

Optimización de ruta corta en el espacio sin corte lineal con algoritmo genético

Short path optimization in space without linear cutting with genetic algorithm

¹Raúl Eduardo Huarote Zegarra

ORCID: 0000-0001-7466-7404

²Yensi Vega Luján

ORCID: 0000-0001-5503-0571

³Mónica Patricia Romero Valencia

ORCID: 0000-0002-7052-9220

RESUMEN

Esta investigación tuvo como objetivo encontrar la ruta más corta de n puntos en el espacio, sin corte o intersección entre las líneas generadas por los caminos. Para lograr esto, se usó el algoritmo genético, donde se aprovechó la ventaja de no competir “todos contra todos”, sino que a partir de una pequeña población puede encontrarse la posible mejor ruta en el espacio, también llamada “búsqueda local”. Para realizar el proceso evolutivo, se consideró el método de la ruleta, cruce por intercambio de 2 puntos, mutación por intercambio y método de parada a la varianza a 25 generaciones. Implementado en Matlab 8.3, se obtuvo como resultado una duración de 24.7 s y 210.6 s, con funciones de adaptación de 0.79 y 0.76, 76 y 206 generaciones a las pruebas realizadas de 10 y 100 puntos, respectivamente. Demostrando que con el algoritmo genético se encontró la posible mejor ruta corta de n puntos en el espacio y que el método de parada ideal para este problema en particular es la varianza, aunque consume más tiempo frente a cantidad de generaciones establecidas.

Palabras clave: Algoritmo genético, espacio, intersección lineal, optimización, ruta corta.

ABSTRACT

This research article has aimed to find the shortest route of points in space and without cutting or intersection between the lines generated by the roads, to achieve this the genetic algorithm was used, where the advantage of not competing all was used against all, if not from a small population is the best possible route in space or also called local search. In order to carry out the evolutionary process, the roulette method, 2-point exchange crossing, exchange mutation and 25-generation variance stop method have been evaluated. Implemented in Matlab 8.3, a duration of 24.7 and 210.6 sec was obtained, with adaptation functions of 0.79 and 0.76, 76 and 206 generations to the tests of 10 and 100 points respectively. Demonstrating that the best possible short route of points in space is found with the genetic algorithm and that the ideal stop method for this particular problem is variance even more time consuming compared to the number of established generations.

Keywords: Genetic algorithm, space, linear intersection, optimization, short route.

¹ Universidad César Vallejo. Lima Norte – Perú. E-mail: rhuarote@ucv.edu.pe

² Universidad Nacional de Trujillo. Perú. E-mail: yensi.vega@gmail.com.

³ Universidad Nacional Federico Villarreal. Perú. E-mail: mromerova@unfv.edu.pe

INTRODUCCIÓN

Desde que se formalizó el problema del agente viajero (traveling salesman problema) por los matemáticos Garey y Jhonson (1979), donde Lawker, Lenstra, Rinnooy y Shmoys (1985) lo resaltaron como un problema donde se debe partir de una ciudad origen, visitar exactamente sola una vez cada ciudad dentro de un conjunto, teniendo en cuenta el orden, y retornar a la ciudad de origen en la menor distancia total posible. Se tienen diferentes tratamientos, como Karl (1932), quien aplicó el método de la fuerza bruta; Holton y Aldred (1999), quienes aplicaron el método hamiltoniano; Liu (2008), quien empleó el algoritmo genético usando el esquema de reinas; Martello (2010), que utilizó el método húngaro; o como Cook (2012), quien aplicó el método simplex. Para Coello (2018), el algoritmo genético ha demostrado tener éxito. Por eso, esta investigación se basa en el tratamiento de n puntos en el espacio con el modelo alternativo, que es el algoritmo genético usando la mejor estrategia propuesta, es decir, traer al mejor individuo de la generación anterior y reemplazarlo con el individuo de la generación actual (el mejor en función de adaptación). Esto para ayudar a continuar con la evaluación de otras posibilidades y teniendo en cuenta la restricción, que las rutas generadas por los puntos en el espacio no se intersecten o corten, sin perder el objetivo que es encontrar la ruta corta en el espacio. El modelo propuesto tiene la complejidad total de $O(n^2)$. Para lograr evidenciar el proceso y resultados se elaboró el simulador en Matlab 8.3.

