Rutas más cortas entre todos los pares de nodos

IIC2133

Yadran Eterovic, Vicente Domínguez

2020-1

Podemos ejecutar $|V|$ veces un algoritmo para rutas más cortas desde un vértice, una vez para cada vértice en el rol de s :

- si los costos de las aristas son no negativos, podemos usar el algoritmo de Dijkstra

  … el tiempo de ejecución sería $O(VE \log V)$

- si las aristas pueden tener costos negativos, debemos usar el algoritmo de Bellman-Ford

  … el tiempo de ejecución sería O(V^2E), que para grafos densos es O(V^4)

Podemos mejorar este último desempeño

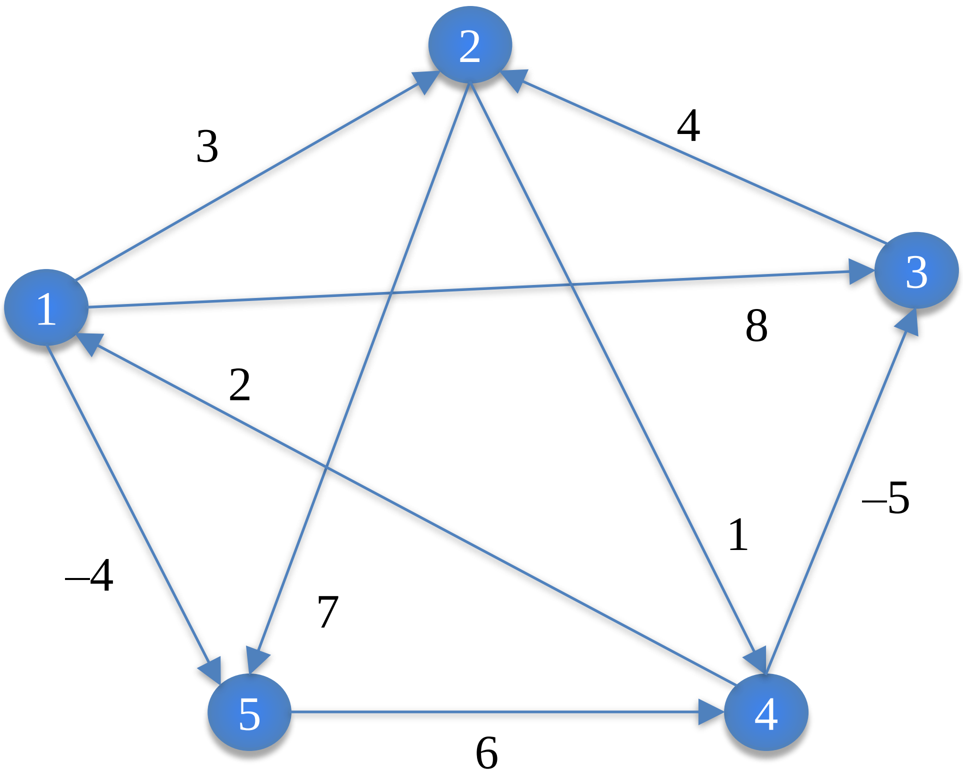
En resumen

- Dijkstra
  Busca el camino más corto de **un nodo a todos los nodos**
- Bellman-Ford
  Busca el camino más corto de **un nodo a todos los nodos**, los **pesos negativos están permitidos**
- ???
  Busca el camino más corto **entre todos los pares de vertices**, **pesos negativos incluidos**

Representaremos G por su *matriz de adyacencias* (en vez de las listas de adyacencias, que hemos usado mayoritariamente)

Si los vértices están numerados 1, 2, …, n (o sea, $|V| = n$),

… el input es una matriz W que representa los costos de las aristas



$$
W = \begin{array}{c}
\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{pmatrix}
0 & 3 & 8 & \infty & -4 \\
\infty & 0 & \infty & 1 & 7 \\
\infty & 4 & 0 & \infty & \infty \\
2 & \infty & -5 & 0 & \infty \\
\infty & \infty & \infty & 6 & 0
\end{pmatrix}
\end{array}
\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
$$

