El camino más corto
Clase 12

Investigación Operativa UTN FRBA 2020
Curso: I4051
Elaborado por: Rodrigo Maranzana
Docente: Martín Palazzo

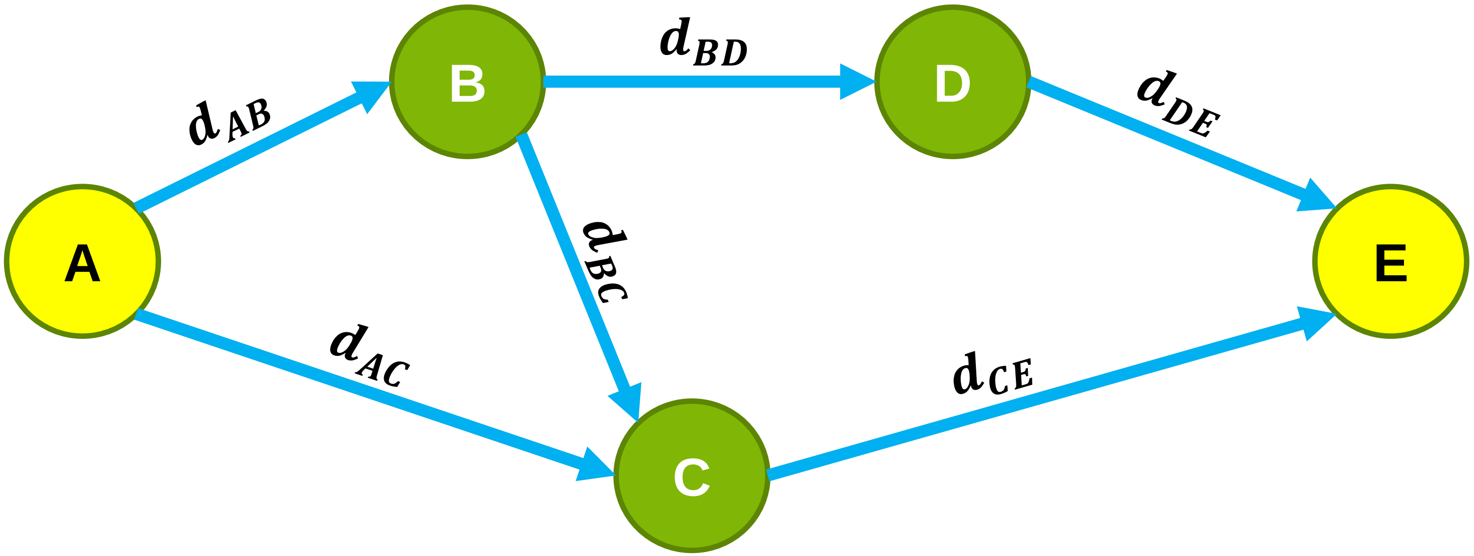
Formalización del camino más corto

Modelo de **programación matemática:**

- **Flujo de Mínimo Costo**

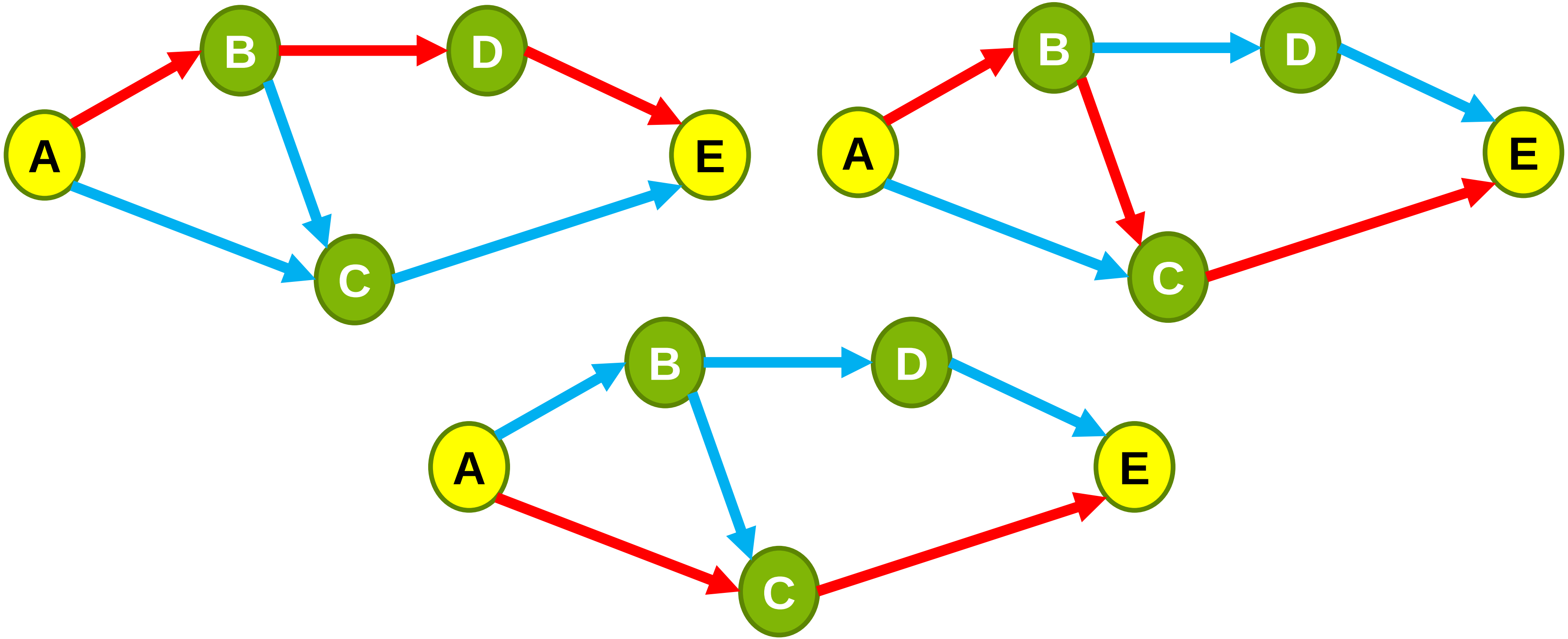
* Resolución simple por algoritmos generalistas.
* Interés matemático teórico.

Formalización del camino más corto

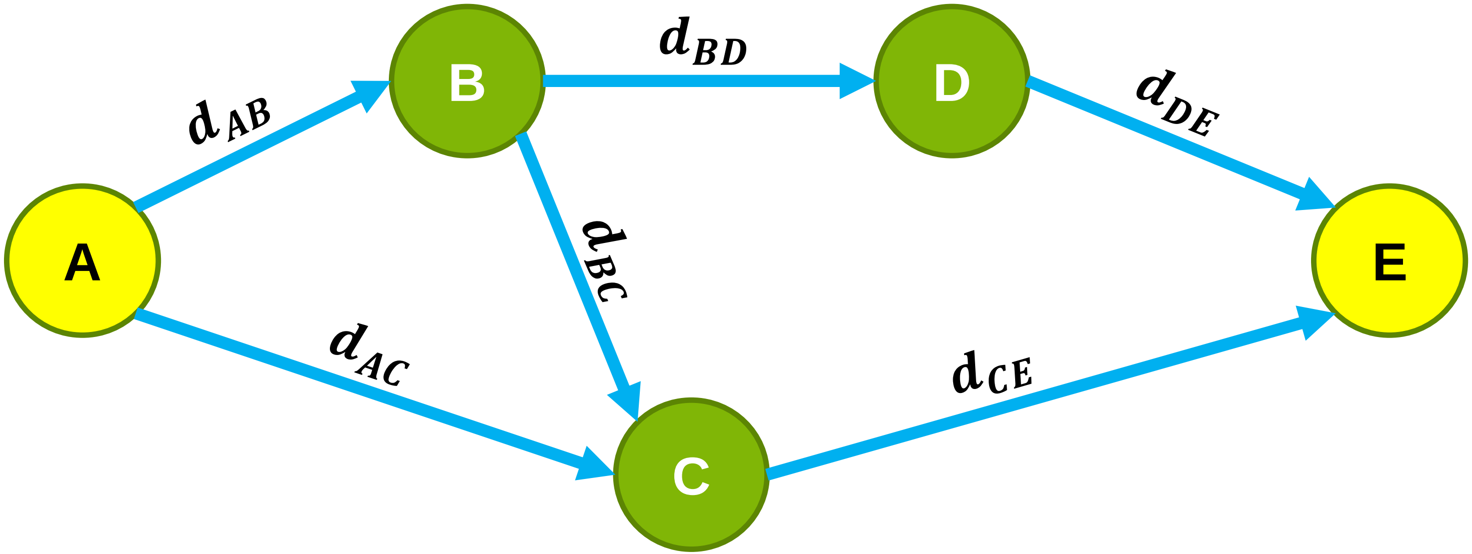


Formalización del camino más corto

Caminos posibles:



Formalización del camino más corto



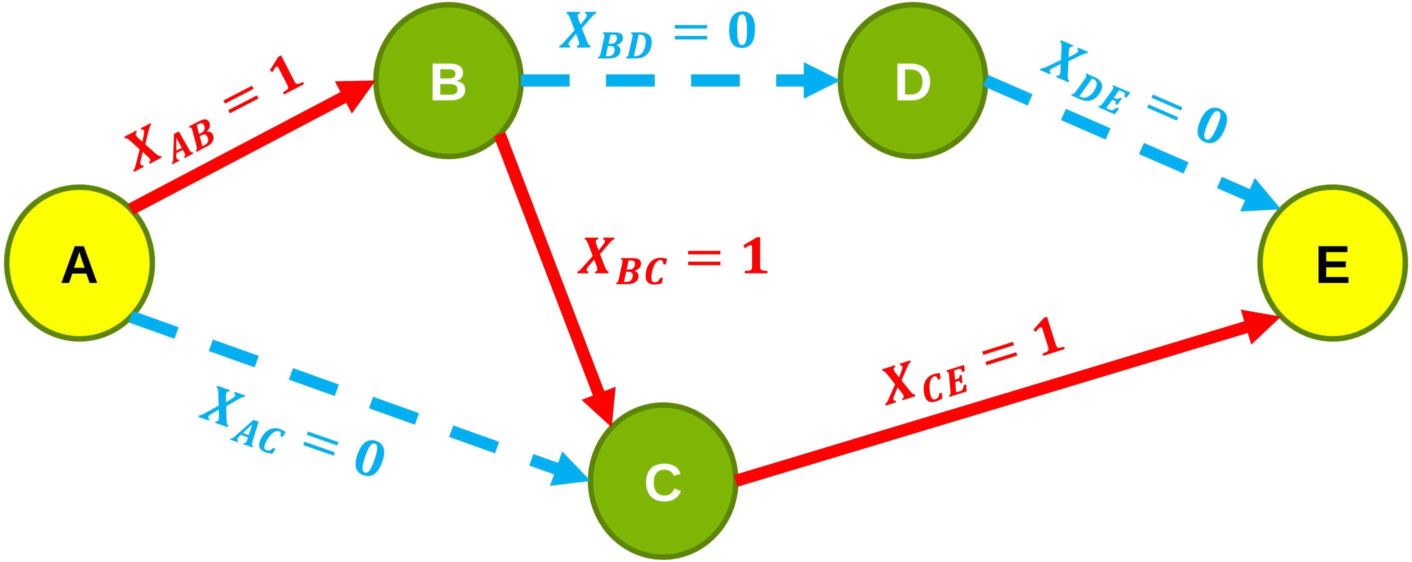
$\boldsymbol{X_{ij}}$ **Variable de decisión binaria de ir por camino i => j**

