

# *Anàlisi de xarxes terrestres multicapa*

Novembre 2024

Presentat per  
**Isaac Bassas, Violeta Bonet, Maria Gil i Lola Monroy**

# *Funcions interessants* —————→

## ————— PostGIS y pgRouting per PostgreSQL

ST\_intersects

ST\_3DClosest\_Point

ST\_3DShortest\_Line

pgr\_aStar

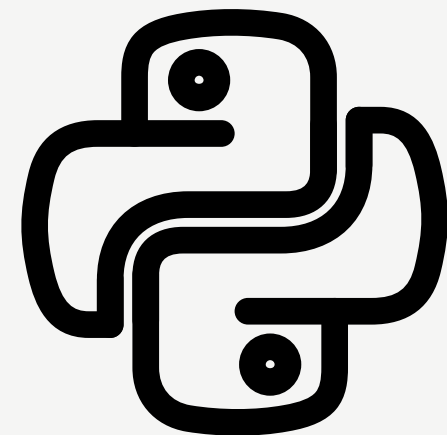
pgr\_aStarCost

ST\_Length

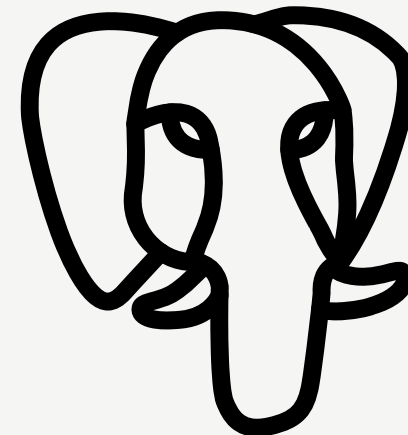
# *Què hem fet servir?*



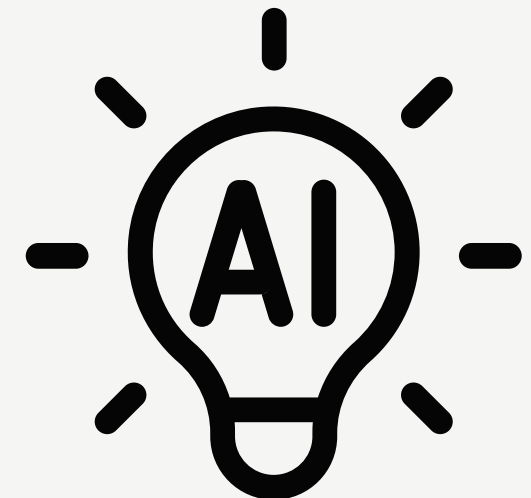
**DBeaver  
(SQL)**



**VisualStudio  
(Python)**



**QGIS  
Postgres  
pgRouting**



**ChatGPT**



# *Algorismes per trobar camins 'òptims'*



## **A\***

- Troba el camí òptim si la funció de cost és admissible i consistent
- Problemes de cerca de rutes
- Costós en memòria i temps i poc escalable

## **BeamSearch**

- No garantitza trobar el camí òptim
- Problemes d'optimització (espai d'estats gran)
- Menys memòria i és molt més eficient (només manté un nombre limitat de nodes actius)

# *Plantejament sencill*

———— **Com ajuntar les capes?**



Merge de les ddbb de punts, afegint una columna pels costos de cada capa

## A\*

- Iterar sobre els punts, fixar un radi i crear una aresta per cada punt a dins d'aquest.
- Afegir com a cost la suma ponderada entre dels costos de les capes on estan els punts.
- Fer l'algoritme A\*

## BeamSearch

- Dur a terme el beam search a la base de dades ajuntada

# *Plantejament més complex*

---

## **Com passar de capa a capa?**

Merge de les ddbb de punts i de línies  
(multistrings)

---

- Trobar les interseccions entre multistrings amb ST\_intersects.
- Afegir els punts d'intersecció a la db de punts.
- Segmentar les antigues línies pels punts d'intersecció, i afegir a la db de línies
- Fer A\* o beam search



# *Desenvolupament web*





# Què fitxers/documents presentem?

## Fitxers SQL

- multicapa.sql
- capa\_unica.sql

## Notebook Python

Fitxer per afegir  
costos a la base de  
dades.

## Frontend

Diseny, interfaç  
simple

## Esquema + PWP + ReadMe

De com ha  
evolucionat el  
nostre treball i idees





*Que ens hagués  
agradat fer?*