$W = (\omega_{ij})$, en que

$$\omega_{ij} = 0 \qquad \text{si } i = j$$

$$= \text{costo de la arista direccional } (i, j) \quad \text{si } i \neq j \text{ y } (i, j) \in E$$

$$= \infty \qquad \text{si } i \neq j \text{ y } (i, j) \notin E$$

Suponemos que *G* **no contiene ciclos de costo negativo**:

- por lo tanto, *las rutas más cortas no contienen ciclos*

Consideremos los vértices intermedios de una ruta más corta

Supongamos que la ruta más corta de *i* a *j* es

$$i \longrightarrow 2 \longrightarrow 6 \longrightarrow 4 \longrightarrow 3 \longrightarrow j$$

Podemos estar seguros (¿por qué?) de que:

- $i \longrightarrow 2 \longrightarrow 6$ es una ruta más corta de *i* a 6, y
- $6 \longrightarrow 4 \longrightarrow 3 \longrightarrow j$ es una ruta más corta de 6 a *j*

Elegimos el vértice 6 para hacer la partición de la ruta, porque es *el de mayor valor numérico* (asignado, obviamente, a priori de manera arbitraria)

En el ej., elegimos el vértice 6 para hacer la partición de la ruta, porque es el de mayor valor numérico:

- así, estamos seguros de que los vértices intermedios de la ruta más corta de *i* a 6 *sólo pueden tener valores numéricos entre 1 y 5*
- … y los vértices intermedios de la ruta más corta de 6 a *j sólo pueden tener valores numéricos entre 1 y 5*

En general, sea *k* el vértice intermedio de mayor valor numérico en la ruta más corta de *i* a *j*

… entonces, los vértices intermedios de la ruta más corta de *i* a *k* sólo pueden tener valores numéricos entre 1 y *k*–1

… y los vértices intermedios de la ruta más corta de *k* a *j* también sólo pueden tener valores numéricos entre 1 y *k*–1

(por supuesto, además, todos estos vértices intermedios deben tener valores numéricos distintos de *i* y *j*)

El algoritmo de Floyd-Warshall

Considera los vértices intermedios de una ruta más corta

Si los vértices de G son $V = \{1, 2, \ldots, n\}$, consideremos el subconjunto $\{1, 2, \ldots, k\}$, para algún k

Para cualquier par de vértices $i, j \in V$,

… consideremos todas las rutas de i a j cuyos vértices intermedios están todos tomados del conjunto $\{1, 2, \ldots, k\}$

… y sea $\boldsymbol{p}$ una ruta más corta entre ellas

Usamos una argumentación que ya hemos visto:

k puede ser o no un vértice (intermedio) de $\boldsymbol{p}$:

- no sabemos, pero sí sabemos cuáles son las implicaciones en cada caso

Si k **no es** un vértice de ***p***,

… entonces todos los vértices (intermedios) de ***p*** están en el conjunto $\{1, 2, \ldots, k-1\}$

⇒ una ruta más corta de i a j con todos los vértices intermedios en $\{1, 2, \ldots, k-1\}$

… es también una ruta más corta de i a j con todos los vértices intermedios en $\{1, 2, \ldots, k\}$

Si k **es** un vértice de ***p***, entonces podemos dividir ***p*** en dos tramos:

el tramo p_1 de i a k

... y el tramo p_2 de k a j

⇒ por el principio de subestructura óptima (o de optimalidad)

... p_1 es una ruta más corta de i a k con todos los vértices intermedios en $\{1, 2, \ldots, k-1\}$

... y p_2 es una ruta más corta de k a j con todos los vértices intermedios en $\{1, 2, \ldots, k-1\}$

Sea $d_{ij}^{(k)}$ el costo de una ruta más corta de i a j, tal que todos los vértices intermedios están en el conjunto $\{1, 2, \ldots, k\}$

