

**Solución del problema de localización-ruteo con ventanas de tiempo (LRPTW) en
condiciones de incertidumbre**

Nicolás Stefan Ruiz Vega

Trabajo de grado para optar por el título de Ingeniero Industrial

Director:

Henry Lamos Díaz

Ph.D. En Física - Matemáticas

Universidad Industrial De Santander

Facultad De Ingenierías Físico-Mecánicas

Escuela De Estudios Industriales Y Empresariales

Bucaramanga

2018

AGRADECIMIENTOS

Agradezco a Dios por llenarme de fortaleza y paciencia para alcanzar este gran logro en mi vida profesional.

Agradezco a mis abuelos María del Carmen y Antonio Ruiz por su apoyo incondicional, compañía y consejo en mi desarrollo como profesional.

Agradezco a mis padres Abraham and María Elsa por su amor y paciencia durante la realización de mi proyecto de grado

Agradezco a Johana Quiñonez por su amor, paciencia y dedicación en el desarrollo de mi proyecto de grado pues gracias a su apoyo incondicional este proceso de formación fue más ameno y tranquilo.

Agradezco a los miembros del grupo OPALO por su acompañamiento, apoyo y guía; especialmente a Fabian Torres y Lina Lozano por su asesoramiento y consejo durante la realización de mi proyecto

Agradezco a Leonardo Roper y Felipe Puentes por su asesoramiento durante el desarrollo de mi proyecto.

Agradezco al profesor Henry Lamos por su continua y efectiva guía, enriqueciendo enormemente este proceso de formación con su dirección.

Tabla de contenido

Introducción	15
1. Planteamiento del problema.....	18
2. Justificación del proyecto.....	19
3. Objetivos	21
3.1. Objetivo general.....	21
3.2. Objetivos específicos	21
4. Revisión de la literatura	22
4.1. Revisión de literatura del LRP.....	22
4.2. Revisión de literatura del LRPTW.....	27
4.3. Revisión de literatura del LRP difuso.....	28
5. Marco Teórico.....	31
5.1. Optimización combinatoria.....	31
5.1.1. Complejidad computacional.....	32
5.1.2. Métodos de solución.....	33
5.1.2.1. Métodos exactos.	33
5.1.2.2. Métodos aproximados.....	33
5.1.2.3. Algoritmos híbridos.	33

5.2. Problema de localización-ruteo (LRP).	34
5.2.1. Problema de localización de instalaciones.	34
5.2.2. Problema de ruteo de vehículo (VRP).	35
5.2.3. Clasificación del LRP.	35
5.3. Lógica difusa.	36
5.3.1. Conjuntos difusos.	37
5.3.2. Números difusos.	38
5.3.3. Medidas difusas.	39
5.3.4. Teoría de la credibilidad.	40
5.3.5. Programación lineal difusa.	43
5.3.6. Agrupamiento difuso.	44
5.3.6.1. Algoritmo fuzzy c-means (FCM).	45
5.4. Algoritmo genético	47
5.4.1. Cromosoma.	49
5.4.2. Tamaño de la población.	49
5.4.3. Criterio de selección	50
5.4.3.1. Selección por ruleta.	50
5.4.3.2. Selección por torneo.	50
5.4.4. Operadores de cruce y mutación	51
5.4.4.1. Cruce.	52

5.4.4.1.1.Cruce de un punto.	52
5.4.4.1.2.Cruce de dos puntos.	53
5.4.4.1.3.Cruce uniforme.	53
5.4.4.2.Mutación.	54
6. Problema de localización-ruteo con ventanas de tiempo (LRPTW) en condiciones de incertidumbre	55
6.4. Descripción del problema	55
6.4.1. La credibilidad y el índice de preferencia del despachador..	56
6.4.2. Fallo de ruta.....	57
6.5. Formulación del problema.....	57
6.5.1. Vectores de decisión. i valores repetidos.	61
6.5.2. Función objetivo.....	61
6.5.3. Restricciones.....	62
7. Descripción del algoritmo genético	64
7.1. Representación de la solución.....	64
7.2. Población inicial	67
7.3. Criterio de selección.	68
7.4. Operador de cruce.....	69
7.4.1. Operador de cruce de un punto.....	69
7.5. Operador de mutación.....	70

