

Tipo de artículo: Artículo original  
Temática: Inteligencia Artificial  
Recibido: 18/12/2022 | Aceptado: 04/04/2023

## Integración de técnicas de visualización a la metaheurística Algoritmo Genético

Integration of visualization techniques to the Genetic Algorithm metaheuristic

José Manuel Cassola Bacallao <sup>1\*</sup> <https://orcid.org/0000-0002-1029-1548>

Andy Morfa Hernández <sup>1</sup> <https://orcid.org/0000-0002-2003-0974>

Carlos Pérez Risquet <sup>1</sup> <https://orcid.org/0000-0002-3547-3049>

<sup>1</sup>Centro de Investigaciones Informáticas. Universidad Central “Marta Abreu” de Las Villas, Carretera Camajuaní km 5 ½, Santa Clara, Villa Clara, Cuba. [jcassola@uclv.cu](mailto:jcassola@uclv.cu), [andym@uclv.cu](mailto:andym@uclv.cu), [cperez@uclv.edu.cu](mailto:cperez@uclv.edu.cu)

\*Autor para la correspondencia. ([jcassola@uclv.cu](mailto:jcassola@uclv.cu))

---

### RESUMEN

El proceso de solución de problemas de optimización constituye una ciencia en constante desarrollo. Las técnicas de interacción visual para apoyar algoritmos metaheurísticos representan una atractiva variante en el progreso de la misma, gracias a las ventajas que pueden ofrecer el análisis e interacción de un usuario en tiempo de ejecución de un algoritmo. Este artículo propone una serie de interacciones y visualizaciones integrada a una variante de la metaheurística Algoritmo Genético, con el objetivo de optimizar los resultados al resolver el Problema del Viajero Vendedor y el Problema de Ruteo de Vehículos. Dicho procedimiento trajo como resultado el desarrollo de una herramienta a la cual se le realizaron las pruebas

pertinentes, mostrando la eficiencia de la integración realizada y la superioridad de los resultados obtenidos con asistencia del usuario.

**Palabras clave:** algoritmo genético; problema del viajero vendedor; problema de ruteo de vehículos; interacción; visualización.

## **ABSTRACT**

The optimization problem solving process is an ever-evolving science. The visual interaction techniques to support metaheuristic algorithms represent an attractive variant in its progress, thanks to the advantages that the analysis and interaction of a user at runtime of an algorithm can offer. This article proposes a series of interactions and visualizations integrated into a variant of the Genetic Algorithm metaheuristic with the aim of optimizing the results when solving the Travelling Salesman Problem and the Vehicle Routing Problem. This procedure resulted in the development of a software tool to which the pertinent tests were carried out, showing the efficiency of the achieved integration and the superiority of the results obtained with the assistance of the user.

**Keywords:** genetic algorithm; travelling salesman problem; vehicle routing problem; interaction; visualization.

---

## **Introducción**

Algunos tipos de problemas de optimización, por sus características, pueden resolverse haciendo uso de técnicas determinísticas que permiten encontrar su solución óptima. Por ejemplo, los problemas o funciones que son lineales, pueden ser resueltos mediante el método Simplex. (López Gutiérrez, 2021)

Sin embargo, la mayoría de los problemas reales no pueden ser resueltos con los algoritmos determinísticos (Chuzhoy et al., 2020), algunos porque no tienen las características para poder usar el método y otros porque el tiempo que necesitan para hallar la solución resulta ser extenso, es decir, el tiempo se hace mayor de acuerdo al tamaño del problema y por lo tanto se deben buscar otras técnicas para encontrar la solución. (Dokeroglu et al., 2019)

Para ello, se hace uso de los métodos metaheurísticos, los cuales resuelven problemas de búsqueda y optimización encontrando una solución que, aunque no necesariamente es la óptima, es una buena solución y se obtiene en un tiempo razonable. (Houssein et al., 2021)

Problemas de estas características son los famosos Problema del Viajero Vendedor (TSP) (Applegate et al., 2006) y Problema de Ruteo de Vehículos (VRP) (Dantzig y Ramser, 1959), y uno de los algoritmos metaheurísticos más famosos y utilizados en diferentes problemáticas es el conocido Algoritmo Genético (GA) (Holland, 1975).

Sin embargo, generalmente los softwares desarrollados que utilizan el Algoritmo Genético u otras metaheurísticas para resolver estos problemas poseen escasa o ninguna interacción visual con el usuario, por lo que deja al mismo con pocas opciones de modificación y/o interacción más allá de la ejecución del algoritmo. Cabe destacar como excepción el modelo de interacciones propuesto e implementado a través del software ACOVis (Morfa Hernández et al., 2018), aunque este usa la metaheurística Optimización con Colonia de Hormigas (Nayar et al., 2021).

El presente trabajo describe el trayecto durante el desarrollo de un software que integre visualización e interacción con la metaheurística Algoritmo Genético para solucionar el Problema del Viajero Vendedor y el Problema de Ruteo de Vehículos. Para ello se realizó el análisis de la literatura relacionada a las líneas de investigación en que se encuentran los temas investigados. Además, se proponen las interacciones visuales a implementar en el producto y finalmente se realizan las pruebas de desempeño del software implementado a través de comparaciones con resultados registrados y pruebas estadísticas.