Cuando $k = 0$,

… una ruta de i a j sin vértices intermedios con número mayor que 0

… simplemente no tiene vértices intermedios,

… y por lo tanto tiene a lo más una arista $\Rightarrow d_{ij}^{(0)} = \omega_{ij}$

Definimos $d_{ij}^{(k)}$ recursivamente por

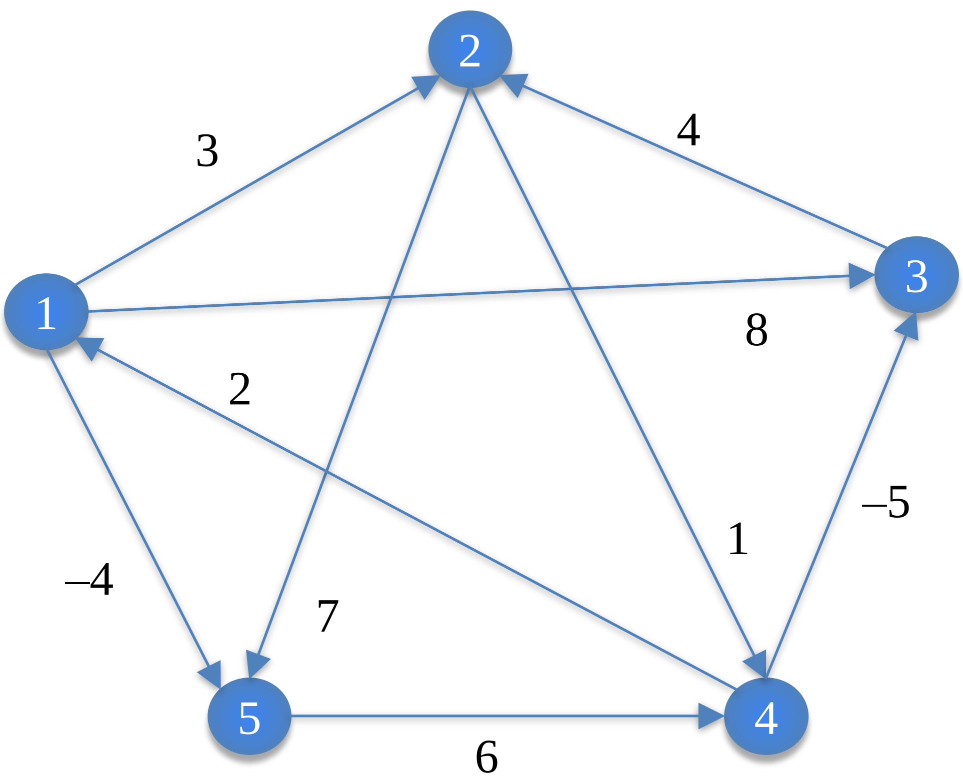
$$
\begin{aligned}
d_{ij}^{(k)} &= \omega_{ij} && \text{si } k = 0 \\
&= \min(\, d_{ij}^{(k-1)}\, ,\, d_{ik}^{(k-1)} + d_{kj}^{(k-1)}\,) && \text{si } k \geq 1
\end{aligned}
$$

La matriz $D^{(n)} = (d_{ij}^{(n)})$ da la respuesta final:

$d_{ij}^{(n)} = \delta(i, j)$ para todo $i, j \in V$

El algoritmo de Floyd-Warshall, *bottom-up*, toma tiempo O(V^3)

