



# **ALGORITMO PARA LA SELECCIÓN DE LA RUTA MÁS CORTA Y SEGURA**

# Presentación del equipo



**Vanessa Vélez  
Restrepo**  
Investigación de  
artículos



**Luis Miguel  
Giraldo**  
Investigador del  
proyecto



**Andrea Serna**  
Revisión de  
la literatura

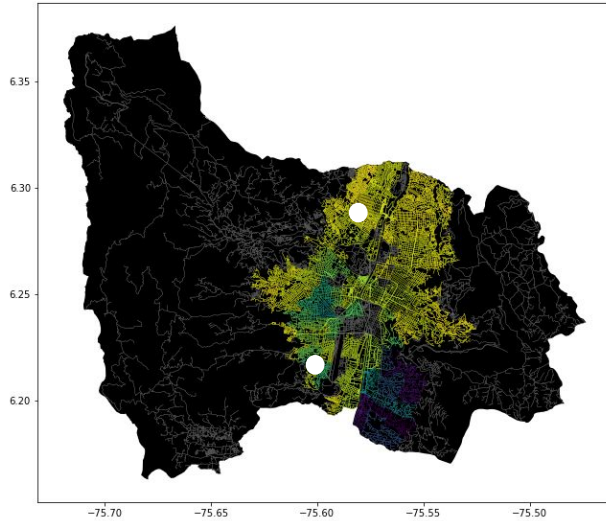


**Mauricio Toro**  
Preparación  
de los datos

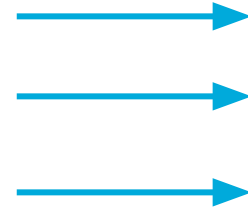


<https://github.com/gotaluism/ST0245-002>

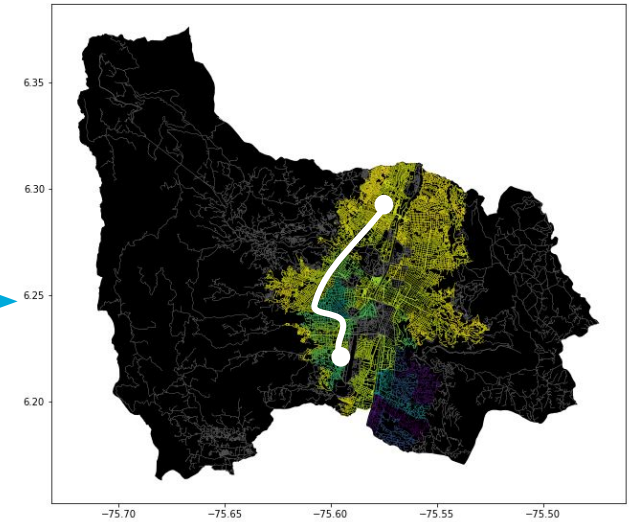
# Planteamiento del problema



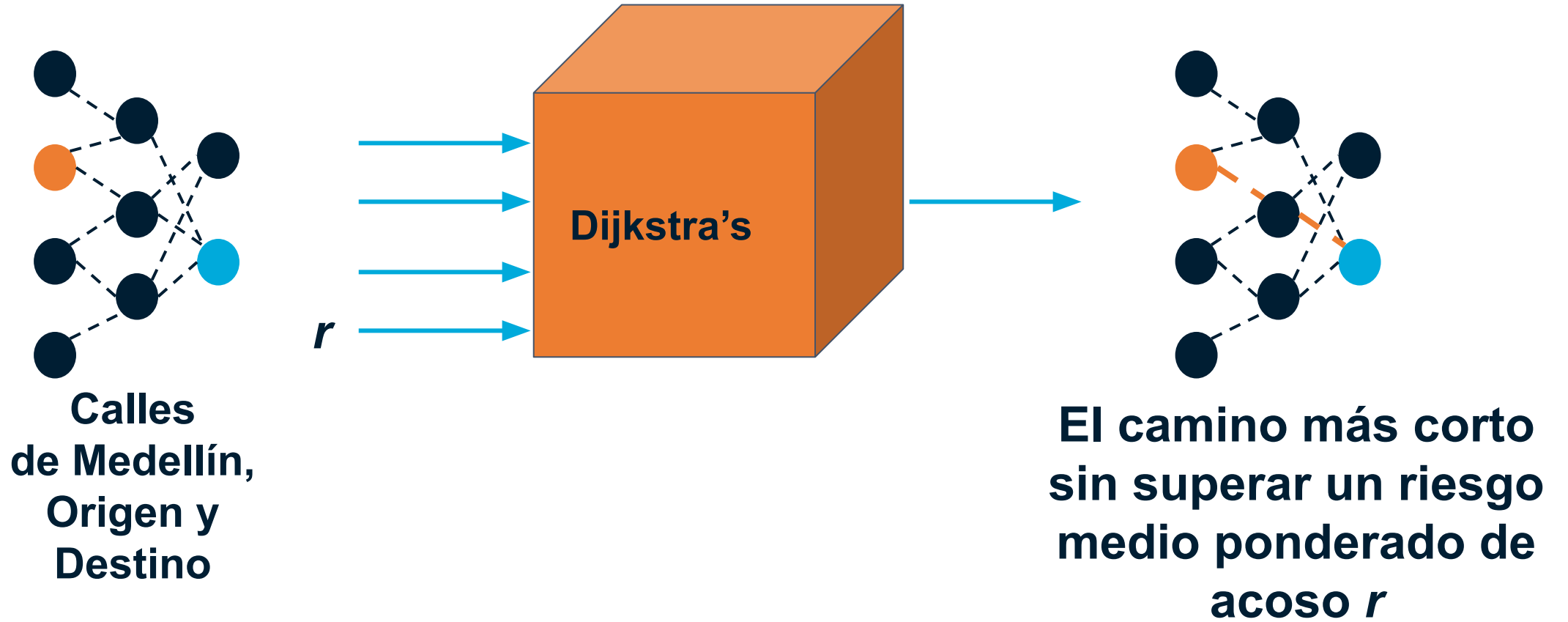
**Calles  
de Medellín,  
Origen y  
Destino**



