Preventing Street Harassment by Using Constrained Shortest Path Algorithms

Samuel Lopera
Universidad Eafit
Colombia
sloperat@eafit.edu.co

Santiago Lopez
Universidad Eafit
Colombia
Slopezc12@eafit.edu.co

Andrea Serna
Universidad Eafit
Colombia
asernac1@eafit.edu.co

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

ABSTRACT

Street harassment or catcalling is a problem that has been affecting our society for a very long time. It primarily consists of unwanted sexualized comments, gestures and actions forced on a stranger in a public place without their consent. This is problematic because it affects a person's sense of freedom and security in the streets. This is a problem that mainly affects women but its not exclusive to them. It is also closely related to sexism because street harassment is usually the consequence of a sexist culture.

Keywords

Constrainted shortest path, street sexual harassment, secure-path identification, crime prevention.

1. INTRODUCTION

In Colombia, women must face constant sexual harassment when they are in public. This makes women feel unsafe, it reduces their sense of freedom, and it affects their health and well-being. More specifically in Medellin, when women were asked about how they feel in public, 60% of them said that they felt that the city was unsafe for them, 50% said that they are afraid of public parks and other public spaces, and the list goes on.[1] This motivates us to bring a solution to help women feel safer in Medellin.

1.1. Problem

To help reduce street harassment in Medellin, we are facing the problem of using an algorithm to find the shortest path possible from one point to another, without exceeding a certain risk of harassment, and finding the path with the least amount of harassment, without exceeding a certain distance.

1.3 Article structure

In what follows, in Section 2, we present related work to the problem. Later, in Section 3, we present the data sets and methods used in this research. In Section 4, we present the algorithm design. After, in Section 5, we present the results. Finally, in Section 6, we discuss the results and we propose some future work directions.

2. RELATED WORK

In what follows, we explain four related works to path finding to prevent street sexual harassment and crime in general.

2.1 Safest Routing Algorithm for Evacuation Simulation in Case of Fire

In this project, the team developed an algorithm that could find the safest route in an evacuation in a fire. In the words of the authors "The algorithm shows a positive impact on the evacuation time and overall, on the safety during an evacuation simulation."[2]

2.2 Urban navigation beyond shortest route: The case of safe paths

This work is very similar to ours. They utilized crime data to provide same urban navigation. Then, the goal is to find a short and low-risk path from one point to another. Their algorithm outputs a small set of paths that provide tradeoffs between distance and safety.[3]

This one is interesting, because it is really similar to our project. .

2.3 Vehicle Routing Problem with elementary shortest path-based column generation

The team involved in this project considered some of the shortest path algorithms (Ex: Dijkstra's Algorithm, A* Algorithm, Floyd-Warshal's Algorithm, etc.) that exists as of today have an extremely high time complexity which makes them impractical for some existing problems. The problem they are trying to solve is related to the vehicle routing problem and they try to check which algorithm is more optimal for their problem.[4]

2.4 Shortest path problem in static navigation situations

This project is related to graph navigation, nonetheless it is still relevant for the problem at hand, considering graph related algorithms relate to quite a lot of problems, including the shortest route problem.[5]

3. MATERIALS AND METHODS

In this section, we explain how data was collected and processed and, after, different constrained shortest-path algorithm alternatives to tackle street sexual-harassment.

3.1 Data Collection and Processing

The map of Medellín was obtained from Open Street Maps (OSM)[1] and downloaded using Python OSMnx API[2]. The (i) length of each segment, in meters; (2) indication wheter the segment is one way or not, and (3) well-known binary representation of geometries were obtained from metadata provided by OSM.

---

[1] https://www.openstreetmap.org/

[2] https://osmnx.readthedocs.io/

For this project, we calculated the linear combination that captures the maximum variance between (i) the fraction of households that feel insecure and (ii) the fraction of households with income below one minimum wage. These data were obtained from the quality of life survey, Medellín, 2017. The linear combination was normalized, using the maximum and minimum, to obtain values between 0 to 1. The linear combination was obtained using principal components analysis. The risk of harassment is defined as one minus the normalized linear combination. Figure 1 presents the risk of harassment calculated. Map is available at Github[3].