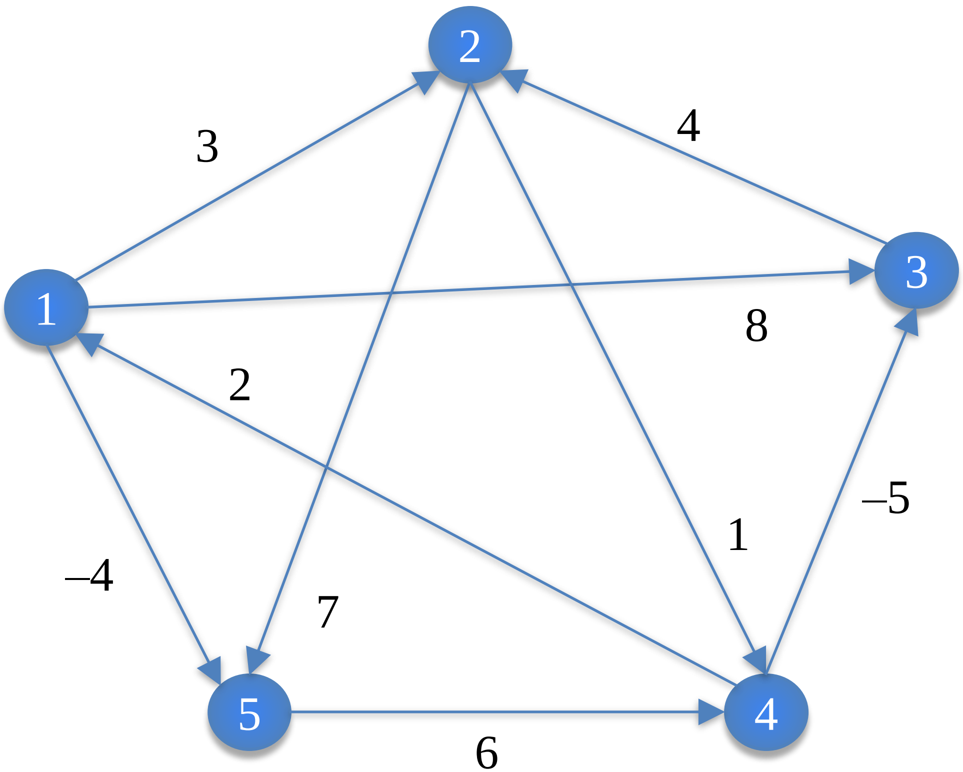
```
D^(0) = W
for k = 1 … n:
    sea D^(k) = (d_ij^(k)) una nueva matriz
    for i = 1 … n:
        for j = 1 … n:
            d_ij^(k) = min(d_ij^(k-1), d_ik^(k-1)+d_kj^(k-1))
return D^(n)
```



$$D^{(0)} = \begin{matrix} 0 & 3 & 8 & \infty & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \infty & \infty \\ 2 & \infty & -5 & 0 & \infty \\ \infty & \infty & \infty & 6 & 0 \end{matrix}$$

$$D^{(1)} = \begin{matrix} 0 & 3 & 8 & \infty & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & \infty & \infty \\ 2 & 5 & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{matrix}$$

$$D^{(2)} = \begin{matrix} 0 & 3 & 8 & 4 & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & 5 & 11 \\ 2 & 5 & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{matrix}$$



$$D^{(3)} = \begin{matrix} 0 & 3 & 8 & 4 & -4 \\ \infty & 0 & \infty & 1 & 7 \\ \infty & 4 & 0 & 5 & 11 \\ 2 & -1 & -5 & 0 & -2 \\ \infty & \infty & \infty & 6 & 0 \end{matrix}$$

$$D^{(4)} = \begin{matrix} 0 & 3 & -1 & 4 & -4 \\ 3 & 0 & -4 & 1 & -1 \\ 7 & 4 & 0 & 5 & 3 \\ 2 & -1 & -5 & 0 & -2 \\ 8 & 5 & 1 & 6 & 0 \end{matrix}$$

$$D^{(5)} = \begin{matrix} 0 & 1 & -3 & 2 & -4 \\ 3 & 0 & -4 & 1 & -1 \\ 7 & 4 & 0 & 5 & 3 \\ 2 & -1 & -5 & 0 & -2 \\ 8 & 5 & 1 & 6 & 0 \end{matrix}$$