7.6. Función de adaptación	71
7.7. Diagrama de flujo del algoritmo genético	73
8. Algoritmo de simulación propuesto	75
9. Validación del algoritmo genético propuesto	76
9.1. Diseño experimental para la instancia Gaskell67-21 x 5.....	78
9.2. Diseño experimental para la instancia Gaskell67-22 x 5.....	80
9.3. Diseño experimental para la instancia Gaskell67-32 x 5.....	82
9.4. Diseño experimental para la instancia Christofides69-50x5	84
9.5. Diseño experimental para la instancia Christofides69-75x10	86
9.6. Diseño experimental para la instancia Christofides69-100x10	87
9.7. Diseño experimental para la instancia Daskin95-150x10.....	89
9.8. Resultados de los diseños experimentales	91
9.9. Ejemplos numéricos.....	95
9.9.1. Problema de prueba 1	96
9.9.2. Problema de prueba 2	100
10. Conclusiones	103
11. Recomendaciones	104
Referencias Bibliográficas	105

Lista de tablas

Tabla 1. Cumplimiento de objetivos del proyecto.	17
Tabla 2. Clasificación del LRP con respecto a su problemática.	36
Tabla 3. Instancias de Barreto (2004)	77
Tabla 4. Factores y niveles del diseño experimental 2^2	77
Tabla 5. Parámetros utilizados en el algoritmo genético para cada clase de instancia.	92
Tabla 6. Comparación de costo utilizando $P_m=0.7$ vs $P_m=0.1$ para las instancias de la literatura	92
Tabla 7. Brecha con la mejor respuesta de la literatura y tiempo computacional en segundos empleado en conseguir la mejor respuesta con el algoritmo propuesto.....	93
Tabla 8. Comparación de resultados con población inicial usando heurísticas vs. población inicial aleatoria.....	94
Tabla 9. Ventanas de tiempo en los nodos (Problema de prueba 1).	96
Tabla 10. Demandas de los nodos de clientes.....	97
Tabla 11. Resultados con diferentes valores de CrV^* (Problema de prueba 1).	98
Tabla 12. Las ventanas de tiempo en los nodos (Problema de prueba 2).	100
Tabla 13. Resultados con diferentes valores de CrV^* (Problema de prueba 2).	101

Lista de figuras

Figura 1. Representación de un conjunto difuso.....	37
Figura 2. Numero difuso que expresa la proporción “cerca de $x = 1.3$ ”.....	39
Figura 3. Una variable triangular difusa..	42
Figura 4. Puntos agrupados por FCM.	45
Figura 5. Cromosoma.....	49
Figura 6. Cruce de un punto.....	52
Figura 7. Cruce de dos puntos.....	53
Figura 8. Cruce uniforme.....	54
Figura 9. Ajuste entre las secciones de la representación y los vectores de decisión del modelo matemático.....	65
Figura 10. Representación de una solución con nueve clientes, cuatro vehículos y tres depósitos.	66
Figura 11. Un ejemplo de aplicación del operador de cruce de un punto.....	70
Figura 12. Un ejemplo de aplicación del operador de mutación.	71
Figura 13. Diagrama de flujo del algoritmo genético.	74
Figura 14 . Diagrama de Pareto de efectos estandarizados de la Gaskell67-21 x 5.	78
Figura 15. Gráfica de Efectos principales para la instancia Gaskell67-21 x 5	78
Figura 16. Gráfica de interacción para Costo de la instancia Gaskell67-21 x 5	79
Figura 17. Diagrama de Pareto de efectos estandarizados de la Gaskell67-22x5.	80
Figura 18. Gráfica de Efectos principales para la instancia Gaskell67-22 x 5.	80

Figura 19. Gráfica de interacción para Costo de la instancia Gaskell67-22 x 5.....	81
Figura 20. Diagrama de Pareto de efectos estandarizados de la Gaskell67-32x5.	82
Figura 21. Gráfica de Efectos principales para la instancia Gaskell67-32 x 5.	82
Figura 22. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.....	83
Figura 23. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5.....	84
Figura 24. Gráfica de Efectos principales para la instancia Christofides69-50x5.....	84
Figura 25. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.....	85
Figura 26. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5.....	86
Figura 27. Gráfica de Efectos principales para la instancia Christofides69-50x5.....	86
Figura 28. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5.....	87
Figura 29. Gráfica de Efectos principales para la instancia Christofides69-50x5.....	88
Figura 30. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.....	88
Figura 31. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5.....	89
Figura 32. Gráfica de Efectos principales para la instancia Christofides69-50x5.....	90
Figura 33. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.....	90
Figura 34. Distancia total en el problema con diferentes valores del CrV * (Problema 1).....	99
Figura 35. Sensibilidad de la solución con respecto al valor del nivel de α (Problema 1).....	99
Figura 36. Distancias en el problema con diferentes valores del CrV * (Problema 2).....	102
Figura 37. Sensibilidad de la solución con respecto al valor del nivel de confianza α (Problema 2).	102