## **Métodos o Metodología Computacional**

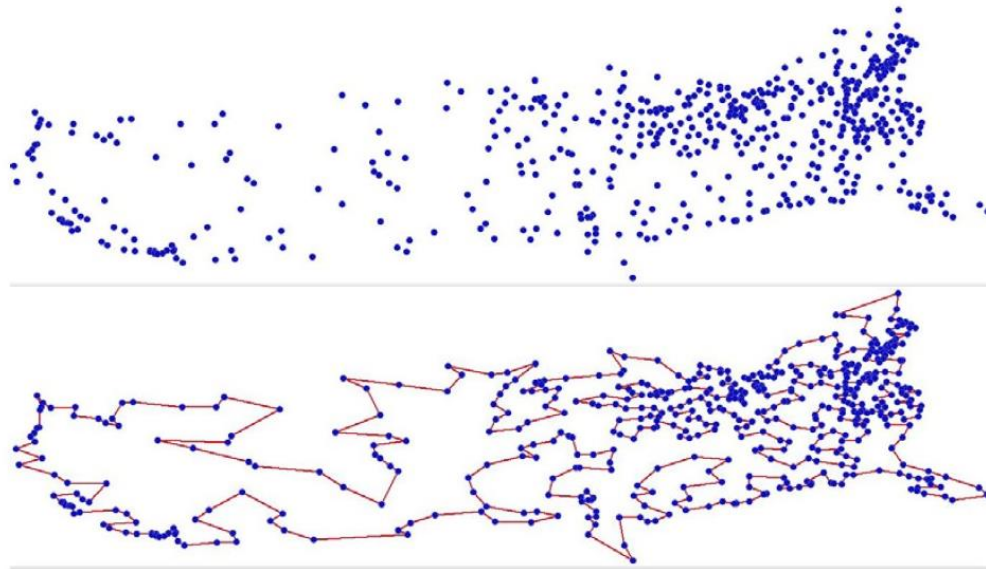
## **Problema del Viajero Vendedor**

Primero se analizó el Problema del Viajero Vendedor, un problema clásico y abierto de la optimización combinatoria, una de las principales subdisciplinas de la investigación de operaciones. (Peres y Castelli, 2021)

El TSP se puede formular de la siguiente manera: “Dado un conjunto de ciudades, de las cuales se conoce para cada par de ellas, la distancia que las separa, un agente viajero ha de partir de una ciudad de origen y debe visitar exactamente una vez cada ciudad del conjunto, y retornar al punto de partida”. (Hariyadi et al., 2020)

Un recorrido con estas características es conocido dentro de este contexto como un *tour* o ciclo hamiltoniano. El problema consiste en encontrar el *tour* para el cual la distancia total recorrida sea mínima (Shi y Zhang, 2022).

El problema fue definido como tal en 1800 por el matemático irlandés William Rowan Hamilton y por el matemático británico Thomas Kirkman. (Fleitas Perdomo, 2018)

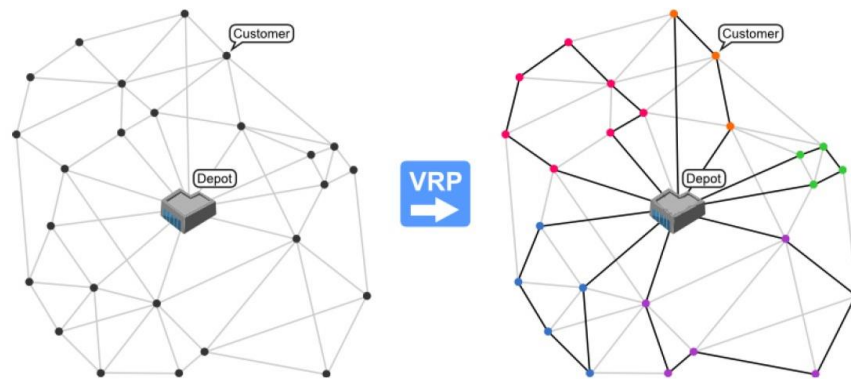


**Fig. 1** – Grafo y recorrido óptimo de una instancia del TSP con 532 ciudades. (Oliinyk et al., 2021)

El problema tiene considerables aplicaciones prácticas, aparte de las más evidentes en áreas de logística de transporte que cualquier negocio de reparto, pequeño o grande, conoce. Por ejemplo, en robótica permite resolver problemas de fabricación para minimizar el número de desplazamientos al realizar una serie de perforaciones en una plancha o en un circuito impreso. También puede ser utilizado en control y operativa optimizada de semáforos, etc. (Greco Chandía, 2020)

### **Problema de Ruteo de Vehículos**

Como segundo problema a resolver con nuestro software tenemos al VRP. El objetivo principal de este es atender un conjunto de clientes con una demanda por servicio a un costo mínimo a través de rutas que tienen un origen y un término en un depósito central. Este problema típicamente considera una flota de vehículos ubicados en un depósito central que pueden ser programados con el fin de proveer algún tipo de servicio a clientes geográficamente dispersos en una región a atender. Así, estos problemas consideran el diseño de rutas de atención (distribución o recolección) desde uno o más depósitos centrales a un conjunto de clientes dispersos. (Sar y Ghadimi, 2023)



**Fig. 2** – Ejemplo de solución de un VRP. (Wang y Sheu, 2019)

En el presente artículo se analizará específicamente el Capacitated Vehicle Routing Problem (CVRP) (Aydemir y Karagul, 2020), una de las variantes del VRP y, de igual manera, la propuesta de solución será enfocada en este.