Algoritmos genéticos

La investigación del biólogo Fraser (1957), sobre los sistemas biológicos en una computadora, toma como referencia la representación de Charles Darwin (1859), donde “sobrevive el más apto”. A su vez, la representación en software fue dada por Bremermann (1962), que posteriormente fue utilizada por Holland (1992), acuñando el nombre de “Algoritmos Genéticos”. Hoy, es una técnica basada en la teoría de adaptación de Darwin (1859). Posteriormente, Golberg (1989) toma esta idea como “métodos adaptativos”, generalmente usados en problemas de búsqueda y optimización de parámetros, basados en la reproducción sexual y en el principio de supervivencia del más apto. Para Arora (2012) y Carlton (2019) el algoritmo genético pertenece a la clase de métodos de optimización de búsqueda estocástica.

Por lo tanto, un algoritmo genético es un modelo computacional de búsqueda de la posible mejor solución, basándose en la adaptación del modelo evolutivo. Para ello, considera los criterios de selección, cruce, mutación, valor de adaptación, etc. También es un modelo digital de adaptación del proceso evolutivo, con el fin de buscar una posible mejor solución, a partir de una muestra para recorrer el universo de posibilidades.

MATERIAL Y MÉTODOS

Datos para procesar

Tener en cuenta que los datos que se van a procesar son posiciones en el espacio de tipo $P_i(x, y, z)$, donde x, y, z son posiciones del i -avo punto y parte de un camino, está reflejado por $P_i P_{i+1}$. Para el proceso, es necesario considerar la restricción $\overline{P_i P_{i+1}}$ con $\overline{P_{i+2} P_{i+3}}$ no se debe intersectar. Para saber esto, se aplicaron las siguientes ecuaciones de la recta en el espacio, con el objeto de encontrar si es que existe intersección entre ambas rectas.

$$\overline{P_0 P_1}(x, y, z) = (x_0, y_0, z_0) + \lambda(x_1 - x_0, y_1 - y_0, z_1 - z_0) \quad (1)$$

$$x = x_0 + \lambda(x_1 - x_0) \quad (2)$$

$$y = y_0 + \lambda(y_1 - y_0) \quad (3)$$

$$z = z_0 + \lambda(z_1 - z_0) \quad (4)$$

Aplicando la ecuación (1) de la recta paramétrica en el espacio para una recta $\overline{P_0 P_1}$, y de la misma manera para la recta $\overline{P_2 P_3}$ por tanto en la ecuación (2,3,4), de existir un punto en común de estas 2 rectas, entonces quedaría expresado para evaluar en la ecuación (5,6,7) para el eje x, y, z respectivamente.

$$x_0 + \lambda(x_1 - x_0) = x_2 + \gamma(x_3 - x_2) \quad (5)$$

$$y_0 + \lambda(y_1 - y_0) = y_2 + \gamma(y_3 - y_2) \quad (6)$$

$$z_0 + \lambda(z_1 - z_0) = z_2 + \gamma(z_3 - z_2) \quad (7)$$

$$\gamma = \frac{\frac{z_2 - z_0}{z_1 - z_0} - \frac{y_2 - y_0}{y_1 - y_0}}{\frac{y_3 - y_2}{y_1 - y_0} - \frac{z_3 - z_2}{z_1 - z_0}} \quad (8)$$

$$\lambda = \frac{y_2 + \gamma(y_3 - y_2) - y_0}{y_1 - y_0} \quad (9)$$

Para comprobar que existe intersección entre 2 rectas en el espacio, se evalúa si es que cumple la

igualdad de la ecuación (7) a partir de los valores obtenidos de la ecuación (8,9), extraídas de la ecuación (5,6) Por tanto, de cumplir la ecuación (7), se puede afirmar que existe intersección entre

las 2 rectas en el espacio, ya que existe un $P(x, y, z)$ en común. Tal como se muestra en la ecuación (10), donde 1 significa que sí se cortan (o intersectan) y 0 no.

$$Secortan(\overline{P_i P_{i+1}}, \overline{P_j P_{j+1}}) = \begin{cases} 1 & z_i + \lambda(z_{i+1} - z_i) = z_j + \gamma(z_{j+1} - z_j) \\ 0 & \text{Caso contrario} \end{cases} \quad (10)$$

Donde $j > i + 1$

Cromosoma

Cada cromosoma presenta una ruta posible, donde cada gen representa un punto en el espacio en específico. Además, el cromosoma tiene un

tamaño definido, y este tamaño se lleva para todo el proceso evolutivo; siendo evaluado de izquierda a derecha la distancia total aplicando la ecuación (11), en la Figura 1 se muestra un modelo de cromosoma.

