

# Algoritmo genético aplicado al turismo de Panamá

## Genetic algorithm applied to tourism in Panama

Trujillo-González, Julio



Julio Trujillo-González

julio.trujillo@up.ac.pa

Universidad de Panamá, Panamá

### Investigación y Pensamiento Crítico

Universidad Católica Santa María La Antigua, Panamá

ISSN: 1812-3864

ISSN-e: 2644-4119

Periodicidad: Cuatrimestral

vol. 12, núm. 1, 2024

[investigacion@usma.ac.pa](mailto:investigacion@usma.ac.pa)

Recepción: 14 Abril 2023

Aprobación: 04 Diciembre 2023

URL: <http://portal.amelica.org/ameli/journal/333/3334711005/>

DOI: <https://doi.org/10.37387/ipc.v12i1.373>



Esta obra está bajo una [Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).

**Resumen:** Este artículo tiene como objetivo determinar una ruta turística en la Ciudad de Panamá mediante la implementación de un algoritmo genético; donde estamos ante un problema clásico de la Teoría de Grafos que es encontrar un camino que pase por varios puntos de interés, con la restricción que pase sólo una vez y terminar en un lugar, y si añadimos otra restricción que sea la ruta más corta, el problema se convierte uno de tipo TSP (Traveling Salesman Problem). Mediante los experimentos computacionales se establece que el algoritmo genético genera una ruta más corta y que satisface las restricciones. Se concluye que el algoritmo genético brinda soluciones eficientes en un tiempo corto, por lo cual se recomienda su implementación en otros tipos de problemas de optimización.

**Palabras clave:** grafo, camino hamiltoniano, traveling salesman problem, algoritmo genético, turismo..

**Abstract:** This article aims to determine a tourist route in Panama City through the implementation of a genetic algorithm; where we are faced with a classic problem of Graph Theory, which is to find a path that passes through several points of interest, with the restriction that it passes only once and ends in one place, and if we add another restriction that is the shortest route, the problem becomes a TSP (Traveling Salesman Problem). Through computational experiments, it is established that the genetic algorithm generates a shortest path that satisfies the restrictions. It is concluded that the genetic algorithm provides efficient solutions in a short time, for which its implementation is recommended in other types of optimization problems.

**Keywords:** graph, hamiltonian path, traveling salesman problem, genetic algorithm, tourism..

## INTRODUCCIÓN

El presente artículo es una continuación de la idea presentada por Trujillo (2019) que tenía como objetivo aplicar los grafos hamiltonianos para encontrar un tour turístico. Utilizaremos los algoritmos genéticos para mostrar una alternativa de modelización y para resolver un problema de ruta, como indica el nombre “Algoritmo genético” constituye una técnica de búsqueda y optimización de parámetros, inspirada en el principio Darwiniano de la selección natural y reproducción genética.

Tomando la definición de Goldberg ‘los algoritmos genéticos son algoritmos de búsqueda basados en la mecánica de selección natural y de la genética natural. Combina la supervivencia del más apto entre estructuras de secuencias con un intercambio de información estructurado, aunque aleatorizado, para

constituir así un algoritmo de búsqueda que tenga algo de las genialidades de las búsquedas humanas' (Booker et. al, 1989).

**Bases biológicas**

La naturaleza, los individuos de una especie que la llamamos población compiten por los recursos, así teniendo más posibilidades de sobrevivir y pasar sus genes a la siguiente generación. Las combinaciones de las características de estos individuos, pueden hacer que los descendientes estén mejor adaptados al ambiente que las generaciones pasadas. De esta forma, las especies evolucionan adaptándose al entorno en cada generación (Beasley et al, 1993).

**Representación**

La representación de las posibles soluciones del espacio de búsqueda de un problema define a la estructura del cromosoma para ser manipulado por el algoritmo.

Los principales tipos de representación son:

**TABLA 1**  
Tipos de representación

Representación	Problema
Binaria	Punto flotante, Enteros
Números reales	Punto flotante
Permutaciones de símbolos	Basado en el Orden
Símbolos repetidos	Agrupamiento

Las primeras representaciones fue codificación binaria presentadas en los trabajos de John Holland, donde se le asigna un determinado número de bits a cada parámetro y se discretiza la variable por cada gen.

La hipótesis del bloque de construcción o teorema del Esquema de Holland que subyace en los algoritmos genéticos establece que la solución óptima del problema estudiado está formada por pequeños bloques de construcción y, a medida que reunimos más bloques de construcción, la solución se va acercando a la solución óptima.

Este teorema implica que los esquemas de orden inferior con aptitud superior a la media tendrán más probabilidades de sobrevivir en la población. Esta es la llamada hipótesis o conjetura del bloque de construcción de Goldberg.

