Algorithms for pedestrian paths that reduce both street harassment and distance

Juan Esteban Barrios Tovar
Universidad EAFIT
Colombia
jebarriost@eafit.edu.co

Mauricio Toro
Universidad EAFIT
Colombia
mtorobe@eafit.edu.co

1. RESUME

The problem addressed in this report is the construction of an algorithm, which allows calculating three possible paths, in order to reduce both the distance and the risk of harassment from one point to another in the streets of the city of Medellín. It is important to address this problem, because in recent years Medellín has lost its reputation for being a safe city and little by little we have seen that street harassment has become a very common topic to talk about, in turn it has been seen as in On certain occasions when we need to reach a destination in the shortest possible time, we take the shortest path without thinking about our safety, which is why it is of vital importance toT provide a solution to this problem.

The algorithm that has been proposed in this report to solve this problem is Dijkstra's algorithm, which has been used taking as reference data in a CSV file of distances and risks of harassment in the streets, where through a combination of these variables d(distance) and r (risk of harassment) we obtain the result of a product, which is taken as a reference in the algorithm for the calculation of the three paths. As a result, the three paths were successfully obtained in an approximate execution time of 1.2 seconds and these paths vary from the shortest and safest to the least short, but safest.

2. INTRODUCTION

Over the years it has been seen how various computer applications have been a fundamental piece to facilitate people's daily activities, such as transportation and mobility, however, during the last period both street harassment and long distances that are generated when transporting us through various paths, have shown the great need for the development of an algorithm that allows the reduction of the previous pair of problems mentioned.

3. PROBLEM

The main objective of this project is the development of an algorithm based on the streets of the city of Medellin, in order to completely eliminate or at least reduce situations of street harassment, which for many years has been one of the main causes of insecurity in the city, affecting both men and women, the latter being the main affected by this situation, in turn through the algorithm it is proposed to reduce the time it takes people to go from one point to another looking for the way in which the distance from the destination is the shortest.

As a final result, it is expected that both problems will be solved through an algorithm, yielding as a final product the safest and shortest path possible provide the user.

4. SOLUTION

As a solution to the problem, Dijkstra's algorithm is used through the Python networkx library, which taking into account the data from the csv file, extracted and converted into a data frame by the Python NumPy extension "Pandas" , allows us to generate the shortest path with the least harassment possible, taking as input an origin X and a destination Y in coordinates that exist in the file (Calles_de_medellin_con_acoso) and in turn graph it.

5. RELATED WORKS

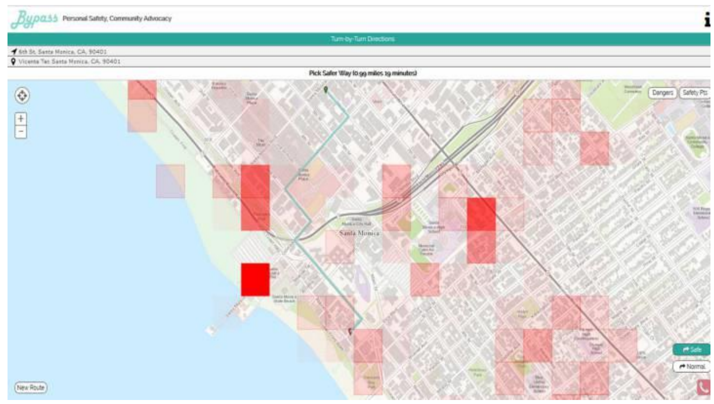
5.1 Always safe

It is an algorithm which was developed to confront the persistent violence against women in Mexico, it was created by a group of activists in the city of Quetaro. The algorithm seeks to generate a mapping to identify the areas where there is a higher incidence of harassment to find the causes and generate actions to, consequently, make this reality visible as a problem that requires public policies in Mexico.