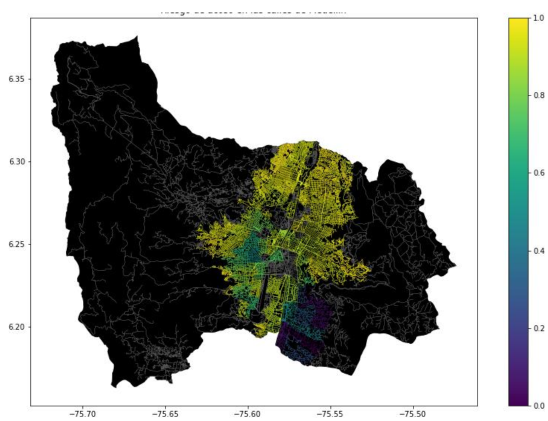


Figure 1. Risk of sexual harassment calculated as a lineal combination of the fraction of households that feel insecure and the fraction of households with income below one minimum wage, obtained from Life Quality Survey of Medellín, in 2017.

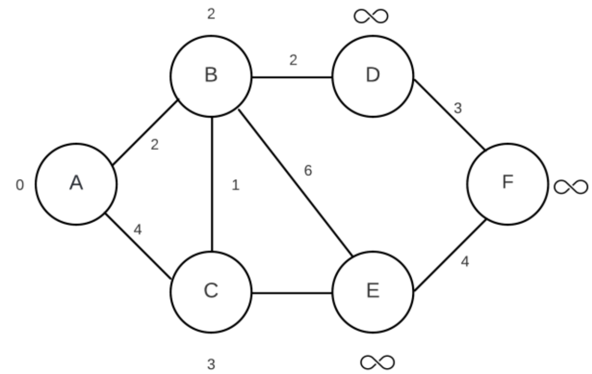
3.2 Constrained Shortest-Path Alternatives

In what follows, we present different algorithms used for constrained shortest path.

3.2.1 Dijkstra's Algorithm

Its an algorithm for finding the shortest path between two nodes in a weighted graph. It evaluates all the distance between nodes, and at the end it gives the least cost path from the start node to the goal node. It also finds the least cost path from the starting node to all the other nodes in the graph [6]

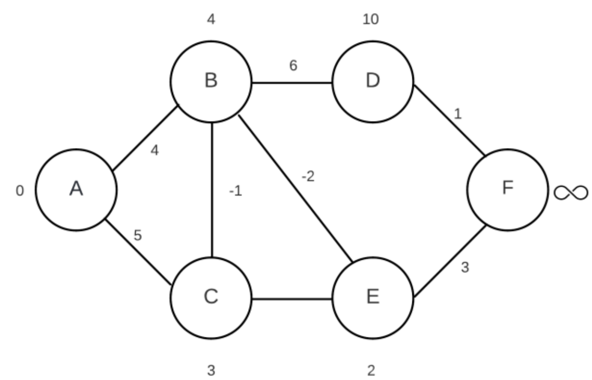
[3]https://github.com/mauriciotoro/ST0245Eafit/tree/master/proyecto/Datasets/