X_{ij} = 0, no elijo el camino

X_{ij} = 1, elijo el camino

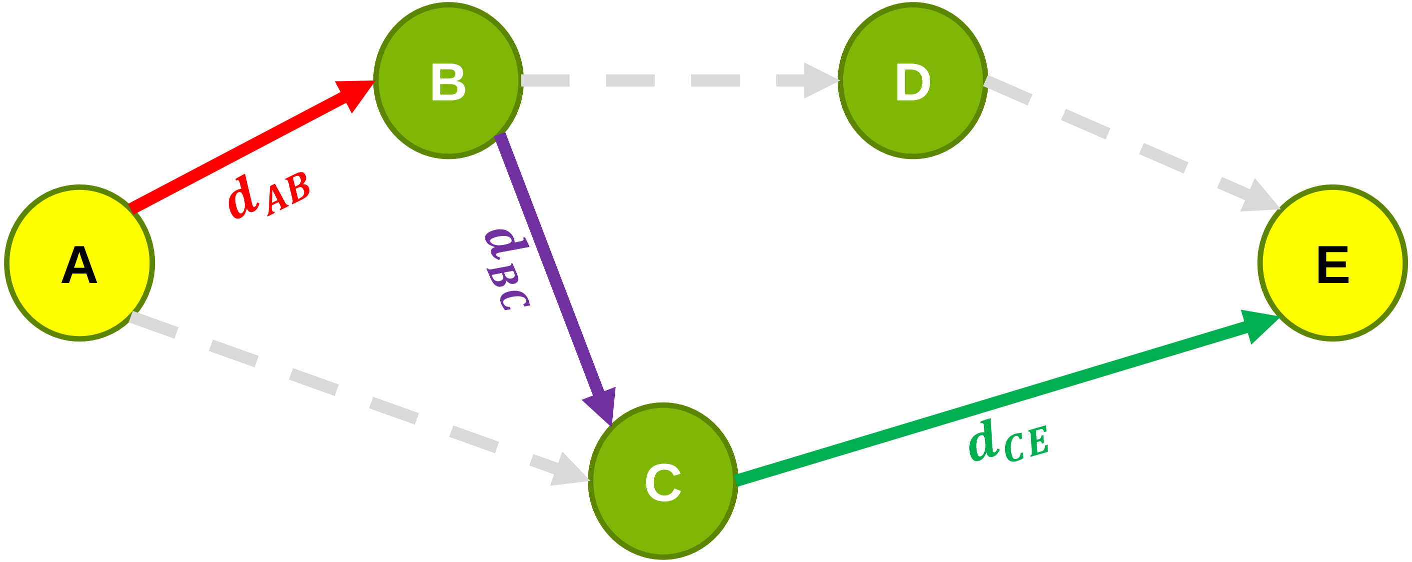
Formalización del camino más corto

Veamos un camino:



Formalización del camino más corto

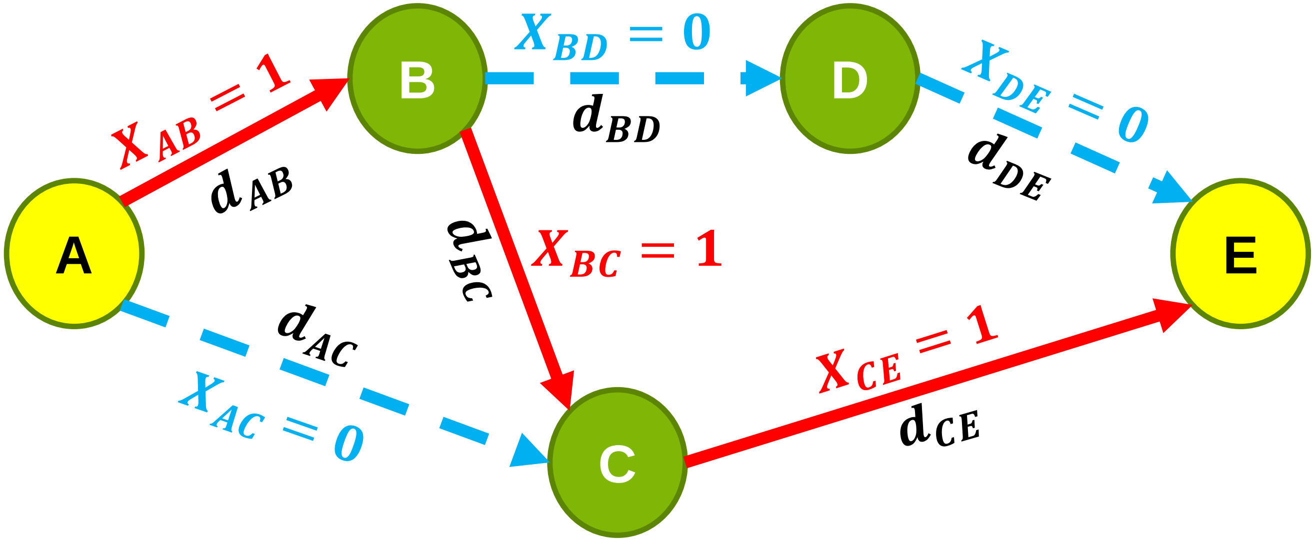
Cálculo de distancia del camino:



$$d_{TOTAL} = d_{AB} + d_{AB} + d_{CE}$$

Formalización del camino más corto

Generalización del cálculo de distancia del camino:



$$d_{TOTAL} = X_{AB} * d_{AB} + X_{BC} * d_{BC} + X_{CE} * d_{CE} +$$
$$+X_{BD} * d_{BD} + X_{AC} * d_{AC} + X_{DE} * d_{DE}$$

Formalización del camino más corto

Generalización del cálculo de distancia del camino:

$$\boldsymbol{d_{TOTAL} = \overset{1}{X}_{AB} * d_{AB} + \overset{1}{X}_{BC} * d_{BC} + \overset{1}{X}_{CE} * d_{CE} +}$$

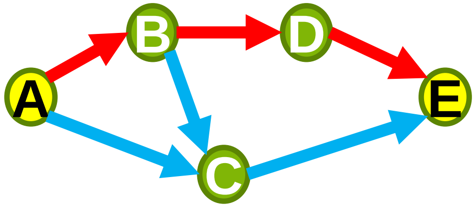
$$\boldsymbol{+ \underset{0}{X}_{BD} * d_{BD} + \underset{0}{X}_{AC} * d_{AC} + \underset{0}{X}_{DE} * d_{DE}}$$

$$\boldsymbol{d_{TOTAL} = d_{AB} + d_{AB} + d_{CE}}$$

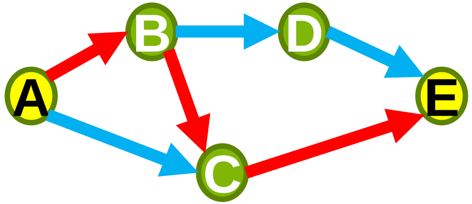
Formalización del camino más corto

$$d_{TOTAL} = X_{AB} * d_{AB} + X_{BC} * d_{BC} + X_{CE} * d_{CE} + $$
$$+ X_{BD} * d_{BD} + X_{AC} * d_{AC} + X_{DE} * d_{DE}$$

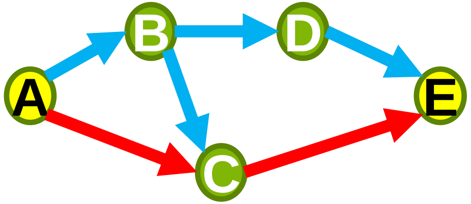
$X_{AB} = X_{BD} = X_{DE} = 1$
$X_{AC} = X_{BC} = X_{CE} = 0$



$X_{AB} = X_{BC} = X_{CE} = 1$
$X_{AC} = X_{BD} = X_{DE} = 0$

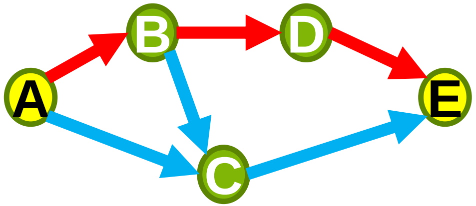


$X_{AC} = X_{CE} = 1$
$X_{AB} = X_{BC} = X_{BD} = X_{DE} = 0$

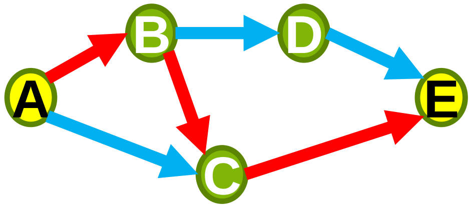


Formalización del camino más corto

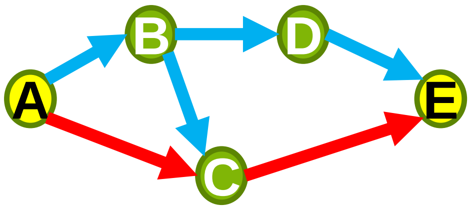
Cálculo del camino más corto:



$d_{TOTAL\,1}$



$d_{TOTAL\,2}$



$d_{TOTAL\,3}$

$$Min\,\{d_{TOTAL\,1}, d_{TOTAL\,2}, d_{TOTAL\,3}\}$$

Formalización del camino más corto

$$d_{TOTAL_k} = X_{AB} * d_{AB} + X_{BC} * d_{BC} + X_{CE} * d_{CE} + \\ + X_{BD} * d_{BD} + X_{AC} * d_{AC} + X_{DE} * d_{DE}$$

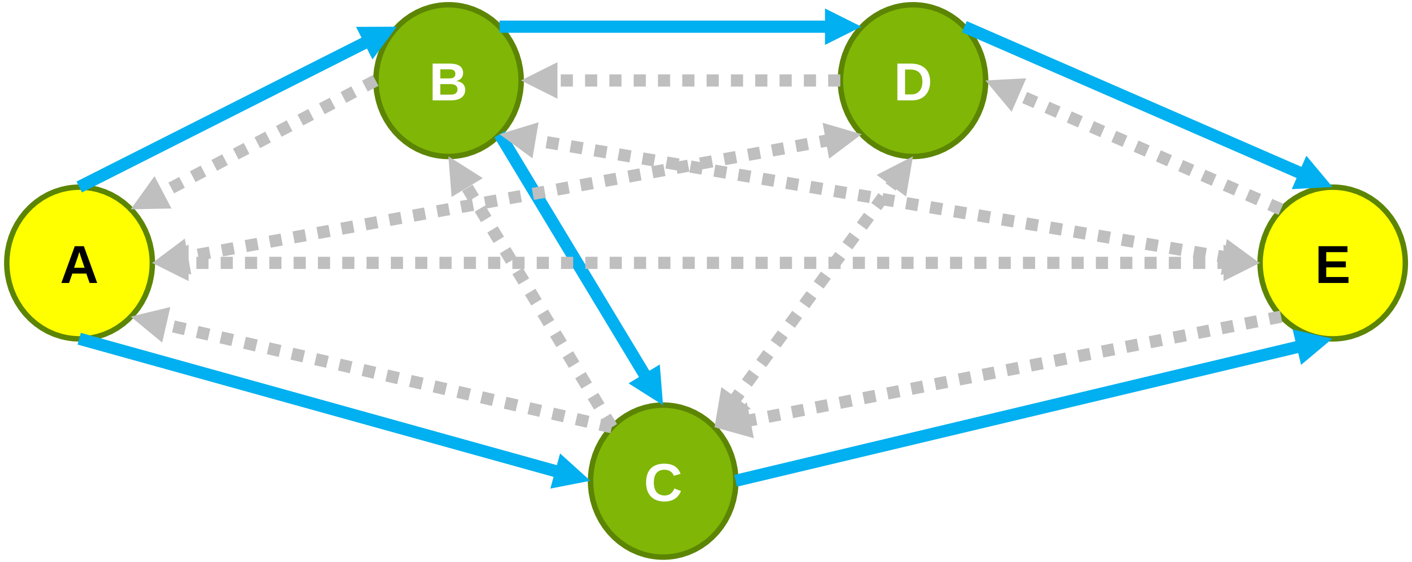
Generalización para encontrar un camino “k”

$$d_{TOTAL_k} = \sum_i \sum_j X_{ij} d_{ij}$$

¿Qué grafo se adapta a esta ecuación?