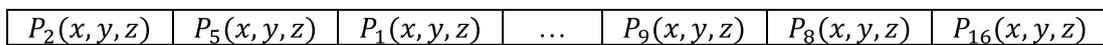


Figura 1. Representación de un cromosoma

$$d_{Cromosoma} = \sum_{i=0}^{tamCromosoma-1} \overline{P_i P_{i+1}} \quad (11)$$

$$= \sum_{i=0}^{tamCromosoma-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}$$

Generar población inicial

Los valores de los genes de cada cromosoma son asignados por las posiciones de un punto P_i , esta asignación es de manera aleatoria y sin ningún orden en específico. Teniendo en cuenta la consistencia del cromosoma, primero que no se repitan 2 puntos o más en un cromosoma y segundo que en 2 genes seguidos que representan una recta $\overline{P_i P_{i+1}}$ con otros 2 genes seguidos que también representan una recta $\overline{P_j P_{j+1}}$ no exista un cruce en la línea que generan como camino.

Selección de padres

La selección de padres se realiza mediante el método de la ruleta, donde los cromosomas de la población tienen un valor porcentual en función a su adaptación (inversamente proporcional a la distancia $\overline{P_i P_{i+1}}$) dando mayor posibilidad de ser seleccionado al que más se adapte (mayor valor), este proceso también se repite para el segundo padre.

Cruce

Por la naturaleza del problema, se tiene que hacer el cruce de 1 punto y, como es de esperar, al realizar el cruce en los nuevos hijos va a existir inconsistencia. Entonces en cada hijo se van a completar los valores que faltan de cada gen, que se repiten de izquierda a derecha, siendo esto necesario para que solo se reemplacen los genes duplicados y se mantenga parcialmente el proceso de cruce.

También se puede realizar el cruce de alternancia de posiciones (AP), propuesto por Larrañaga y col (1999), donde se seleccionan las ciudades alternativamente del padre 1 al padre 2 en el orden que ocupan, omitiendo las ciudades que ya se han colocado; para el segundo hijo la alternancia es del padre 2 al padre 1. Cabe resaltar que este proceso se cumple si es que está en el ámbito de la probabilidad de cruce (P_c).

Mutación

Por la naturaleza del problema se tiene que hacer la mutación por intercambio entre 2 puntos seleccionados al azar, ya que el alterar solo a un

gen implica la repetición de un punto y la exclusión de otro punto, perdiendo la consistencia de la ruta a buscar. Cabe resaltar que este proceso se cumple si es que está en el ámbito de la probabilidad de mutación (Pm).

Adaptación (Fitness)

Este cálculo permite evaluar, por cada

$$Fitness_{Cromosoma} = 1 - \frac{d_{cromosoma}}{máximaDistancia * tamañoCromosoma} \tag{12}$$

$$FitnessNormalizado_c = \frac{Fitness_{Cromosoma}}{\sum_{c=1}^{TamPoblación} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}} \tag{13}$$

Condición de parada

Se va a considerar y analizar 2 métodos de parada para el proceso evolutivo, el primero es cuando se establece la cantidad de generaciones y otra es cuando el sistema evalúa si es que no existe cambios en los valores de las funciones de adaptación de los mejores individuos de las últimas *k* generaciones, por tanto se usa la ecuación (14) de Cartusia (1998), que evalúa la varianza de datos, donde *f_i* es el valor de la función de adaptación del *i*-avo cromosoma y *f* es el promedio de la función de adaptación de los *k* cromosomas.

cromosoma, el valor de adaptación; y esto va en función a la distancia del cromosoma como se refleja en la ecuación (11), comparado con la probabilidad del peor de los casos, tal como se refleja en la ecuación (12), a su vez estos valores son normalizados y expresados en la ecuación (13).

$$s^2 = \frac{1}{k} \sum_{i=1}^k (f_i - \bar{f})^2 \tag{14}$$

Estrategia del algoritmo genético para encontrar la ruta óptima de *n* puntos en el espacio

Para encontrar la ruta corta, se propone el siguiente algoritmo, siendo una adaptación al modelo presentado originalmente por Abuiziah (2013).

