



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE CIENCIAS

HEURÍSTICAS DE OPTIMIZACIÓN COMBINATORIA

Proyecto 1: Recocido simulado

Semestre 2026-1

Alumno:

Medina Peralta Joaquín—— 320202513

Fecha de Entrega: 15 de septiembre de 2025

Introducción

Para este proyecto introductorio a las heurísticas de optimización combinatoria, se decidió implementar el recocido simulado por umbrales para resolver el problema del agente viajero, siendo un clásico problema de teoría de gráficas y un problema NP -completo, es decir, no existe un algoritmo que lo resuelva de manera óptima en un tiempo razonable en instancias de ciudades grandes.

De este modo, se estará trabajando en instancias de tamaño 40 y 150, donde el tamaño de la gráfica es $40!$ y $150!$ respectivamente, haciendo usando el algoritmo de dijkstra con complejidad de $O(|E|\log(|V|))$ tomando el tiempo de la edad del universo para poder resolverse.

Por lo que, usando la heurística propuesta, podremos obtener trayectorias con resultados óptimos en poco tiempo pero sin asegurar que esa solución al problema sea la mejor para esa instancia de ciudades. Por último, la implementación de la heurística será realizado en el lenguaje de programación **Rust** por su ventaja del uso de vectores y su velocidad para acceder en memoria estos elementos.

Problema

Como se menciona en la sección anterior, el problema a resolver para este proyecto es el del agente viajero, el cual consiste en encontrar el recorrido más corto que visite a todas las ciudades una sola vez, donde habrá un conjunto de n ciudades con una ciudad de origen y una ciudad destino.

De este modo, intentar resolver el problema usando fuerza bruta, lo único que provocara que tenga una complejidad de $O(n!)$ o con el algoritmo de dijkstra teniendo una complejidad de $O(|E|\log(|V|))$, por lo que resolverlo con este enfoque será imposible y es necesario una heurística de optimización combinatoria.

Dicho esto, tendremos que el conjunto de ciudades estarán dadas por los nuestro planeta, donde para medir la distancia entre dos ciudades se usará distancia geodésica.

De este modo, sean dos puntos en la superficie terrestre (φ_1, λ_1) y (φ_2, λ_2) con latitudes y longitudes en radianes. Definimos

$$\Delta\varphi = \varphi_2 - \varphi_1, \quad \Delta\lambda = \lambda_2 - \lambda_1.$$

La cantidad intermedia usada en Haversine es

$$a = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos \varphi_1 \cos \varphi_2 \sin^2\left(\frac{\Delta\lambda}{2}\right).$$

La distancia (longitud del gran círculo) se puede escribir como

$$d = R \cdot C, \quad C = 2 \operatorname{atan2}(\sqrt{a}, \sqrt{1-a}).$$

Es decir,

$$d = 2R \operatorname{atan2}(\sqrt{a}, \sqrt{1-a})$$

Por lo que, se usará una gráfica donde los vértices serán las ciudades (conectadas solo si existe una conexión en el planeta) y las artistas estarán dadas por la distancia natural entre ellas.

Heurística

Para el proyecto, como se menciona en la introducción, se utiliza una variante de la heurística del recocido simulado, donde la aceptación de soluciones se basa en un umbral que controla la calidad mínima aceptable, donde una solución s' aceptada está dada por:

$$\Delta \leq \Theta.$$

donde:

- $f(s)$: función objetivo evaluada en la solución s .
- s : solución actual.
- s' : solución candidata.
- $\Delta = f(s') - f(s)$: cambio en la función objetivo.
- Θ : valor del umbral en la iteración actual.

Esto significa que el algoritmo permite aceptar soluciones peores siempre que la pérdida no sea mayor al umbral actual.

Donde el escoger un vecino será por números pseudo aleatorios con una semilla y una instancia inicial, para generar un bloque b_1 con una cantidad c y compararlo con otro bloque b_2 con la misma cantidad.

Análisis del problema

Diseño de la solución

Experimentación

Resultados