Ejemplo. Sea # = 1 # 0 ## 1 # es un esquema que representa un conjunto de cromosomas con 2. genes diferentes. Donde un esquema es una cadena que consiste de sólo 0, 1 o \* (Conroy, 1991).

**Algoritmo**

El algoritmo toma como punto de partida una población de individuos. Cada individuo es una posible solución al problema a modelar, ellos están asociados un ajuste de acuerdo a la bondad. En la naturaleza sería una medida de eficiencia del individuo en que se adapta al ambiente.

La generación se obtiene por dos medios que son:

- Cruce o reproducción. Para crear un par de individuos, seleccionamos dos individuos usualmente de la generación actual, y parte de sus cromosomas son intercambiados para crear descendencia.
- Copia. Un grupo de individuos pasa a la siguiente generación sin sufrir ningún cambio directamente.

El flujo de la descendencia (figura 1) que vamos a considerar es la presentada por Reina et. al (2020)

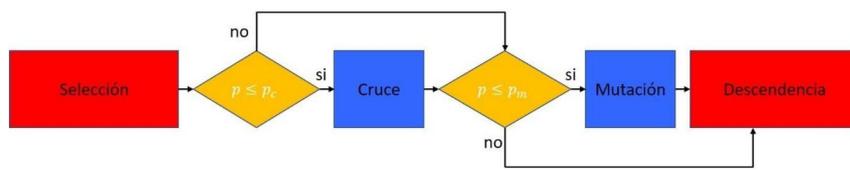


Figura 1. Offspring de una población

**FIGURA 1**  
Offspring de una población

El operador mutación en la figura 1 tiene el propósito de actualizar periódica y aleatoriamente la población, introduciendo nuevos patrones en los cromosomas y fomentar las búsquedas en las áreas inexploradas del espacio de la solución.

La mutación se realiza en una porción de los cromosomas, y se realiza en una porción aleatoria de los genes.

## MÉTODO

### Objetivos:

- Determinar una ruta turística en la Ciudad de Panamá con la distancia mínima de recorrido. Aplicar el algoritmo genético para encontrar una solución.
- Ilustrar el uso de la Matemática para resolver un problema real de la sociedad panameña.

### Población y Muestra

La población se puede considerar los centros turísticos en Panamá, tomando una muestra de 15 sitios por conveniencia que son los siguientes : 1-Multicentro, 2-Casco Viejo, 3- Cinta costera, 4-Centro de visitantes de Miraflores, 5-Panamá vieja, 6-Albrook Mall, 7-Centro de visitantes del Parque Natural Metropolitano, 8-Centro Natural Punta Culebra, 9-Isla Flamenco, 10-Biomuseo, 11-Mirador de las Américas, 12-Cerro Ancón, 13- Calzada de Amador, 14-Museo del Canal Interoceánico de Panamá, 15-Parque Municipal de Summit.

### Instrumento

Teniendo en cuenta que debemos pasar por cada sitio sin repetir, además teniendo la ruta más corta con respecto a la distancia en kilómetros. Se construye la tabla 2 que son las distancia entre los puntos usando Google Maps.

Tabla 2. Datos de la muestra por curso y área de conocimiento

**TABLA 2**  
 Datos de la muestra por curso y área de conocimiento

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4.1	3.1	12.4	6.1	5.3	4.7	13.8	14.9	11.5	10.6	5.6	13.5	4.4	22.1
4.1	0	2.3	10.2	11.1	5.6	5.1	10.7	11.9	7.7	7.5	2.8	10.5	0.29	27.5
3.1	2.3	0	10.8	8.8	6.2	5.7	12.1	13.3	9.9	8.9	4	11.9	2.8	20.2
12.4	10.2	10.8	0	19.7	8.6	12.4	16.4	17.6	14.2	13.2	8.7	16.2	12	11.2
6.1	11.1	8.8	19.7	0	9.8	8	17.5	18.6	15.2	14.2	11.2	17.2	10.1	22.2
5.3	5.6	6.2	8.6	9.8	0	1.6	12	13.2	9.8	8.8	4.3	11.8	7.6	18.3
4.7	5.1	5.7	12.4	8	1.6	0	12.2	13.3	9.9	9	7.4	11.9	7.7	21.8
13.8	10.7	12.1	16.4	17.5	12	12.2	0	2.1	3.2	8	8.4	0.7	7.9	25.7
14.9	11.9	13.3	17.6	18.6	13.2	13.3	2.1	0	4.3	9.1	9.6	1.8	9.1	26.9
11.5	7.7	9.9	14.2	15.2	9.8	9.9	3.2	4.3	0	4.9	5.3	3	5.1	22.6
10.6	7.5	8.9	13.2	14.2	8.8	9	8	9.1	4.9	0	8.5	10.6	8	25.4
5.6	2.8	4	8.7	11.2	4.3	7.4	8.4	9.6	5.3	8.5	0	7.7	3.5	18.6
13.5	10.5	11.9	16.2	17.2	11.8	11.9	0.7	1.8	3	10.7	7.7	0	7.2	25.6
4.4	0.29	2.8	12	10.1	7.6	7.7	7.9	9.1	5.1	8	3.5	7.2	0	20
22.1	27.5	20.2	11.2	22.2	18.3	21.8	25.7	26.9	22.6	2.5	18.6	25	20	0

