Universitat Autònoma de Barcelona

Optimization

A-star algorithm

Rocco Ballester (1502124)
Sergi Domènech (1496219)

27 December 2022

The two codes presented in the attached files compute the minimal distance from *Basílica de Santa Maria del Mar* in Barcelona to the *Giralda* in Sevilla using the A^* algorithm (seen in the lectures). To do so we have used two main things: on the first hand we have chosen the haversine formula (i.e. the formula that determines the great-circle distance between two points on a sphere given their longitudes and latitudes) to be our heuristic function, since it is the way that this distance is computed with less error. On the other hand, and as the delivery's sheet recommended, we have implemented a file (*write.c*) to convert csv files into binary files.
In the following picture one can see how this last file should be compiled and execute and the time it took us to convert *catalunya.csv* and *spain.csv* into binary files:

```
rocco@rocco-VirtualBox:~/Desktop/Opt/Delivery2/roccobb$
rocco@rocco-VirtualBox:~/Desktop/Opt/Delivery2/roccobb$
rocco@rocco-VirtualBox:~/Desktop/Opt/Delivery2/roccobb$ gcc write.c -o write.exe -lm
rocco@rocco-VirtualBox:~/Desktop/Opt/Delivery2/roccobb$ time ./write.exe catalunya.csv

real    0m11,839s
user    0m10,946s
sys     0m0,743s
rocco@rocco-VirtualBox:~/Desktop/Opt/Delivery2/roccobb$ time ./write.exe spain.csv

real    31m18,748s
user    30m40,885s
sys     0m16,711s
rocco@rocco-VirtualBox:~/Desktop/Opt/Delivery2/roccobb$
```

Figure 1: Compilation and execution of the *write.c* file to convert the csv files to binary files.

Regarding the *Astar.c* file, we decided to organize it into three sections. On top of it one can find all structures and functions used in our algorithm later on. Following them, one can find the A^* algorithm in an isolated function. At the end, one can find the main function where the binary file is read and the algorithm is called. As we comment in the code, all functions have been written following the

delivery's main sheet, the subject's notes and help from the internet.
Moreover, we have created a function called *output.txt* that creates a third file called *spain_ SROutput.txt* (or *catalunya_ SROutput.txt* depending on which map you are working on) where the optimal path found by the algorithm is displayed:

Open

spain_SROutput.txt
~/Desktop/Opt/Delivery2/roccobb

```
1 # Distance from 240949599 to 195977239: 958814.883524 meters.
2 # Optimal path:
3 Id = 240949599 | 41.383341 | 2.181774 | Dist = 0.000000
4 Id = 240944785 | 41.383476 | 2.181506 | Dist = 0.026927
5 Id = 240936347 | 41.383541 | 2.181366 | Dist = 0.040717
6 Id = 240936348 | 41.383570 | 2.180862 | Dist = 0.082864
7 Id = 30647274 | 41.383608 | 2.180647 | Dist = 0.101262
8 Id = 240939090 | 41.383621 | 2.180584 | Dist = 0.106756
9 Id = 30647313 | 41.383652 | 2.180399 | Dist = 0.122532
10 Id = 240934220 | 41.383709 | 2.180116 | Dist = 0.147014
11 Id = 240934857 | 41.383761 | 2.179822 | Dist = 0.172166
12 Id = 240936350 | 41.383837 | 2.179353 | Dist = 0.212191
13 Id = 240936351 | 41.383873 | 2.179055 | Dist = 0.237407
14 Id = 240936353 | 41.383896 | 2.178848 | Dist = 0.254873
```

Figure 2: Output file that contains the optimal path.

This output file tries to mimic the one given by the professors, showing all nodes from the source to the destination along with their Ids, longitudes, latitudes and distance travelled.
In addition, one can use this output file to generate the path on a map thanks to the *mapplot.py* file:

