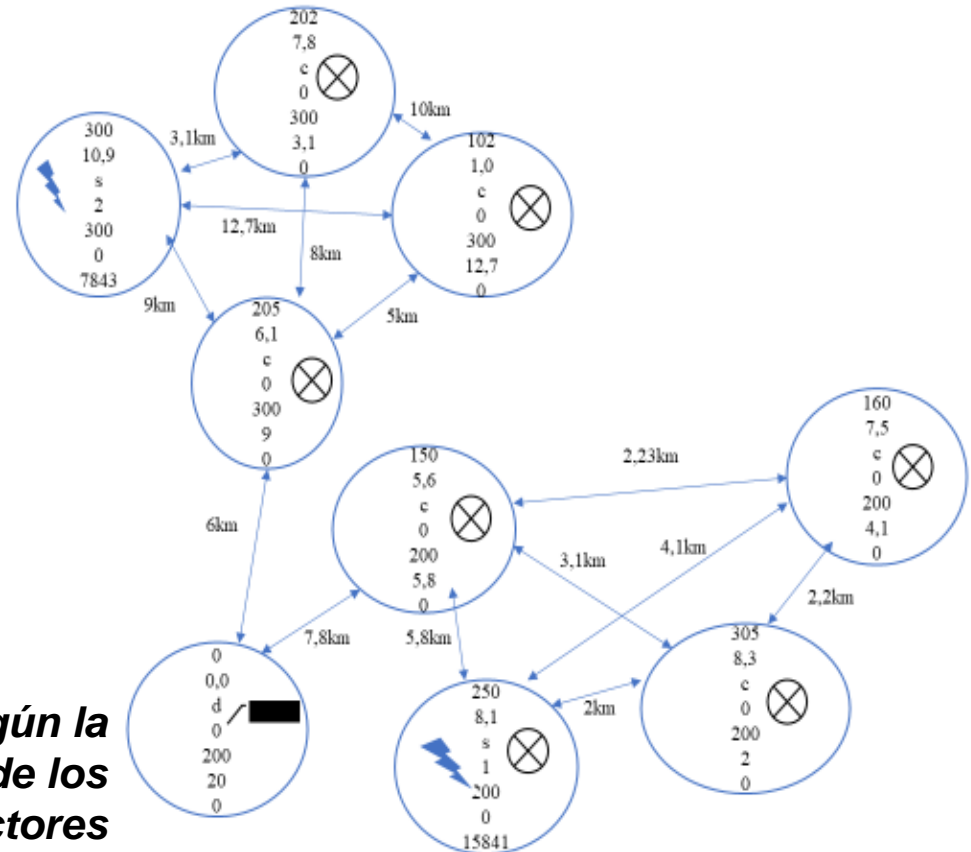


Gráfico 1: Almacenamiento de los datos de entrada, cada posición del vector contiene un nodo con sus respectivas características.

Gráfico 2: División por regiones según la estación mas cercana y construcción de los grafos, cada grafo es un vector de vectores de node.



