

ALGORITMO PARA LA SELECCIÓN DE LA MEJOR RUTA: EVITANDO EL ACOSO CALLEJERO

Agustín Rico Piedrahita
Universidad EAFIT
Colombia
aricop@eafit.edu.co

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

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

RESUMEN

Se trabaja el problema de la selección de la mejor ruta posible teniendo en cuenta criterios de seguridad en la ruta, específicamente los relacionados al abuso callejero. La importancia en la solución de dicho problema radica en poder dar una alternativa a los problemas de inseguridad y su influencia en aspectos básicos y fundamentales de la vida de las personas, por ejemplo, sus desplazamientos por la ciudad. Los problemas relacionados son todos aquellos que respondan al problema de la selección de la mejor ruta teniendo en cuenta criterios involucrados en la comodidad o seguridad de la persona que hace la ruta, por ejemplo, factores climáticos.

Palabras clave

Camino más corto restringido, acoso sexual callejero, identificación de rutas seguras, prevención del crimen.

1. INTRODUCCIÓN

El acoso callejero es un fenómeno que aqueja a gran parte de la población, especialmente a las mujeres. Después de un evento de acoso callejero, las víctimas suelen quedar en un escenario muy incómodo y posiblemente con ciertas secuelas físicas y psicológicas. Poder dar alternativas evasivas a este problema es importante para darle lugar y tiempo de acción a estrategias de seguridad más directas que puedan garantizar la seguridad en los desplazamientos de las personas.

1.1. Problema

El problema consiste en la selección de la mejor ruta, teniendo en cuenta datos de acoso callejero en ellas. La solución de este problema brinda una ventaja inmediata y es la de evitar o intentar reducir el número de eventos de acoso callejero, pero además puede ser el punto de partida para análisis profundos que puedan permitir enfocar acciones y estrategias en lugares específicos.

1.2 Solución

El algoritmo que gobierna la solución al problema de recorrer una distancia mínima con un valor de acoso, se resolvió usando iteraciones del algoritmo de Dijkstra. En cada iteración se penaliza la distancia del arco con mayor valor de acoso y se ejecuta una nueva corrida hasta encontrar una ruta más corta que respete el criterio del máximo valor del acoso.

1.3 Estructura del artículo

A continuación, en la Sección 2, presentamos trabajos relacionados con el problema. Posteriormente, en la Sección 3, presentamos los conjuntos de datos y los métodos utilizados en esta investigación. En la Sección 4, presentamos el diseño del algoritmo. Después, en la Sección 5, presentamos los resultados. Finalmente, en la Sección 6, discutimos los resultados y proponemos algunas direcciones de trabajo futuro.

2. TRABAJOS RELACIONADOS

A continuación, explicamos cuatro trabajos relacionados con la búsqueda de caminos para prevenir el acoso sexual callejero y la delincuencia en general.

2.1 Selección de la mejor ruta para peatones y ciclistas en condiciones de invierno

[1] Se investiga la relación entre las condiciones del pavimento en temporadas de invierno y la selección de la mejor ruta para ciclistas y peatones. Para analizar dicha relación, se llevaron a cabo encuestas que preguntaban a peatones y ciclistas por sus rutinas de transporte, sus actitudes y preferencias a la hora de decidirse por una ruta. Los resultados de la encuesta fue que alrededor del 50% de los encuestados cambia su ruta con mucha frecuencia durante el invierno, con respecto a la ruta que toman en verano.

2.2 Modelo basado en GPS para la selección de rutas para ciclistas en San Francisco, California

[2] La selección de la mejor ruta para ciclistas suele diferir de la ruta más corta o la mejor para ir en automóvil. Recolectando datos de los recorridos realizados por ciclistas en San Francisco (California), se entrenó un modelo de regresión logística multinomial que asociaba las probabilidades a las preferencias realizadas por los ciclistas y así sugerir una ruta.

2. Predicción en la ruta de un peatón

[3] Un peatón es sujeto de posibles complicaciones en su ruta de desplazamiento tales como viajar por un vecindario inseguro o desplazarse a la deriva en condiciones climáticas incómodas. Microsoft publicó una patente que exhibe un modelo de inteligencia basado en procesos estocásticos y redes bayesianas con el propósito de satisfacer ciertas preferencias particulares de los peatones en la búsqueda de la mejor ruta.

2.4 Detección de crímenes basado en estimación de rutas y análisis de comportamiento

[4] El objetivo del trabajo fue investigar si era aplicable el uso de rastreos geográficos y de comportamiento para detectar posibles criminales. El método principal de investigación fue un algoritmo que rastreaba las rutas de los crímenes cometidos e identificaba aquellas rutas similares, tanto geográficamente como aspectos como la hora del día y la duración del recorrido. Fue posible concluir y deducir ciertas dinámicas en el comportamiento de las personas y sus rutas al momento de realizar un crimen.