**Procedimiento de recogida y análisis de datos**

En esta sección se presenta los detalles de los experimentos computacionales realizado con Python, los paquetes random, json, matplotlib y deap, y se ejecutó en una computadora portátil con procesador Intel Core i7-8750H de 2.20 GHz, y 16.0 GB de memoria RAM.

En la tabla 3 se presenta los parámetros de control para el algoritmo genético, los cuales se obtuvieron de forma exploratorio probando valores y experiencia de estudios de algoritmos genéticos.

Tabla 3. *Parámetros de control del algoritmo genético*

**TABLA 3**  
 Parámetros de control del algoritmo genético

Nombre del parámetro	Valor
Tamaño de la población inicial	100
Método de selección de padre	Torneo
Estrategia de cruzamiento	Cruce ordenado
Probabilidad de cruzamiento	0.6
Estrategia de mutación	Mezcla de índices
Probabilidad de mutación	0.4
Generaciones	30

El algoritmo genético considerado es EaMuPlusLambda que se basa en cada nueva generación se extrae a partir de la población extendida, donde se puede definir como la combinación de la población actual (tamaño #) y los descendientes (tamaño #); por eso el algoritmo se llama # + #.

## RESULTADOS

En la figura 2 se puede observar la evolución del algoritmo. La convergencia es aceptable, a partir de la generación 29 no existe un cambio significativo en el fitness. Cuando avanza las generaciones la población suele parecerse entre ellos, por lo tanto, el fitness tiende a converger.

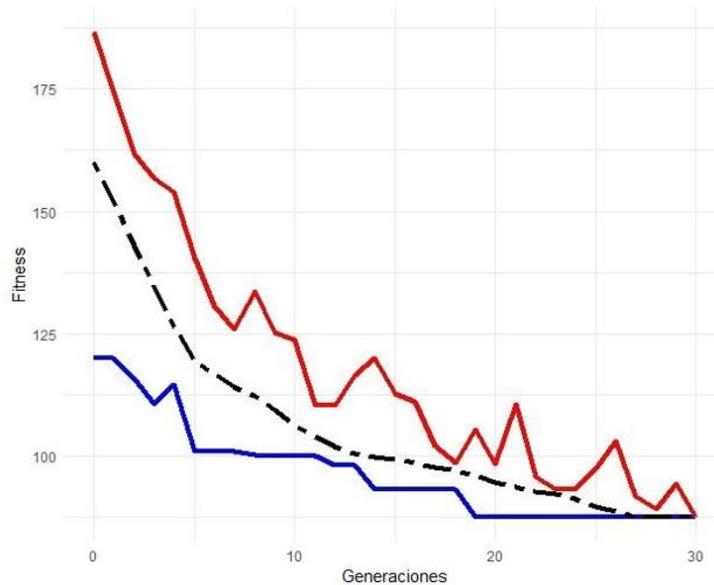


Figura 2. Evolución TSP

FIGURA 2  
Evolución TSP

Los picos que se presenta en la figura 2 es debido de la aparición de individuos que son peores que sus antepasados, afectando los resultados del máximo y el promedio de la población. Trujillo (2019) obtiene una solución que es 1-5-7-6-11-8-13-9-10-14-2-12-15-4-3 con una distancia total de 88.09 Km, que en comparación de la solución del algoritmo genético que es 11-12-4-15-6-7-5-1-3-2-14-13-8-9- 10 con un recorrido de 87.29 Km una diferencia de 800 metros.

Hay que tener en cuenta que existe otras configuraciones que nos pueden dar una solución mejor, lo cual se puede ver en la tabla 4.