Example of Dijkstra's Algorithm

3.2.2 Bellman–Ford Algorithm

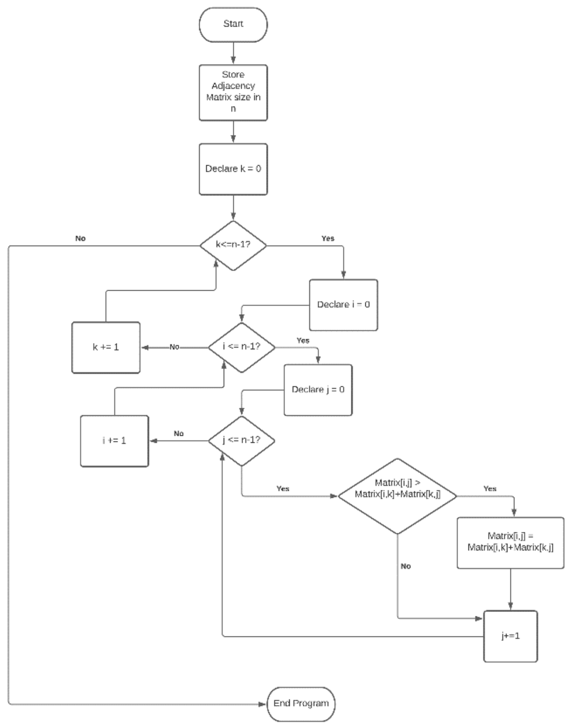
It gives the same result as Dijkstra's Algorithm because it also gives you the shortest path from one node to all other nodes in the graph. The difference between both algorithms is that Bellman-Ford's algorithm works with negative edge weights.[7]



Example of Bellman-Ford Algorithm

3.2.3 Floyd-Warshall Algorithm

The Floyd-Warshall algorithm is designed to find the shortest route from any node to any node in weighed directed graphs allowing negative weights, it iterates over the adjacency matrix of the graph showing the weights of each of the direction of one edge to another, if there an edge is not directed to another its considered “infinite", it iterates k times over the size of the adjacency matrix (which has a size of nxn) it iterates a second time i times over the size of the matrix, and a last j times over the size of the matrix, when it's on the last cycle it checks if the weight in the position (i, j) is bigger than the weights of the position (i,k) and (k,j) added; if the condition is true, it assigns the lowest weight of them; when it has finished iterating it returns a matrix where any position (i,j) is the shortest distance from the node $O(n^3)$ to the node j. It has a worst-case complexity of 2^{64}, and for inputs bigger than 2^{64}tends to overflow.[8]



3.2.4 A* Algorithm

The A* algorithm is designed to be a complete algorithm, if there is a solution it will always find it; it analyzes considering the heuristic value of a node, and after analyzing all the adjacent nodes it returns the shortest possible route. It has a time complexity of $O(b^d)$ where b is the branching factor (Average number of successors per state) and d is the shortest route possible.[9]

The authors are grateful to Prof. Juan Carlos Duque, from Universidad EAFIT, for providing data from Medellín Life Quality Survey, from 2017, processed into a Shapefile.

REFERENCES

1. Alcaldía de Medellín. Ciudades Y Espacios Públicos Seguros Para Las Mujeres Y Las Niñas, 2018. URL: https://www.medellin.gov.co/sicgem_files/adff8f26-2f7e-4308-af10-4b9351f35e6c.pdf
2. Shikhalev, D. Khabibulin, R. and Ulrich Kemloh Wagoum A. 2014. Development of a Safest Routing Algorithm for Evacuation Simulation in Case of Fire. Proceedings of the 6th International Conference on Agents and Artificial Intelligence, 685-690.
3. Galbrun, E. Pelechrinis, K. Terzi, E. 2016. Urban navigation beyond shortest route: The case of safe paths. Information Systems, Volume 57, 160-171.

4. Alain Chabrier,Vehicle Routing Problem with elementary shortest path-based column generation, Madrid,2005

5. Mariusz Dramski, Shortest path problem in static navigation situations in Metody Informatyki Stosowanej, Szczecin , 2011, 173-180

6. Jaakko. 2016. Dijkstra's algorithm explained. URL: https://www.youtube.com/watch?v=CL1byLngb5Q

7. Sambol, M. 2015. Bellman-Ford in 4 minutes — Theory. URL: https://www.youtube.com/watch?v=9PHkk0UavIM

8. Stefan Hougardy, The Floyd–Warshall algorithm on graphs with negative cycles,Bonn, 2010

9.Multiple Authors, A* search algorithm in Wikipedia: the free encyclopedia, https://en.wikipedia.org/wiki/A*_search_algorithm