Algoritmo	Complejidad temporal
poblacionInicial = generarPoblaciónInicial()	O(n)
ListaFitness = tamListaFiness	O(1)
f = 0	O(1)
dimElite = 2	O(1)
condicionParada=True	O(1)
numeroCruces = (dimPoblacion - dimElite) /2	O(1)
mientras (condicionParada==True)	O(n)
nuevaPoblación = vacio	O(1)
[CromosomaElite1, CromosomaElite2]	O(n)
evaluarPoblacion(poblacionInicial)	
nuevaPoblacion.insertar(CromosomaElite1, CromosomaElite1)	O(2)
Desde i = 0 hasta numeroCruces	O(n)
[Padre1, Padre2] = seleccionarPadres(poblacionInicial)	O(n)
[Hijo1, Hijo2] = cruce(Padre1,Padre2)	O(n)
Hijo1 = mutacion(Hijo1)	O(5)
Hijo2 = mutacion(Hijo2)	O(5)
nuevaPoblacion.insertar(Hijo1,Hijo2)	O(2)

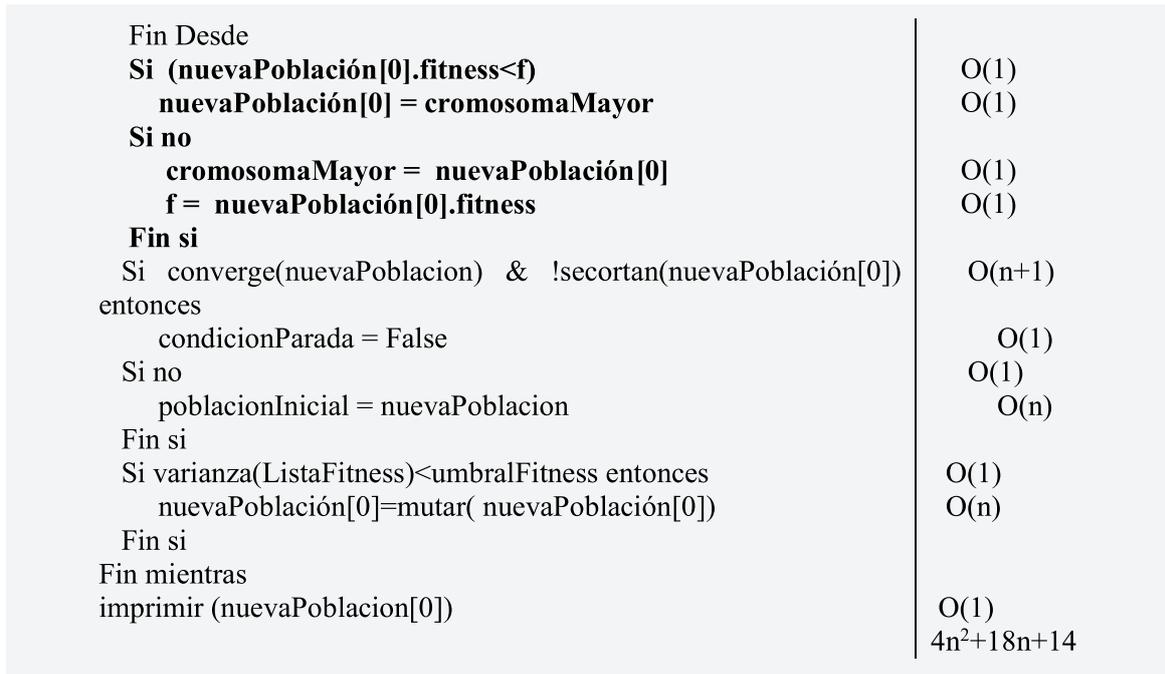


Figura 2. Estructura del algoritmo genético para encontrar la ruta optima de n puntos en el espacio.