Explicación del algoritmo y su complejidad

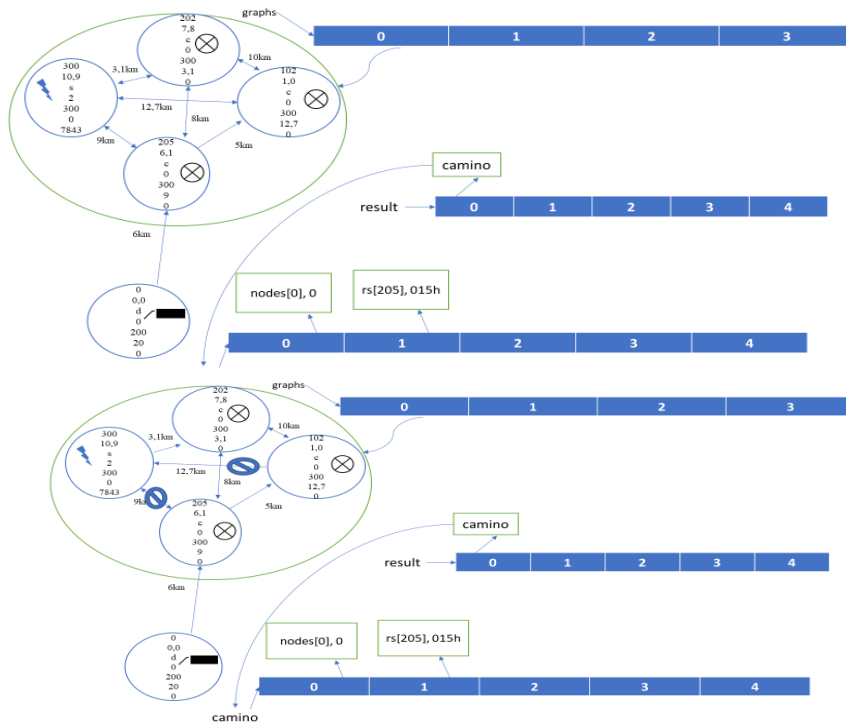


Gráfico 3: recorrido de un camión por una subregión, mostrando un caso de recarga de batería.

SUB PROBLEMA	COMPLEJIDAD
Creación de la estructura	$O(r*(m^2))$
Tsp para cada subregión	$O(r*(m^2))$
Imprimir las rutas	$O(t*(k))$
Complejidad Total	$O(2(r*m^2) + t*k)$

Tabla 1: Complejidad del algoritmo para cada una de las operaciones por cada archivo de datos, donde r representa la cantidad de regiones, m es la cantidad de nodos en tal región, t es el numero de rutas creadas o encontradas y k la cantidad de nodos en cada ruta encontrada.

Criterios de Diseño del Algoritmo

- El tiempo de ejecución para tsp agente viajero es $O(n^2)$
- Dividiendo en subgrafos podíamos tener control tanto del tiempo del camión como de la estación de carga mas conveniente
- El uso de map nos facilita el reconocimiento de cada región trabajando los procedimientos con $O(1)$
- Al trabajar con vectores también nos posibilita el acceso en $O(1)$ a cualquier posición
- Uno de los principales criterios en el diseño es la posibilidad de satisfacer todas las condiciones sin sacrificar complejidad grande en el algoritmo.

Consumo de Tiempo y Memoria

Algoritmo para ruteo de vehículos eléctricos	Tsp(vecino mas cercano)
Espacio en le Heap	2MB
Ruteo para 24f0.txt	2,6ms
Ruteo para 24t0.txt	2,6ms
Ruteo para 38f0.txt	2,5ms
Ruteo para 38t0.txt	2,6ms

	24f0.txt	24t0.txt	38f0.txt	38t0.txt
Mejor caso	19.6ms	20.8ms	19.2ms	21.3ms
Caso promedio	20.5ms	21.53ms	22.2ms	22.6ms
Peor caso	21.4ms	22.9ms	24.7ms	25.2ms

	24f0.txt	24t0.txt	38f0.txt	38t0.txt
Consumo de memoria	2 MB	3 MB	3 MB	3 Mb

En las graficas anteriores se observan muy buenos tiempos de ejecución así como un uso eficiente de espacio y de memoria.

Software en funcionamiento

```
~/RepoUni/datos2/ST0247-032/proyecto/CodigoFinal/CodigoConPruebas
342 (0.637331)
97 (0.332951)
0 (2.00554)
costo Local: 9.11204

Ruta 46:
0 (0)
72 (1.26066)
98 (0.775472)
99 (0.0822443)
342 (0.433429)
0 (1.67321)
costo Local: 7.09837

Ruta 47:
0 (0)
78 (0.0578278)
116 (0.21436)
115 (0.0916038)
117 (0.0861983)
113 (0.349302)
114 (0.0480883)
112 (0.0715306)
0 (0.5401)
costo Local: 4.95901

Ruta 48:
0 (0)
319 (0.611222)
320 (0.0598635)
17 (0.136658)
16 (0.130324)
18 (0.0587633)
257 (0.108946)
61 (0.287627)
62 (0.0506779)
176 (0.152458)
175 (0.148433)
177 (0.0951381)
344 (0.777062)
0 (0.815556)
costo Local: 9.59174

Ruta 49:
0 (0)
319 (0.611222)
52 (0.224426)
169 (0.550877)
171 (0.0421878)
170 (0.132959)
0 (0.795467)
costo Local: 5.15222

Costo Total: 378.492

danys@DESKTOP-D0J0PKG ~/RepoUni/datos2/ST0247-032/proyecto/CodigoFinal/CodigoConPruebas
$
```

Gráfico 4: Algoritmo para el ruteo de vehículos eléctricos.