

# El problema del agente viajero

Mauricio Trejo

Alejandra

Andrés

7 de diciembre de 2019

## 1 Planteamiento del problema

Supongamos que tenemos  $k$  ciudades conectadas por rutas bidireccionales.

Un agente de ventas que reside en la ciudad 1 debe viajar a cada una de las ciudades al menos una vez y después regresar a su hogar.

El problema del agente viajero consiste en encontrar el trayecto que atraviese todas las ciudades y cuya distancia sea mínima.

### 1.1 Planteamiento gráfico

Sea  $\mathcal{G}$  la gráfica que representa este problema, entonces

1. el conjunto de los nodos de  $\mathcal{G}$  queda dado por  $\mathcal{N}_{\mathcal{G}} = \{n_1, n_2, \dots, n_k\}$  donde cada nodo representa una ciudad.
2. el conjunto de los arcos de  $\mathcal{G}$  queda dado por  $\mathcal{A}_{\mathcal{G}} = \{a_{ij} \mid i \neq j, i, j = 1, 2, \dots, k\}$  donde cada elemento  $a_{ij}$  representa la ruta entre la ciudad  $i$  y la ciudad  $j$ .
3. el conjunto de distancias  $\mathcal{D}$  queda dado por  $\mathcal{D} = \{d_{ij} \mid i \neq j, i, j = 1, 2, \dots, k\}$  donde  $d_{ij}$  representa la distancia del arco  $a_{ij}$ .

### 1.2 Planteamiento algebraico

Sea  $x_{ij}$  para  $i, j = 1, 2, \dots, k, i \neq j$  tal que

$$x_{ij} = \begin{cases} 1 & \text{si el arco } a_{ij} \text{ está en la ruta óptima} \\ 0 & \text{e.o.c} \end{cases}$$

El problema del agente viajero se puede plantear como un PLE de la manera siguiente:

$$\begin{aligned}
 &\text{minimizar} && \sum_{i=1}^k \sum_{i \neq j=1}^k d_{ij} x_{ij} \\
 &\text{sujeto a} && \sum_{i=1}^k \sum_{i \neq j=1}^k x_{ij} \geq k \\
 &&& d_{ij} \in \mathbb{N}, x_{ij} \in \{0, 1\} \quad \forall i, j = 1, 2, \dots, k, i \neq j
 \end{aligned} \tag{1}$$

## 2 Algoritmos genéticos

La mayoría de los algoritmos de optimización y búsqueda clásicos utilizan un procedimiento determinístico para alcanzar una solución óptima: comienzan desde un punto cualquiera y determinan la dirección de búsqueda mediante una regla predefinida [1].

La naturaleza determinística de los métodos tradicionales tiene las siguientes desventajas:

1. La convergencia a una solución optimal depende de la solución inicial de la que se parte.
2. Pueden quedar atrapados en óptimos locales.
3. No son eficientes en el proceso de solución de problemas con un espacio de búsqueda discreto.

Los dos primeros problemas se pueden resolver mediante la aleatoriedad, pero el último no.

La *computación evolutiva* se refiere a un conjunto de algoritmos estocásticos de búsqueda que se basan de una forma u otra en la idea del proceso de la evolución de Darwin. Su mayor atributo es que no sufren de los males que aquejan a los algoritmos clásicos.

El funcionamiento de los algoritmos evolutivos se puede resumir de manera general en cuatro pasos [1]:

1. Se mantiene una población de posibles soluciones para el problema.
2. Se modifican las soluciones almacenadas y se descartan algunas de acuerdo con una medida de adaptación al entorno.

3. El proceso de modificación y selección obliga a la población a evolucionar a través de las mejores regiones del espacio de búsqueda.
4. Las modificaciones hechas a la población permiten mezclar información de los padres que se transmite a los descendientes (*cruce*) o introducir innovación dentro de la población (*mutaciones*).

Los algoritmos genéticos se distinguen de los demás algoritmos evolutivos por tres razones: su esquema binario, el método de selección que utilizan y el uso del cruce de información para producir variaciones en la población [1].

**Algoritmo genético (reescribir como lista):** el sistema parte de una población inicial de individuos que codifican, mediante alguna representación genética, soluciones candidatas al problema propuesto. Esta población de individuos (a los que se denomina cromosomas) evoluciona en el tiempo a través de un proceso de competición y variación controlada. Cada cromosoma de la población tiene asociada una medida de adaptación para determinar qué cromosomas serán seleccionados para formar parte de la nueva población en el proceso de competición. La nueva población se creará utilizando operadores genéticos de cruce y mutación. Este ciclo evolutivo continúa hasta que se verifique una determinada condición de parada: que se hayan realizado un determinado número máximo de evaluaciones de individuos, que la población haya evolucionado durante un número máximo de generaciones, que se haya alcanzado una solución con un determinado valor de la función de adaptación (o de parte de ella), que se estacione la población por no generarse individuos nuevos durante un determinado número de generaciones, etc.

### 3 Explicación del código

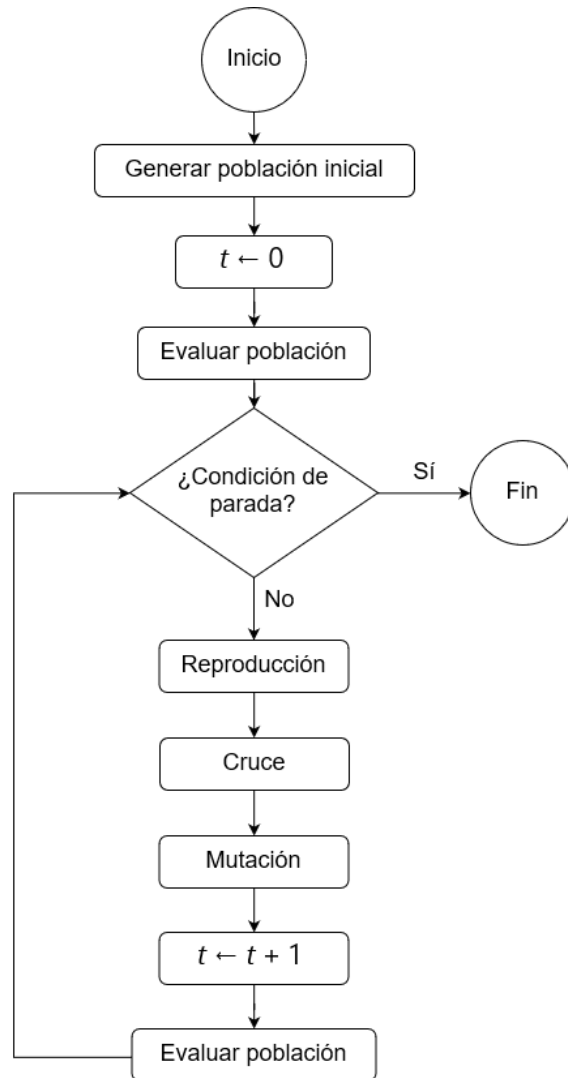


Figure 1: Diagrama de flujo de un Algoritmo Genético básico