Tal como se refleja en la Figura 2, la estrategia está en reemplazar un cromosoma de mejor función de adaptación en función de la generación anterior. Para eso, se considera adicionalmente si cumple con la restricción de intersección en el espacio. De

cumplir de manera global, se va a incrementar el 'nivel' de la generación futura en busca de la mejor solución. Con esta estrategia lo que hace es incrementar el valor de adaptación de cada generación.

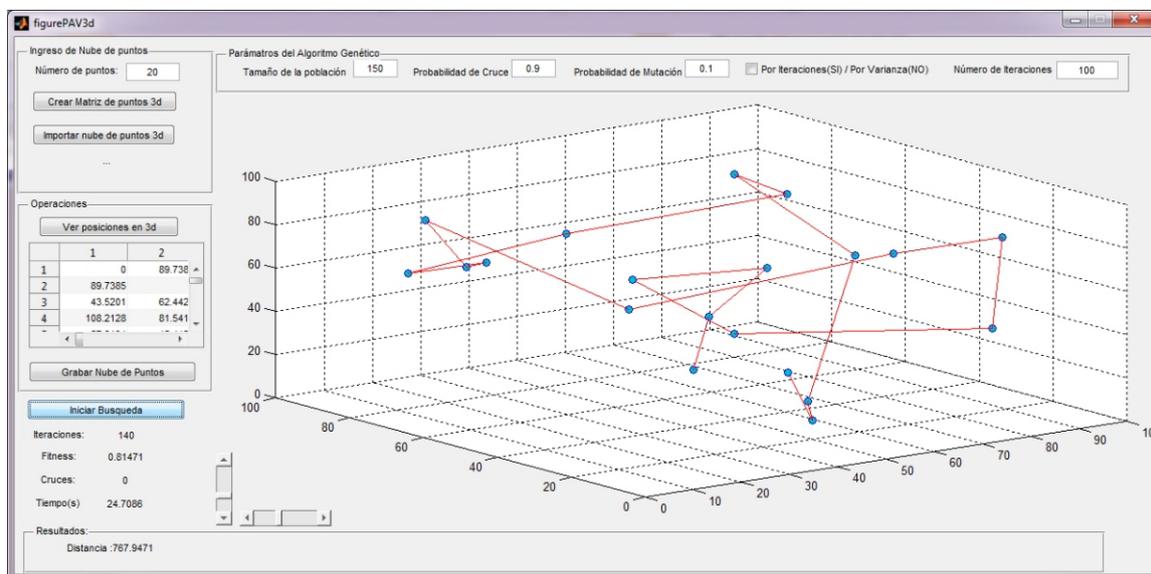


Figura 3. Prototipo funcional para encontrar la ruta optima de n puntos en el espacio.

En la Figura 3 se muestra el prototipo funcional implementado en Matlab 8.3, donde se aprecian los valores de entrada del tamaño de población inicial, siendo este 150; la probabilidad de cruce, 0.95; la probabilidad de mutación, 0.1; el método de parada es la varianza.

RESULTADOS

Los resultados de la evaluación refleja que se encontró la ruta corta mediante el algoritmo genético y con la restricción, para ello se analizaron casos de 10 hasta 100 puntos, la Figura

4 muestra el crecimiento de la función de adaptación. Asimismo, la Figura 5 muestra la cantidad de intersecciones ocurridas durante el

proceso evolutivo para el método de parada de la varianza, pero todos termina en 0 intersecciones.