Formalización del camino más corto

Generalización del grafo sin ciclos:



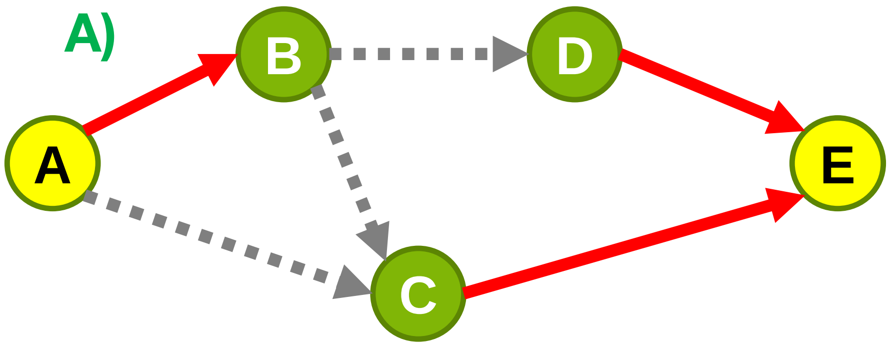
Arcos siempre apagados

Formalización del camino más corto

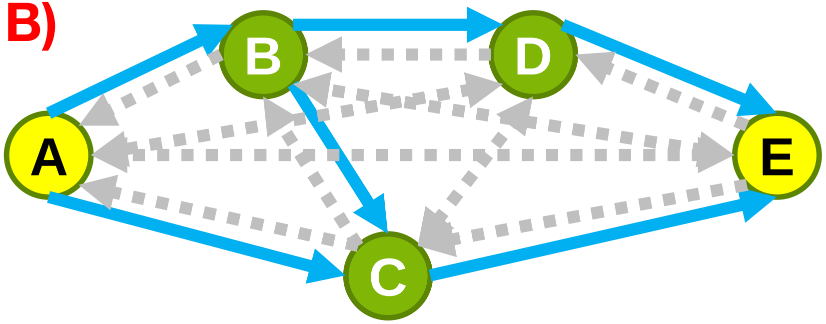
Problemas de la ecuación anterior:

- A) ¿Cómo explico al modelo qué es un camino?
- B) ¿Cómo le digo al modelo qué arcos existen?

A)



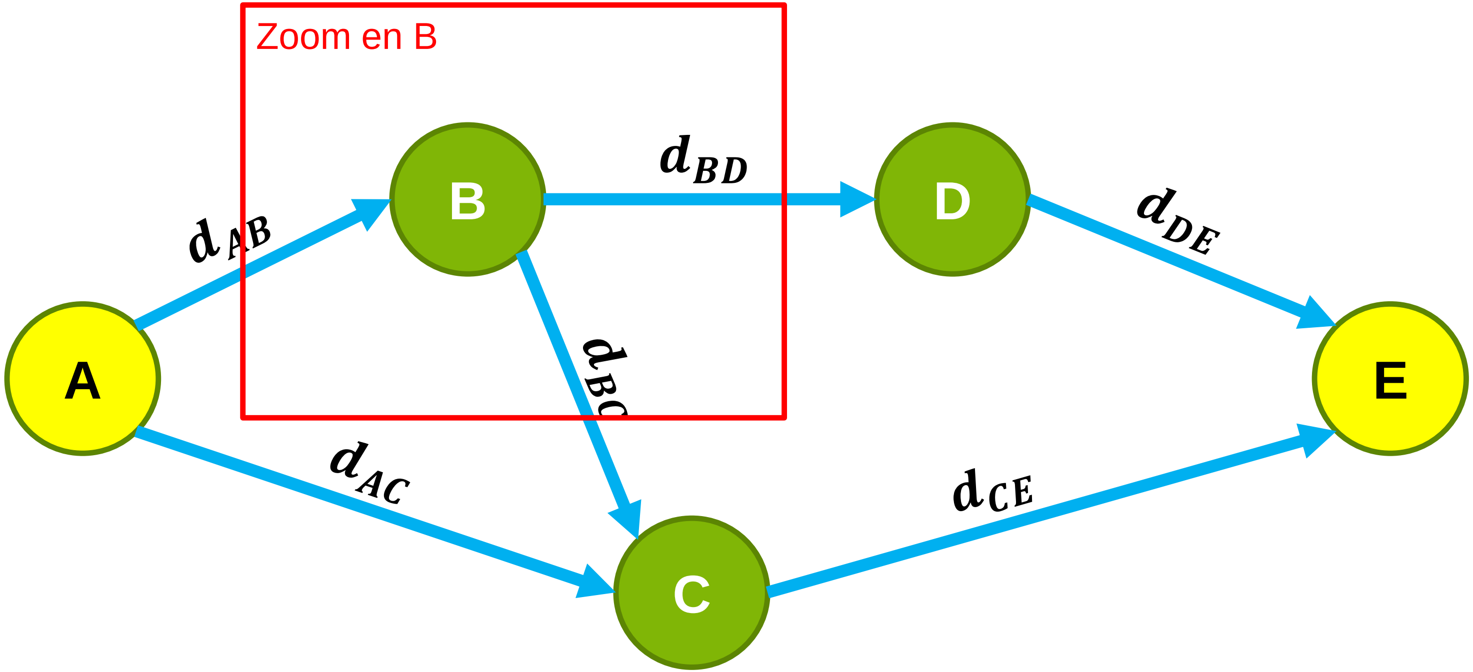
B)



La clave está en agregar: **Restricciones**

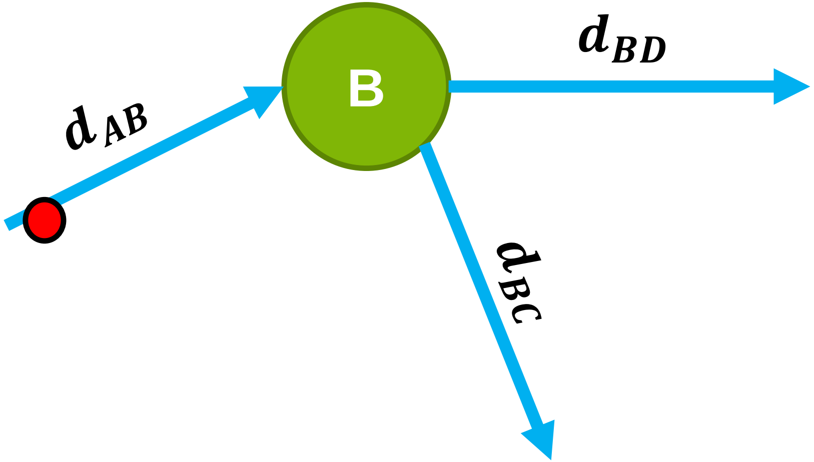
Formalización del camino más corto

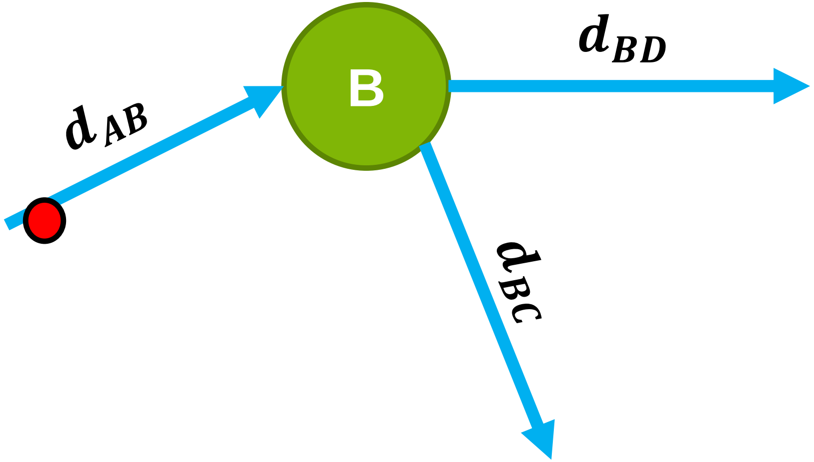
Resolvemos problema A



Formalización del camino más corto

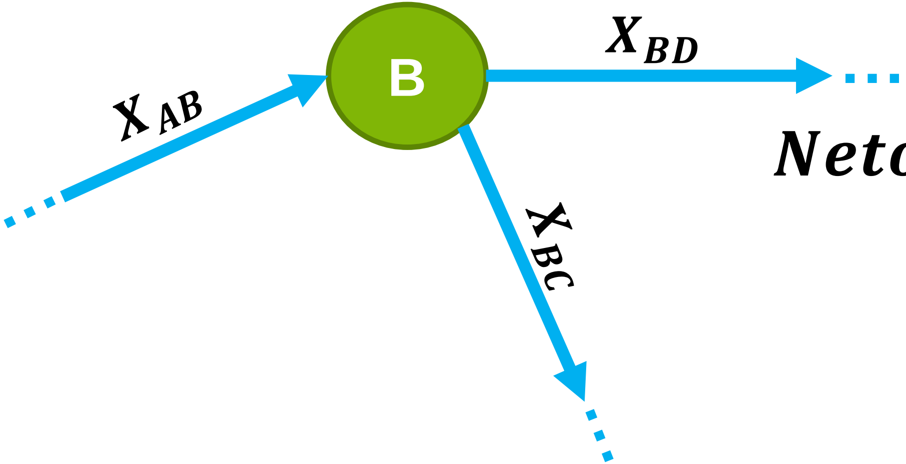
Posibilidades de un agente viajando por el camino:





Formalización del camino más corto

Entradas y salidas de B (1 persona viaja):