5.2 Bypass

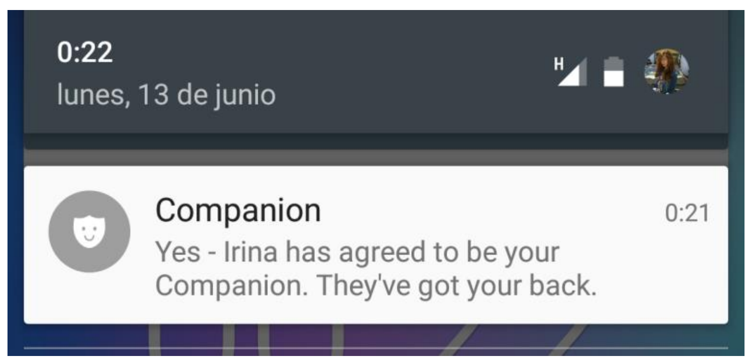
Bypass es una aplicación la cual permite informar It a las personas que zonas de la ciudad se deben evitar en una ruta. Esta aplicación realiza un análisis a partir de informaciones que se publican en la red y opiniones de usuarios.

El sistema de la aplicación es aproximado e identifica con cuadrados rojos de mayor o menor intensidad las calles por las que es mejor no pasar, por lo que si en la ruta llega a aparecer una de estas zonas el sistema calculara una mejor ruta con el trayecto mas corto.



5.3 Companion

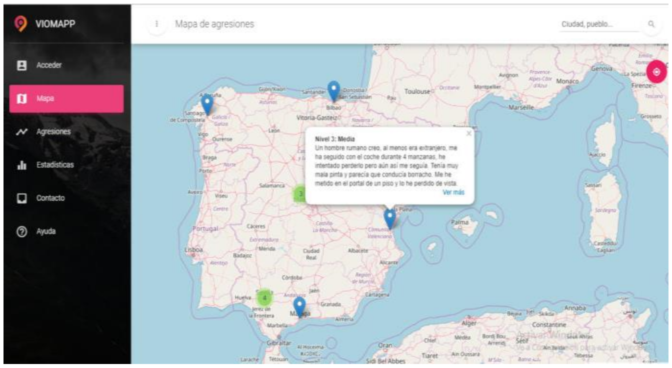
It is a free app devised by five students from the University of Michigan, the app consists of geolocating the user and generating the shortest route, along the way the application allows the user to have access to a virtual companion from his contact list, the which receives notifications if the user is in a dangerous situation through sudden movements detected by his cell phone.



5.4 Viomapp

This an application that was designed by the engineer Joaquín Vásquez in Spain, this application uses an algorithm that allows women to record on the map a sexual assault, street harassment or inappropriate behavior that they have suffered on their usual or night routes.

The main intention of the app is focused on women being able to review a route before traveling it, thus checking if it is an unsafe area, marked with previous incidents by other people.



The foundation on which this algorithm is based is the principle of optimizing: if the shortest path between vertices “u” and “v” passes through vertex “w”, then the part of the path that goes from “w” to “v” must be the shortest path among all paths from “w” to “v”. In this way, the paths of minimum cost are successively built from an initial vertex to each of the vertices of the graph, and the paths obtained are used as part of the new paths.

The time complexity of Dijkstra’s algorithm is O(n*n) where n represents the number of vertices

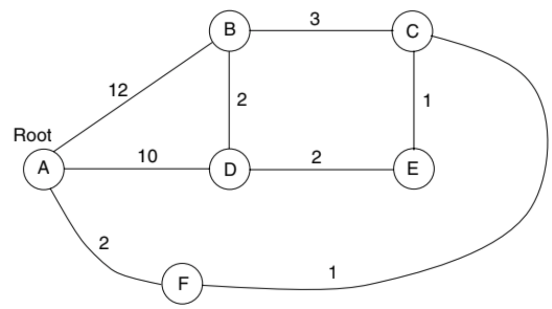


Fig. 4.2. A network graph with link costs indicated

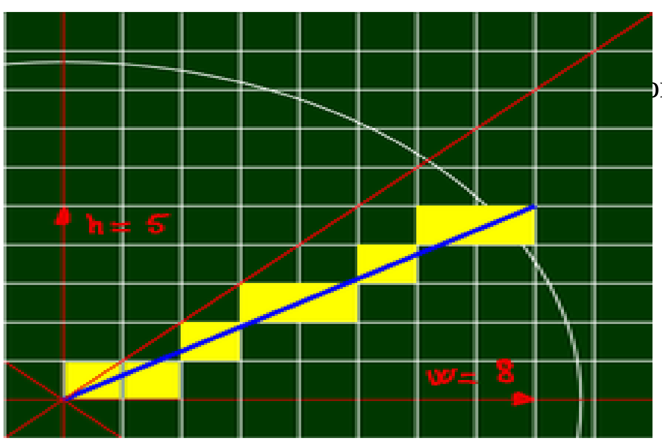
6 ALTERNATIVE SOLUTIONS

6.1 Bresenha Line Algorithm:

It is a line drawing algorithm that determines the points of an n-dimensional raster that should be selected to form a close approximation of a straight line between two points.