El CVRP es considerado la restricción más básica sobre el problema VRP clásico y fue por tanto uno de los primeros tipos de VRP en ser analizado. En el problema VRP con restricciones de capacidad (CVRP),  $n$  clientes deben ser atendidos desde un *depot* normalmente identificado por el nodo 0. Cada cliente  $i$  demanda una cierta cantidad  $m_i$  positiva ( $m_i > 0$ ) de un mismo producto y para cada par de clientes existe una distancia  $m_{ij}$  dada. Los clientes son atendidos por una flota de vehículos, cada uno de los cuales tiene una capacidad  $C$ . El principal objetivo en CVRP es encontrar el conjunto de rutas que minimicen el tiempo total de viaje de forma que (Jiang et al., 2022):

- i. Un cliente sea servido una sola vez por un único vehículo.
- ii. La ruta seguida por cada uno de los vehículos empiece y termine en el *depot*.
- iii. El total de demandas cubiertas por cada vehículo no pueda exceder su capacidad.

El problema CVRP es por tanto *NP-hard* (Li et al., 2020) ya que contiene el problema TSP como subproblema. Por ello, en la práctica resolver el problema VRP es más difícil, ya que implica resolver dos problemas anidados.

## Algoritmo Genético

Para la solución de ambos problemas usaremos la metaheurística Algoritmo Genético. Los algoritmos genéticos presentan una propuesta eficaz motivada en la observación de que la evolución natural ha sido extraordinariamente exitosa en desarrollar especies complejas y bien adaptadas por medio de un mecanismo simple, por lo que fue la seleccionada para este proyecto. La técnica emula la evolución natural para explorar con eficiencia el espacio de búsqueda con el supuesto de que unos individuos con ciertas características son aptos para sobrevivir y transmiten estas características a su descendencia. (Mirjalili, 2019)

Los algoritmos genéticos operan sobre una población o conjunto de soluciones representadas como cadenas binarias o cromosomas. Durante la ejecución, el algoritmo cruza los individuos de mayor aptitud para renovar la población y elimina los de menor aptitud. Al final, el cromosoma de mayor aptitud es la solución al problema. (Papazoglou y Biskas, 2023)

### Terminología del Algoritmo Genético

Según (Mitchell, 1998) algunos conceptos básicos en algoritmos genéticos son:

- **Cromosoma.** Cadena binaria que representa un individuo o solución, donde cada elemento en la cadena se conoce como gen (Fig. 3a).
- **Población.** Conjunto finito de cromosomas (Fig. 3b).
- **Aptitud.** Criterio que evalúa la calidad de un cromosoma. A mayor aptitud, mejor la solución y mayor la probabilidad de que sobreviva y transmita sus características a su descendencia.
- **Cruzamiento.** Operación por medio de la cual se producen nuevos descendientes a partir de dos cromosomas padre seleccionados al azar (Fig. 4a).
- **Mutación.** En esta operación se seleccionan al azar y se cambian uno o más genes en el cromosoma; ocurre con probabilidades muy bajas (Fig. 4b).

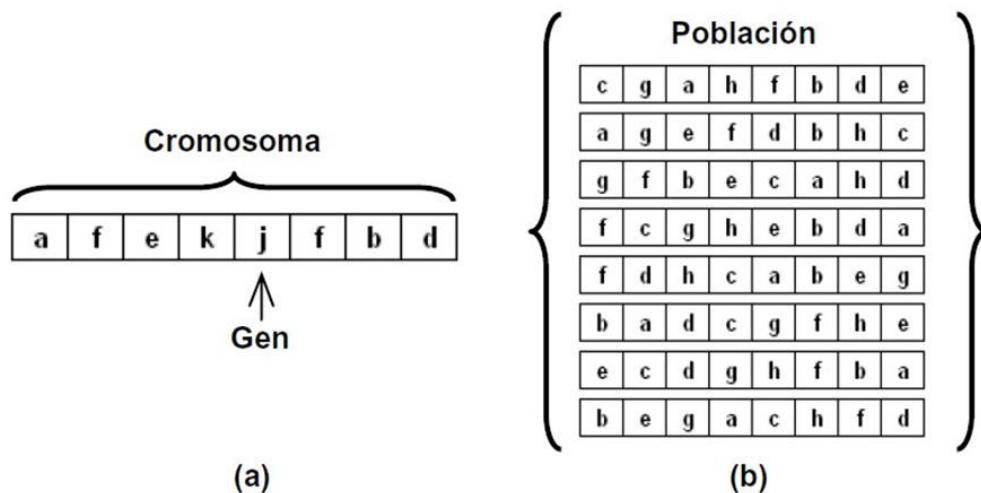


Fig. 3 – Representación de la población en un algoritmo genético.