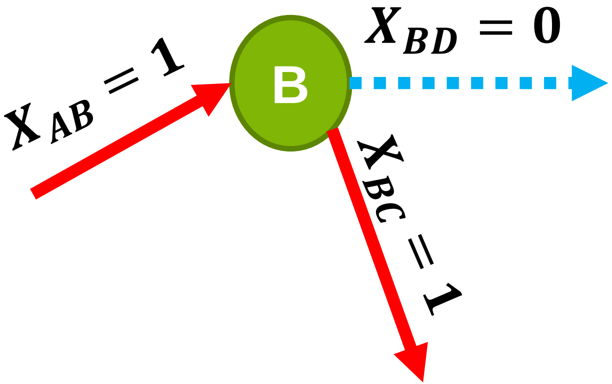
$$Neto_B = Salidas_B - Entradas_B$$

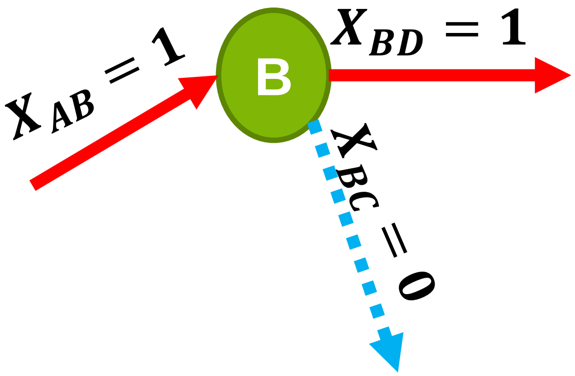
$$0 = Salidas_B - Entradas_B$$

$$0 = X_{BD} + X_{BC} - X_{AB}$$

Formalización del camino más corto

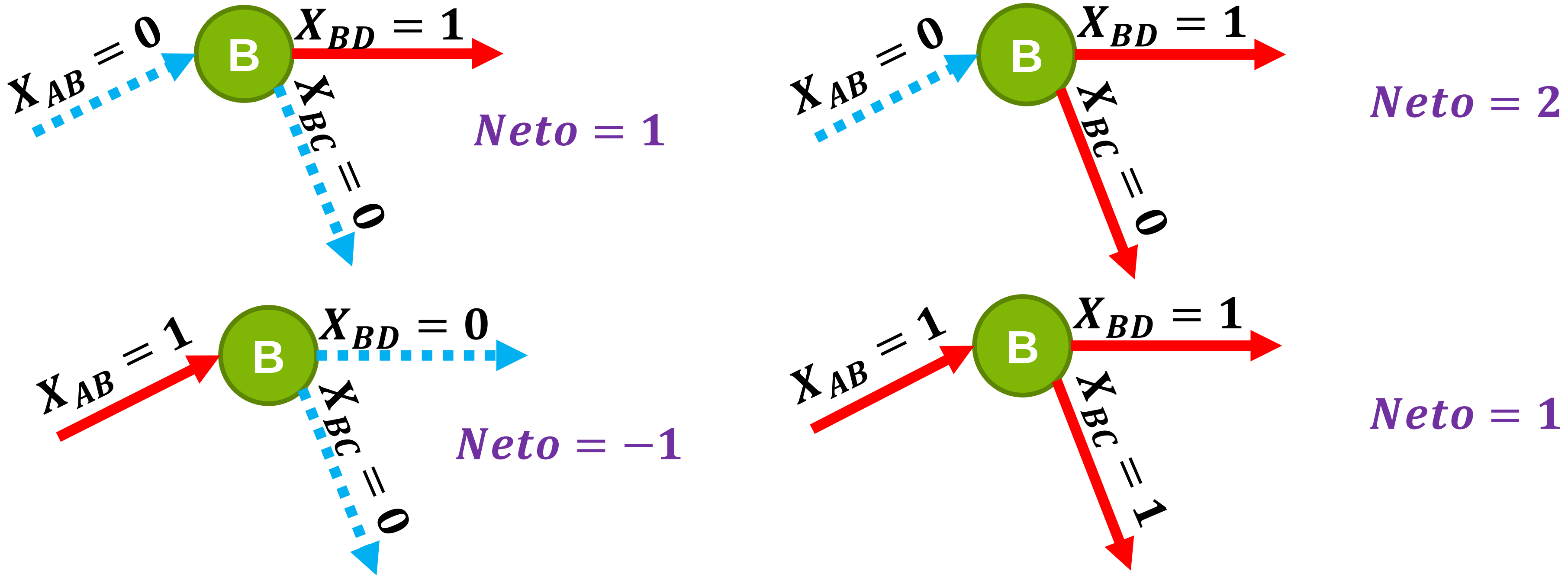
$$0 = Salidas_B - Entradas_B$$





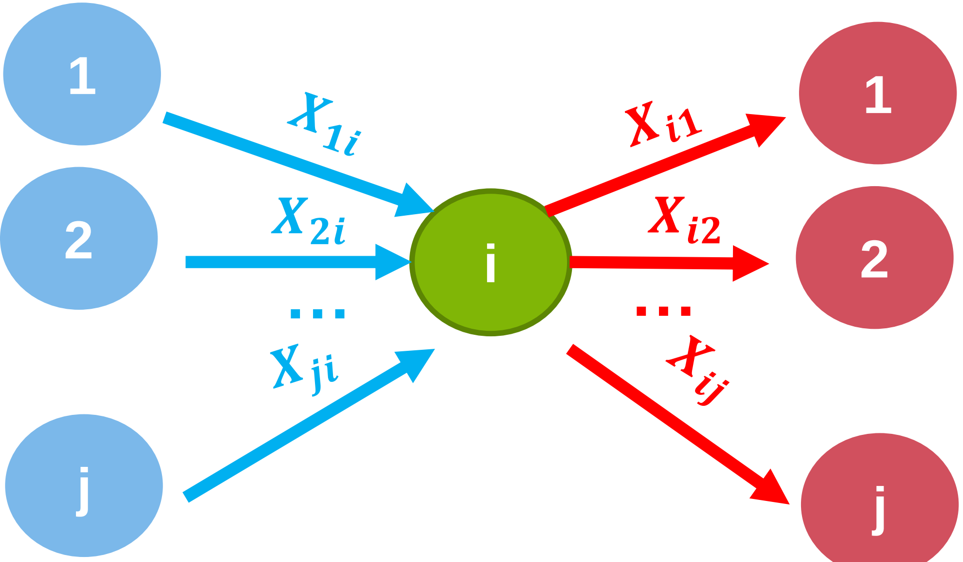
Formalización del camino más corto

$$0 \neq Salidas_B - Entradas_B \; (no\ se\ cumple\ restricción)$$



Formalización del camino más corto

Generalizamos la restricción



$$\mathbf{0} = \color{red}{Salidas_i} - \color{cyan}{Entradas_i}$$

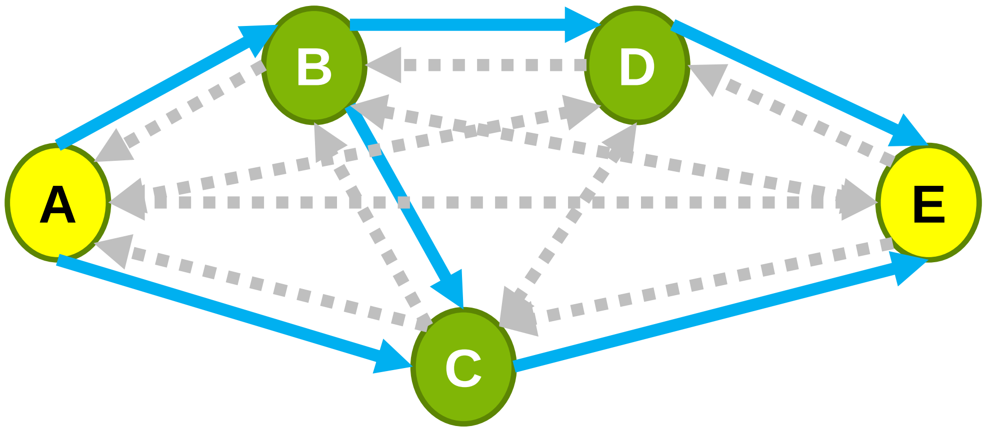
$$\mathbf{0} = \sum_{j} X_{ij} - \sum_{j} X_{ji} \qquad ; \ \forall i$$

¡Pero con esto surge el mismo **problema B!**

¿Cómo le digo qué arcos existen?

Formalización del camino más corto

Generalización del grafo sin ciclos:



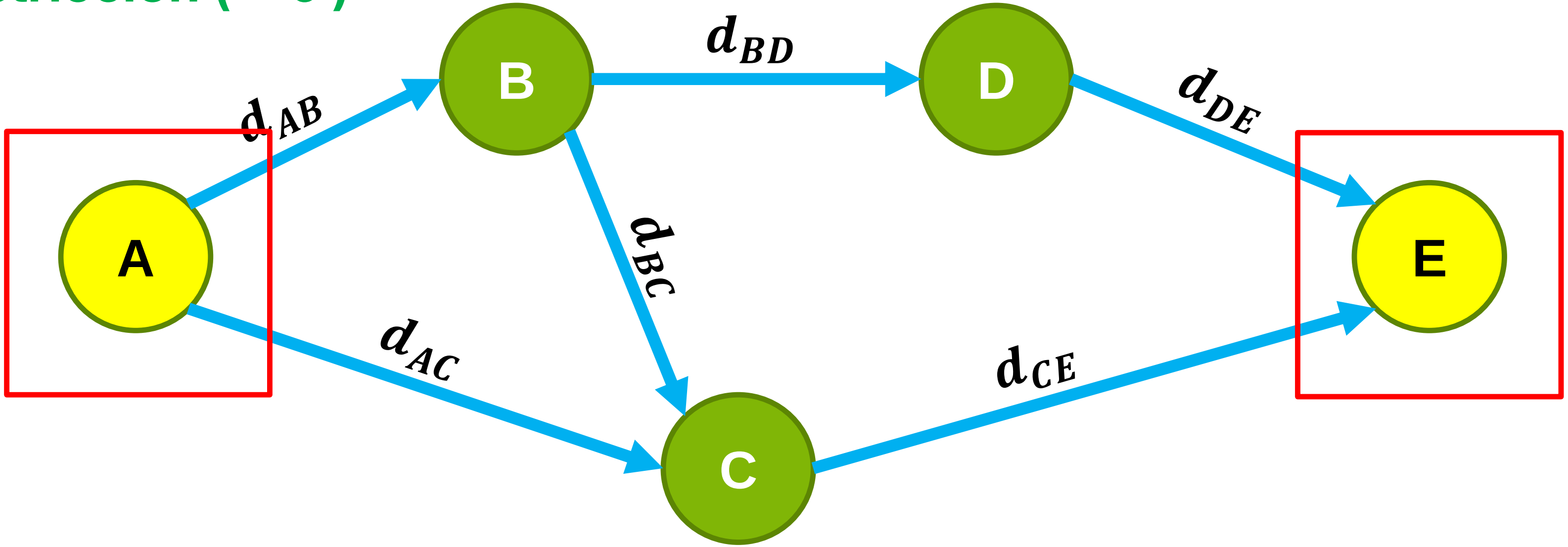
$$A = \begin{array}{c|ccccc} & A & B & C & D & E \\ \hline A & 0 & 1 & 1 & 0 & 0 \\ B & 0 & 0 & 1 & 1 & 0 \\ C & 0 & 0 & 0 & 0 & 1 \\ D & 0 & 0 & 0 & 0 & 1 \\ E & 0 & 0 & 0 & 0 & 0 \end{array}$$

$$0 = \sum_{j,\ ij \subset A} X_{ij} - \sum_{j,\ ji \subset A} X_{ji} \quad ; \forall i$$

Formalización del camino más corto

Un problema más:

Los nodos A (inicio) y E (final) no tienen forma de cumplir la restricción (= 0)



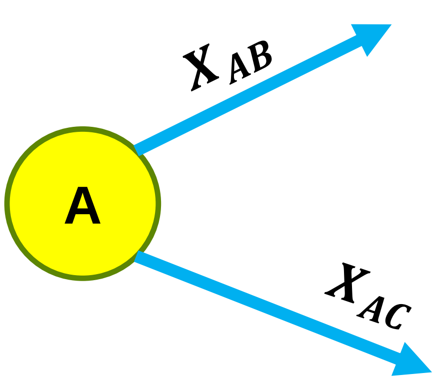
Formalización del camino más corto

$$\color{red}{1} = Salidas_A$$

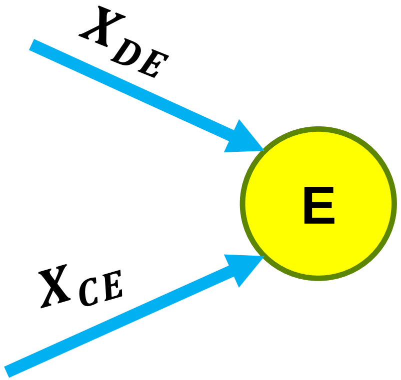
$$\color{red}{1} = X_{AB} + X_{AC}$$

$$\color{red}{-1} = Entradas_E$$

$$\color{red}{-1} = -X_{DE} - X_{CE}$$



Elegir solo un arco



1 = “Sale de A”

-1 = “Llega a E”

Formalización del camino más corto

Generalizamos la restricción

$$\boldsymbol{b_i} = \sum_{j,\ \ ij \subset A} \boldsymbol{X_{ij}} - \sum_{j,\ \ ji \subset A} \boldsymbol{X_{ji}} \quad ; \forall \boldsymbol{i}$$

$\boldsymbol{b_i} = 1$ si es inicio
$\boldsymbol{b_i} = -1$ si es final
$\boldsymbol{b_i} = \mathbf{0}$ si es intermedio

Formalización del camino más corto

$$\boldsymbol{Min} \sum_{i} \sum_{j} X_{ij} d_{ij}$$

Sujeto a la restricción:

$$b_i = \sum_{j,\ ij \subset A} X_{ij} - \sum_{j,\ ji \subset A} X_{ji} \quad ; \forall i$$

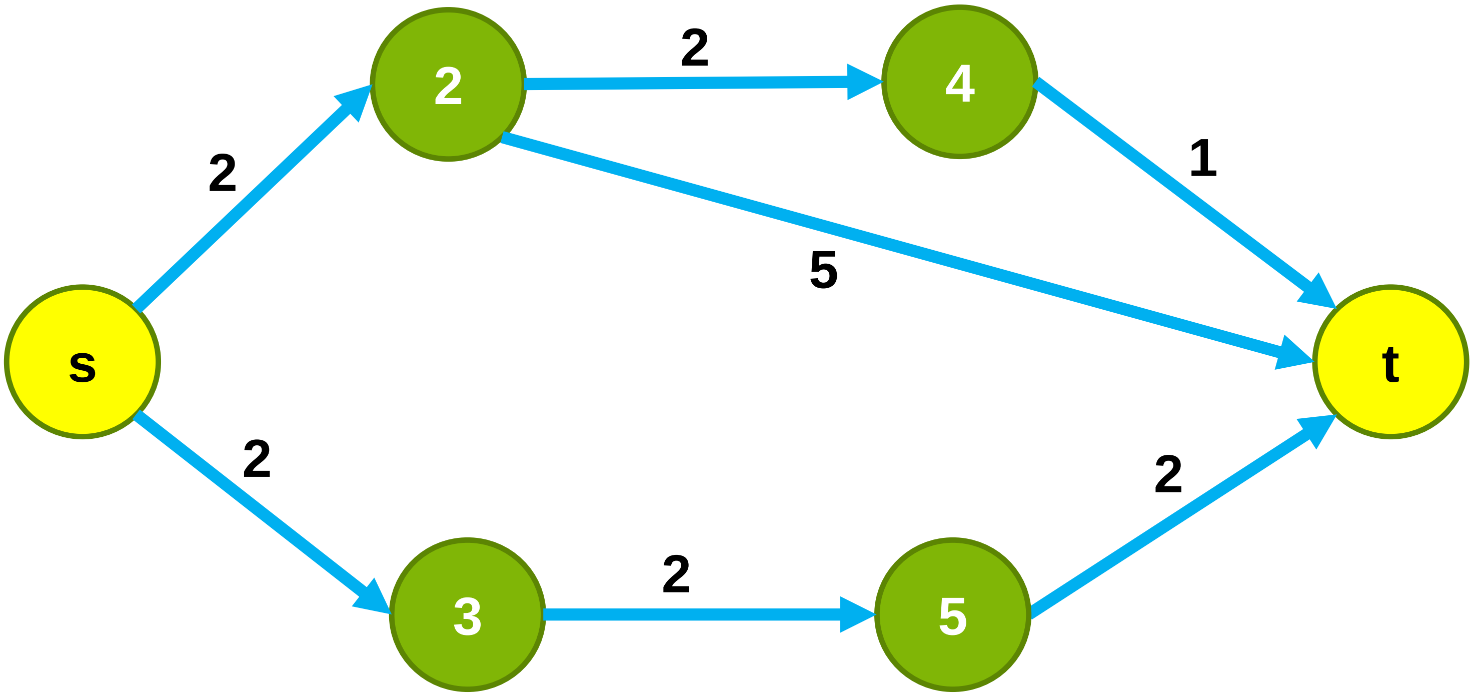
¡ Más adelante vamos a ver cómo resolver esto con programación matemática !