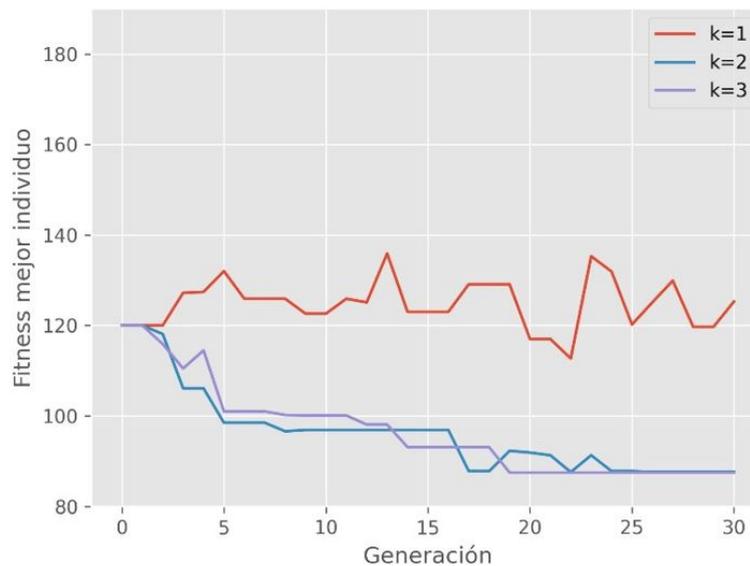
Tabla 4. *Análisis de los parámetros probabilidad de cruce y mutación*

**TABLA 4**  
Análisis de los parámetros probabilidad de cruce y mutación

Métrica	Cruce	Mutación	Fitness
Max	0.8	0.2	92.39
Min	0.8	0.2	87.09
Avg	0.8	0.2	89.981
Sdt	0.8	0.2	1.5078226
Max	0.6	0.4	92.69
Min	0.6	0.4	87.69
Avg	0.6	0.4	89.2
Sdt	0.6	0.4	2.17873817
Max	0.7	0.3	91.89
Min	0.7	0.3	87.49
Avg	0.7	0.3	89.94
Sdt	0.7	0.3	1.64392822

Por último, se analizó la convergencia del algoritmo genético utilizando el tamaño del torneo. En este proceso los mejores individuos se utilizan más veces en las operaciones genéticas que generan la siguiente población, teniendo en cuenta el tiempo computacional y los recursos que tenemos disponible.

En la figura 3 se muestra la evolución del mejor individuo para nuestro problema con distintos tamaños de torneo. Si aumentamos el tamaño del torneo aceleramos la convergencia del algoritmo; en otros tipos de problemas se podría dar el caso de que se empeora el resultado del mejor individuo. Se recomienda utilizar un tamaño de torneo mayor cuando nuestro problema es complejo y no tenemos el tiempo necesario para esperar un resultado satisfactorio.



*Figura 3. Tamaño del torneo*

**FIGURA 3**  
Tamaño del torneo

## DISCUSIÓN Y CONCLUSIONES

En este artículo se propuso el desarrollo de un algoritmo genético para la solución del problema de una ruta turística en la Ciudad de Panamá con la menor distancia posible. La propuesta pretende igual que en el artículo Trujillo (2019) demostrar la aplicabilidad de la Matemática en un contexto nacional y que tenga un impacto económico.

A través de los experimentos computacionales se demuestra que el algoritmo genético estudiado genera una reducción de 800 metros con respecto a la solución propuesta por Trujillo (2019), implicando una reducción costo y tiempo para una posible compañía de tour turístico.

Cabe destacar que no sólo se aplicó el algoritmo genético, también se analizó algunos parámetros importantes para su convergencia como la probabilidad de mutación, probabilidad de cruce y el tamaño del torneo. Demostrando que algunos cambios de estos parámetros representan una mejor solución y una convergencia más rápida.

### Conflicto de intereses

Los autores declaran no tener conflicto de intereses.

## REFERENCIAS

- Beasley, D., Bull, D. R., & Martin, R. R. (1993). An overview of genetic algorithms: Part 1, fundamentals. *University computing*, 15(2), 56-69. <https://mat.uab.cat/~alseda/MasterOpt/Beasley93GA1.pdf>
- Booker, L. B., Goldberg, D. E., & Holland, J. H. (1989). Classifier systems and genetic algorithms. *Artificial intelligence*, 40(1-3), 235-282. [https://doi.org/10.1016/0004-3702\(89\)90050-7](https://doi.org/10.1016/0004-3702(89)90050-7)
- Conroy, G. (1991). Handbook of genetic algorithms by Lawrence Davis (Ed.), Chapman & Hall, London, 1991, pp 385, £32.50. *The Knowledge Engineering Review*, 6(4), 363-365. <https://doi.org/10.1017/S0269888900006068>
- Reina, D., Nozal, A. y Córdoba, A. (2020). Algoritmos Genéticos con Python. Editorial Marcombo. ISBN:9788426729859
- Trujillo, J. (2019). Grafos hamiltonianos aplicado al turismo de Panamá. *Investigación y Pensamiento Crítico*, 7(1), 109–113. <https://doi.org/10.37387/ipc.v7i1.12>