III. El algoritmo de Dijkstra:

Def: Un <u>grafo dirigido con pesos</u> es un grafo dirigido G junto con una función

$$\ell : E(G) \longrightarrow \mathbb{R}$$

que a cada arista $(u,v) \in E(G)$ le asigna un costo $\ell(u,v)$.

Obs: Los costos $\ell(u,v)$ típicamente se interpretan como "longitudes" o "tiempos" y típicamente, (aunque no siempre) son números positivos.

Ejemplo: En googlemaps cada localización es un vértice de un grafo

$\ell(u,v) =$ "Tiempo promedio de desplazamiento"

la novedad de Waze fue "aprender" la estructura de ese grafo.

Ambos resuelven el mismo problema:

PROBLEMA: Dados dos vértices s y t de un grafo con pesos (G, ℓ) encuentre:

(i) La mínima <u>distancia con pesos</u> $d_\ell(s,t)$ desde s hasta t y

(ii) Una ruta óptima (de peso mínimo).

El extraordinario algoritmo de Dijkstra
(Edsger Dijkstra 1956 (9?))

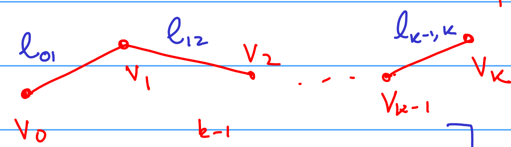
- Matemático y físico
- Primer programador de Holanda
- Ganó el premio Turing en 1972
- Muere en 2002
- Inventó la recursión mediante un stack...

Ver biografía en wikipedia...

Página siguiente...

Si (G, ℓ) es un grafo con pesos y

Def: $P = (v_0, v_1, \ldots, v_k)$ es un camino en G



$$\ell(P) := \underbrace{\sum_{i=0}^{k-1} \ell(v_i, v_{i+1})}_{\text{Peso del camino } P}$$

Si $u, v \in V(G)$ definimos

$$\underbrace{d_\ell(u,v)}_{} = \min \left\{ \ell(P) : \begin{array}{l} P \text{ es un camino} \\ \text{desde } u \text{ hasta } v \\ \text{en } G \end{array} \right\}$$

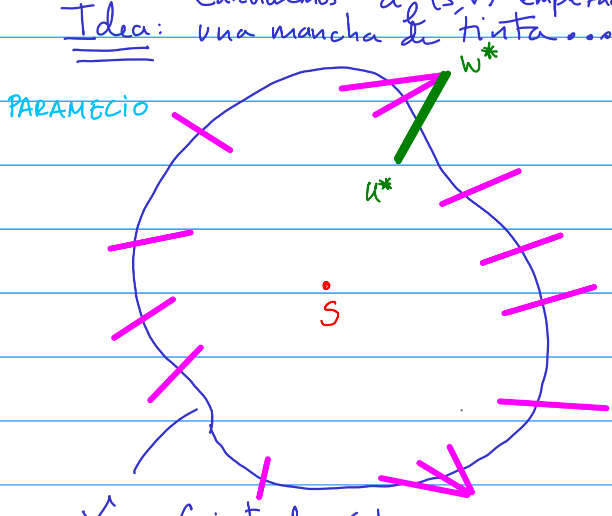
Peso mínimo de un camino desde u hasta v en G

"distancia con pesos de u a v".

El algoritmo de Dijkstra fija un vértice inicial s y calcula $d_\ell(s, v)$ para todo vértice $v \in V(G)$.
ASUMIENDO que los pesos $\ell(a,b) \geq 0$ para toda arista $(a,b) \in E(G)$.

Iniciamos en s y vamos "creciendo" lentamente alrededor de s anotando los mínimos costos disponibles

__Idea:__ Calcularemos $d_\ell(s,v)$ empezando en s y yendo hacia afuera, como una mancha de tinta...



X = Conjunto de vértices x donde <u>conocemos</u> $d_\ell(s,x)$

¿Cómo crece X?