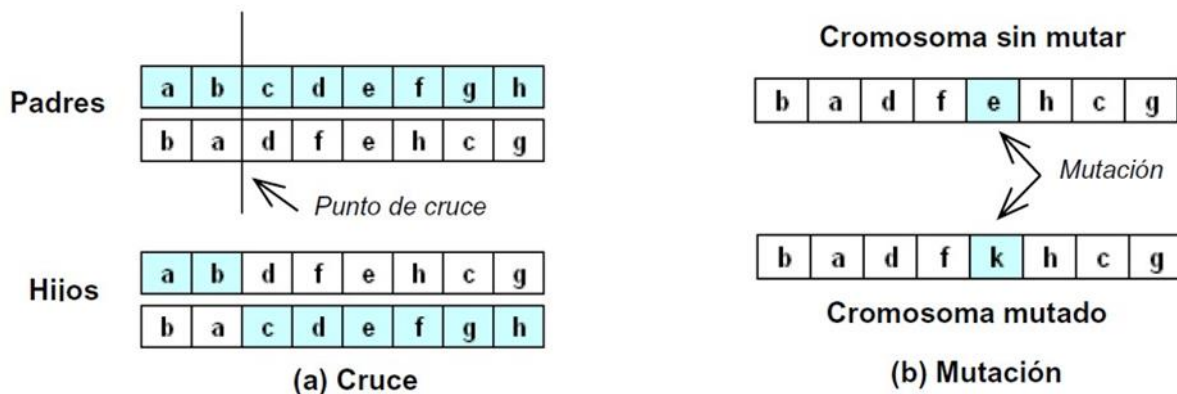
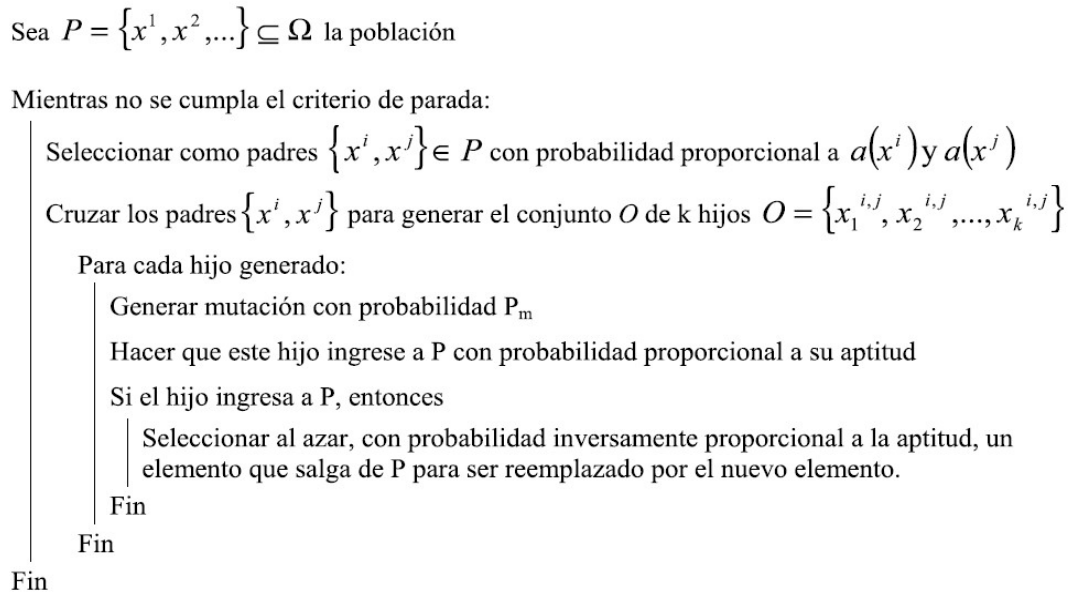


Fig. 4 – Operaciones básicas en algoritmos genéticos.

### Pseudocódigo del Algoritmo Genético

La idea básica de un algoritmo genético y la que se implementó en el sistema es la siguiente:





**Fig. 5** – Pseudocódigo del GA. (Jia y Lichti, 2017)

## Visualización de Grafos

En este trabajo se utilizó la visualización de grafos, una de las subáreas de la Visualización de la Información, para el despliegue y posterior interacción de nuestros mapas.

El dibujado de grafos direcciona el problema de visualizar información estructural o relacional construyendo representaciones visuales geométricas de grafos o redes que son los modelos subyacentes en una gran cantidad de datos abstractos. (Chen et al., 2019)

Es común que un grafo se represente con el objetivo de ser visualizado. Su visualización suele realizarse como un conjunto de vértices y aristas colindantes entre los puntos que en no muy pocos casos, representan un escenario de la vida real (Sánchez, 2017). Se debe tener en cuenta que un grafo tiene infinidad de formas de ser esbozado, esto depende del contexto en el que sea utilizado. (Asif et al., 2021)

El problema de dibujado de un grafo puede plantearse simplemente del siguiente modo (Martig y Castro, 2000):

“Dado un conjunto de nodos y un conjunto de arcos que representan las relaciones entre los nodos, obtener la posición de los nodos y la curva que debe ser dibujada para cada arco.”

### **Interacciones y visualizaciones**

Las interacciones y visualizaciones que se implementaron en el sistema con el objetivo de optimizar la búsqueda de los resultados son las siguientes:

- Representar los nodos (ciudades/clientes) según sus respectivas coordenadas extraídas de un fichero *dataset*.
- Modificar parámetros de ejecución del Algoritmo Genético antes de comenzar o luego de ser detenido por el usuario y permitiéndole al mismo probar las combinaciones de parámetros con los que se alcanzan mejores soluciones.
- Visualizar las rutas que se obtienen durante la ejecución del Algoritmo Genético sobre el *dataset* importado.
- Detener el Algoritmo Genético en un instante determinado para hacer variaciones en los parámetros o en el área de representación.
- Visualizar los resultados numéricos de las soluciones encontradas durante y finalizado el Algoritmo Genético.
- Añadir/eliminar aristas (rutas) en determinadas situaciones, ya sea cuando se detiene el algoritmo o cuando se alcanza la solución final.
- Contabilizar el tiempo de ejecución del Algoritmo Genético para tener un balance del tiempo que se utilizó en encontrar una determinada solución.
- Exportar un archivo de imagen de determinada ruta alcanzada por el algoritmo.
- Exportar un fichero con los resultados de todas las ejecuciones que se hagan del algoritmo.