Resolución con algoritmos puntuales

Algoritmo de Dijkstra

- Encuentra siempre la solución óptima.
- La complejidad depende del problema. Pero es muy eficiente.
- No sirve si los pesos de los arcos **son negativos**

Algoritmo de Dijkstra



Algoritmo de Dijkstra

Pseudo código

Inicializar etiquetas de todos los nodos a infinito

Para cada nodo del grafo no explorado:

    nodo_i = nodo actual

    Para cada vecino del nodo_i:

        nodo_j = vecino actual

        nueva_etiqueta(**vecino_j**) = etiqueta(**nodo_i**) + distancia(**nodo_i**, **nodo_j**)

        Si nueva_etiqueta(**vecino_j**) < etiqueta(**vecino_j**)

            Actualizar etiqueta(**vecino_j**)

Algoritmo de Dijkstra

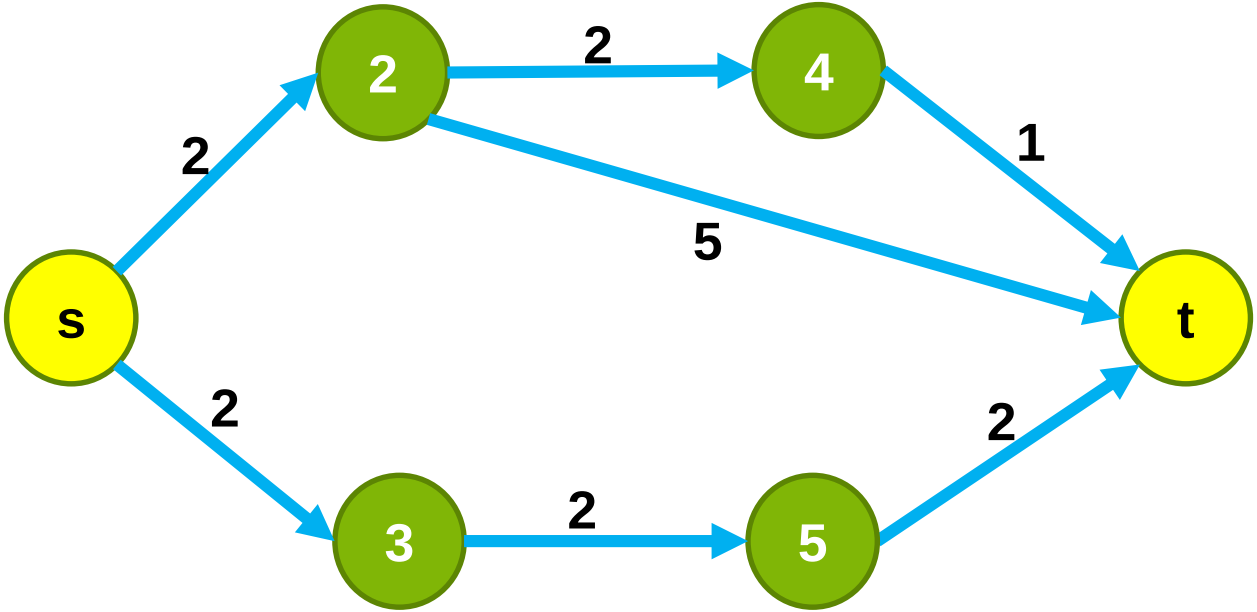


Tabla de precedencias

S	2	3	4	5	t

Algoritmo de Dijkstra

INICIALIZACIÓN

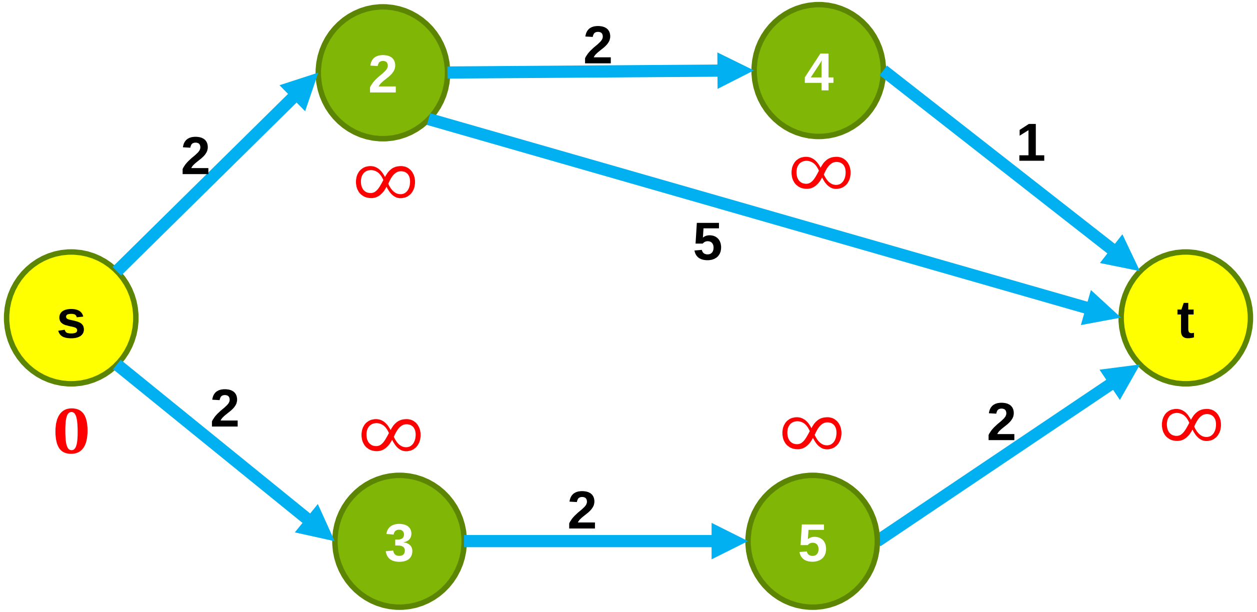


Tabla de precedencias

S	2	3	4	5	t

Algoritmo de Dijkstra

VECINOS DE “s”

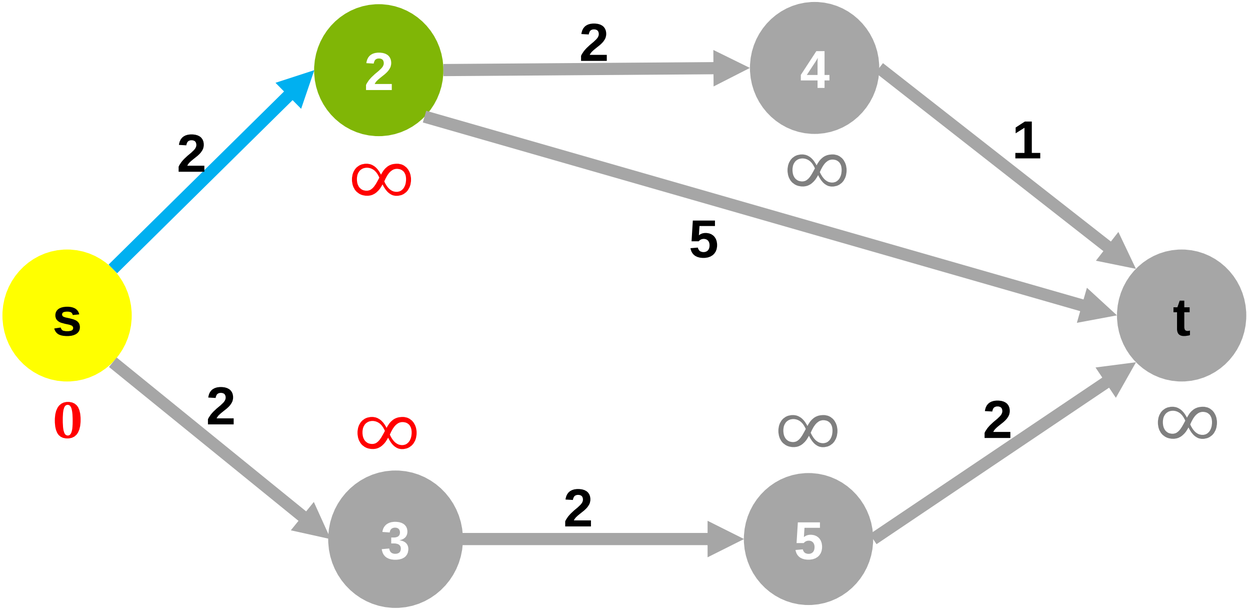


Tabla de precedencias

S	2	3	4	5	t

Algoritmo de Dijkstra

VECINOS DE “s”

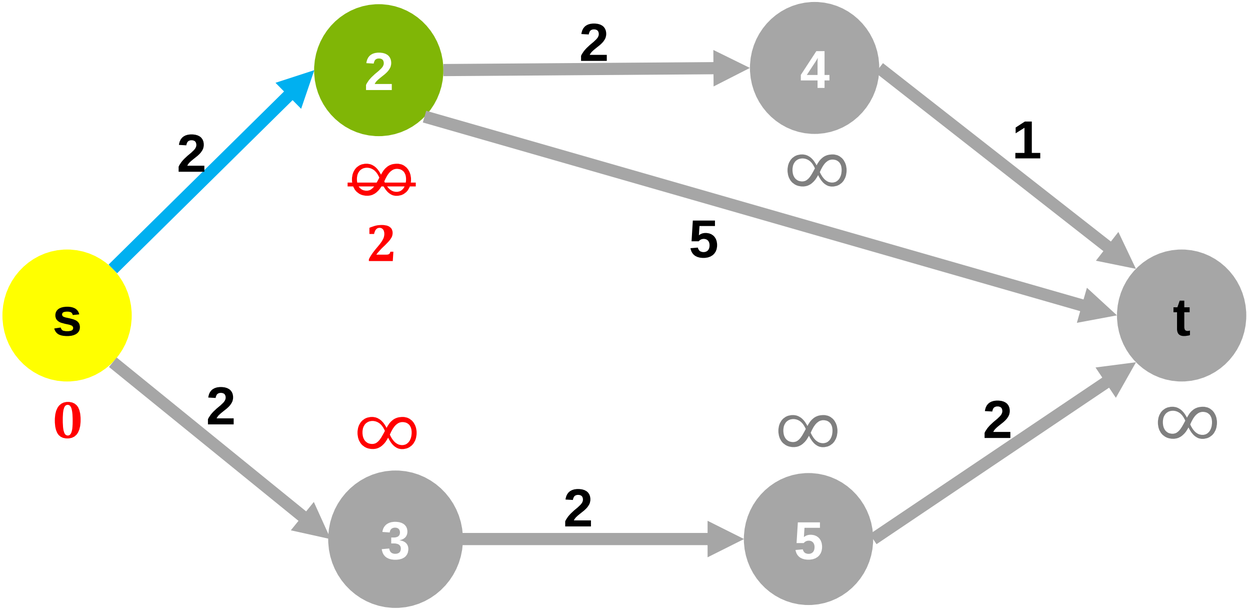


Tabla de precedencias

S	2	3	4	5	t
	s				

Algoritmo de Dijkstra

VECINOS DE “s”