Lista de apéndices

(Los apéndices están adjuntos en el CD y puede visualizarlos en base de datos de la biblioteca UIS).

Apéndice A. Análisis Bibliométrico.

Apéndice B. Código en Matlab.

Apéndice C. Artículo de carácter publicable.

Apéndice D. Datos de diseño de experimentos.

Apéndice E. Datos de ejemplos numéricos.

Apéndice F. Tiempos de viaje difuso para instancia “20-5-1a”.

RESUMEN

Título: Solución del problema de localización-ruteo con ventanas de tiempo (LRPTW) en condiciones de incertidumbre*

Autor: Nicolás Stefan Ruiz Vega **

Palabras clave: Problema de localización-ruteo, Ventanas de tiempo, Incertidumbre, Lógica difusa, Algoritmo genético.

Descripción:

En la presente investigación se aborda el problema de localización-ruteo con ventanas de tiempo (LRPTW por sus siglas en inglés) en condiciones de incertidumbre. Decisiones importantes para la gestión de la cadena de suministro (SCM por sus siglas en inglés), como la localización de instalaciones y el diseño de una red de distribución, se incluyen como partes principales del problema. Cabe señalar que la incertidumbre de la información es uno de los retos principales en la SCM, por lo que se ha asumido que la demanda de los clientes y los tiempos de viaje de los vehículos son variables difusas en el LRPTW.

Para dar solución a este problema se utiliza un modelo matemático que involucra la teoría de la credibilidad. Así mismo, se presenta un algoritmo genético (GA por sus siglas en inglés) donde se implementa un criterio de selección por torneo, un cruce de un punto y una mutación que modifica la asignación de clientes a los depósitos. Además, parte de la población inicial se genera mediante la combinación de la heurística fuzzy c -means (FCM por sus siglas en inglés) y el método del barrido.

Para validar el método de solución propuesto, en primer lugar, se realiza un diseño de experimentos 2^2 con el fin de determinar la influencia de los factores en el valor de la función objetivo. Luego se compara la solución alcanzada por el algoritmo genético con la mejor solución encontrada para algunas instancias en la literatura, dando como resultado mejores soluciones en las instancias de menor tamaño. Finalmente, se realiza un ejemplo numérico, en el cual se utiliza un algoritmo de simulación para determinar un valor aproximado de distancias adicionales debido a fallos de ruta y un valor de credibilidad en el cumplimiento de las ventanas de tiempo.

* Proyecto de Grado.

** Facultad de Ingenierías Físico Mecánicas. Escuela de Estudios Industriales y Empresariales. Programa de Ingeniería Industrial. Director PhD. Henry Lamos Diaz.

ABSTRACT

Title: Solution for the localization-routing problem with time windows (LRPTW) in conditions of uncertainty*

Author: Nicolás Stefan Ruiz Vega **

Key words: Localization-routing problem, Time windows, Uncertainty, Diffusive logic, Genetic Algorithm.

Description:

The current research project addresses the localization-routing problem with time windows (LRPTW) in conditions of uncertainty. Vital decisions on supply chain management (SCM), such as facilities location and a distribution network design are included as main parts of the problem. It is important to highlight that since the uncertainty in the information is one of the main challenges in the SCM, customers demand and vehicles traveling time have been considered to be fuzzy variables in the LRPTW.

In order to ease and solve this problem, a mathematical model which includes the use of the credibility theory has been applied. Likewise, a genetic algorithm (GA) which implements a tournament selection criterion, a single-point crossover and a mutation that modifies the customers-depots assignation has been administered. Besides, a proportion of the initial sample is generated by the combination of the heuristic fuzzy c- means (FCM) and the sweep method.

