ROUTE CALCULATION FOR TRAVELLING ON CITIES WITH HIGHER HARASSMENT RATE

Andrés Felipe Caro Gómez
Universidad Eafit
Colombia
afcarog@eafit.edu.co

Andrea Serna
Universidad Eafit
Colombia
Asernac1@eafit.edu.co

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

ABSTRACT

Considering recent events and according to current statistics, street harassment has peaked on the past few years.[1] to counter this social issue competent authorities can implement specialized pathfinding software for safer transit routes. With this kind of technology, we could effectively reduce the median risk a person may take by walking on the city's streets. In this project, we'll be diving into the possibilities pathfinding software may bring to safety regulations and systems.

Keywords

Harassment, pathfinding, route, planning, algorithm, safety.

1. INTRODUCTION

As of 2021, many studies have shown the alarming increase on street harassment incidents,[2] people no longer feel safe walking by themselves, even at daytime or at work hours. Now, the problem in question is, in fact, socially oriented which means that a complete solution may be more complicated than expected. Nonetheless, we can afford to solve the problem temporarily if we use another type of techniques.

By implementing algorithms that plan safe routes based on harassment risk and minimum distance we can effectively reduce the overall incidences.

1.1 Problem

The concrete issue we are trying to tackle is the lack of competent implementation of safety measures to avoid harassment. This is vitally important because authorities have the resources to ensure public safety, but the analog nature of our current system lacks versatility. We will be, then, opening the possibilities to algorithmically solve complex social issues by showing how efficient pathfinding can be.

1.2 Solution

To solve this issue, we have implemented an algorithm that, firstly, reads a data frame containing all the information relevant for calculation of a shortest route from point A to point B. then implements a BFS (Breadth first search) algorithm to return the fastest route based on distance solely.

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

Up next, we explain four related works to pathfinding to prevent street sexual harassment and risks of common crime.

2.1 Preventing Sexual Harassment Using Nearby Search

In this article, the author makes use of a Nearby Search API and a Directions API to implement a safe route pathfinding algorithm based on a heatmap of sexual harassment cases. The idea was to compute the safest route based on the risk associated with each possible path to prevent the probability of an incident happening. The calculations are made by measuring Euclidean distances between point A and point B on a given map.

2.2 Safest route prediction using crime and accidental data

In this work, the author focuses mainly on prediction of safest routes by considering a crime database in the city of New York. It implements a risk score system based on the collective average of the nearest clusters when moving from a point A to a point B and it functions by analyzing the safest route (which would be the one with the lowest risk score) in parallel with the shortest route.

2.3 Safe routing for motorized tourists

For this paper, the author emphasizes on the implementation of Volunteered Geographic Information (VGI) to calculate a crime mapping of a city with high risk of accidentality.

With this, one could calculate safety indexes which represent relative risk in the investigated area. Also, the author uses certain empirical data of urbanistic crime incidents that happened before to mark routing obstacles.

2.4 Safety indexes incorporated to pathfinding

This work approaches the street harassment problematic by proposing a series of algorithms by incorporating safety measures on travelers' pathfinding processes.

The inner working of this algorithm intends to find the most efficient route considering travel time and safety. It also implements integrals for safety indexes and traffic conditions, for example

$$D_L = \int_{s=0}^{s=l} sf(s)\,ds = \int_{s=0}^{s=l} sde^{-ds}ds = \frac{1}{d}\left(1 - e^{-dl} - dle^{-dl}\right)$$

Where, *s* is the distance between two successive vehicles, *d* is the vehicle density along a given link and *l* is the link length.

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

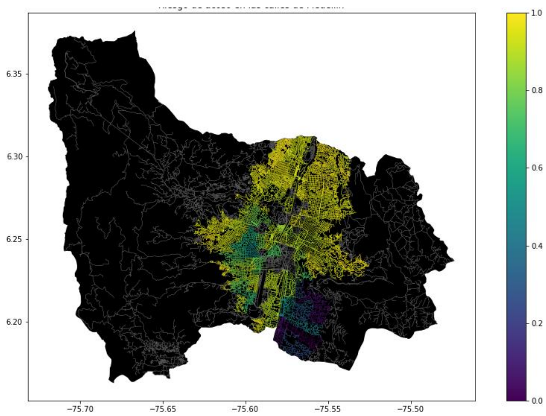
The map of Medellín was obtained from Open Street Maps (OSM)[1] and downloaded using Python OSMnx API[2]. The (i) length of each segment, in meters; (2) indication whether the segment is one way or not, and (3) well-known binary representation of geometries was obtained from metadata provided by OSM.

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

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

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

normalized linear combination. Figure 1 presents the risk of harassment calculated. Map is available at GitHub[3].



3.2 Compact Shortest-path alternatives

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

3.2.1 Backtracking matrix

The idea with this algorithm is to recursively find the shortest path in a matrix given certain conditions such as movement restrictions and borderline limits.