Hacer zoom, manipular el área de representación y restablecer la misma para tener conocimiento de las cercanías y relación de distancia entre los nodos.

## Resultados y discusión

El principal objetivo del estudio fue establecer una comparación entre los resultados oficiales registrados, disponibles en TSPLIB (<http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsplib.html>) y CVRPLIB (<http://vrp.atd-lab.inf.puc-rio.br/index.php/en/>) y los alcanzados por el sistema, ejecutando el software desarrollado sin asistencia del usuario y con asistencia de este, a través de las interacciones implementadas, para valorar la efectividad y eficacia del sistema.

Para la selección de los ficheros se tomó en cuenta la diferencia de los parámetros propios del problema entre los ficheros: cantidad de ciudades en el TSP y de clientes y vehículos en el VRP.

El sistema se ejecutó en un ordenador con las siguientes propiedades:

- Procesador: Intel(R) Core (TM) i7-6700HQ CPU @ 2.60GHz (8CPUs)
- Memoria: 8192MB RAM
- Sistema Operativo: Windows 10 Pro 64 bits (10.0, compilación 18362)

En las pruebas desarrolladas se utilizan diferentes combinaciones de parámetros que proporciona el sistema para la ejecución del Algoritmo Genético, ofreciendo así un mayor número de posibilidades para la comparación según la efectividad de los mismos.

Los parámetros necesarios para la ejecución del Algoritmo Genético son los siguientes:

- **Población:** Es el conjunto de cromosomas (soluciones) generados después de una iteración.
- **Generaciones:** Se forman nuevas poblaciones a partir de poblaciones anteriores, ya sea seleccionando un cromosoma completo (generalmente de alta aptitud), modificando un poco un cromosoma existente (mutación) o, más comúnmente, generando nuevos cromosomas a partir de cromosomas “padres” (cruzamiento).

- **Selector Natural:** Su función es garantizar que los cromosomas de mejor rendimiento (mayor aptitud) tengan una mayor probabilidad de ser utilizados para reproducir la próxima generación.
- **Cruzamiento:** Se selecciona al azar dos cromosomas de la población y se “cruzan”, escogiendo de manera aleatoria un gen y luego intercambiando dicho gen y todos los genes subsecuentes entre los dos cromosomas. El resultado será dos descendientes que combinan las características de ambos cromosomas “padres”.
- **Mutación:** Modifica al azar parte (genes) del cromosoma de los individuos (sin incumplir las restricciones del problema) y permite alcanzar zonas del espacio de búsqueda que no estaban cubiertas por los individuos de la población actual. Es una estrategia para escapar de los mínimos locales potenciales y de generar nuevo material de cruzamiento.
- **Proporción:** Es la probabilidad de que ocurra la funcionalidad del parámetro al que está vinculado (selección natural, cruzamiento, mutación).

Para la ejecución sin asistencia del usuario del TSP se seleccionaron los siguientes parámetros:

- Generaciones: 5000
- Población: acorde a la cantidad de ciudades (calculado por el sistema)
- Selector Natural: Ranking
- Cruzamiento: Grafenstettes
- Mutación: Segment Swapping
- Proporción (Selector Natural y Mutación): 50%

Para la ejecución con asistencia del usuario se seleccionaron 3500 generaciones, el resto de los parámetros iniciales fueron los mismos que los de la ejecución sin asistencia del usuario.

Además, fueron realizadas diversas paradas en la ejecución del proceso durante las cuales se realizaron diferentes interacciones con el algoritmo, como la modificación de parámetros y sus respectivas

proporciones, así como variaciones en la ruta obtenida a través de la inserción y/o eliminación de aristas, apoyándose en todo momento en la visualización (manipulación y zoom) del mapa obtenido.

**Tabla 1** - Resultados de las pruebas en datasets TSP.

Fichero	Ciudades	Soluciones			Tiempo de ejecución sin asistencia (s)	Tiempo de ejecución con asistencia (s)
		TSPLIB	GAVis (sin asistencia)	GAVis (con asistencia)		
ei151	51	426.0	437.0	426.0	104.668	251.667
berlin52	52	7542.0	7722.0	7542.0	152.609	386.937
st70	70	675.0	737.0	682.0	457.739	1143.587
ei176	76	538.0	571.0	547.0	538.448	1345.225
kroA100	100	21282.0	24964.0	21388.0	992.416	2389.391
ch150	150	6528.0	7194.0	6567.0	2881.334	6637.245
a280	280	2579.0	2869.0	2586.0	3594.115	8199.158

Para la ejecución sin asistencia del usuario del VRP se seleccionaron los siguientes parámetros:

- Generaciones: 5000
- Población: 250
- Selector Natural: Ranking
- Cruzamiento: Default
- Mutación: *Two Way*
- Proporción (todos): 50%

Para la ejecución con asistencia del usuario se seleccionaron 3500 generaciones, el resto de los parámetros iniciales fueron los mismos que los de la ejecución sin asistencia del usuario.