To validate the proposed solution approach, firstly, a 2^2 experiment design is done to determine the factors influence in the objective function value. Then, the solution achieved by the genetic algorithm is compared to the best solution found for instances in literature, resulting into better solutions for minor instances. Finally, a numerical example which applies a simulation algorithm to determine an approximate value for the additional distances originated due to routing failures and a credibility value in the time windows accomplishment has been presented.

* Graduation Project

** Faculty of physicomechanical Engineering. School of Industrial and Business Studies. Industrial Engineering. Directed by PhD. Henry Lamos Diaz.

Introducción

En algunas ocasiones es conveniente usar la modelación matemática para poder tomar buenas decisiones que ayuden a cumplir los objetivos de una empresa, tales objetivos generalmente apuntan al aumento de las utilidades y la satisfacción de los requerimientos de los clientes. Actualmente las empresas continúan asumiendo el reto de atender un cliente más exigente en el cumplimiento de las cantidades y los tiempos de entrega pactados de manera eficiente.

. La gestión de la cadena de suministro (SCM) es el proceso de planificación, ejecución y control de las actividades que van desde el aprovisionamiento de materias primas hasta la distribución final de productos a los clientes, así como los flujos de información relacionados (Ballou, 2004). Debido a la complejidad y las consecuencias que surgen al tomar decisiones que conduzcan a beneficios en una cadena de suministro, es imprescindible el uso de la investigación de operaciones como método para generar y resolver modelos matemáticos con base a problemas logísticos reales.

El transporte de mercancía hacia los clientes representa uno de los eslabones más importantes de la cadena de suministro. “Algunos factores que se deben considerar al diseñar una red de distribución son los siguientes: las necesidades del cliente que se satisfacen, y los costos relacionados a la satisfacción de estas necesidades” (Chopra, 2007, párr. 11). Programar una red de distribución involucra decisiones que van desde la determinación de las rutas hasta encontrar la ubicación idónea de los depósitos.

El problema de localización-ruteo (LRP por sus siglas en inglés) abarca tanto decisiones de tipo estratégico como decisiones de tipo táctico. Los problemas de localización de instalaciones y ruteo de vehículos han sido estudiados generalmente de manera separada, debido a que la localización es una decisión estratégica tomada por un largo período de tiempo, mientras que el ruteo es una

decisión táctica que puede ser modificada dinámicamente muchas veces en un período corto de tiempo. J.W. (Escobar, Linfati & Jaimes, 2015). En la literatura científica se ha comprobado que darle solución al problema de forma simultánea reduce de manera significativa los costos logísticos en la cadena de suministro.

La incertidumbre presente al tomar decisiones frente al LRP puede ser modelada por variables estocásticas o variables difusas. En la mayoría de los trabajos encontrados en la literatura científica sobre el diseño de redes de cadena de suministro en condiciones de incertidumbre, la incertidumbre se modela a través de la programación estocástica basada en escenarios; sin embargo, una gran cantidad de escenarios utilizados para representar la incertidumbre puede conducir a problemas computacionalmente desafiantes (Pishvae & Torabi, 2010). Cuando los datos disponibles para modelar el LRP no son confiables o son propensos a errores, el uso de las variables difusas es de mayor utilidad.

Este trabajo de investigación pretende solucionar el problema localización-ruteo con ventanas de tiempo (LRPTW), donde el cliente está disponible para ser atendido en una ventana de tiempo determinada. Además, se hace uso de la matemática difusa y la teoría de la credibilidad para representar la incertidumbre en las demandas de los clientes y los tiempos de viaje de los vehículos. El problema ha sido tratado con anterioridad por Zarandi, Hemmati, Davari & Turksen (2013), quienes utilizan una metaheurística de recosido simulado (SA por sus siglas en inglés) para resolver el problema. Entre las perspectivas de investigaciones futuras propuestas por Zarandi et al., (2013), está la posibilidad de utilizar otras heurísticas o metaheurísticas para resolver el mismo problema, por ello el presente trabajo de investigación pretende comprobar la fortaleza del algoritmo genético en la solución del LRPTW.

La presente propuesta está organizada de la siguiente manera: en el capítulo 4 se encuentra la revisión de literatura relevante sobre el LRPTW y la lógica difusa. Luego en el capítulo 5 se presenta el marco teórico. En el capítulo 6 se describe el problema a tratar en este trabajo, el LRPTW en condiciones de incertidumbre. En el capítulo 7 se presenta el algoritmo genético propuesto para resolver el problema. En el capítulo 9 se presenta la validación del algoritmo propuesto y se muestran los resultados computacionales. Por último, en el capítulo 10 y el capítulo 11 se muestran respectivamente las conclusiones del presente trabajo y las recomendaciones para futuras investigaciones.