3. MATERIALES Y MÉTODOS

En esta sección, explicamos cómo se recogieron y procesaron los datos y, después, diferentes alternativas de algoritmos del camino más corto restringido para abordar el acoso sexual callejero.

3.1 Recogida y tratamiento de datos

El mapa de Medellín se obtuvo de Open Street Maps (OSM) ¹y se descargó utilizando la API ² OSMnx de Python. La (i) longitud de cada segmento, en metros; (2) la indicación de si el segmento es de un solo sentido o no, y (3) las representaciones binarias conocidas de las geometrías se obtuvieron de los metadatos proporcionados por OSM.

Para este proyecto, se calculó la combinación lineal (CL) que captura la máxima varianza entre (i) la fracción de hogares que se sienten inseguros y (ii) la fracción de hogares con ingresos inferiores a un salario mínimo. Estos datos se obtuvieron de la encuesta de calidad de vida de Medellín, de 2017. La CL se normalizó, utilizando el máximo y el mínimo, para obtener valores entre 0 y 1. La CL se obtuvo mediante el análisis de componentes principales. El riesgo de acoso se define como uno menos la CL normalizada. La Figura 1 presenta el riesgo de acoso calculado. El mapa está disponible en GitHub³.

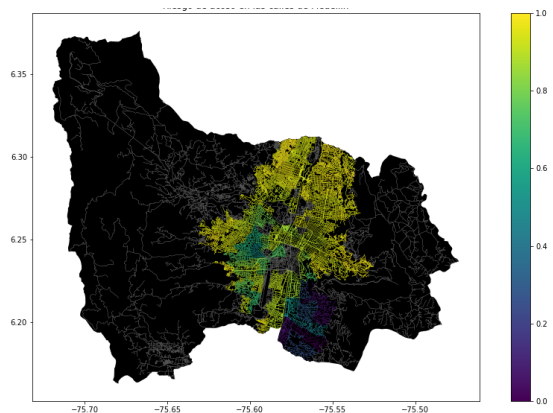


Figura 1. Riesgo de acoso sexual calculado como una combinación lineal de la fracción de hogares que se sienten inseguros y la fracción de hogares con ingresos inferiores a un salario mínimo, obtenida de la Encuesta de Calidad de Vida de Medellín, de 2017.

3.2 Alternativas de camino más corto con restricciones

A continuación, presentamos diferentes algoritmos utilizados para el camino más corto restringido.

3.2.1 Algoritmo de Dijkstra

El algoritmo funciona explorando todos los caminos más cortos que parten desde el origen y que llevan a todos los otros vértices. Una vez se obtiene el camino más desde el vértice de origen y hasta el nodo final de interés, el algoritmo se detiene. Su complejidad es de $O(E + V \log V)$

3.2.2 Algoritmo de Bellman-Ford

El algoritmo tiene un funcionamiento al de Dijkstra pero es aplicable a grafos con ejes de pesos negativos. Su complejidad es de $O(VE)$

3.2.3 Algoritmo de búsqueda A*

Es un algoritmo de búsqueda que asocia una función de evaluación heurística a cada nodo. En cada paso del algoritmo, para cada nodo no evaluado, se calcula la función heurística para todos sus hijos. Su complejidad es $O(b^d)$, donde b es el promedio de nodos hijos por nodo y d es la profundidad del nodo inicial al final.

3.2.4 Algoritmo de Floyd-Warshall

¹ <https://www.openstreetmap.org/>

² <https://osmnx.readthedocs.io/>

³ <https://github.com/mauriciotoro/ST0245Eafit/tree/master/proyecto/Datasets/>

El algoritmo compara todos los posibles caminos a través del grafo entre cada par de vértices, utilizando una matriz de adyacencia haciendo uso de la transitividad entre las conexiones de los nodos. Dicha comparación se actualiza con la mejor solución hasta haber hecho el recorrido total del grafo. Su complejidad es de $O(V^3)$.

4. DISEÑO E IMPLEMENTACIÓN DEL ALGORITMO

A continuación, explicamos las estructuras de datos y los algoritmos utilizados en este trabajo. Las implementaciones de las estructuras de datos y los algoritmos están disponibles en Github⁴.

4.1 Estructuras de datos

Para el grafo, se tiene una implementación con listas de adyacencia, en donde cada elemento de cada lista contiene el nodo destino y el peso, valor de acoso y nombre del arco correspondiente, tal como se muestra en la Figura 2.

Ahora, se describirán las estructuras usadas cada una de las corridas del algoritmo de Dijkstra: se utilizan dos arreglos que almacenan tanto la distancia como el nodo predecesor de la ruta hasta cada nodo, desde el nodo origen. Ambos arreglos son de tamaño V (V = número de nodos).