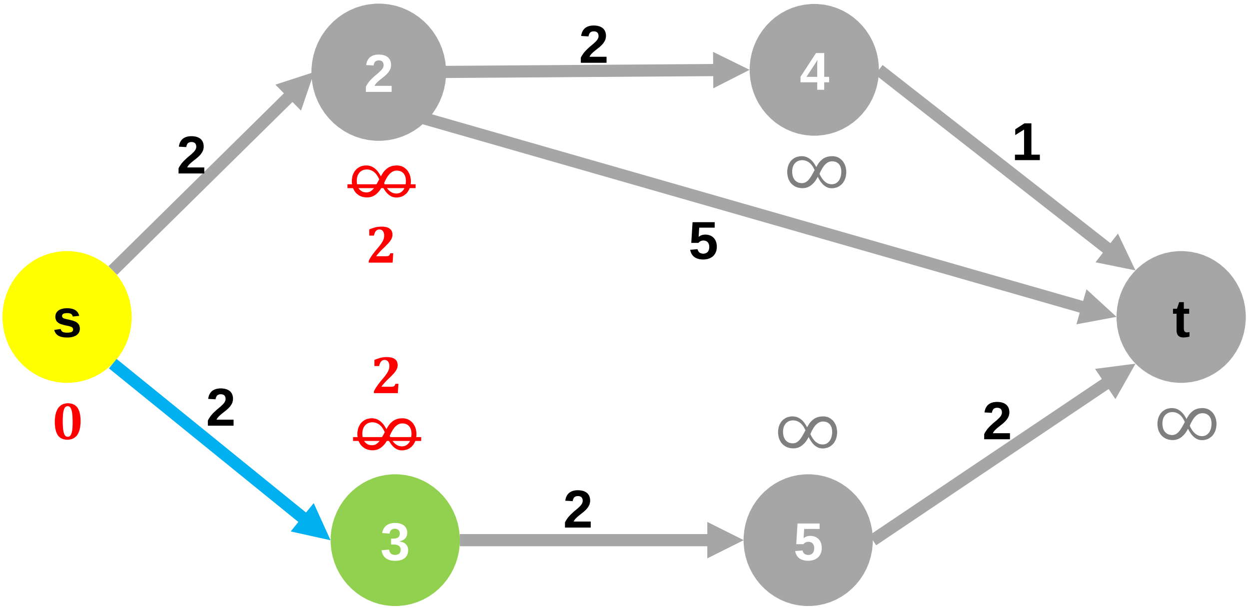


Tabla de precedencias

S	2	3	4	5	t
	s	s			

Algoritmo de Dijkstra

S ya explorado

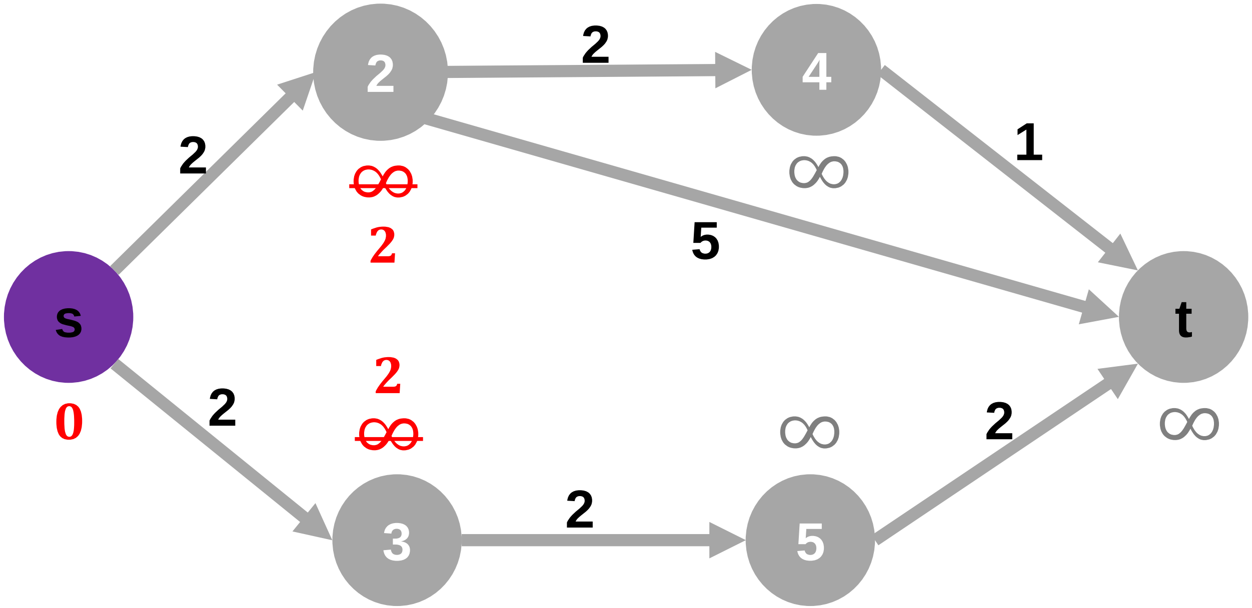


Tabla de precedencias

S	2	3	4	5	t
	s	s			

Algoritmo de Dijkstra

VECINOS DE “2”

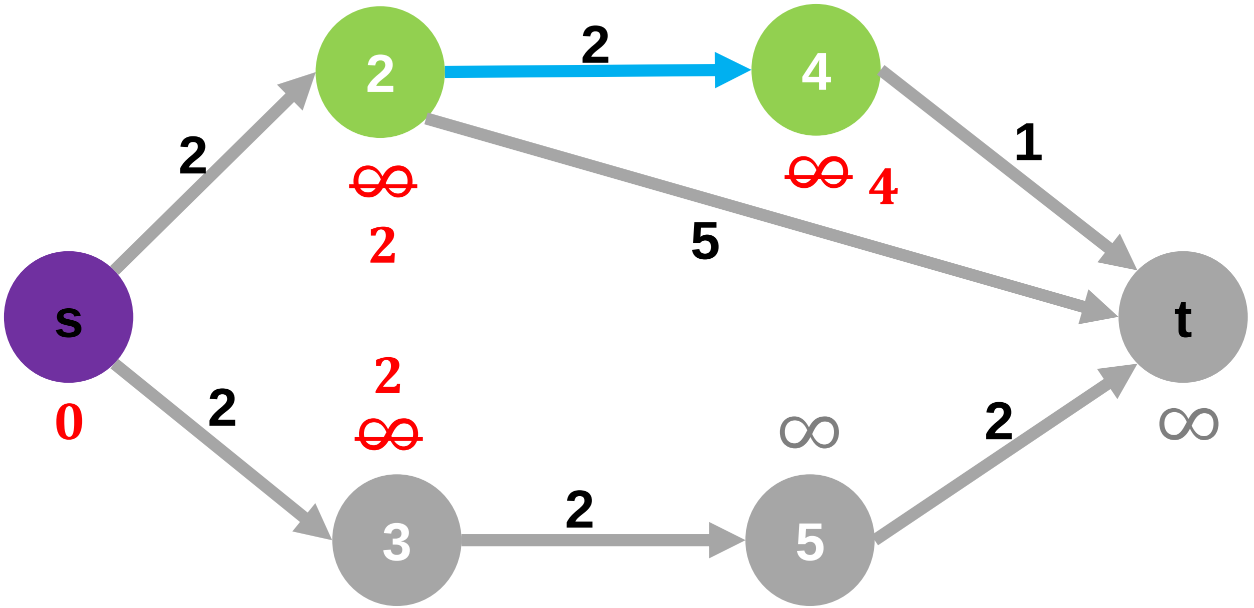


Tabla de precedencias

S	2	3	4	5	t
	s	s	2		

Algoritmo de Dijkstra

VECINOS DE “2”

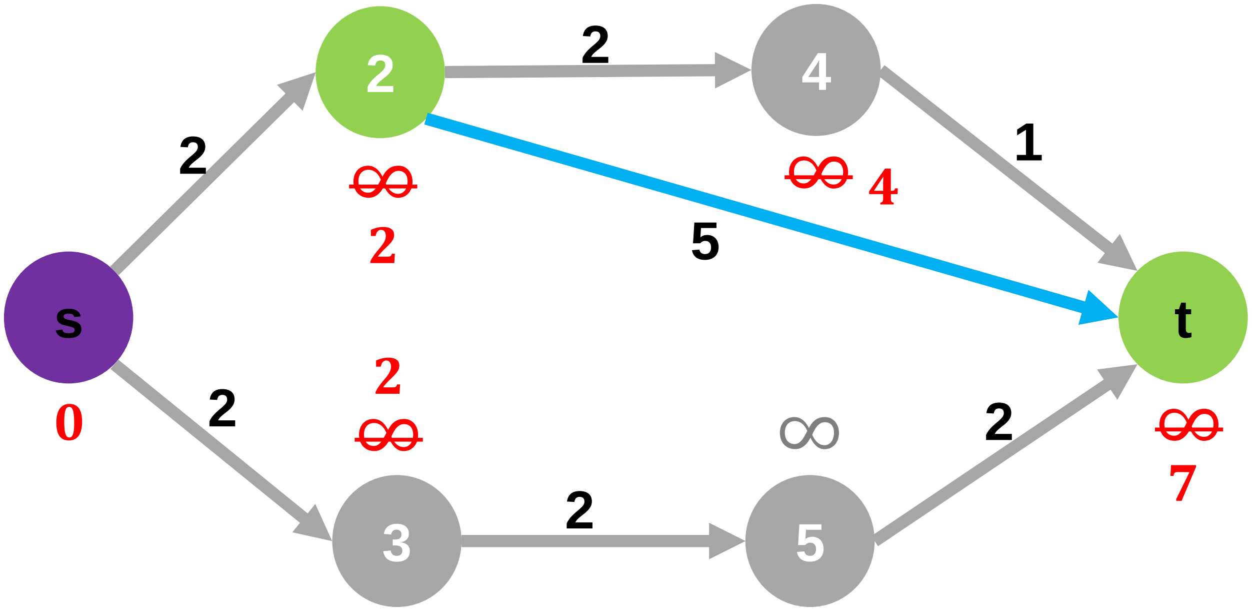


Tabla de precedencias

S	2	3	4	5	t
	s	s	2		2

Algoritmo de Dijkstra

“2” ya explorado

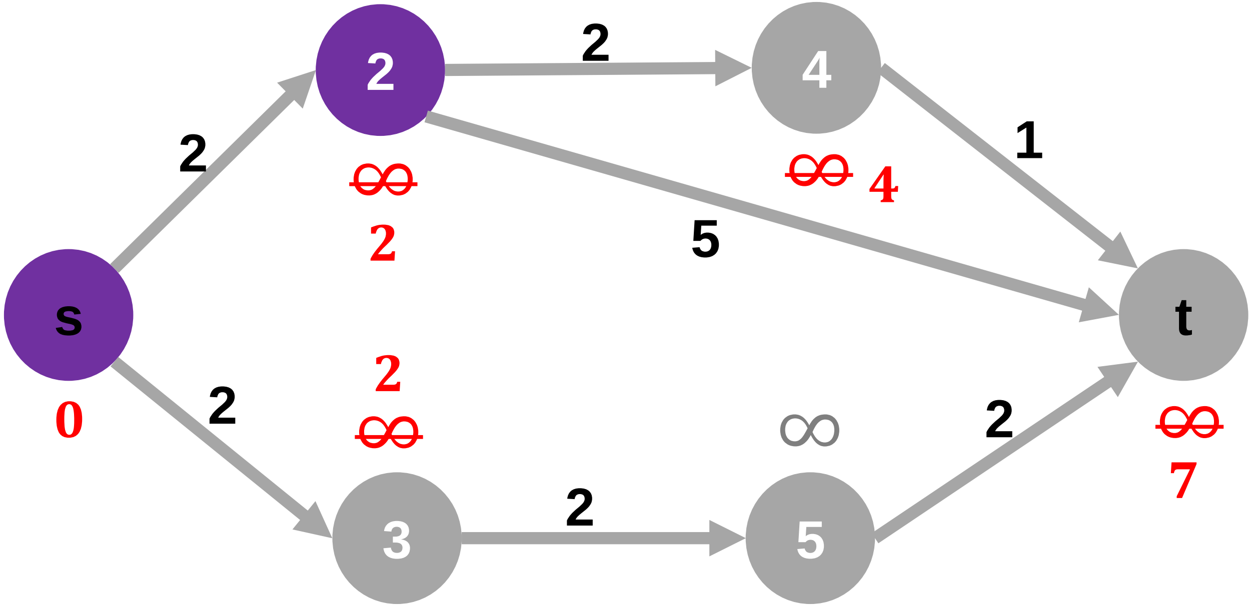


Tabla de precedencias

S	2	3	4	5	t
\	s	s	2		2
\					
\					
\					
\					

Algoritmo de Dijkstra

VECINOS DE "3"