Tabla 1.

Cumplimiento de objetivos del proyecto.

Objetivo	Cumplimiento
Realizar una revisión de la literatura sobre el problema de localización y ruteo con ventanas de tiempo (LRPTW) en condiciones de incertidumbre.	Capítulo 4
Desarrollar un algoritmo genético para la solución del problema de LRPTW en condiciones de incertidumbre generando soluciones iniciales mediante la heurística fuzzy <i>c</i> -means (FCM) y el método de barrido.	Capítulo 7
Implementar el algoritmo genético utilizando un software.	Capítulo 9
Validar la solución alcanzada con el algoritmo genético mediante experimentos numéricos que permitan medir su eficiencia.	Capítulo 9
Realizar un artículo académico de carácter publicable basado en el trabajo de investigación realizado.	Apéndice C

1. Planteamiento del problema

El LRP clásico parte de un conjunto de ubicaciones potenciales para los depósitos, una flota homogénea de vehículos con capacidad limitada y un conjunto de clientes con demandas determinadas. Prodhon & Prins (2014) afirman que la solución al LRP consiste en abrir un subconjunto de depósitos, asignarles clientes y determinar rutas de vehículos, para minimizar el costo total que incluye el costo de los depósitos abiertos, los costos fijos de los vehículos utilizados y el costo del desplazamiento.

Por otro lado, en algunas circunstancias reales al realizar la planeación de las rutas, la demanda de los clientes no es conocida exactamente por el despachador, y el tiempo de trayecto de un vehículo en una ruta es impreciso debido a la congestión en el tráfico o el estado de las vías. De manera que aplicar el LRP en situaciones reales genera una alta incertidumbre en los parámetros del problema. (Zarandi et al, 2013) refiere que un problema bajo incertidumbre puede ser modelado usando diversos métodos, tales como: el uso de variables aleatorias o variables difusas. La utilización de variables difusas o aleatorias en un modelo depende directamente de la semántica del problema y también de la disponibilidad de datos fiables.

En la mayoría de los casos, la imprecisión de la información es tal que no es posible establecer una distribución de probabilidad confiable; por tal motivo, es preferible representar la incertidumbre en los parámetros del problema a través de las variables difusas. En cuanto al LRP, cuando algunos de sus elementos son inciertos y son representados por variables difusas, se le llama: problema de localización-ruteo difuso (FLRP por sus siglas en inglés). Por consiguiente, la presente investigación plantea un FLRP con demandas difusas, y tiempos de viaje difusos, en donde se utiliza una flota homogénea de vehículos con limitaciones en su capacidad para atender la demanda de un grupo de clientes dentro de una ventana de tiempo determinada (FLRPTW).

En el FLRPTW, además de las limitaciones de capacidad de los vehículos y los parámetros difusos; los clientes tienen ventanas de tiempo donde es necesario satisfacer sus demandas, lo que limita las rutas factibles del problema (Brito, 2013). Ahora bien, debido a la complejidad tipo NP-*Hard* del FLRPTW, no se puede obtener soluciones óptimas en tiempos de computación razonables por medio de métodos exactos, por tanto, es necesario el uso de heurísticas y metaheurísticas para encontrar soluciones cercanas a la óptima.

Zarandi et al. (2013) realizan una investigación sobre el FLRPTW, en donde utilizan la lógica difusa para tratar con la incertidumbre de los parámetros, además, implementan un algoritmo de recocido simulado para solucionar el problema. Con la intención de abarcar el mismo problema desde una perspectiva diferente, el presente proyecto de investigación utiliza un algoritmo genético. Sin embargo, al igual que, Zarandi et al. (2013), se recurre a los algoritmos heurísticos FCM y al método de barrido para obtener soluciones iniciales. Finalmente, la eficiencia de la solución resultante se evalúa a través de la realización de experimentos numéricos.

2. Justificación del proyecto

Actualmente el problema de localización de instalaciones y ruteo de vehículos (LRP) ha sido objeto de estudio, debido a que las empresas asumen constantemente el reto de atender un cliente más exigente en tiempos de entrega y cantidades pactadas. Además, solucionar este problema conduce a reducciones sustanciales en los costos logísticos y conlleva al desarrollo de una ventaja competitiva en las organizaciones.

