



All pairs shortest paths

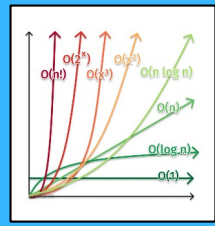
mejora al algoritmo inicial



por Vpode

Teoría de Algoritmos
Víctor Podberezski
vpodberezski@fi.uba.ar

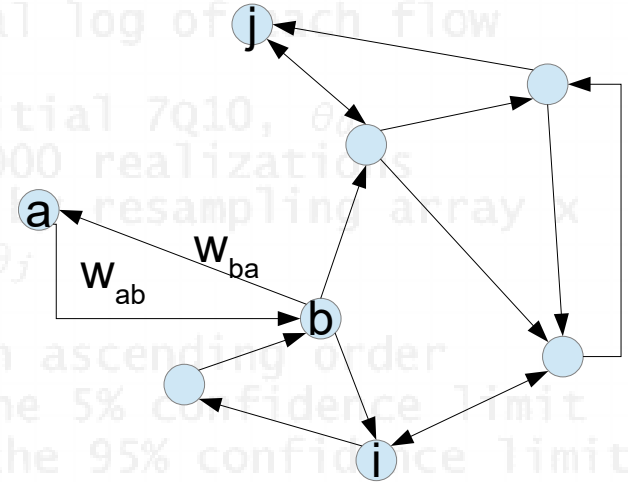
Problema



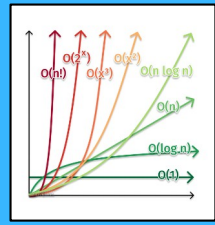
- Sea
 - $G=(V,E)$ un grafo ponderado dirigido
 - Sin ciclos negativos

Donde

- cada vértice $e=(a,b) \in E$ tiene un peso w_{ab}
- Queremos saber
 - para cada par $i,j \in V$ el camino mínimo entre ellos



Relación de recurrencia



- En resumen

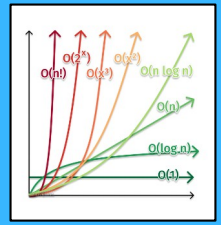
$$L_{ij}^{(0)} = \begin{cases} 0 & \text{si } i=j \\ \infty & \text{si } i \neq j \end{cases}$$

$$L_{ij}^{(1)} = W$$

$$L_{ij}^{(m)} = \min_{1 \leq k \leq n} (L_{ik}^{(m-1)} + w_{kj})$$

- Si $m=|V|-1$ conoceremos el camino mínimo entre todos los vértices

Solución iterativa



```
ObtenerCaminoMinimos(W)
```

```
  Sea  $n = |V|$ 
```

```
  Sea  $L[1] = W$ 
```

```
  Desde  $m=2$  a  $n-1$ 
```

```
    Sea  $L[m]$  una nueva matriz de  $n \times n$ 
```

```
     $L[m] = \text{calculo\_camino\_minimo}(L[m-1], W)$ 
```

```
  next  $j$ 
```

```
  Retornar  $L[n-1]$ 
```

```
  conf_5=theta(50)
```

```
  conf_95=theta(951)
```

```
END MAIN PROGRAM
```

```
FUNCTION GET_7Q10(input array:  $x(1..n)$ )
```

```
  for  $i=1$  to  $n$ 
```

```
     $\text{sum} = \text{sum} + x(i)^2$ 
```

```
  next  $i$ 
```

```
  mean =  $\text{sum} / n$ 
```

```
  //compute
```

```
//x is an array of  $n$  annual low flows  
//loop through all years  
//take the natural log of each flow
```

```
//compute the initial 7Q10,  $\theta_0$   
//loop through 1000 realizations
```

```
//create array  $y$  by resampling  $x$   
//compute 7Q10,  $\theta_j$ 
```

```
//sort  $\theta_1.. \theta_{1000}$  in ascending order  
//report  $\theta_{50}$  as the 5% confidence limit
```

```
//report  $\theta_{951}$  as the 95% confidence limit
```

```
//loop through years
```

```
//compute
```

$\Theta(n)$

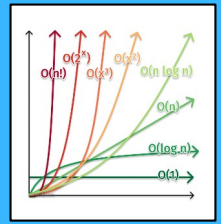


Complejidad
temporal:

