

Pablo Emilio Lopez Avila Matricula 1740087

Descripción del problema de Agente viajero.

Es un problema común dentro de la logística y de la programación en el cual se propone que un hombre tiene que viajar a distintos puntos en el mundo; el problema consiste en que el hombre tiene que ir a todos los lugares de su ruta una única vez y regresar al punto de partida, pero se mide un aspecto que puede ser distancia, o tiempo o gasto; mediante algoritmos podemos encontrar una ruta que minimice al máximo uno de esos tres aspecto que se busca reducir, lo más **difícil** de este problema es su complejidad al menos a mi parecer ya que se maneja con un  $(n-1)!$  De complejidad no es la mejor pero aquí entra un caso más allá de eso, si las distancias entre las ciudades o lugares que debe visitar son simétricas (en la vida real es así) se divide el total de casos posibles entre 2 y si no es así se deben buscar los  $(n-1)!$  Posibles casos para poder evaluarlos.

Algoritmos de aproximación:

Existen casos donde nuestro algoritmo se tardará mucho para resolver nuestro problema por lo que debemos recurrir a “recursos desesperados” en los cuales permiten encontrar una solución aceptable mas no la mejor la cual se podrá obtener en el tiempo que se necesita; estos casos son algoritmos con una complejidad polinomio; Exactamente eso es una algoritmo de aproximación un algoritmo que hace lo mismo que otros en un menor tiempo de ejecución dándonos una respuesta aceptable mas no se puede asegurar que la mejor.

Algoritmo de Kruskal:

Es un algoritmo que se aplica en árboles, con pesos en cada una de sus aristas y es conexo (para cada nodo existe un conector) este utiliza las propiedades de los arboles generando “arcos” con esto asegura que se busca el peso más pequeño entre los grafos generando un subgrupo. Dicho por el mismo Joseph B. Kruskal “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.”

Árbol de Expansión mínima:

Primero que nada un árbol es grafo que tiene peso en sus aristas y no dirigido que contiene todas los nodos y posiblemente todas las aristas, la única diferencia entre todos los árboles que se pueden generar y el de expansión mínima: es que el árbol

de expansión mínima a siempre se asegura que las distancias entre los nodos siempre será la más pequeña.

**El algoritmo** en que trabajamos tomamos como base un Grafo con 10 nodos conectados los 10 todos entre si; este algoritmo se basa en la utilización de la mayoría de los que ya teníamos, agregándole el Algoritmo de Kruskal y contador de tiempo.

En el caso del kruskal como tal nos arroja el siguiente resultado:

MST con peso 14738: {'EU', 'Egipto', 'Chec', 'China', 'Canada', 'Mexico', 'UK', 'Italia', 'Spain', 'Francia'}

En el caso de aplicar “Shortest” inicie desde el nodo de “UK” hacia todos los países restantes.

{'UK': (0, 'UK', ['UK']), 'Spain': (1028, 'Spain', ['UK', 'Spain']), 'Chec': (1270, 'Chec', ['UK', 'Chec']), 'EU': (1300, 'EU', ['UK', 'EU']), 'Francia': (1300, 'Francia', ['UK', 'Francia']), 'Mexico': (1330, '**Mexico**', ['UK', 'EU', 'Mexico']), 'Italia': (1740, 'Italia', ['UK', 'Italia']), 'Canada': (2325, '**Canada**', ['UK', 'Francia', 'Canada']), 'Egipto': (4060, 'Egipto', ['UK', 'Egipto']), '**China**': (7970, 'China', ['UK', 'Chec', 'China'])}

Mediante este algoritmo dentro de nuestro algoritmo mayor vemos como en su mayoría la arista principal es la que es más corta solo en los casos de ir de UK a: México, Canadá y China si existen rutas más cortas que la “directa”

Estos resultados se pueden considerar “Exactos”.

Lo que después imprimio el algoritmo fue su vecino mas cercano:

('EU', 'Mexico') 30

('Mexico', 'EU') 30

('Francia', 'Spain') 530

('Spain', 'Francia') 530

('Italia', 'Chec') 815

('Chec', 'Italia') 815

('Italia', 'Francia') 925

('Francia', 'Italia') 925

('Canada', 'Francia') 1025

('Francia', 'Canada') 1025

('UK', 'Spain') 1028

('Spain', 'UK') 1028

('UK', 'EU') 1300

('EU', 'UK') 1300

('Egipto', 'Italia') 2385

('Italia', 'Egipto') 2385

('China', 'Chec') 6700

('Chec', 'China') 6700

**Ahora** el algoritmo mediante aproximaciones o estimaciones creó rutas distintas a partir de un nodo inicial aleatorio en el cual toma un nodo y toma a su vecino más cercano y de esa manera llegar a todos los nodos y regresar al nodo que se seleccionó al principio de manera aleatoria dándonos estos resultados:

Spain Francia 530

Francia Italia 925

Italia Chec 815

Chec China 6700

China Egipto 8440

Egipto Canadá 9366

Canadá UK 5300

UK EU 1300

EU México 30

México Spain 9045

**costo 42451**

Chec Italia 815

Italia Francia 925

Francia Canadá 1025

Canadá Spin 5505

Spain UK 1028

UK EU 1300

EU México 30

México Egipto 12000

Egipto China 8440

China Chec 6700

**costo 37768**

China Chec 6700

Chec Italia 815

Italia Francia 925

Francia Canadá 1025

Canadá Spain 5505

Spain UK 1028

UK EU 1300

EU México 30

México Egipto 12000

Egipto China 8440

**costo 37768**

Canadá Francia 1025

Francia Italia 925

Italia Chec 815

Chec China 6700

China Egipto 8440

Egipto Spin 3566

Spain UK 1028

UK EU 1300

EU México 30

México Canadá 4035

**costo 27864**

UK Spain 1028

Spain Francia 530

Francia Italia 925

Italia Chec 815

Chec China 6700

China Egipto 8440

Egipto Canadá 9366

Canadá EU 2705

EU México 30

México UK 8535

**costo 39074**

UK Spain 1028

Spain Francia 530

Francia Italia 925

Italia Chec 815

Chec China 6700

China Egipto 8440

Egipto Canadá 9366

Canadá EU 2705

EU México 30

México UK 8535

**costo 39074**

Chec Italia 815

Italia Francia 925

Francia Canadá 1025

Canadá Spain 5505

Spain UK 1028

UK EU 1300

EU México 30

México Egipto 12000

Egipto China 8440

China Chec 6700

**costo 37768**

Italia Francia 925

Francia Canadá 1025

Canadá Spain 5505

Spain UK 1028

UK EU 1300

EU México 30

México Chec 9035

Chec China 6700

China Egipto 8440

Egipto Italia 2385

**costo 36373**

Francia Italia 925

Italia Chec 815

Chec China 6700

China Egipto 8440

Egipto Canadá 9366

Canadá Spain 5505

Spain UK 1028

UK EU 1300

EU Mexico 30

México Francia 9100

**costo 43209**

EU UK 1300

UK Spain 1028

Spain Francia 530

Francia Italia 925

Italia Chec 815

Chec China 6700

China Egipto 8440

Egipto Canadá 9366

Canadá México 4035

México EU 30

**costo 33169**

**Conclusión:**

Realmente existe una gran diferencia entre los algoritmos de aproximación y los que se dedican a hacer únicamente lo que deben; en algunos casos realmente si es una respuesta u opción aceptable pero en las otras es totalmente excesivo, por lo cual si hay el tiempo de checar todo bin es mejor usar el algoritmo original y obtener una buena o la mejor opción, ya que una aproximada si nos podría disparar costos, precios o distancias.