De acuerdo con la revisión de literatura, en los últimos tiempos, con el fin de adaptar el problema a condiciones más reales, es necesario considerar condiciones especiales como: la disponibilidad del cliente para ser atendido, la incertidumbre presente en los tiempos de viaje y la

incertidumbre en las demandas de los clientes. Por tanto, el presente trabajo de investigación aborda el problema de localización de instalaciones y ruteo de vehículos con ventanas de tiempo duras (LRPTW), presentando la demanda de los clientes y los tiempos de viaje como variables difusas. Además, se soluciona el problema por medio de un algoritmo genético, con el fin de determinar una solución factible que presente una mejor función objetivo (minimización del costo).

Para el grupo de investigación Optimización de Sistemas Productivos, Administrativos y Logísticos (OPALO) de la Universidad Industrial de Santander, es de gran interés contribuir con el desarrollo de las empresas colombianas al avanzar en el estudio de soluciones efectivas a problemas de operación logística como el LRPTW y en la inclusión de la lógica difusa como una herramienta moderna y de implementación importante en la solución de problemas de investigación de operaciones en las organizaciones.

3. Objetivos

3.1. Objetivo general

Resolver el problema de localización-ruteo con ventanas de tiempo en condiciones de incertidumbre por medio de un algoritmo genético.

3.2. Objetivos específicos

- Realizar una revisión de la literatura sobre el problema de localización y ruteo con ventanas de tiempo (LRPTW) en condiciones de incertidumbre.
- Desarrollar un algoritmo genético para la solución del LRPTW en condiciones de incertidumbre generando soluciones iniciales mediante la heurística fuzzy *c*-means (FCM) y el método de barrido.
- Implementar el algoritmo genético utilizando un software.
- Validar la solución alcanzada con el algoritmo genético mediante experimentos numéricos que permitan medir su eficiencia.
- Realizar un artículo académico de carácter publicable basado en el trabajo de investigación realizado.

4. Revisión de la literatura

La siguiente revisión de literatura se divide en tres partes: Revisión de literatura del LRP, revisión de literatura del LRPTW y revisión de literatura del LRP difuso.

Según Hassanzadeh, Mohseninezhad, Tirdad, Dadgostari, & Zolfagharinia, (2009) el LRP es un área de investigación dentro del análisis de localización de instalaciones, con la característica distintiva de prestar especial atención al ruteo de vehículos, es decir, el LRP se puede definir como una extensión al problema de ruteo de vehículos (VRP por sus siglas en inglés). Por lo anterior, esta revisión de literatura incluye estudios sobre el VRP.

4.1. Revisión de literatura del LRP

Dantzig & Ramser (como se citó en Oviedo y Valdivieso, 2016) trabajaron el VRP por primera vez, en 1959, al asignar estaciones de gasolina a camiones que transportaban combustibles, de tal manera que se cumpliera con las demandas de cada estación y se minimizara la distancia recorrida por la flota vehicular. Con respecto a la formulación del problema, los investigadores plantearon un procedimiento basado en la programación lineal; sin embargo, no realizaron aplicaciones prácticas del modelo.

Inicialmente los problemas de localización de instalaciones y ruteo de vehículos se trabajaban de manera individual; sin embargo, con los avances en investigación, fue notoria la necesidad de combinarlos para lograr soluciones más satisfactorias. Boventer (1961), pionero en la relación entre las decisiones de localización y ruteo, implementó el método de reducción de matrices para resolver el problema de distribución, destacando el rol del costo en el alquiler de los puntos de

producción. Años más tarde, en 1973, Gandy & Dohrn (como se citó en Prodhon & Prins, 2014) fueron posiblemente los primeros en considerar de manera simultánea la visita a los clientes y la ubicación de los depósitos; lo que contribuyó en gran medida en la definición del LRP clásico.

Entre las primeras aplicaciones del LRP se encuentra el trabajo de Or & Pierskalla (1979), quienes proponen un modelo de localización-asignación y transporte para afrontar el problema de algunos bancos de sangre en la distribución del líquido a un conjunto de hospitales. Para adaptar el problema a situaciones reales, los autores plantean que los vehículos que hacen la entrega tengan restricciones en su capacidad. También, utilizan algoritmos de ordenamiento por inserción para solucionar el problema. Finalmente, prueban la eficacia de los algoritmos en una muestra de 3 bancos de sangre y 117 hospitales en la ciudad de Chicago, donde obtienen resultados satisfactorios.