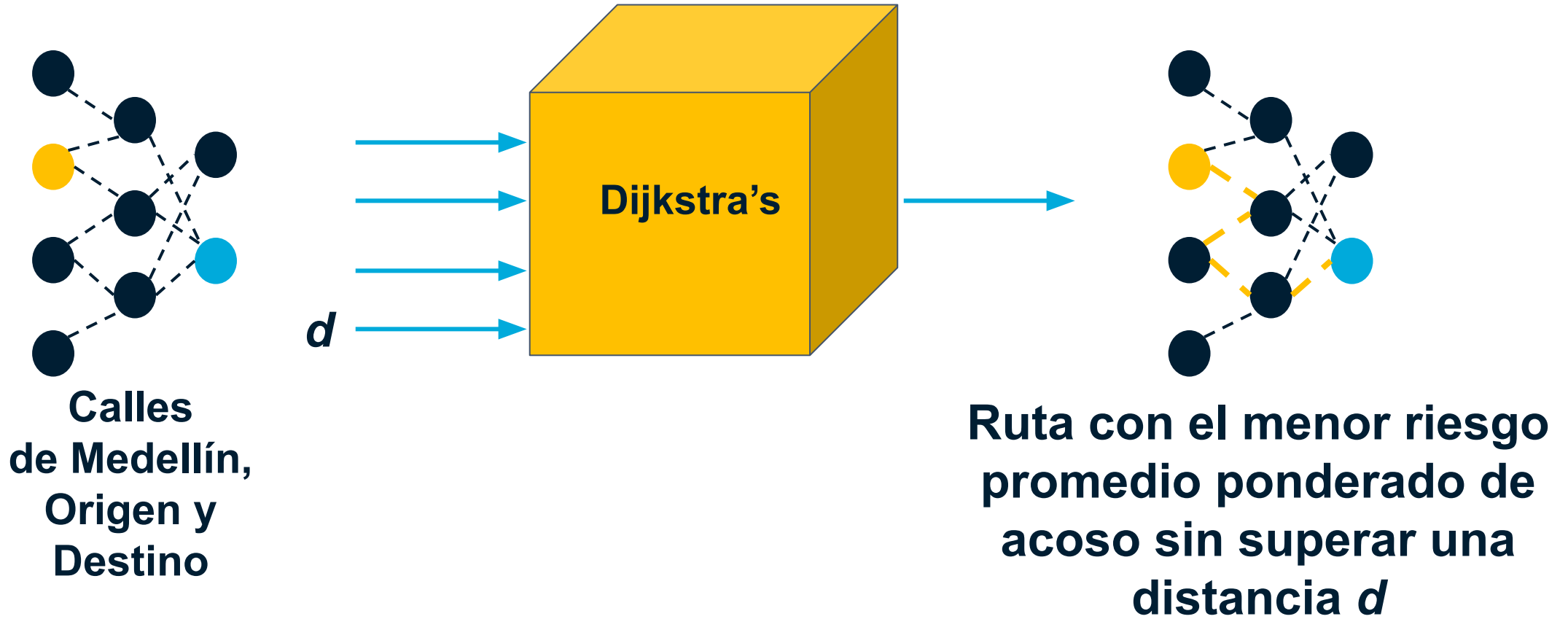
**Algoritmo  
del camino  
más corto  
restringido**