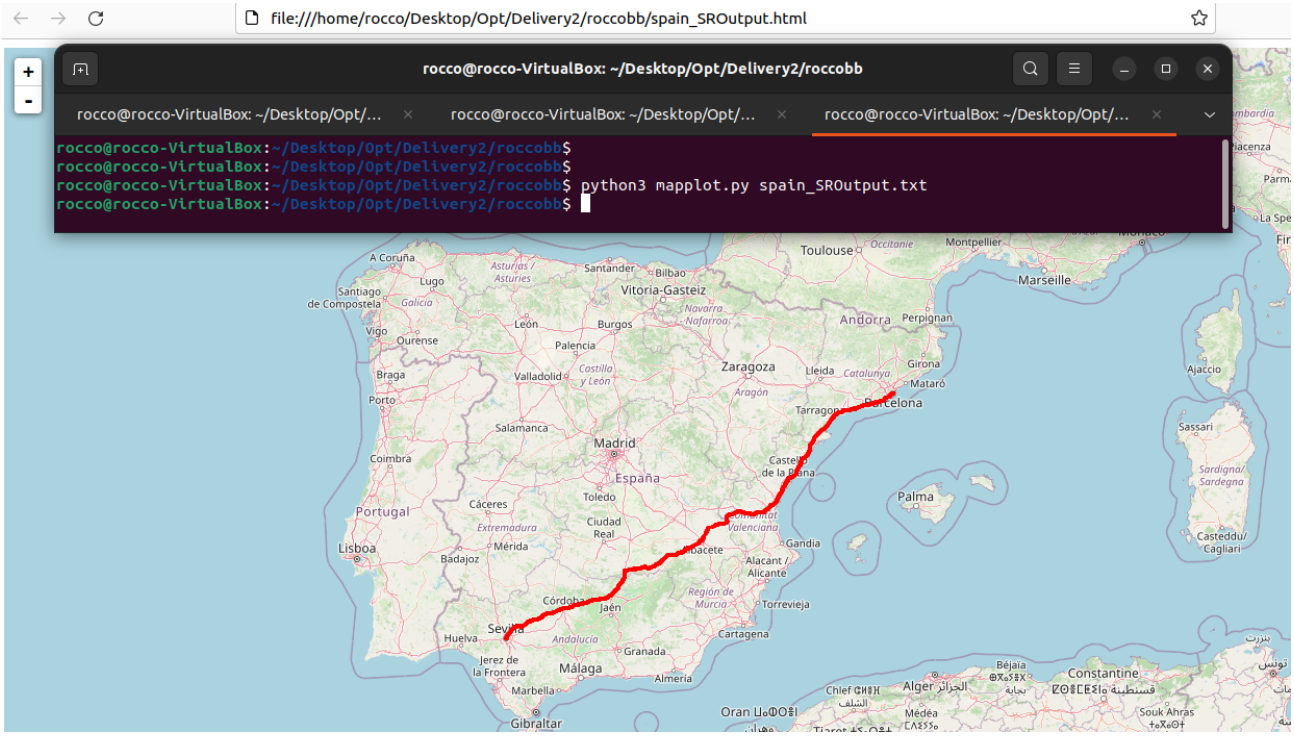

Figure 3: Image of the optimal path found from *Basílica de Santa Maria del Mar* in Barcelona to the *Giralda* in Sevilla using the A^* algorithm.

All in all, and as we can see in the following picture, we have been able to compute the minimal distance between the *Basílica de Santa Maria del Mar* in Barcelona and the *Giralda* in Sevilla using the A^* algorithm (notice also that if we want to use another source or another destination we just

need to change lines 275 and 276 in the *Astar.c* file with their respective id). In our case, the optimal distance found by the algorithm is about 958814 meters, our program runs in approximately 7 seconds and the A^* algorithm takes about 5 seconds to run.

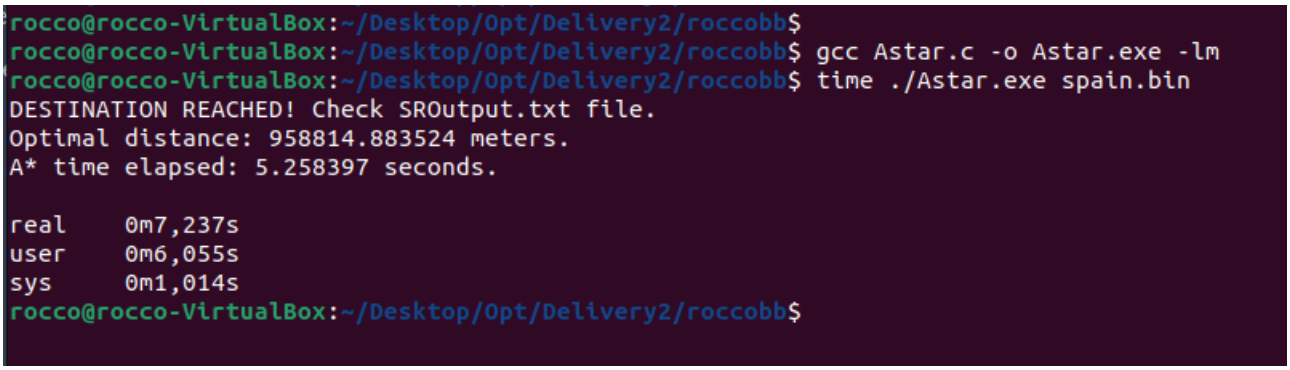

Figure 4: Compilation and execution of the main file.

The file is compiled as usually but notice that to execute it one needs to pass a binary file as an argument!