En 1986, Laporte, Nobert, & Arpin (como se citó en Baldacci, Hadjiconstantinou & Mingozzi, 2004), fueron los primeros en presentar una solución al problema de localización-ruteo con limitaciones de capacidad (CLRP por sus siglas en inglés), utilizando un algoritmo exacto mediante una programación lineal entera (ILP por sus siglas en inglés) y un método de relajación de restricciones.

Tuzun & Burke (1999) presentan una metaheurística de búsqueda tabú (TS por sus siglas en inglés) de dos fases para la solución del LRP y resaltan que la arquitectura de dos fases hace posible buscar el espacio de solución de manera eficiente, produciendo así buenas soluciones sin un cálculo excesivo. Prueban la eficiencia del algoritmo TS al comparar sus soluciones con las soluciones que proporciona una heurística encontrada en la literatura que resuelve el mismo problema.

Barreto (2004) divide su trabajo de investigación en dos partes. En la primera realiza una revisión bibliográfica intensiva sobre documentos dedicados al estudio del LRP y presenta un estudio que demuestra su complejidad *NP-Hard*. En la segunda parte define el LRP con restricciones de capacidad y expone muchas aplicaciones prácticas del problema. El autor crea instancias de prueba para el LRP con el objetivo que sirvan como referencia para futuras investigaciones sobre el problema. Además, propone un algoritmo de resolución exacta que permite resolver algunos problemas de pequeña dimensión. También uso técnicas de agrupamiento en la construcción de algoritmos heurísticos para resolver el problema y evaluó la potencialidad de los métodos jerárquicos y no jerárquicos. Finalmente, la recomendación de la investigación es aprovechar los beneficios de cruzar dos áreas científicas aparentemente distintas, análisis de grupos e investigación de operaciones, y encontrar así nuevas oportunidades de investigación.

Prins, Prodhon & Calvo (2006a) presentan una metaheurística para resolver el LRP con restricciones en capacidad en vehículos y depósitos. En la primera parte del trabajo se ejecuta una versión extendida de un GRASP y se implementa un proceso de aprendizaje sobre la elección de los depósitos. En la segunda parte del trabajo genera nuevas soluciones utilizando un reencadenamiento de rutas o *path-relinking*. El método de solución al problema es evaluado con un conjunto de instancias generadas aleatoriamente, y es comparado con otras heurísticas. Finalmente, los autores afirman obtener soluciones en un tiempo razonable para el problema. Ese mismo año, Prins, Prodhon, & Calvo (2006b) presentan un algoritmo memético con gestión en la población (MA | PM por sus siglas en inglés) para resolver el LRP con restricciones de capacidad en rutas y depósitos. El método de solución al problema es evaluado en tres conjuntos de instancias y se compara con otras heurísticas. Finalmente, los autores afirman obtener resultados bastante prometedores ya que el MA | PM encuentra las mejores soluciones en varias instancias. Un año

más tarde, Prins, Prodhon, Ruiz, Soriano & Calvo (2007) presentan una metaheurística cooperativa para resolver el problema LRP con restricciones de capacidad en vehículos y depósitos. En la primera fase del trabajo, las rutas y clientes se agregan a súper clientes, lo que lleva a un problema de localización de instalaciones, que luego se resuelve mediante una relajación lagrangiana. En la segunda fase del trabajo, las rutas resultantes del problema de ruteo de vehículos con múltiples depósitos (MDVRP por sus siglas en inglés) se mejoran usando una heurística de búsqueda tabú granular (GTS por sus siglas en inglés). El método de solución es evaluado en tres conjuntos de instancias generadas aleatoriamente y se compara con otras heurísticas. Finalmente, los autores afirman obtener soluciones en un tiempo razonable y demuestran que la metaheurística utilizada supera a otros métodos de solución para el mismo problema.

Yücenur & Demirel (2011) desarrollan un algoritmo genético basado en una técnica de agrupamiento para la etapa de asignación de clientes a los depósitos en el MDVRP. El algoritmo propuesto fue probado en problemas de referencia que se obtuvieron en la literatura y se pudo demostrar que la solución, con el algoritmo genético, puede ser igual a las soluciones aplicando el algoritmo del vecino más cercano, con la diferencia de una reducción en el tiempo de cálculo.

