Instituto Politécnico Nacional

Unidad Profesional Interdisciplinaria De Ingeniería Zacatecas

Hilario Abraham Rodarte España

Análisis de Algoritmos

TSP

Introducción

El problema del vendedor ambulante (TSP) es uno de los métodos más importantes en la aplicación en ciencia del transporte. TSP puede ilustrarse como un vendedor que debe viajar a través de todas las ciudades designadas por las distancias más cortas, donde cada ciudad solo puede atravesarse una vez. La solución del TSP es el camino recorrido por el vendedor. Seguramente la mejor u óptima solución de este problema es el camino con la distancia más corta o se puede llamar con un mínimo de rutas de viaje.

Objetivos

Desarrollar un algoritmo para resolver el problema TSP de manera dinámica.

Desarrollo

Pasos iniciales:

- Determinar la matriz de adyacencia que represente el grafo a recorrer
- Calcular $f(i,S)$ Para $S = 1$, donde $f(i,S)$ es igual a la distancia entre nodos de donde te encuentras y el nodo *i*.
- Así obtener $f(i,S)$ hasta $S = n - 1$
- Teniendo los resultados anteriores calcular la ecuación de la relación recursiva:

  $$f(1, v-\{1\}) = \min_{3<k<n+1}\{c_{1k} + f(k, V-\{1,k\})\}$$

  donde :

  f (1 , v-{1}) es la longitud optima del recorrido

  V es el conjunto de vértices no vacíos finitos del grafo

  v-{1} es el conjunto de vértices del grafo menos uno

  k son los vértices conectados al nodo inicial

  c_{1k} es el peso o longitud del vértice del nodo inicial al nodo k

- Después de calcular la ecuación recursiva, obtendremos el peso/longitud óptima para el recorrido.

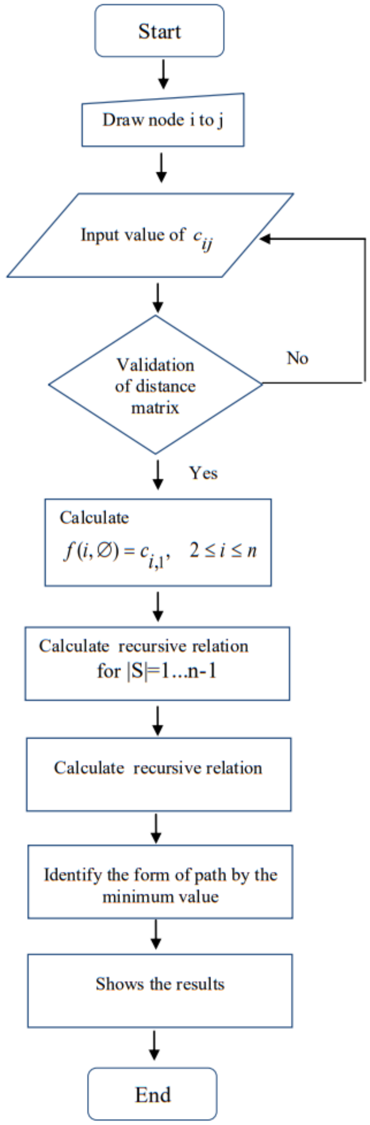
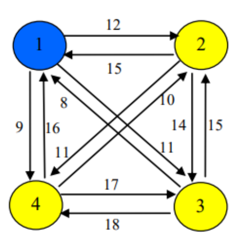


Ilustración 1 Diagrama de flujo de la programación dinámica para TSP

Ejemplo numérico

Teniendo el siguiente grafo



En donde su matriz de adyacencia es la siguiente:

$$c = \begin{bmatrix} 0 & 12 & 11 & 16 \\ 15 & 0 & 15 & 10 \\ 8 & 14 & 0 & 18 \\ 9 & 11 & 17 & 0 \end{bmatrix}$$

Step 1

Calculate $f(i,\varnothing) = c_{i,1}, \quad 2 \le i \le n$

We get:

$$f(2,\varnothing) = c_{21}, = 15$$

$$f(3,\varnothing) = c_{31}, = 8$$

$$f(4,\varnothing) = c_{41}, = 9$$

Step 2

For | S | = 1, calculate $f(i,s) = \min_{j \in S}\left\{c_{ij} + f(j, S - \{j\})\right\}$

We get:

$$f(2,\{3\}) = \left\{c_{23} + f(3,\varnothing)\right\} = 15 + 8 = 23$$

$$f(3,\{2\}) = \left\{c_{32} + f(2,\varnothing)\right\} = 14 + 15 = 29$$

$$f(4,\{2\}) = \left\{c_{42} + f(2,\varnothing)\right\} = 11 + 15 = 26$$

$$f(2,\{4\}) = \left\{c_{24} + f(4,\varnothing)\right\} = 10 + 9 = 19$$

$$f(3,\{4\}) = \left\{c_{34} + f(4,\varnothing)\right\} = 18 + 9 = 27$$

$$f(4,\{3\}) = \left\{c_{43} + f(3,\varnothing)\right\} = 17 + 8 = 25$$

For | S | = 2, calculate $f(i,s) = \min_{j \in S} \{c_{ij} + f(j, S-\{j\})\}$

We get:

$$f(2,\{3,4\}) = \min\{c_{23}+f(3,\{4\}), c_{24}+f(4,\{3\})\}$$
$$= \min\{15+27, 10+25\}$$
$$= \min\{42,35\} = 35$$

$$f(3,\{2,4\}) = \min\{c_{32}+f(2,\{4\}), c_{34}+f(4,\{2\})\}$$
$$= \min\{14+19, 27+11\}$$
$$= \min\{33,38\} = 33$$

$$f(4,\{2,3\}) = \min\{c_{42}+f(2,\{4\}), c_{43}+f(3,\{2\})\}$$
$$= \min\{11+22, 17+29\}$$
$$= \min\{33,46\} = 33$$

Step 3

By using the equation, $f(1, v-\{1\}) = \min_{2 \le k \le n} \{c_{1k} + f(k, V-\{1,k\})\}$

We get:

$$f(1,\{2,3,4\}) = \min\{c_{12}+f(2,\{3,4\}), c_{13}+f(3,\{2,4\}), c_{14}+f(4,\{2,3\})\}$$
$$= \min\{12+35, 11+33, 16+33\}$$
$$= \min\{47,44,49\} = 44$$

Así se tiene que el camino más corto es 44, para obtener el camino basta con hacer backtraking para obtener los nodos por los que pasó el vendedor con algunas banderas.

Conclusiones

En la aplicación del algoritmo de TSP de manera dinámica esto produce la ruta optima para seguir con una ejecución optima del programa, ya que el tiempo de ejecución, comparado con una algoritmo de fuerza bruta es comparable, y más aun para grafos más grandes.