The general idea behind this algorithm is: given an initial endpoint of a line segment, the next grid point it traverses to reach the other endpoint is determined by evaluating where the line segment intersects relative to the midpoint (above or below) of the two possible grid point options.

The time complexity of the algorithm is O(n*n) since the algorithm goes through a while loop n*n time



6.2 Dijkstra’s algorithm:

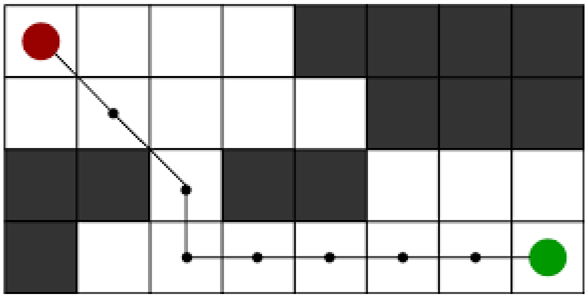
It is an algorithm that consists of finding the shortest distances from a source vertex to all other vertices in a connected weighted graph. It should be emphasized that the edges must have positive values and that this algorithm does not work for negative values.

6.3 A-Star Algorithm:

This algorithm is an improvement developed to the postulates of the Dijkstra algorithm that is responsible for finding the shortest paths within a graph. In this modification, the observation of informed searches within the graph that allow us to make optimal decisions about the paths that must be taken to traverse the graph efficiently is taken as a central point.

The application of this algorithm we must understand how to proceed to divide the cost of the route. In this case it is divided into two parts where g(n) represents the cost of the path from its origin to some node n within the graph. We also have that h(n) represents the estimated cost of the path from node n to the destination node, calculated by an intelligent guess.

The A-Star algorithm, being specially designed to detect the least expensive paths within a complex graph, is normally used to find short paths between individual pairs of locations. This is why one of its most common applications is to detect geo-locations in which satellite location coordinates are known.



The complexity of the algorithm is closely related to the quality of the heuristics used in the problem. In the worst case, with a poor quality heuristic, the complexity will be exponential, while in the best case, with a good heuristic(h'(n)), the algorithm will execute in linear time.

6.4 Euclidean distance algorithm

It is an algorithm that is based on the application of a mathematical formula that allows to measure the distance in a straight line between two points in an n-dimensional space.

$$d(\mathbf{p},\mathbf{q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2}$$

Within the functionalities of the Euclidean distance algorithm, its usefulness to determine the similarity between two things or pairs of data stands out. Proceeding from this, the similarity calculated from this algorithm can be used as an integral part of recommendation query systems. With it we can obtain data schemes that identify elements that have similar characteristics, such as a score or an assessment so that the user of the information can decide. An example of this can be applied in a movie recommendation system so that we obtain suggestions based on the ratings received.

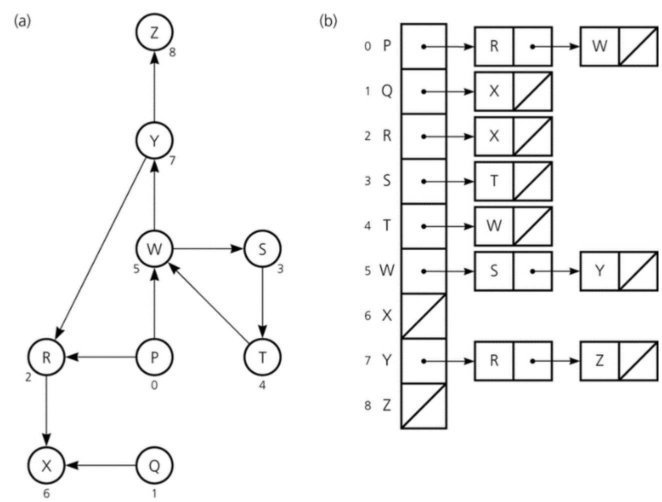
The complexity of the Euclidean distance algorithm is O (1) since there are no loops in the algorithm and no recursion is used.