Además, fueron realizadas diversas paradas en la ejecución del proceso durante las cuales se realizaron diferentes interacciones con el algoritmo, como la modificación de parámetros y sus respectivas proporciones, apoyándose en todo momento en la visualización (manipulación y zoom) del mapa obtenido.

**Tabla 2 - Resultados de las pruebas en datasets VRP.**

Fichero	Vehículos	Soluciones			Tiempo de ejecución sin asistencia (s)	Tiempo de ejecución con asistencia (s)
		CVRPLIB	GAVis (sin asistencia)	GAVis (con asistencia)		
B-n34-k5	5	788.0	804,33	796,17	294.886	322.172
A-n53-k7	7	1010.0	1217,20	1157,86	330.156	373.205
A-n69-k9	9	1159.0	1518,94	1382,27	857.772	957.485
B-n78-k10	10	1221.0	1784,30	1592.45	1335.523	1480.752
E-n101-k8	8	815.0	1824,06	1579.14	549.532	610.416
M-121-k7	7	1034.0	3043,60	2868.68	723.965	808.128

Para arribar a una conclusión con respecto a las pruebas realizadas se utilizó la prueba de rangos con signo de Wilcoxon, una prueba no paramétrica para comparar el rango medio de dos muestras relacionadas y determinar si existen diferencias entre ellas (Manoukian, 2022). La prueba de rangos con signo de Wilcoxon es más sensata que la prueba t. Asume la conmensurabilidad de las diferencias, pero sólo cualitativamente: las mayores diferencias siguen contando más, lo que probablemente se desea, pero las magnitudes absolutas se ignoran. Desde el punto de vista estadístico, la prueba es más segura, ya que no asume distribuciones normales. Además, los valores atípicos (rendimientos excepcionalmente buenos o malos en unos pocos conjuntos de datos) tienen menos efecto en el Wilcoxon que en la prueba t. (de Barros, Hidalgo y de Lima Cabral, 2018)

Para realizar la prueba se utilizó el software estadístico SPSS.

Se definieron las hipótesis, definiéndose como hipótesis alterna que la asistencia del usuario influye sobre los resultados.

Hipótesis definidas:

- H0 (hipótesis nula): La asistencia del usuario no influye en el resultado del algoritmo.
- H1 (hipótesis alterna): La asistencia del usuario influye en el resultado del algoritmo.

Los resultados obtenidos se muestran a continuación:

**Rangos**

		N	Rango promedio	Suma de rangos
TSPConAsistencia - TSPSinAsistencia	Rangos negativos	7 <sup>a</sup>	4,00	28,00
	Rangos positivos	0 <sup>b</sup>	,00	,00
	Empates	0 <sup>c</sup>		
	Total	7		

- a. TSPConAsistencia < TSPSinAsistencia
- b. TSPConAsistencia > TSPSinAsistencia
- c. TSPConAsistencia = TSPSinAsistencia

**Estadísticos de contraste<sup>a</sup>**

	TSPConAsistencia - TSPSinAsistencia
Z	-2,366 <sup>b</sup>
Sig. asintót. (bilateral)	,018

- a. Prueba de los rangos con signo de Wilcoxon
- b. Basado en los rangos positivos.

**Fig. 6** – Resultados de la prueba de Wilcoxon en los resultados con *datasets* TSP.

**Rangos**

		N	Rango promedio	Suma de rangos
VRPConAsistencia - VRPSinAsistencia	Rangos negativos	6 <sup>a</sup>	3,50	21,00
	Rangos positivos	0 <sup>b</sup>	,00	,00
	Empates	0 <sup>c</sup>		
	Total	6		

a. VRPConAsistencia < VRPSinAsistencia

b. VRPConAsistencia > VRPSinAsistencia

c. VRPConAsistencia = VRPSinAsistencia

**Estadísticos de contraste<sup>a</sup>**

	VRPConAsistencia - VRPSinAsistencia
Z	-2,201 <sup>b</sup>
Sig. asintót. (bilateral)	,028

a. Prueba de los rangos con signo de Wilcoxon

b. Basado en los rangos positivos.

**Fig. 7** – Resultados de la prueba de Wilcoxon en los resultados con *datasets* VRP.

Luego de la realización de la prueba en ambos resultados se obtuvo como Significación asintótica bilateral los valores 0,018 y 0,028 respectivamente. Por tanto, como el valor de la Significación asintótica bilateral es menor que 0,05 podemos afirmar que se rechaza la hipótesis nula (H0) y se acepta la hipótesis alterna (H1): la asistencia del usuario influye en el resultado del algoritmo.

## Conclusiones

Durante la presente investigación se realizó un análisis de los problemas combinatorios TSP y VRP, así como de la metaheurística Algoritmo Genético, logrando definir una serie de interacciones y visualizaciones en los mismos que fueron posteriormente integradas mediante la implementación de una herramienta. Un estudio experimental con la misma mostró que cuando el usuario interactúa con el algoritmo de optimización en tiempo de ejecución puede lograr soluciones de mejor calidad (y en ocasiones cercanas a las registradas como óptimas) y en períodos de tiempo aceptables, siempre dependiendo del conocimiento y



habilidad del usuario. La integración de técnicas de visualización durante la solución de problemas combinatorios ofrece considerables ventajas, por lo que se recomienda continuar su estudio.