Posterior a cada corrida de Dijkstra, se ejecuta un algoritmo que obtiene la ruta ordenada hasta el origen, su distancia total y su valor de acoso total. El del tamaño del arreglo de la ruta es igual al número de nodos que la componen, y tanto la distancia como el valor del acoso son *float*.

Se implementó la versión del algoritmo de Dijkstra que utiliza una cola de prioridad. Para esta cola de prioridad se utiliza un montículo binario (binary heap).

Finalmente, el algoritmo principal que reúne los dos anteriores, conserva sus mismas estructuras de datos.

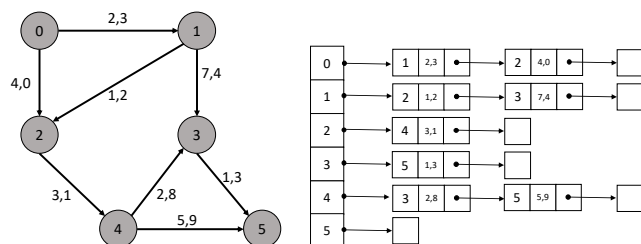


Figura 2: Un ejemplo de mapa de calles y su representación como lista de adyacencia

4.2 Algoritmos

En este trabajo, proponemos algoritmos para el problema del camino más corto restringido. El primer algoritmo calcula el camino más corto sin superar un riesgo medio ponderado de acoso r . El segundo algoritmo calcula el camino con el menor riesgo medio ponderado de acoso sin superar una distancia d .

4.2.1 Primer algoritmo

El primer algoritmo realiza iteraciones del algoritmo de Dijkstra hasta encontrar una ruta más corta que no supere una media ponderada de riesgo de acoso r . Para ello, en cada iteración del algoritmo, se calcula la ruta más corta sin tener en cuenta el criterio de acoso, luego se calcula el riesgo de acoso para dicha ruta y, de ser sobrepasado, se penaliza el peso (distancia) del arco con mayor nivel de acoso ponderado de la ruta (dándole el valor de infinito). Este proceso se repite hasta encontrar una ruta más corta (cuyo ninguno de sus arcos haya sido penalizado previamente) que respete el criterio de riesgo máximo de acoso. En las Figuras 3 y 4 se ilustra el algoritmo. La ruta verde es la primera ruta que encuentra el algoritmo, y las flechas azules y rojas son las penalizaciones que se realizan en las corridas posteriores hasta dar como resultado la ruta amarilla, que respeta un valor de acoso total de $r = 1.5$. Los valores que marcan cada arco son la distancia y el valor de acoso correspondiente.

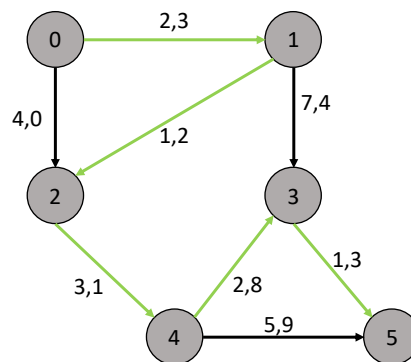


Figura 3: Grafo inicial y ruta más corta original.

⁴ <http://www.github.com/?????????/.../proyecto/>

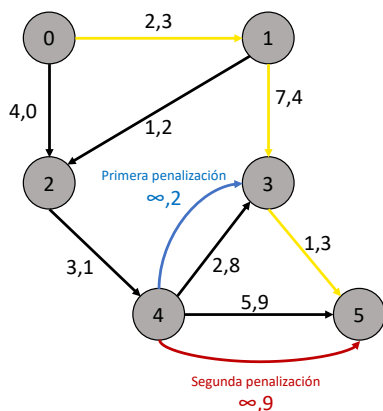


Figura 4: Grafo con penalizaciones ejecutadas en las primeras dos iteraciones (flecha azul y roja, respectivamente) y ruta final que respeta el criterio de máximo acoso (amarilla).

4.2.2 Segundo algoritmo

El funcionamiento del segundo algoritmo es idéntico al del primero, con la diferencia de que se invierten los objetivos de la distancia y la cantidad de acoso. Es decir, las penalizaciones de modificar el acoso de un arco a un valor de infinito se hacen sobre el arco de mayor distancia en la ruta.