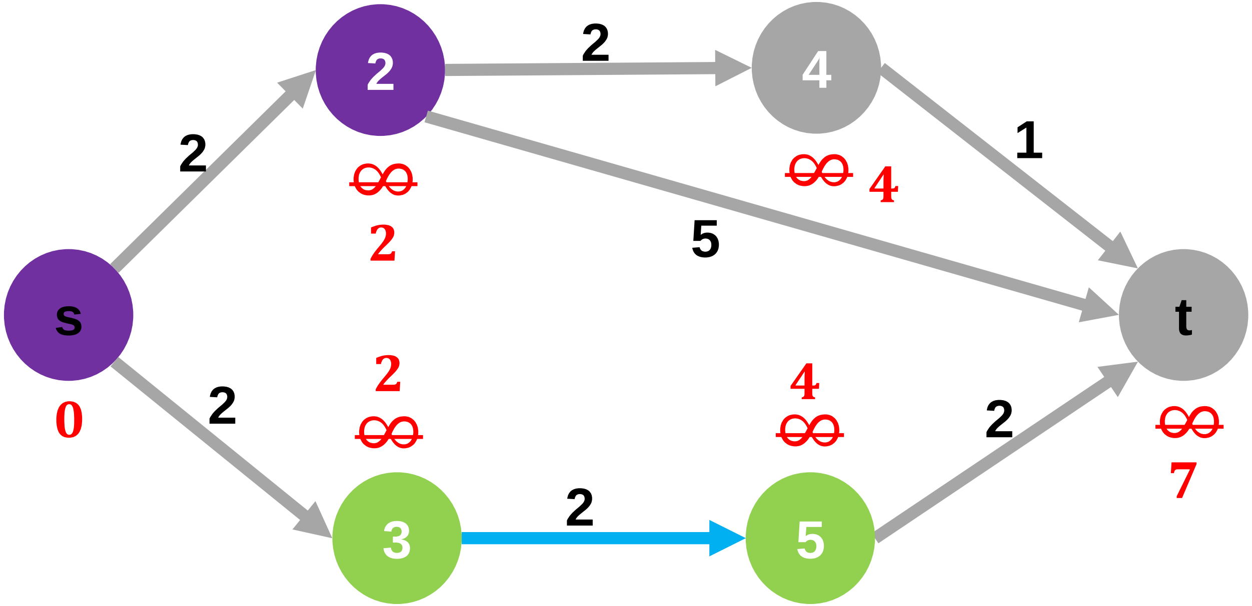


Tabla de precedencias

S	2	3	4	5	t
\	s	s	2	3	2
\					
\					
\					
\					

Algoritmo de Dijkstra

“3” ya explorado

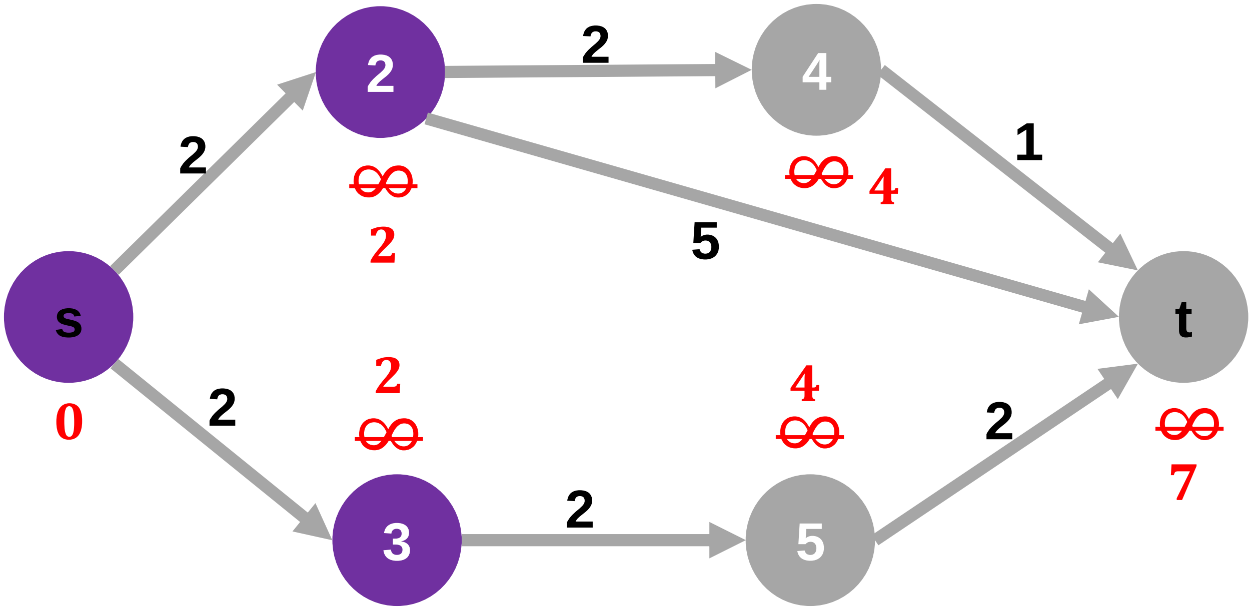


Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	2

Algoritmo de Dijkstra

VECINOS DE “4”

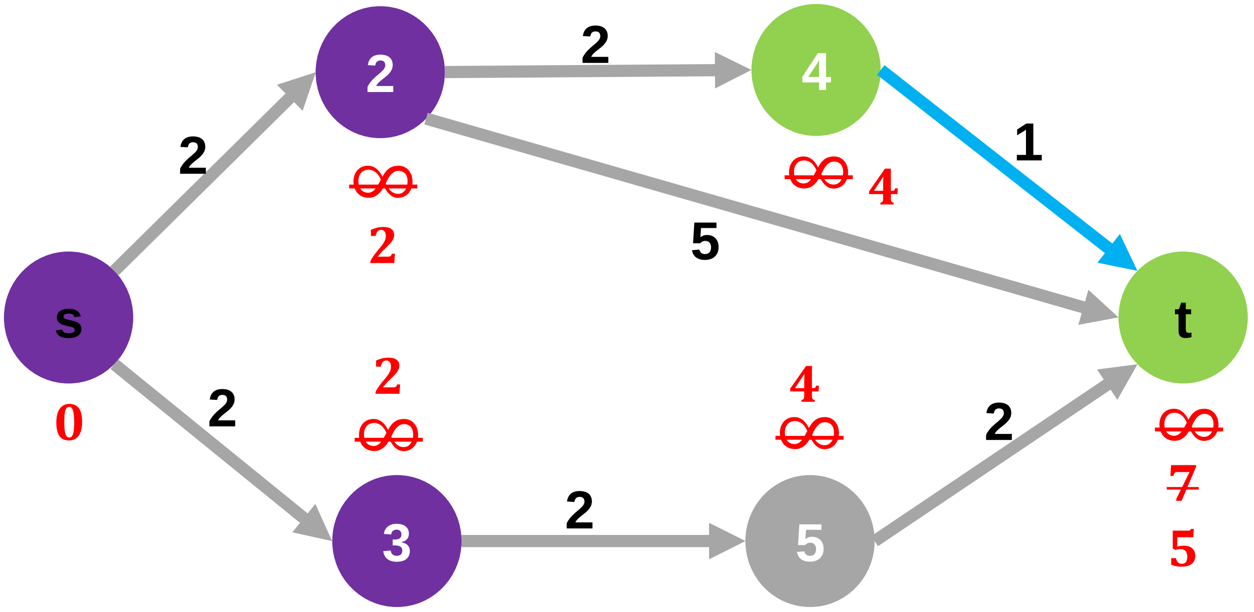


Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	~~2~~
					4

Algoritmo de Dijkstra

“4” ya explorado

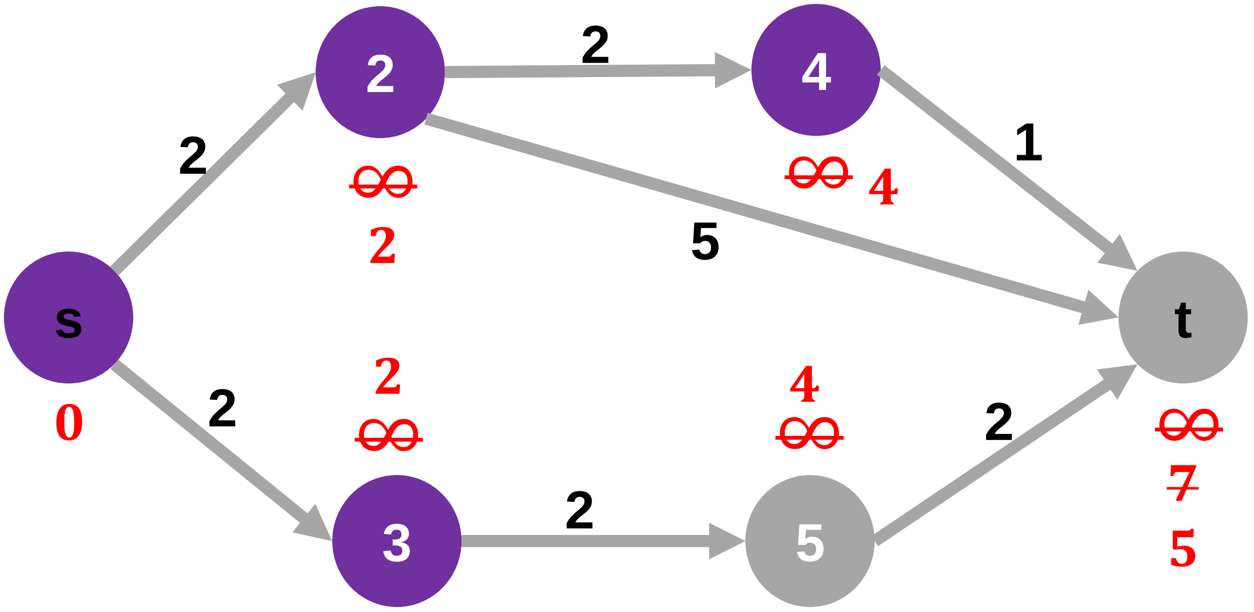


Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	~~2~~
					4

Algoritmo de Dijkstra

VECINOS DE "5"