It works by updating the shortest path with every iteration starting from the destination as the recursive stop and going to the beginning, then it highlights which directions to take.

3.2.2 Langrangian relaxation method

The Langrangian relaxation method is a less complex (computationally) iteration of the kSP (*k* shortest path) based algorithm, which consists of a series of deterministic routing paths from the origin point to the destination point. This algorithm approaches the problem by implementing summation methods defined by a function, given for any X ∈ Y:

$$f_3(X) = \sum_{(i,j)\in A} t_{ij}x_{ij} - T.$$

With this the constrained shortest path can be simplified to be attained by solving a dual problem which is:

$$L^* = L(u^*) = \max_{u\geqslant 0} L(u).$$

3.2.3 The pulse algorithm

This algorithm is based on the idea of propagating pulses through a network from a start node to an end node. The algorithm builds a partial path to form a cumulative path with the summation of all given iterations. With each passing calculation the feasible path becomes more effective, ensuring that the optimal path is always found.

3.2.4 The Lee Algorithm

This algorithm is one possible solution for the maze routing problem based on the Breadth-first search. Even though this algorithm provides a optimal solution 100% of the time, it is arguably slower than most and requires more capacity.

The algorithm consists of a queue that enqueues and empties itself by checking each cell from the origin until it reaches the destination node.

4. ALGORITHM DESIGN AND IMPLEMENTATION

In what follows, we explain the data structures and the algorithms used in this work. The implementations of the data structures and algorithms are available at GitHub[4]

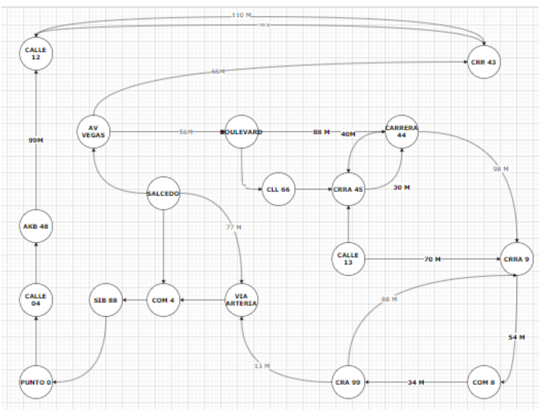
[3]https://github.com/mauriciotoro/ST0245Eafit/tree/master/proyecto/Datasets

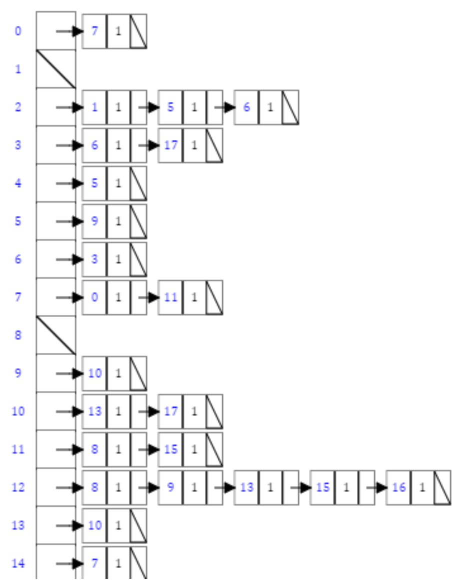
[4] https://github.com/WhiteTeaCup/ST0245-001

4.1 Data structures

As expected for route calculation and pathing algorithms, the selected data structure we chose is a Graph based off an adjacent list.

A Graph could be conceived as a "disordered" data structure consisting of edges and vertices. The edges have assigned values which, in this case, refer to the respective lengths in meters of the streets, while the vertices contain coordinates of locations in the streets.





4.2 Algorithms

In this work, we propose algorithms for the constrained shortest-path problem. The first algorithm calculates the shortest path. The second algorithm calculates the path with the lowest weighted-average risk of harassment.

4.2.1 First algorithm

The algorithm is designed to calculate the shortest path by using BFS, a blind search procedure used to find a certain element in a graph by exploring each subsequence of neighbors uniformly.

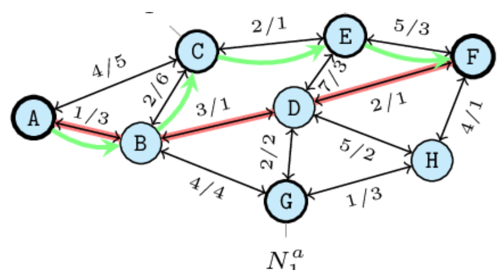


Figure 3: A graph noting a solution to the shortest path problem by implementing Breadth First Search

REFERENCES

1. https://www.nsvrc.org/resource/facts-behind-metoo-movement-national-study-sexual-harassment-and-assault