Buscamos una arista (u^*, w^*) con:

(1) $u^* \in X$, $w^* \notin X$

(2)

$$d_\ell(s,u^*) + \ell(u^*, w^*)$$
$$\|$$
$$\min \{ d_\ell(s,a) + \ell(a,b) : a \in X, b \notin X \}$$

Hacemos crecer X $X \leftarrow X \cup \{w^*\}$

y definimos $\varphi(w^*) = \varphi(u^*) + \ell(u^*, w^*)$

Veremos que $\boxed{\varphi(w^*) = d_\ell(s, w^*)}$

Más concretamente, ALGORITMO DE DIJKSTRA:

INPUT: Grafo G con pesos $\ell \geq 0$, vértice de inicio s.

Output: diccionario φ con $\varphi[v] = d_\ell(s, v)$

INICIALIZACIÓN:

$X = [s]$

$\varphi[s] = 0$

Ejecución:

while hay al menos una arista (a,b) $a \in X$, $b \notin X$

(1) $(u^*, w^*) = \underset{\substack{(a,b) \in E(G) \\ a \in X \\ b \notin X}}{\operatorname{argmin}} \left[\varphi(a) + \ell(a,b)\right]$

(2) $X.\text{append}(w^*)$

$\varphi(w^*) := \varphi(u^*) + \ell(u^*, w^*)$

Teorema [Dijkstra, 1956] Si $\ell \geq 0$

En todo paso del algoritmo φ está definida en los vértices $v \in X$ y para ellos tenemos que $\varphi(v) = d_\ell(s, v)$

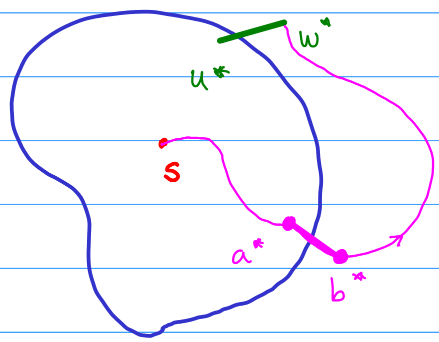
Dem: Demostraremos la siguiente afirmación por inducción en j:

$\Lambda(j) =$ "Para todo grafo con pesos $\ell \geq 0$ G, todo vértice inicial s y toda ejecución del algo de Dijkstra con $|X| \leq j$ tenemos $\varphi(v) = d_\ell(s, v)$"

Base: $j=1$ ✓ pues es la inicialización del algo.

Ind: Supongamos que $|X| = j+1$ y sea (u^*, w^*) la arista que nos llevó a incluir el último elemento de X.

Sea $X' = X \setminus \{w^*\}$. Como $|X'| = j$ por inducción tenemos



X'

$\varphi(x) = d_\ell(s,x) \quad \forall x \in X'$

Claim:

$\varphi(w^*) = d_\ell(s, w^*)$

Vamos a mostrar que cualquier otro camino de s hasta w^* tiene más peso.

Suponga que $\underbrace{s = v_0, v_1, \ldots, v_{k-1}, v_k}_{P} = u^*$ es un camino. (—). El camino P inicia dentro de X' y termina afuera de X' así que en algún momento debe salir. Sea (a^*, b^*) la primera vez que sale, es decir $a^* \in X'$, $b^* \notin X'$.

Como $\underbrace{\ell \geq 0}$ tenemos

por lo que sabemos de inducción

$$\ell(P) \geq \ell(a^*, b^*) + d_\ell(s, a^*) =$$

$$= \ell(a^*, b^*) + \varphi(a^*) \geq \ell(u^*, w^*) + \varphi(u^*)$$

por la manera como Dijkstra escoge arista

Como P es un camino cualquiera de s a w^* concluimos

$$d_\ell(s,w^*) \geq \ell(u^*,w^*) + \varphi(u^*) \quad \text{I}$$

Por inducción, como $u^* \in X'$ sabemos que $\varphi(u^*) = d_\ell(s,u^*)$ luego hay un camino Q de s a u^* con peso $\varphi(u^*)$ y concatenándolo con (u^*,w^*) al final hemos construido un camino con peso $\ell(u^*,w^*) + \varphi(u^*)$ y por eso