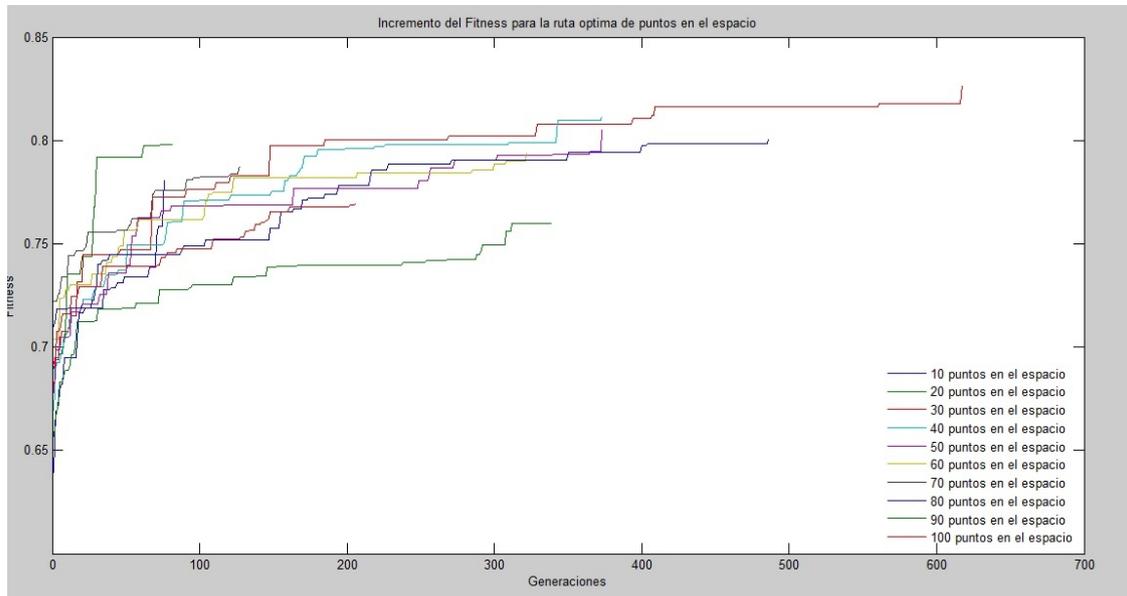


Figura 4. Para las 10 evaluaciones el valor función de la función de adaptación para la cantidad de generaciones.

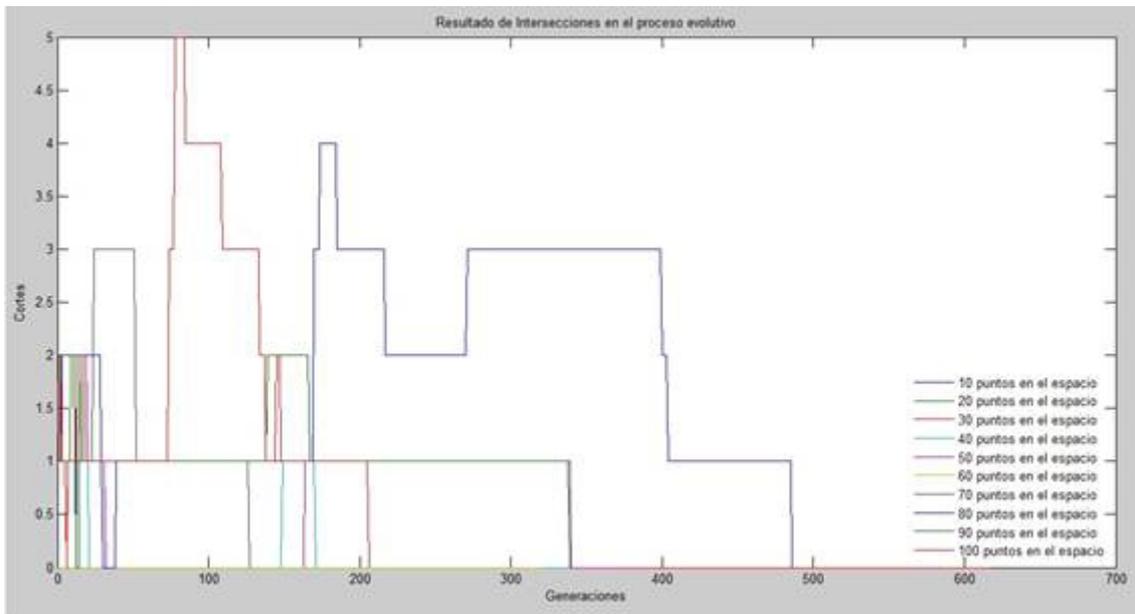


Figura 5. Comparativa de las 10 evaluaciones la cantidad de cortes o intersección conforme a la cantidad de generaciones.

El tiempo para el desarrollo del proceso es variable, la Figura 6 muestra la comparativa de los tiempos de las 10 pruebas realizadas,

considerando el método de la varianza y el método por generaciones establecidas, con los datos extraídos de la Tabla 1.