$\Theta(n^4)$

$\Theta(n^3)$

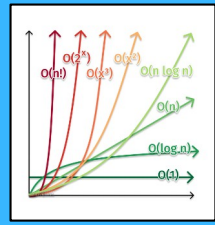
Cálculo $L^{(m)}$ con $L^{(m-1)}$



```
read x(1..n)
for i=1 to n
  calculo_camino_minimo(L, W)
next i
Sea n=|V|
Sea L' una nueva matriz de nxn
for j=1 to 1000
  Desde i=1 a n
    Desde j=1 a n
      L'[i,j]=∞
      Desde k=1 a n
        L'[i,j] = min ( L'[i,j] , L[i,k] + W[k,j] )
      next k
    next j
  sort(theta)
  conf_5=theta(5)
  conf_95=theta(95)
next j
Retornar L'
```

$L_{ij}^{(m)} = \min_{1 \leq k \leq n} (L_{ik}^{(m-1)} + w_{kj})$

Similitud con multiplicación de matrices

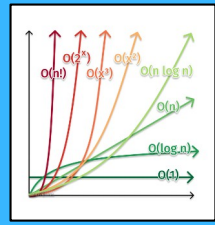


$$L_{ij}^{(m)} = \min_{1 \leq k \leq n} (L_{ik}^{(m-1)} + w_{kj})$$

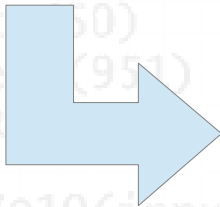
$$L' = L \diamond W$$

$$L'_{ij} = \min (L_{i1} + w_{1j}, L_{i2} + w_{2j}, \dots, L_{in} + w_{nj})$$

En búsqueda de una mejora...



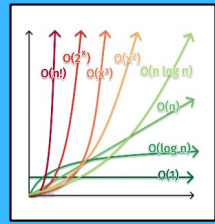
- Para el cálculo de cada $L[m]$ usamos $\text{calculo_camino_minimo}(L[m-1], W)$
 - $L[1]=W$
 - $L[2]=\text{calculo_camino_minimo}(W, W)=W \diamond W$
- Generalizando



$L[1] =$	W	$W^{''1''}$
$L[2] =$	$W \diamond W$	$W^{''2''}$
$L[3] =$	$W \diamond W \diamond W$	$W^{''3''}$
$L[4] =$	$W \diamond W \diamond W \diamond W = W^{''2''} \diamond W^{''2''}$	$W^{''4''}$
$L[r=a+b] =$	$(W^{''a''}) \diamond (W^{''b''})$	$W^{''r''}$

(!) Solo si \diamond es asociativa

Asociatividad



- Sean

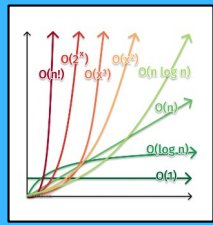
- X, Y, Z matrices de $n \times n$

- $X \diamond Y = (X, Y)_{ij} = \min_k (x_{ik} + y_{kj})$ para todo i, j

- Queremos ver que

- $(X \diamond Y) \diamond Z = X \diamond (Y \diamond Z)$

Asociatividad: Demostración



read x(1..n) // x is an array of n annual low flows
 for i=1 to n // loop through all years
 x(i)=ln(x(i)) // take the natural log of each flow
 next i
 7Q10=exponent[GET_7Q10(x)] //compute the initial 7Q10, θ_0
 for j=1 to 1000 //loop through 1000 realizations
 y=GET_RESAMPLE(x) //create array y by resampling array x
 theta(j)=exponent[GET_7Q10(y)] //compute 7Q10, θ_j
 next j
 sort(theta) //sort $\theta_1.. \theta_{1000}$ in ascending order
 conf_5=theta(50) //report θ_{50} as the 5% confidence limit
 conf_95=theta(951) //report θ_{951} as the 95% confidence limit
 END MAIN PROGRAM

 FUNCTION GET_7Q10(input array: x(1..n))
 for i=1 to n //loop through years
 sum=sum+x(i)
 sum2=sum2+x(i)^2
 next i
 mean=sum/n
 var=(sum2/n)-(mean^2)
 stddev=sqrt(var)
 //compute 7Q10
 return exponent[7*stddev+mean]