**El más camino más corto  
restringido**



## Segundo algoritmo

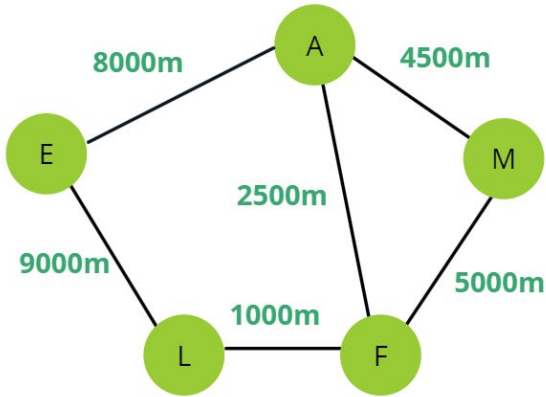




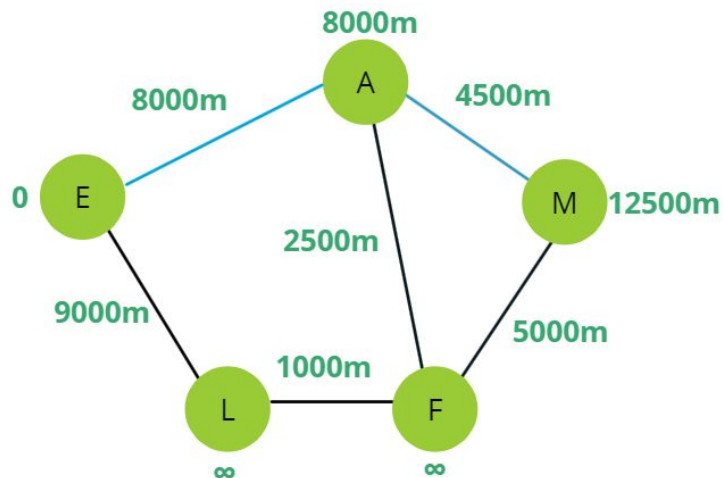
# Explicación del algoritmo



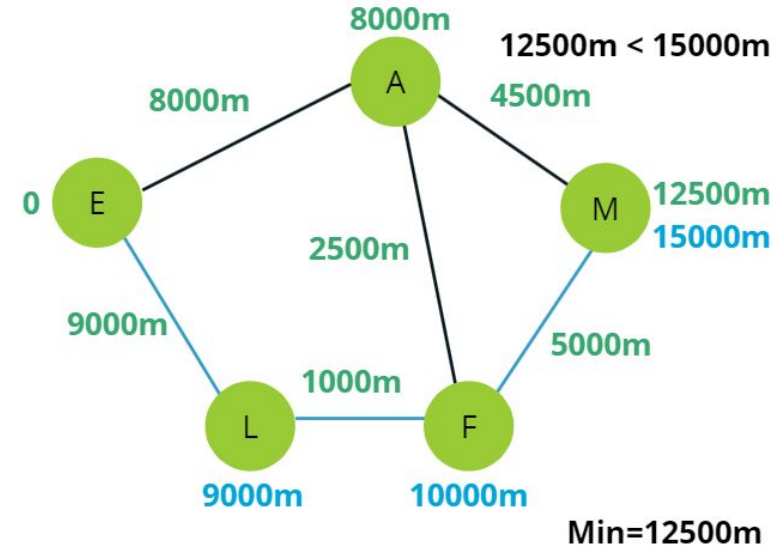
1.



2.



3.



Algoritmo de Dijkstra

*Este algoritmo toma el vértice inicial con un peso de 0, mira a su vecino y le asigna el valor del vértice inicial + el valor del vértice del vecino (0+8000) y así se va sumando por ejemplo en la segunda imagen suma 8000+4500 que es el valor del peso del vecino y al final compara los valores finales de las posibles rutas y retorna la más corta de todas.*

	Complejidad temporal	Complejidad de la memoria
Algoritmo de Dijkstra	$O(E * \log(V))$	$O(V * E * 2^E)$

Complejidad en tiempo y memoria del algoritmo de Dijkstra.  $V$  es el vertice de 1 a  $n$ ; y  $E$  es el número total de aristas.