## Referencias

- Applegate, D.L., Bixby, R.M., Chvátal, V. Y Cook, W.J., 2006. *The Travelling Salesman Problem*. 2006. S.L.: S.N.
- Asif, N.A., Sarker, Y., Chakraborty, R.K., Ryan, M.J., Ahamed, M.H., Saha, D.K., Badal, F.R., Das, S.K., Ali, M.F., Moyeen, S.I., Islam, M.R. Y Tasneem, Z., 2021. Graph Neural Network: A Comprehensive Review On Non-Euclidean Space. *Ieee Access*, Vol. 9, Pp. 60588-60606. Doi 10.1109/Access.2021.3071274.
- Aydemir, E. Y Karagul, K., 2020. Solving A Periodic Capacitated Vehicle Routing Problem Using Simulated Annealing Algorithm For A Manufacturing Company. *Brazilian Journal Of Operations & Production Management*, Vol. 17, No. 1, Pp. 1–13. Doi 10.14488/Bjopm.2020.011.
- Chen, Y., Guan, Z., Zhang, R., Du, X. Y Wang, Y., 2019. A Survey On Visualization Approaches For Exploring Association Relationships In Graph Data. *Journal Of Visualization*, Vol. 22, No. 3, Pp. 625-639. Issn 1875-8975. Doi 10.1007/S12650-019-00551-Y.
- Chuzhoy, J., Gao, Y., Li, J., Nanongkai, D., Peng, R. Y Saranurak, T., 2020. A Deterministic Algorithm For Balanced Cut With Applications To Dynamic Connectivity, Flows, And Beyond. *2020 Ieee 61st Annual Symposium On Foundations Of Computer Science (Focs)*. S.L.: S.N., Pp. 1158-1167. Doi 10.1109/Focs46700.2020.00111.
- Dantzig, G.B. Y Ramser, J.H., 1959. *The Truck Dispatching Problem*. 1959. S.L.: S.N.
- De Barros, R.S.M., Hidalgo, J.I.G. Y De Lima Cabral, D.R., 2018. Wilcoxon Rank Sum Test Drift Detector. *Neurocomputing*, Vol. 275, Pp. 1954-1963. Issn 0925-2312. Doi <https://doi.org/10.1016/j.neucom.2017.10.051>.
- Dokeroglu, T., Sevinc, E., Kucukyilmaz, T. Y Cosar, A., 2019. A Survey On New Generation Metaheuristic Algorithms. *Computers & Industrial Engineering*, Vol. 137, Pp. 106040. Issn 0360-8352. Doi <https://doi.org/10.1016/j.cie.2019.106040>.

- Fleitas Perdomo, A.A., 2018. *Integración De Técnicas De Visualización Para El Algoritmo De Búsqueda Metaheurística Recocido Simulado*. S.L.: Universidad Central «Marta Abreu» De Las Villas.
- Greco Chandía, M., 2020. *Resolviendo El Problema De Cobertura Del Vendedor Viajero Generalizado Con Búsqueda Heurística*. S.L.: Universidad Andrés Bello.
- Hariyadi, P.M., Nguyen, P.T., Iswanto, I. Y Sudrajat, D., 2020. Traveling Salesman Problem Solution Using Genetic Algorithm. *Journal Of Critical Reviews*, Vol. 7, No. 1, Pp. 56-61.
- Holland, J., 1975. *Adaptation In Natural And Artificial Systems*. University Of Michigan Press. Michigan, 1975.
- Houssein, E.H., Mahdy, M.A., Shebl, D. Y Mohamed, W.M., 2021. A Survey Of Metaheuristic Algorithms For Solving Optimization Problems. En: D. Oliva, E.H. Houssein Y S. Hinojosa (Eds.), *Metaheuristics In Machine Learning: Theory And Applications* [En Línea]. Cham: Springer International Publishing, Pp. 515-543. Issn 978-3-030-70542-8. Disponible En: [https://doi.org/10.1007/978-3-030-70542-8\\_21](https://doi.org/10.1007/978-3-030-70542-8_21).
- Jia, F. Y Lichti, D., 2017. A Comparison Of Simulated Annealing, Genetic Algorithm And Particle Swarm Optimization In Optimal First-Order Design Of Indoor Tls Networks. *Isprs Annals Of Photogrammetry, Remote Sensing & Spatial Information Sciences*, Vol. 4, No. 2, Pp. 18–22.
- Jiang, H., Lu, M., Tian, Y., Qiu, J. Y Zhang, X., 2022. An Evolutionary Algorithm For Solving Capacitated Vehicle Routing Problems By Using Local Information. *Applied Soft Computing* [En Línea], Vol. 117, Pp. 108431. Issn 1568-4946. Doi <https://doi.org/10.1016/j.asoc.2022.108431>. Disponible En: <https://www.sciencedirect.com/science/article/pii/S1568494622000126>.
- Li, W., Ding, Y., Yang, Y., Sherratt, R.S., Park, J.H. Y Wang, J., 2020. Parameterized Algorithms Of Fundamental Np-Hard Problems: A Survey. *Human-Centric Computing And Information Sciences* [En Línea], Vol. 10, No. 1, Pp. 29. Issn 2192-1962. Doi 10.1186/S13673-020-00226-W. Disponible En: <https://doi.org/10.1186/S13673-020-00226-W>.
- López Gutiérrez, A., 2021. *Métodos Matemáticos De Optimización. El Algoritmo Simplex* [En Línea]. 2021. S.L.: S.N. Disponible En: <http://hdl.handle.net/10902/22965>.
- Manoukian, E.B., 2022. *Mathematical Nonparametric Statistics*. S.L.: Taylor & Francis.
- Martig, S.R. Y Castro, S.M., 2000. *Visualización De Grafos*. Bahía Blanca:
- Mirjalili, S., 2019. Genetic Algorithm. En: S. Mirjalili (Ed.), *Evolutionary Algorithms And Neural*

*Networks: Theory And Applications*. Cham: Springer International Publishing, Pp. 43-55. Isbn 978-3-319-93025-1.