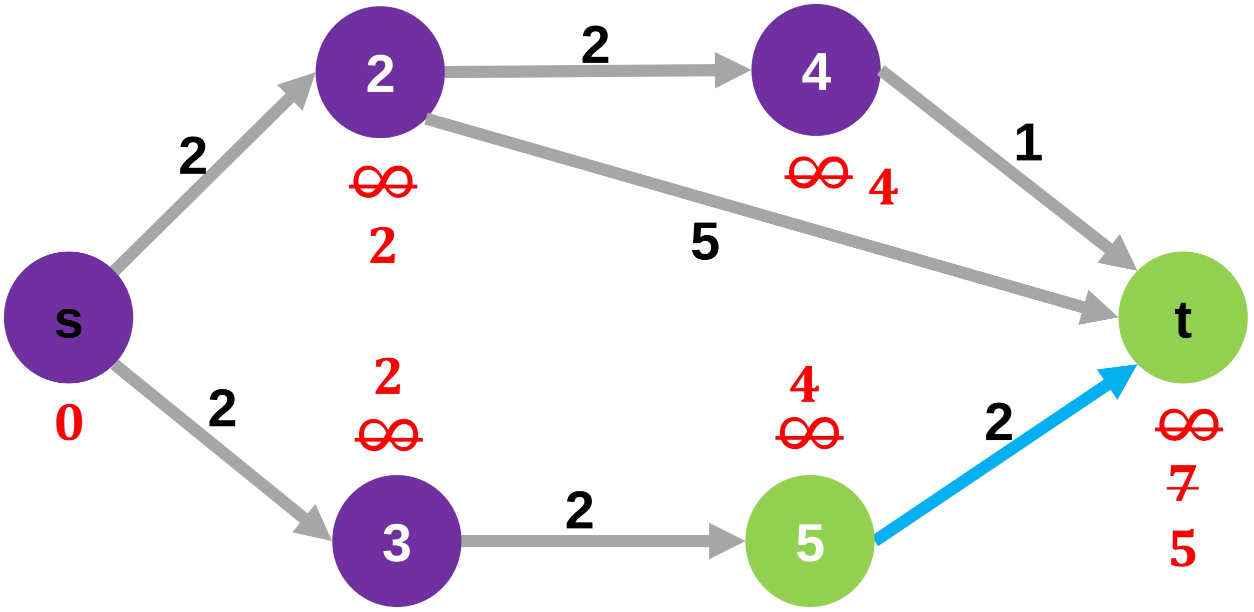


Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	~~2~~
					4

6 No es menor que 5, no actualiza!

Algoritmo de Dijkstra

“5” ya explorado

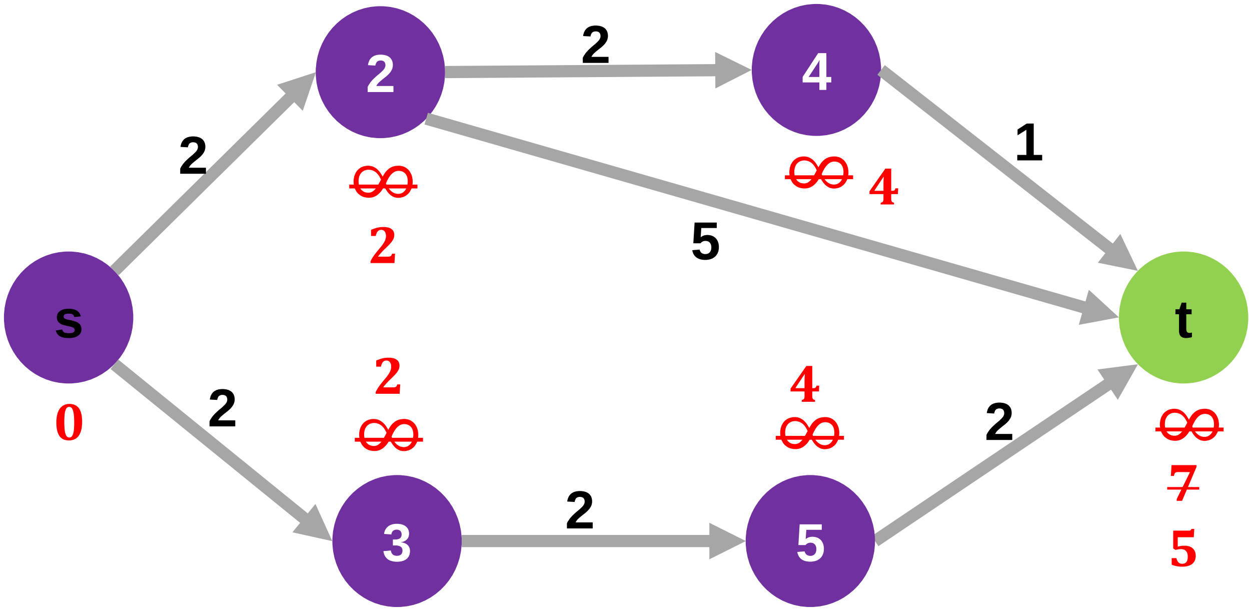


Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	~~2~~
					4

Algoritmo de Dijkstra

Nodo “t” final, sin vecinos. Fin del algoritmo

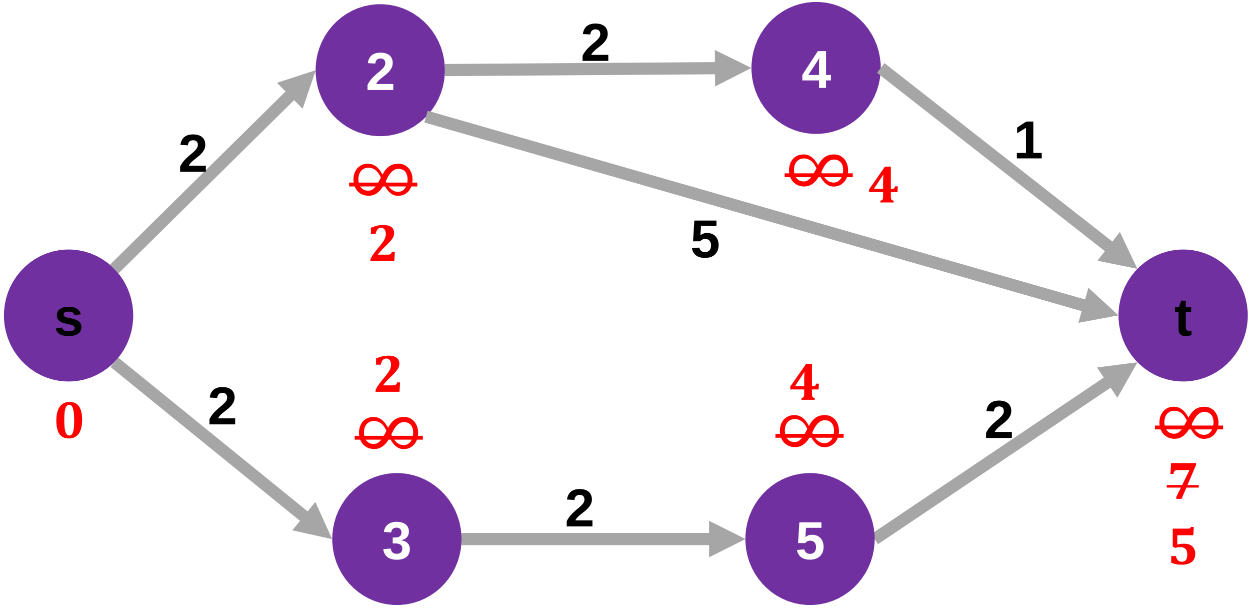
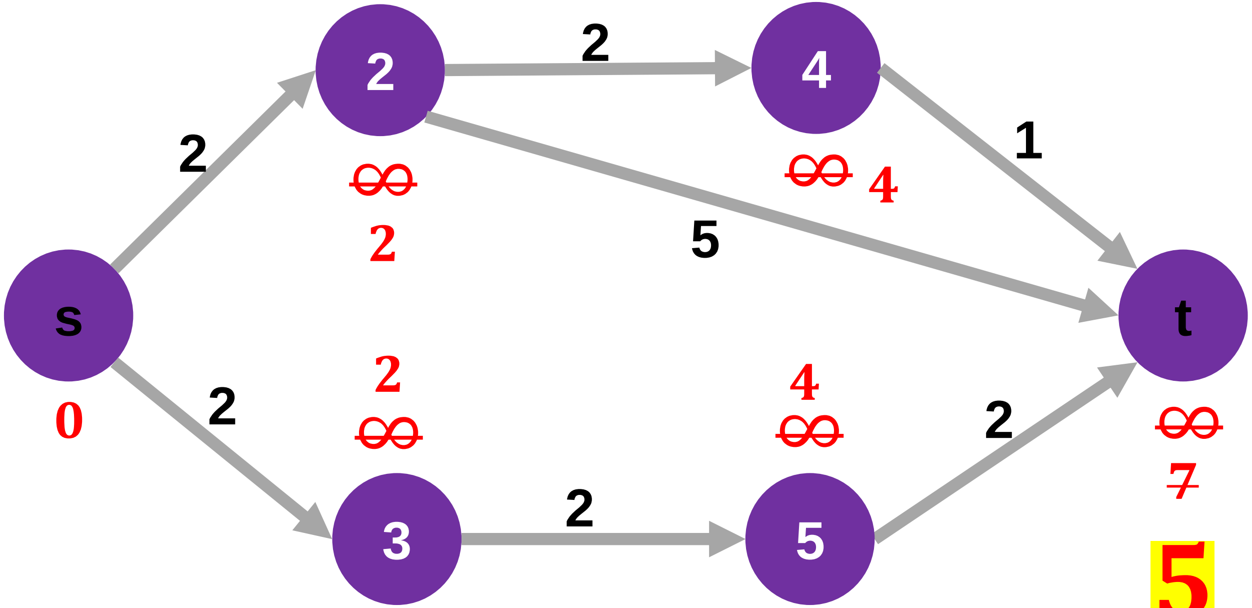


Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	~~2~~
					4

Algoritmo de Dijkstra



5

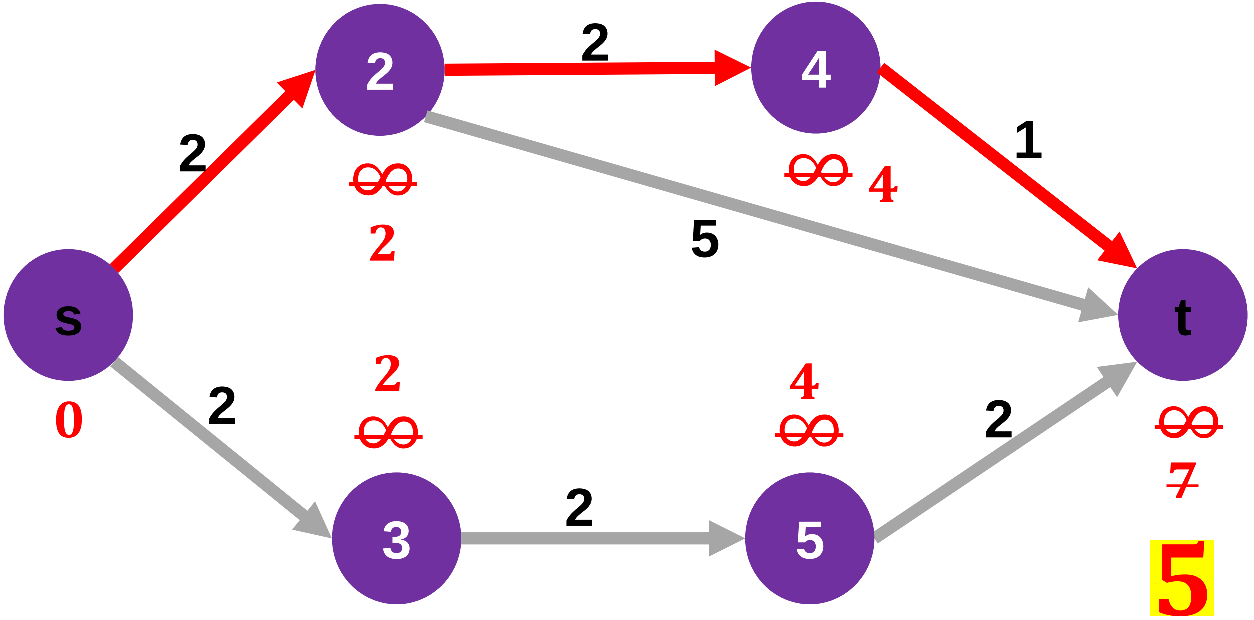
Distancia mínima s-t

Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	~~2~~
					4

Algoritmo de Dijkstra

Reconstrucción del camino más corto



Distancia mínima s-t

Tabla de precedencias

S	2	3	4	5	t
	s	s	2	3	~~2~~
					4

Resolución con algoritmos puntuales

Comentarios adicionales:

- Dijkstra puede resolver todos los caminos más cortos desde cualquier nodo.
- No tiene heurística.
- Para acelerarlo, agregar heurística. Ej: Algoritmo A*