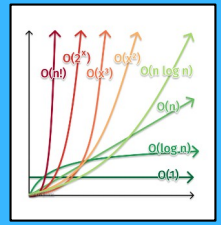
$$((X, Y), Z)_{ij} = \min_k (\min_{k'} (X_{ik} + Y_{kk'}) + Z_{k'j})$$

$$= \begin{matrix} \begin{matrix} i \\ \vdots \\ j \end{matrix} \\ \left[\begin{matrix} \vdots \\ \vdots \\ \vdots \end{matrix} \right] \end{matrix} = \begin{matrix} \begin{matrix} i \\ \vdots \\ j \end{matrix} \\ \left[\begin{matrix} \vdots \\ \vdots \\ \vdots \end{matrix} \right] \end{matrix} \diamond \begin{matrix} \begin{matrix} k' \\ \vdots \\ k' \end{matrix} \\ \left[\begin{matrix} \vdots \\ \vdots \\ \vdots \end{matrix} \right] \end{matrix} \diamond \begin{matrix} \begin{matrix} j \\ \vdots \\ j \end{matrix} \\ \left[\begin{matrix} \vdots \\ \vdots \\ \vdots \end{matrix} \right] \end{matrix}$$

$$= \begin{matrix} \begin{matrix} i \\ \vdots \\ j \end{matrix} \\ \left[\begin{matrix} \vdots \\ \vdots \\ \vdots \end{matrix} \right] \end{matrix} \diamond \begin{matrix} \begin{matrix} k' \\ \vdots \\ k' \end{matrix} \\ \left[\begin{matrix} \vdots \\ \vdots \\ \vdots \end{matrix} \right] \end{matrix} \diamond \begin{matrix} \begin{matrix} j \\ \vdots \\ j \end{matrix} \\ \left[\begin{matrix} \vdots \\ \vdots \\ \vdots \end{matrix} \right] \end{matrix}$$

X **Y** **Z**

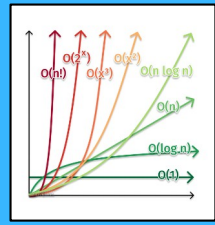
Asociatividad: Demostración (cont.)



$$\begin{aligned}
 & \text{read } x(1..n) \quad //x \text{ is an array of } n \text{ annual low flows} \\
 & \text{for } i=1 \text{ to } n \quad //loop through all years \\
 & \quad x(i)=\ln(x(i)) \quad //take the natural log of each flow \\
 & \text{next } i \\
 & 7Q10=\text{exponent}[\text{GET_7Q10}(x)] \quad //compute the initial 7Q10, \theta_0 \\
 & \text{for } j=1 \text{ to } 1000 \quad //loop through 1000 realizations \\
 & \quad y=\text{GET_RESAMPLE}(x) \quad //create array y by resampling array x \\
 & \quad \theta(j)=\text{exponent}[\text{GET_7Q10}(y)] \quad //compute 7Q10, \theta_j \\
 & \text{next } j \\
 & \text{sort}(\theta) \quad //sort \theta_1.. \theta_{1000} \text{ in ascending order} \\
 & \text{conf_5}=\theta(50) \quad //report \theta_0 \text{ as the 5\% confidence limit} \\
 & \text{conf_95}=\theta(951) \quad //report \theta_{951} \text{ as the 95\% confidence limit} \\
 & \text{END MAIN PROGRAM} \\
 & \text{FUNCTION GET_7Q10(input array: } x(1..n)) \\
 & \quad \text{for } i=1 \text{ to } n \quad //loop through years \\
 & \quad \quad \text{sum}=\text{sum}+x(i)^2 \\
 & \quad \text{next } i \\
 & \quad \text{mean}=\text{sum}/n \quad //compute mean
 \end{aligned}$$

$$\begin{aligned}
 & \left(X, (Y, Z) \right)_{ij} = \min_k \left(X_{ik'} + \min_k (Y_{k'k} + Z_{kj}) \right) \\
 & \left[\begin{matrix} i & j \\ \vdots & \vdots \end{matrix} \right] = \left[\begin{matrix} i & \vdots \\ \vdots & \vdots \end{matrix} \right] \diamond \left[\begin{matrix} j \\ \vdots \end{matrix} \right] \\
 & \qquad \qquad \qquad \mathbf{X} \qquad \qquad \qquad \mathbf{Y} \diamond \mathbf{Z} \\
 & = \left[\begin{matrix} i & \vdots \\ \vdots & \vdots \end{matrix} \right] \diamond \left[\begin{matrix} k' \\ \vdots \end{matrix} \right] \diamond \left[\begin{matrix} j \\ \vdots \end{matrix} \right] \\
 & \qquad \qquad \qquad \mathbf{X} \qquad \qquad \qquad \mathbf{Y} \qquad \qquad \qquad \mathbf{Z}
 \end{aligned}$$

Asociatividad: Demostración (cont.)



