ffm

Algoritmo de Dijkstra

(IdeO otoño 2020, Moyra Núñez)

Sea u_i la distancia más corta del nodo origen al nodo i. Sea $d_{ij} \geq 0$ la longitud del arco entre los nodos i, j. La etiqueta para un nodo j es la siguiente:

$$[\underbrace{u_j}_{\text{distancia más corta del nodo } j \text{ al origen}}, \underbrace{i}_{\text{nodo inmediato anterior al nodo } j}] = [\underbrace{u_i + \overbrace{d_{ij}}^{\text{longitud del arco entre los nodos } i,j}}_{= u_j}, i] \qquad d_{ij} \geq 0$$

distancia más corta del nodo j al origen

Vamos a ir diferenciando entre etiquetado temporal y etiquetado permanente para ir calculando la distancia más corta nodo a nodo.

Algoritmo

Paso 0: etiquetar al nodo origen (nodo 0) con la etiqueta permanente $[0, -]$.

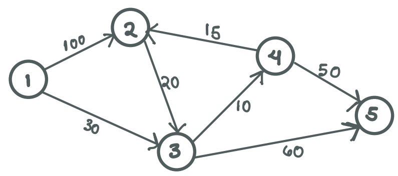
Paso general i:

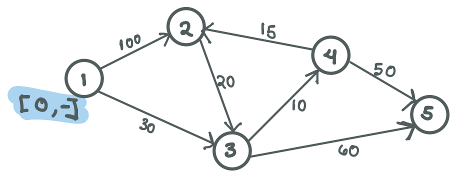
se cambia a etiquetado permanente si ya visitamos todos sus vecinos.

a) Calcular las etiquetas temporales $[u_i + d_{ij}, i]$ para cada nodo j con $d_{ij} \geq 0$ siempre y cuando el nodo j no esté etiquetado permanentemente. Si el nodo j está etiquetado temporalmente, existe $[u_j, k]$ y si se cumple $u_i + d_{ij} < u_j = u_k + d_{kj}$ entonces reemplazar $[u_j, k]$ por $[u_i + d_{ij}, i]$.

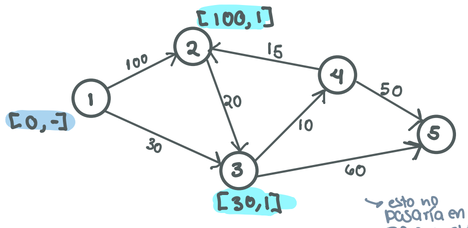
b) Si todos tienen etiquetado permanente se detiene el proceso.

Por ejemplo: considerando:





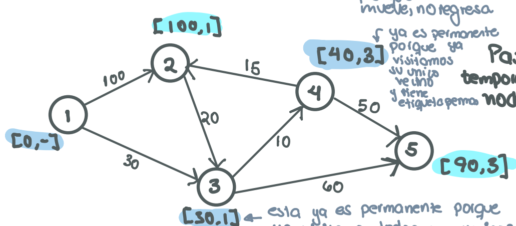
Paso 0: etiquetar al nodo origen (nodo 0) con la etiqueta permanente.



ask Sam

Paso 1: calcular las etiquetas temporales $[u_1 + d_{1j}, 1]$ para cada nodo $j = 2, 3$ (porque son sus vecinos).
Como no hay etiquetas temporales aún entonces todas las etiquetas temporales son aceptadas.

esto no pasaría en FMM porque el frente se mueve, no regresa

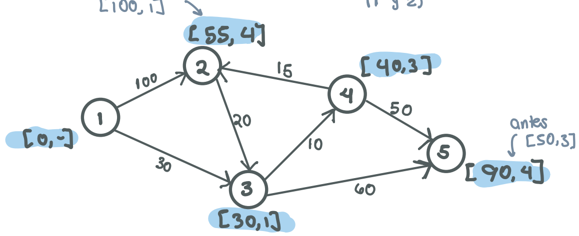


ya es permanente porque ya visitamos su único vecino y tiene etiqueta perma.

Paso 2: Calcular las etiquetas temporales $[u_i + d_{ij}, i]$ para cada nodo j con $d_{ij} \geq 0$

esta ya es permanente porque ya visitamos todos sus vecinos (1 y 2)

antes [100, 1]



antes [50,3]

Fin porque todas las etiquetas son permanentes.