$$\ell(u^*,w^*) + \varphi(u^*) \geq d_\ell(s,w^*) \quad \text{II}$$

Combinando I y II concluimos

$$\varphi(w^*) = d_\ell(s,w^*).$$

Obs: Sea T el grafo con vértices $V(G)$ y $E(T) =$ "Aristas verdes (u^*,w^*) descubiertas en cada paso del algoritmo"

entonces T no tiene ciclos y todo vértice v admite ≤ 1 camino de s a v en T

ese camino es de peso mínimo para G

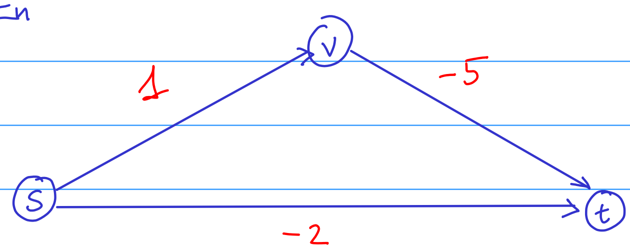
así que Dijkstra no solo encuentra la distancia mínima sino también la ruta óptima.

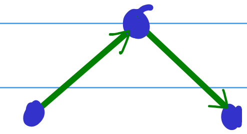
Es natural preguntarse si la restricción de que $\ell \geq 0$ que usamos fuertemente en la demostración de validez es realmente necesaria.

Una primera idea es preguntar si podríamos reducir el problema al caso positivo, sumándole una constante K a los pesos para volverlos positivos. ESTA IDEA NO FUNCIONA, COMO MUESTRA EL SIGUIENTE... !

... EJEMPLO :

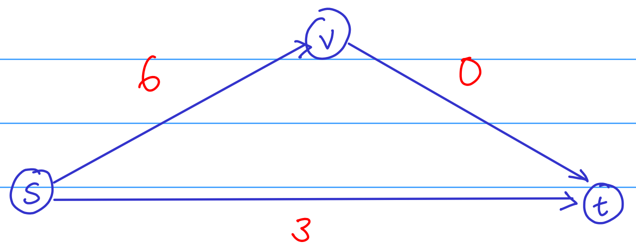
En

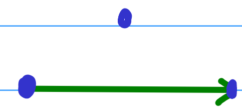




tiene peso $-4 < -2$ luego es el de peso mínimo

Si le sumamos 5 a todos para que sea ≥ 0 queda:



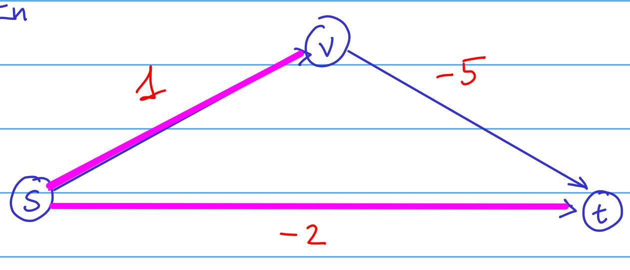


El camino más corto CAMBIA

El problema con nuestra "Reducción" es que el camino de longitud mínima no siempre tiene el mínimo número de aristas luego nuestro truco de sumar K distorsiona el problema

- Si usamos el algoritmo de Dijkstra en este ejemplo ¿qué sucede?

En



$X = \{s\}$

Buscamos en las aristas marcadas la que minimiza

$0 + 1 = 1$

$0 + (-2) = -2 \quad \Rightarrow \quad X = \{s, t\}$ $[\varphi(t) = -2]$ EQUIVOCADO!!

Así que si HAY ARISTAS CON PESO NEGATIVO el algoritmo puede producir resultados ERRÓNEOS.