4.4 Análisis de la complejidad de los algoritmos

La complejidad del algoritmo de Dijkstra utilizando colas de prioridad es de $O(E + V \log(V))$ y el algoritmo iterativo, en el peor de los casos, realiza tantas iteraciones como arcos (E). Por lo tanto, la complejidad es:

Algoritmo	Complejidad temporal
Dijkstra Iterativo	$O(E^2 + EV \log(V))$

Tabla 1: Complejidad temporal de *Dijkstra Iterativo*. Donde V es el número de nodos y E el número de arcos

Estructura de datos	Complejidad de la memoria
Grafo en Listas de Adyacencia	$O(V^2)$
Cola de prioridad, arreglos de pesos y predecesores	$O(V)$

Tabla 2: Complejidad de memoria de *Dijkstra iterativo*. Donde V es el número de nodos y E el número de arcos

4.5 Criterios de diseño del algoritmo

Explique por qué el algoritmo fue diseñado de esa manera. Utilice criterios objetivos. Los criterios objetivos se basan en la eficiencia, que se mide en términos de tiempo y memoria. Ejemplos de criterios NO objetivos son: "estaba enfermo", "fue la primera estructura de datos que encontré en Internet", "lo hice el último día antes del plazo", "es más fácil", etc. Recuerda: Este es el 40% de la calificación del proyecto.

5. RESULTADOS

En esta sección, presentamos algunos resultados cuantitativos sobre el camino más corto y el camino con menor riesgo.

5.1.1 Resultados del camino más corto

A continuación, presentamos los resultados obtenidos para el camino más corto, sin superar un riesgo medio ponderado de acoso r , en la Tabla 3.

Origen	Destino	Distancia más corta	Sin exceder r
Universidad EAFIT	Universidad de Medellín	2397.34	35
Universidad de Antioquia	Universidad Nacional	9605.61	90
Universidad Nacional	Universidad Luis Amigó	1649.56	19

Tabla 3. Distancias más cortas sin superar un riesgo de acoso medio ponderado r .

5.1.2 Resultados de menor riesgo de acoso

A continuación, presentamos los resultados obtenidos para el trayecto con menor riesgo de acoso medio ponderado, sin superar una distancia d , en la Tabla 4.

Origen	Destino	Acoso más bajo	Sin exceder d
Universidad EAFIT	Universidad de Medellín	28.42	3500
Universidad de Antioquia	Universidad Nacional	41.19	45000
Universidad Nacional	Universidad Luis Amigó	17.90	1750

Tabla 3. Menor riesgo de acoso ponderado sin superar una distancia d (en metros).

5.2 Tiempos de ejecución del algoritmo

En la Tabla 4, explicamos la relación de los tiempos medios de ejecución de las consultas presentadas en la Tabla 3.

	Tiempos medios de ejecución (s)
Universidad EAFIT a Universidad de Medellín	3.89 s
De la Universidad de Antioquia a la Universidad Nacional	5.70 s
De la Universidad Nacional a la Universidad Luis Amigó	1.53 s

Tabla 4: Tiempos de ejecución del nombre del *Dijkstra Iterativo*

6. CONCLUSIONES

Los caminos obtenidos usando uno de los algoritmos son considerablemente diferentes a los obtenidos usando el otro. Es decir, es muy claro el cambio del objetivo entre los dos algoritmos. Sin embargo, para valores de tolerancias (tanto de acoso como distancia mínimos), es posible no obtener rutas que cumplan dicho criterio y se retorna la última ruta explorada mas no la mejor obtenida en toda la corrida del algoritmo.

6.1 Trabajos futuros

Como se dijo antes, en caso de no lograr el valor de tolerancia deseado, los algoritmos retornan la última ruta explorada. Dicha característica puede ser una mejora para el futuro: no retornar la última ruta explorada sino la mejor encontrada en todo el recorrido del algoritmo.

AGRADECIMIENTOS

Los autores agradecen al profesor Juan Carlos Duque, de la Universidad EAFIT, por facilitar los datos de la Encuesta de Calidad de Vida de Medellín, de 2017, procesados en un *Shapefile*.

REFERENCIAS

1. Magne Fossum & Eirin Olaussen Ryeng (2022) Pedestrians' and bicyclists' route choice during winter conditions, Urban, Planning and Transport Research, 10:1, 38-57, DOI: 10.1080/21650020.2022.2034524
2. Jeffrey Hood, Elizabeth Sall & Billy Charlton (2011) A GPS-based bicycle route choice model for San Francisco,

California, Transportation Letters, 3:1, 63 75, DOI: 10.3328/TL.2011.03.01.63-75

3. Ivan J. Tashev, Jeffrey D. Couckuyt, Neil W. Black, John C. Krumm, Ruston Panabaker, and Michael Lewis Seltzer. 2012. Pedestrian route production. (January 2012).
4. Borg, Anton & Boldt, Martin & Eliasson, Johan. (2017). Detecting Crime Series Based on Route Estimation and Behavioral Similarity. 1-8. 10.1109/EISIC.2017.10.