Pseudo código
(Gracias Mayra)

[SAM: lists with values, not [[a, b], ...])

```
G ← grafo
S ← inicio/fuente
for V in G do:
    distancia[V] ← infinito
    previo[V] ← NULL
    if V != S :
        agregar V a lista de prioridad Q
distancia[S] ← 0

while Q no está vacío :
    U ← extract min de Q
    for each vecino no visitado V de U
        etiqueta_temporal ← distancia[U] + peso_vértice(U,V)
        if etiqueta_temporal < distancia[V]
            distancia[V] ← etiqueta_temporal
            previo[V] ← U
```

inicializar

iniciar etiqueta de cero

la etiqueta permanente se pone hasta que se visita cada nodo

cálculo de la etiqueta temporal

si con otra ruta se minimiza renombra la etiqueta temporal

Dijkstra

Ingredientes

- function to determine at node i which nodes to visit (i.e. which nodes j are connected to i). This is the priority queue Q
- Initialization step, label all the shortest distances as "infinity", label all the previous ones as NULL
- Initialization of the first node's distance as zero
- Function to calculate the minimum distance in Q (useful if I want to jump from this to FMM)

Fast Marching Method

$|\nabla T| = \frac{1}{f}$

Intialize

- "Alive points", $A = \{ (i,j) : j = 1 \}$, $T_{ij} = 0 \ \forall \ (i,j) \in A$
- Narrow band points
- Far away points

Marching forward

- (i_{min}, j_{min}) in Narrow band with the smallest value for T
- Add (i_{min}, j_{min}) to A
- Tag neighbours

If $|\nabla T|^2 = 1 \iff \left| \begin{bmatrix} \delta T / \delta x_1 \\ \delta T / \delta x_2 \end{bmatrix} \right|^2 = 1 \iff \left(\frac{\delta T}{\delta x_1}\right)^2 + \left(\frac{\delta T}{\delta x_2}\right)^2 = 1$

Using one-sided finite difference, find T_{ij} such that

$$\frac{(T_{ij} - T_{x_1})^2}{h^2} + \frac{(T_{ij} - T_{x_2})^2}{h^2} = 1$$

Which can be forward or backward 1-side finite difference (depends on its value)

$$\left(\frac{T_{ij} - T_{x_1}}{h}\right)^2 = \begin{cases} \left(\frac{T(x_1 + h, x_2) - T(x_1, x_2)}{h}\right)^2 \\ \text{or} \\ \left(\frac{T(x_1, x_2) - T(x_1 - h, x_2)}{h}\right)^2 \end{cases}$$

$$\left(\frac{T_{ij} - T_{x_2}}{h}\right)^2 = \begin{cases} \left(\frac{T(x_1, x_2 + h) - T(x_1, x_2)}{h}\right)^2 \\ \text{or} \\ \left(\frac{T(x_1, x_2) - T(x_1, x_2 - h)}{h}\right)^2 \end{cases}$$

$$T_{x_1} = \min \{ T(x_1 + h, x_2), T(x_1 - h, x_2) \}$$

$$T_{x_2} = \min \{ T(x_1, x_2 + h), T(x_1, x_2 - h) \}$$

This is a quadratic formula, find T_{ij}:

$$\frac{(T_{ij} - T_{x_1})^2 + (T_{ij} - T_{x_2})^2}{h^2} = 1$$

$$\iff T_{ij}^2 - 2T_{ij}T_{x_1} + T_{x_1}^2 + T_{ij}^2 - 2T_{ij}T_{x_2} + T_{x_2}^2 - h^2 = 0$$

$$\iff 2T_{ij}^2 + T_{ij}(-2T_{x_1} - 2T_{x_2}) + (T_{x_1}^2 + T_{x_2}^2 - h) = 0$$

$$\iff T_{ij}^2 + T_{ij}(-T_{x_1} - T_{x_2}) + \frac{T_{x_1}^2 + T_{x_2}^2 - h}{2} = 0$$

$$\iff T_{ij} = \frac{(T_{x_1} + T_{x_2}) \pm \sqrt{(T_{x_1} + T_{x_2})^2 - 2(T_{x_1}^2 + T_{x_2}^2 - h^2)}}{2}$$

$$\Rightarrow T_{ij} = \frac{(T_{x_1} + T_{x_2})}{2} + \frac{1}{2}\sqrt{(T_{x_1} + T_{x_2})^2 - 2(T_{x_1}^2 + T_{x_2}^2 - h^2)}$$

↳ This is the Eikonal update formula for $f \equiv 1$