

# El Problema del Viajante Resuelto con el Bioalgoritmo de las Luciérnagas

## 1 Introducción

El Problema del Viajante (TSP) es uno de los problemas más estudiados en la teoría de la optimización combinatoria. El objetivo es encontrar la ruta más corta posible que permita a un viajante visitar un conjunto de ciudades exactamente una vez y regresar al punto de origen. Este problema pertenece a la clase NP-difícil, lo que significa que no existe un algoritmo eficiente conocido para resolverlo en tiempo polinómico para todas las instancias.

El TSP tiene aplicaciones prácticas en logística, planificación de rutas y diseño de circuitos. Dada la cantidad exponencial de rutas posibles conforme aumenta el número de ciudades, encontrar una solución óptima es un desafío computacional significativo.

## 2 ¿Por qué este problema y no otro?

El TSP ha sido elegido por varias razones:

- **Relevancia Práctica:** La optimización de rutas es crucial en áreas como la logística y la planificación de rutas de transporte. Resolver el TSP puede mejorar la eficiencia en la gestión de recursos y reducir costos operativos.
- **Desafío Computacional:** Su naturaleza NP-difícil lo convierte en un banco de pruebas ideal para nuevas técnicas de optimización como el algoritmo de las luciérnagas.
- **Aplicación del Algoritmo:** El algoritmo de las luciérnagas es adecuado para problemas de optimización complejos y multimodales como el TSP. ( ? )

## 3 Herramientas a Utilizar

El *Algoritmo de las Luciérnagas* ref es un método bioinspirado que se basa en el comportamiento de las luciérnagas para encontrar soluciones óptimas en problemas de optimización. Se utilizarán las siguientes herramientas:

- **Lenguaje de programación:** Python, por su facilidad para implementar algoritmos de optimización y por la variedad de bibliotecas que ofrece.
- **Entorno de desarrollo:** Se usará Jupyter Notebook para implementar y probar el algoritmo de las luciérnagas.
- **Dataset:** Un conjunto de ciudades y las distancias entre ellas, facilitado para resolver el problema mediante el algoritmo.

jy?

## 4 Modelo Matemático del Problema del Viajante

El Problema del Viajante (TSP) se puede formular matemáticamente como sigue:

- Un conjunto de  $n$  ciudades  $C = \{c_1, c_2, \dots, c_n\}$
- Una matriz de distancias  $D$ , donde  $D_{ij}$  representa la distancia entre las ciudades  $c_i$  y  $c_j$

El objetivo es minimizar la función de costo:

$$Z = \sum_{i=1}^n \sum_{j=1}^n D_{ij} X_{ij}$$

matriz Simétrica  
Dirección

Donde:

- $X_{ij}$  es una variable binaria que toma el valor 1 si el viajante viaja de la ciudad  $i$  a la ciudad  $j$ , y 0 en caso contrario.

### 4.1 Restricciones

El modelo debe cumplir las siguientes restricciones:

ref

- Cada ciudad debe ser visitada exactamente una vez.
- Se deben eliminar subciclos para evitar trayectorias no deseadas.

## 5 Optimización con el Algoritmo de las Luciérnagas

El comportamiento del algoritmo se basa en la siguiente formulación:

ref

$$x_i(t+1) = x_i(t) + \beta e^{-\gamma d_{ij}} (x_j(t) - x_i(t)) + \alpha \cdot rand$$

Donde:

- $x_i(t)$  es la posición de la luciérnaga  $i$  en el tiempo  $t$ .

- $\beta$  es la intensidad de atracción.
- $\gamma$  es el coeficiente de absorción de luz.
- $d_{ij}$  es la distancia entre las luciérnagas  $i$  y  $j$ .
- $\alpha$  es el factor de aleatoriedad.
- $rand$  es un término aleatorio.

Este algoritmo permite explorar el espacio de soluciones de manera eficiente, evitando quedarse atrapado en óptimos locales.

## 6 Soluciones Propuestas

Para abordar el TSP, existen dos formulaciones comunes: la formulación MTZ (Miller, Tucker y Zemlin) y la formulación DFJ (Dantzig, Fulkerson y Johnson).

### 6.1 Formulación MTZ

La formulación MTZ busca minimizar el costo total del viaje, sujetándose a restricciones de no generar subciclos.

$$\min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

### 6.2 Formulación DFJ

La formulación DFJ impone restricciones adicionales para evitar sub-recorridos y generar una única solución continua.

$$\min \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

Las tres primeras restricciones aseguran que cada ciudad sea visitada exactamente una vez.

## 7 Relación con el Dataset

El dataset de ciudades de la India proporcionado es ideal para instanciar el Problema del Viajante. Cada ciudad representa un nodo en el grafo, y las distancias entre ellas se pueden calcular utilizando métricas como la distancia euclidiana. Estos valores alimentan directamente el algoritmo de las luciérnagas para resolver el TSP, permitiendo encontrar la ruta más corta que visita todas las ciudades una vez.

El conjunto de datos ofrece una representación real de los desafíos logísticos que enfrenta el TSP, lo que refuerza la aplicación práctica de este algoritmo bioinspirado.

## 8 Referencias

- González-Santander, G. "Tres métodos diferentes para resolver el problema del viajante." *Baobab Soluciones*, 1 de octubre de 2020. Disponible en: <https://baobabsoluciones.es/blog/2020/10/01/problema-del-viajante/>.
- "Problema del viajante." *Wikipedia, la enciclopedia libre*, 8 de septiembre de 2024. Disponible en: [https://es.wikipedia.org/wiki/Problema\\_del\\_viajante](https://es.wikipedia.org/wiki/Problema_del_viajante).
- Vázquez, J. "Revisión de los Algoritmos Bioinspirados." *Universidad Nacional Autónoma de México*, julio de 2014. Disponible en: [https://www.fis.unam.mx/~javazquez/files/Met\\_num/Algoritmosbioinspiradosfinal.pdf](https://www.fis.unam.mx/~javazquez/files/Met_num/Algoritmosbioinspiradosfinal.pdf).
- Wikipedia. (n.d.). *Algoritmo firefly*. Recuperado el 8 de septiembre de 2024, de [https://es.wikipedia.org/wiki/Algoritmo\\_firefly](https://es.wikipedia.org/wiki/Algoritmo_firefly).
- Visión electrónica. (2021). *Firefly algorithm for facility layout problem optimization*. 15(2), 218–225. <https://doi.org/10.14483/22484728.17474>.