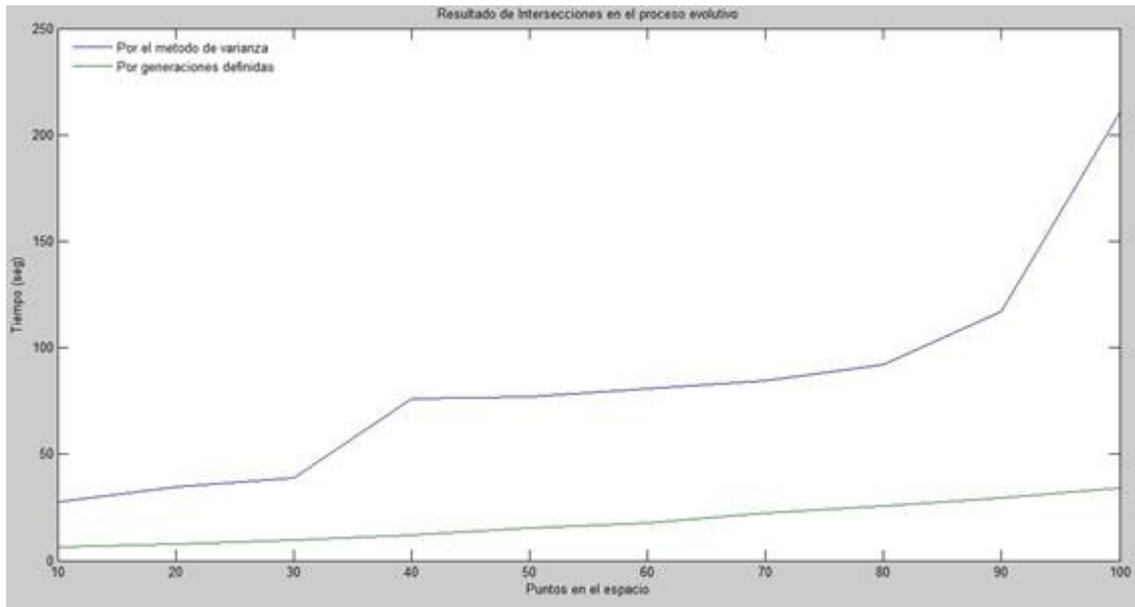


Figura 6. Comparativa de los tiempos por el método de la varianza y por la cantidad de generaciones establecidas.

Tabla 1. Resultados de las iteraciones, función de adaptación, intersecciones y tiempos de las 10 evaluaciones con el método de la varianza y por el método definido de generaciones

Nro Puntos3d	Iteraciones			Fitness			Intersecciones			Tiempo (s)		
	Varianza	Definido	Diferencia	Varianza	Definido	Diferencia	Varianza	Definido	Diferencia	Varianza	Definido	Diferencia
10	76	100	-24	0,7809	0,73	0,0509	0	0	0	24,7	6,2	18,5
20	82	100	-18	0,7979	0,75	0,0479	0	0	0	34,7	7,7	27,0
30	617	100	517	0,8268	0,7662	0,0606	0	0	0	38,8	9,7	29,1
40	373	100	273	0,8117	0,71425	0,09745	0	0	0	76,1	11,9	64,2
50	373	100	273	0,8056	0,708	0,0976	0	0	0	77,0	15,1	61,9
60	322	100	222	0,7941	0,732	0,0621	0	0	0	80,9	17,7	63,2
70	127	100	27	0,7875	0,69	0,0975	0	0	0	84,7	22,2	62,5
80	486	100	386	0,801	0,732	0,069	0	1	-1	92,2	25,5	66,7
90	339	100	239	0,7601	0,728	0,0321	0	2	-2	117,0	29,5	87,5
100	206	100	106	0,7694	0,73	0,0394	0	2	-2	210,6	34,2	176,4

La Tabla 1 se muestra en función de los 10 casos evaluados, al realizar la prueba con el prototipo funcional, y se grafica en las figuras 4,5 y 6.