Mitchell, M., 1998. *L. D. Davis, Handbook Of Genetic Algorithms*. 1998. S.L.: Elsevier Science B.V.

Morfa Hernández, A., Oves García, R., Vázquez Rodríguez, R. Y Pérez Risquet, C., 2018. Integration Of Visualization Techniques To Algorithms Of Optimization Of The Metaheuristics Ant Colony. *Computación Y Sistemas*, Vol. 22, No. 1, Pp. 215-222. Doi 10.13053/Cys-22-1-2769.

Nayar, N., Gautam, S., Singh, P. Y Mehta, G., 2021. Ant Colony Optimization: A Review Of Literature And Application In Feature Selection. En: S. Smys, V.E. Balas, K.A. Kamel Y P. Lafata (Eds.), *Inventive Computation And Information Technologies*. Singapore: Springer Singapore, Pp. 285-297. Isbn 978-981-33-4305-4.

Oliinyk, A., Fedorchenko, Ievgen Stepanenko, A., Rud, M. Y Goncharenko, D., 2021. Implementation Of Evolutionary Methods Of Solving The Travelling Salesman Problem In A Robotic Warehouse. *Data-Centric Business And Applications*. S.L.: S.N., Pp. 263-292. Isbn 978-3-030-43069-6.

Papazoglou, G. Y Biskas, P., 2023. Review And Comparison Of Genetic Algorithm And Particle Swarm Optimization In The Optimal Power Flow Problem. *Energies* [En Línea], Vol. 16, No. 3. Issn 1996-1073. Doi 10.3390/En16031152. Disponible En: <https://www.mdpi.com/1996-1073/16/3/1152>.

Peres, F. Y Castelli, M., 2021. Combinatorial Optimization Problems And Metaheuristics: Review, Challenges, Design, And Development. *Applied Sciences* [En Línea], Vol. 11, No. 14. Issn 2076-3417. Doi 10.3390/App11146449. Disponible En: <https://www.mdpi.com/2076-3417/11/14/6449>.

Sánchez, M., 2017. Visualización De Datos Utilizando Grafos: Ejemplos De Aplicación. *Centro De Estudios De Administración*, Vol. 1, No. 1, Pp. 31-45.

Sar, K. Y Ghadimi, P., 2023. A Systematic Literature Review Of The Vehicle Routing Problem In Reverse Logistics Operations. *Computers & Industrial Engineering* [En Línea], Vol. 177, Pp. 109011. Issn 0360-8352. Doi <https://doi.org/10.1016/j.cie.2023.109011>. Disponible En: <https://www.sciencedirect.com/science/article/pii/S0360835223000359>.

Shi, Y. Y Zhang, Y., 2022. The Neural Network Methods For Solving Traveling Salesman Problem. En: The 8th International Conference On Information Technology And Quantitative Management (Itqm 2020 & 2021): Developing Global Digital Economy After Covid-19, *Procedia Computer Science* [En

Línea], Vol. 199, Pp. 681-686. Issn 1877-0509. Doi <https://doi.org/10.1016/J.Procs.2022.01.084>.  
Disponible En: <https://www.sciencedirect.com/science/article/pii/S1877050922000850>.

Wang, Z. Y Sheu, J.-B., 2019. Vehicle Routing Problem With Drones. *Transportation Research Part B: Methodological* [En Línea], Vol. 122, Pp. 350-364. Issn 0191-2615. Doi <https://doi.org/10.1016/J.Trb.2019.03.005>. Disponible En: <https://www.sciencedirect.com/science/article/pii/S0191261518307884>.

### **Conflicto de interés**

Los autores de este trabajo remitimos el manuscrito titulado «Integración de técnicas de visualización a la metaheurística Algoritmo Genético» y se autoriza la distribución y uso del mismo.

### **Contribuciones de los autores**

1. Conceptualización: José Manuel Cassola Bacallao, Andy Morfa Hernández, Carlos Pérez Risquet
2. Curación de datos: José Manuel Cassola Bacallao
3. Análisis formal: José Manuel Cassola Bacallao, Andy Morfa Hernández
4. Adquisición de fondos: No aplica
5. Investigación: José Manuel Cassola Bacallao, Andy Morfa Hernández
6. Metodología: José Manuel Cassola Bacallao, Andy Morfa Hernández
7. Administración del proyecto: Andy Morfa Hernández, Carlos Pérez Risquet
8. Recursos: José Manuel Cassola Bacallao, Andy Morfa Hernández
9. Software: José Manuel Cassola Bacallao
10. Supervisión: José Manuel Cassola Bacallao, Andy Morfa Hernández, Carlos Pérez Risquet
11. Validación: José Manuel Cassola Bacallao, Andy Morfa Hernández, Carlos Pérez Risquet
12. Visualización: José Manuel Cassola Bacallao, Andy Morfa Hernández
13. Redacción – borrador original: José Manuel Cassola Bacallao
14. Redacción – revisión y edición: José Manuel Cassola Bacallao, Andy Morfa Hernández