7 DESIGN AND IMPLEMENTATION OF THE ALGORITHM

7.1 Data structures

To represent the map of the city of Medellín, a data frame is used, which analyzes the data from the csv file (calles_de_medellin_con_acoso), and with the help of the networkx library, a list of the edges with their respective vertices is created, which have as attribute both the length and the risk of harassment, considering the point of origin and the point of destination.

https://github.com/jebarriost3/ST0245-002

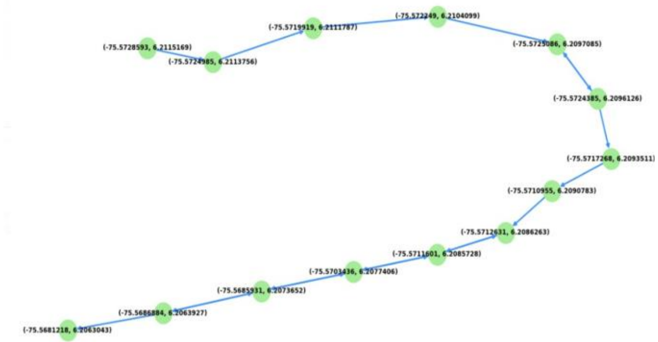


7.2 Algorithms

In this article, an algorithm is proposed for a path that minimizes both the distance and the risk of street sexual harassment.

7.2.1 Algorithm for a path that reduces both the distance and the risk of street sexual harassment

W When generating the data frame with the data from the csv file (Calles_de_medellin_con_acoso), a graph is created from the source, destination and weight data, which would be a new variable which is the multiplication of the data from the length and harassment risk columns, to obtain the weight of each edge. After this we convert the data frame into a dictionary and add it as an attribute of the nodes. Finally, through the nexworkx library, the Dijkstra algorithm is called, which searches for the smallest value of the weight variable to obtain the shortest path between the origin and the destination given in the input.



7.2.2 Calculation of two other ways to reduce both the distance and the risk of street sexual harassment

For the second path, the definition of the variable V is preserved as the product between the distance and the harassment, however, the first path is taken as a reference and the middle node is eliminated from it, this causes the Dijkstra algorithm to perform a calculation of a new path, which does not take into account the node removed from the previous path and thus results in a different path which retains the harassment average of the previous path but its distance in meters increases.

For the third path, the definition of the variable V is preserved again as the product between the distance and the harassment, however, to obtain a path that reduces V different from the first and the second, a modification is made to the data frame index, taking this as the origin coordinates of the streets, taking this into account the graph is created and through a for cycle various edges of each column of the file are added and after that the Dijkstra algorithm is called so that with the data modified frame can generate a new path, which is characterized by having a greater distance than the previous paths but the average harassment of this path is lower than them.

The algorithm is exemplified in Figure 4

Figure 4: Map of the city of Medellín showing three paths for pedestrians that reduce both the risk of sexual harassment and the distance in meters between the Medellín Clinic Carrera 43A with Calle 7 and the Road Intersection Calle 10A with Carrera 36.

7.3 Algorithm complexity analysis

Table 1: Time Complexity of Dijkstra's Algorithm

Algorithm	Temporal complexity
Dijkstra Algorithm	O(V*V+E)

The complexity of the algorithm is O(V*V+E) where V is the number of vertices and E is the number of edges. The calculation of the complexity is obtained through the defined complexity formula of the Dijkstra algorithm which is O(V*V) +O(E) since all the vertices and all the edges of the graph are traversed.

Table 2: Memory complexity of the data structure

name used by your algorithm

Data Structures	Memory complexity
Dataframe	O(V)

In the algorithm, a data frame was used, which is a data structure, which can store data of different types, in this case the data of the CSV file.

7.4. Algorithm design criteria

The algorithm was designed in such a way that the data was read from the CSV file correctly and quickly, relying on the networkx library so that through it a simple call of Dijkstra's algorithm could be made and with it giving an origin as input and a destination can search for the most optimal routes taking into account both distance and harassment risk data, this is reflected in the time it takes for the algorithm to run, which is approximately 1.2 seconds.