DISCUSIÓN

En función de los resultados obtenidos en la Tabla 1, se evidencia que, para el problema en particular, hasta en 30 puntos se puede usar el método de generaciones definidas, en adelante el método de parada de la varianza. En todos los casos, el valor de la función de adaptación es mayor usando el método de parada de la varianza. En todos los casos, el método de parada de la varianza no tiene intersecciones de líneas al finalizar el proceso frente al método de generaciones definidas;

mientras aumenta la cantidad de puntos, a partir de 80 puntos termina con intersecciones. En contraposición, es recomendable el método de parada de generaciones establecidas por el tiempo de ejecución.

Para realizar el proceso, se ha considerado como parámetro del algoritmo genético a la probabilidad de cruce de 0.95, probabilidad de mutación de 0.1, 150 cromosomas de tamaño de población, el tamaño del cromosoma va en función de la cantidad de puntos.

CONCLUSIONES

Al realizar el modelo del algoritmo genético para encontrar la ruta corta sin corte lineal, se

encontraron los siguientes resultados:

- Se encuentra la posible mejor solución con un valor de función de adaptación de 0.8268.
- Tiene una complejidad computacional de $O(n) = 4n^2 + 18n + 14$.
- En función de los datos de la Tabla 1, la tendencia del tiempo en función de la cantidad de puntos está dado por la ecuación $t(n) = 0.0178n^2 - 0.04161n + 37.9$, mientras que para el método de parada de las generaciones establecidas tiene la ecuación $t(n) = 0.0018n^2 - 0.113n + 4.7417$
- Se incrementó el valor de la función de adaptación, realizando unas modificaciones en el método de la varianza en 0.065.
- Permitió encontrar la ruta óptima de n puntos en el espacio sin intersección o corte lineal generada por las rutas usando el algoritmo genético.

REFERENCIAS

- ABUIZIAH, N. A Review of Genetic Algorithm Optimization: Operations and Applications to Water Pipeline Systems, World Academy of Science. *Engineering and Technology International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, 7(12), 2013.
- ARORA, S. *Introduction to Optimum Design*, Elsevier Inc. USA, Chapter 16, 643, 2012.
- BREMERMANN, H J. *Optimization through evolution and recombination*, In Self-organization system, M. C. Yovitts et al. Spartan Books, Washinton, D. C. 93-106, 1962.
- CARLTON, J. S. *Marine Propeller and Propulsion*, Butterworth-Heinemann. Fourth Edition, USA, 469-497, 2019.
- COELLO A. Introducción a la Computación Evolutiva, CINVESTAV-IPN. Departamento de computación. México D. F. 07300, 243, 2018.
- COOK, J. In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation Princeton University press. 2012.
- DARWIN, C. El origen de las especies. London College of Cambridge. 1859.
- DEREK, H. & ALDRED, R. Planar Graphs, Regular Graphs, Bipartite Graphs and Hamiltonicity. Department of Mathematics and Statistics University of Otago, 111-131, 1999.
- FRASER, Alex. Simulation of genetic systems by automatic digital computers Introduction, *Aust. J. Biol Sci*, 10, 484-491, 1957.
- GAREY, M. R. & JOHNSON, D. S. Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, pp. 211-212, 1979. ISBN 0-7167-1045-5
- GOLBERG, D. *Genetics Algorithms in search, optimization and machine learning*, MA: Addison-Wesley Professional. 1989.
- HOLLAND, J. *Adaptation in Natural and Artificial Systems*. Cambridge, MA: MIT Press. 1992.
- LARRANAGA, P.; KUIJPERS, C; MURGA, R.; INZA, I. Y DIZDAREVICH, S. Evolutionary algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review*, 13, 129-170, 1999.
- LAWLER, E.; LENSTRA, J.; RINNOOY, K. & SHMOY, D. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. 1985.
- LIU, Y. Solving the Probabilistic Travelling Salesman Problem Based on Genetic Algorithm with Queen Selection Scheme. Department of Civil Engineering. National Chi Nan University Taiwan. 2008.
- MARTELLO, S. From the origins of the Hungarian algorithm to satellite communication. *Central European: Journal of Operation Research*, 18, 47-58, 2010.
- MENGER, K. (ed.) *Ergebnisse eines Mathematisthen Kolloquiums 2, Kolloquium 5.11.1930*, Teubner Leipzig. 1932.