- Si partimos de

$$((X, Y), Z)_{ij} = \min_{k'} (\min_k (X_{ik} + Y_{kK'}) + Z_{k'j})$$

- Podemos ver que $Z_{k'j}$ no depende de k

$$((X, Y), Z)_{ij} = \min_{k'} (\min_k (X_{ik} + Y_{kK'} + Z_{k'j}))$$

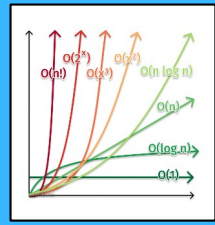
- Renombramos $k' \rightarrow a$ y $k \rightarrow b$

$$((X, Y), Z)_{ij} = \min_a (\min_b (X_{ib} + Y_{ba} + Z_{aj}))$$

- Invertimos el calculo de los mínimos

$$((X, Y), Z)_{ij} = \min_b (\min_a (X_{ib} + Y_{ba} + Z_{aj}))$$

Asociatividad: Demostración (cont.)



- Si partimos de

$$(X, (Y, Z))_{ij} = \min_{k'} (X_{ik'} + \min_k (Y_{k'k} + Z_{kj}))$$

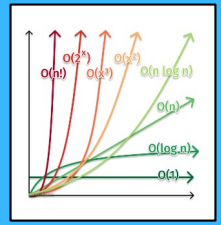
- Podemos ver que $X_{ik'}$ no depende de k

$$(X, (Y, Z))_{ij} = \min_{k'} (\min_k (X_{ik'} + Y_{k'k} + Z_{kj}))$$

- Renombramos $k' \rightarrow b$ y $k \rightarrow a$

$$(X, (Y, Z))_{ij} = \min_b (\min_a (X_{ib} + Y_{ba} + Z_{aj}))$$

La operación es asociativa



- Como:

$$(X, (Y, Z))_{ij} = \min_b (\min_a (X_{ib} + Y_{ba} + Z_{aj}))$$

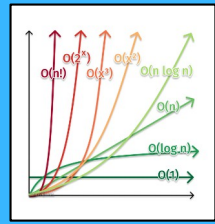
$$((X, Y), Z)_{ij} = \min_b (\min_a (X_{ib} + Y_{ba} + Z_{aj}))$$

- Entonces

$$(X, (Y, Z))_{ij} = ((X, Y), Z)_{ij}$$

- Y la operación es asociativa (c.q.d)

Regresando a la mejora

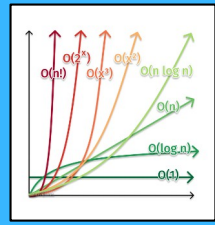


- Utilizando las potencias de 2

$L[1] =$	W	$W^{1''}$
$L[2] =$	$W \diamond W$	$W^{2''}$
$L[4] =$	$W^{2''} \diamond W^{2''}$	$W^{4''}$
$L[8] =$	$W^{4''} \diamond W^{4''}$	$W^{8''}$
$L[2^x] =$	$W^{2^{x-1}} \diamond W^{2^{x-1}}$	$W^{r''}$

- Recordamos que para todo $m \geq n-1$, $L[m] = L[n-1]$
 - Por lo tanto alcanza que $2^x \geq n-1$ para hallar el resultado final
 - Usaremos $x = \lceil \log_2 n - 1 \rceil$

Solución iterativa (mejorada)



```
read x(1..n)
for i=1 to n
  //x is an array of n annual low flows
  //loop through all years
  //take the natural log of each flow
next i
ObtenerCaminosMinimos(W)
  Sea n=|V|
  Sea L[1] = W
  for i=1 to 1000
    //compute the initial 7Q10,  $\theta_0$ 
    //loop through 1000 realizations
    //create array y by resampling array x
    Desde m < n-1
      //compute 7Q10,  $\theta_1$ 
      Sea L[2m] una nueva matriz de nxn
      L[2m] = calculo_camino_minimo(L[m], L[m])
      m = 2m
    }  $\Theta(\log n)$ 
  Retornar L[m]
  //sort  $\theta_1.. \theta_{1000}$  in ascending order
  //report  $\theta_{50}$  as the 5% confidence limit
  //report  $\theta_{951}$  as the 95% confidence limit
END MAIN PROGRAM
FUNCTION GET_7Q10(input array: x(1..n))
  for i=1 to n
    //loop through years
    //compute  $\theta_i$ 
  next i
  //compute  $\theta_{50}$  and  $\theta_{951}$ 
  return  $\theta_{50}$  and  $\theta_{951}$ 
END FUNCTION
```

$\Theta(n^3)$

$\Theta(n^3 \log n)$