8. RESULTS

8.1 Results of the road that reduces both the distance and the risk of street sexual harassment

Table 3

Origin	Destination	Distance	Harassment Risk
Medellín Clinic Carrera 43A with Calle 7	Road Intersection Calle 10A with Carrera 36	1.164,14 m	4,5325
Medellín Clinic Carrera 43A with Calle 7	Road Intersection Calle 10A with Carrera 36	1.207,73 m	5,1369
Medellín Clinic Carrera 43A with Calle 7	Road Intersection Calle 10A with Carrera 36	1.654,02 m	4,2442

Table 3. Distance in meters and risk of street sexual harassment (between 0 and 1) to go from the Medellín Clinic Carrera 43A with Calle 7 to the Road Intersection Calle 10A with Carrera 36 walking.

8.2 Algorithm execution times

Table 4

Calculation of v	Average execution times (s)
v = 5.276,59	1,0087 s
v = 6.204,03	1,0068 s
v = 7.020,06	1,0110 s

Table 4: Execution times of the Dijkstra Algorithm for each of the three paths calculated between the Medellín Clinic Carrera 43A with Calle 7 and Road Intersection Calle 10A with Carrera 36.

9. CONCLUSIONS

Fro from the algorithm it can be concluded that the paths generated are totally different since, although the first two paths have the same risk of harassment, the distances are different. In the third path it is possible to appreciate that the risk of harassment is low, however, the distance in meters of said path is greater than that of the two previous paths, the first being the most optimal both for its distance and for its risk of bullying.

It is also possible to conclude that the execution times are reasonable, therefore there would be no problem applying this in a real situation.

10. FUTURE WORKS

I would like to improve the algorithm in such a way as to take it to a mobile application, since this would be, in my opinion, the best way to tackle the problem in reality, because as I mentioned before, it is very important to solve this problem and do it in a way Similar to how waze is calculating the shortest path of a route, it would be the best option.

ACKNOWLEDGEMENTS

I thank Professor Alexander Narvaez Berrio from

EAFIT University for providing me with practical examples for the elaboration of the algorithm and my colleague Alejandro García for helping me in the elaboration of the map with the three paths. Also, I thank Professor Juan Carlos Duque, from the EAFIT University, for providing the data from the 2017 Medellin Quality of Life Survey, processed in a Shapefile file.

REFERENCES

1. Related works:

[1] Forbes Staff. (2021). Siempre Seguras: la app para mapear el acoso sexual callejero en México[online].Available:https://www.forbes.com.mx/tecnologia-siempre-seguras-app-mapear-acoso-sexual-callejero-mexico/

[2] Betech(2016,11 Oct). Una app que te dice que zonas de tu ciudad debes evitar[online]. Available:https://as.com/meristation/2016/10/11/betech/1476219792_538454.html

[3] Trendencias(2016, 26 sep). Companion: una aliada contra el acoso callejero[online]. Available: https://www.trendencias.com/psicologia-1/companion-una-aliada-contra-el-acoso-callejero

[4] La vanguardia. Nace Viomapp, una aplicación que muestra a tiempo real las agresiones y el acoso callejero a las mujeres[online]. Available: https://www.lavanguardia.com/tecnologia/20180422/442694999230/nace-viomapp-aplicacion-muestra-real-agresiones-acoso-callejero-mujeres.html#:~:text=Viomapp%20es%20%E2%80%9Cuna%20herramienta%20preventiva,cotejando%20as%C3%AD%20si%20se%20trata

2. ALTERNATIVE SOLUTIONS

[5] Omdena. (2020). Preventing Sexual Harassment Through a Path Finding Algorithm Using Nearby Search. [online]. Available: https://omdena.com/blog/path-finding-algorithm/

[6] Geeksforgeeks. (2022). Dijkstra's Shortest Path Algorithm | Greedy Algo-7.[online].Available:https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/

[7] Geeksforgeeks. (2022). A* Search Algorithm.[online].Available:https://www.geeksforgeeks.org/a-search-algorithm/

[8]Graph Analytics o la v2.0 del IA. (n.d). [online]. Available: https://www.grapheverywhere.com