Ting & Chen (2013) desarrollan el algoritmo de optimización de colonia de hormigas múltiples (MACO por sus siglas en inglés) para resolver el CLRP con restricciones de capacidad en depósitos y rutas. Los investigadores descomponen el CLRP de forma jerárquica en el problema de localización de instalaciones (FLP por sus siglas en inglés) y en el MDVRP, donde el segundo problema se trata como un subproblema dentro del primer problema. El algoritmo MACO se evalúa en cuatro conjuntos diferentes de instancias y se compara con otros algoritmos encontrados en la literatura. Los resultados computacionales indican que MACO es capaz de obtener soluciones

óptimas o casi óptimas en un tiempo de cálculo razonable y es competitivo con otros algoritmos bien conocidos.

Mokhtarinejad, Ahmadi, Karimi & Rahmati (2015) abordan el problema de localización-ruteo para un sistema de distribución bajo modalidad *cross-docking* (CD) considerando el envío directo de los fabricantes a los clientes. El diseño del sistema CD se realiza con la finalidad de reducir los costos de transporte y satisfacer la demanda del cliente en el tiempo mínimo. Para resolver el problema se propone un método heurístico basado en el aprendizaje de máquinas (MLBM por sus siglas en inglés), en el que los clientes, fabricantes y ubicaciones de los centros de distribución se agrupan mediante un enfoque de *bi-clustering*.

Entre las más recientes investigaciones sobre el LRP se encuentra el trabajo de Rabbani, Heidari, Farrokhi-Asl & Rahimi (2018), el cual aborda el problema LRP para el manejo de residuos peligrosos industriales considerando restricciones en la compatibilidad entre algunos tipos de desechos. Considerando las condiciones del problema, los autores proponen un modelo multi-objetivo donde simultáneamente se minimizan tres criterios: el costo total, el riesgo total de transporte de desechos peligrosos relacionados con la exposición de la población y el riesgo de la localización de cada instalación de tratamiento de desechos. El método de solución al problema consiste en implementar dos algoritmos evolutivos: el algoritmo genético de ordenación no dirigida (NSGA-II por sus siglas en inglés) y la optimización de enjambre de partículas multiobjetivo (MOPSO por sus siglas en inglés). Los autores realizan experimentos en varias instancias del problema generadas aleatoriamente y confirman la superioridad como método de solución al NSGA-II sobre la MOPSO.

4.2. Revisión de literatura del LRPTW

En 1967, Pullen & Webb (como se citó en Minh, Van Hoai & Nguyet, 2013) realizaron una de las primeras investigaciones sobre el problema de ruteo de vehículos con ventanas de tiempo (VRPTW por sus siglas en inglés), en donde se estudia la programación de horarios de servicio para algunos conductores de furgonetas de correo en el Centro de Londres, Inglaterra. El objetivo del proyecto era ajustar el horario de cada conductor con el servicio que podía prestar, en base a la reducción del tiempo de inactividad en el trabajo de los carteros.

Barkaoui & Gendreau (2013) abordan los problemas de determinación, control, ajuste y combinación de los parámetros del algoritmo genético (cruce y mutación) para permitir que el algoritmo entregue buenas y robustas soluciones para toda una familia de problemas similares. Los investigadores introducen un enfoque evolutivo adaptativo que utiliza un algoritmo genético para solucionar el problema de ruteo dinámico de vehículos con ventanas de tiempo (DVRPTW por sus siglas en inglés).

Taş, Jabali, & Van Woensel (2014) presentan el problema de ruteo de vehículos con ventanas de tiempo suaves (VRPFSTW por sus siglas en inglés); el cual consiste en permitir que los vehículos no cumplan estrictamente con las ventanas de tiempo del cliente, bajo cierta tolerancia, buscando una reducción en los costos operacionales de transporte. Con respecto al método de solución, los investigadores aplican el algoritmo de búsqueda tabú y la heurística del vecino más cercano.

Un año más tarde, Li-ying & Yuan-bin (2015) presentan el problema de localización de instalaciones de carga y ruteo de vehículos eléctricos con restricciones de capacidad con ventanas de tiempo. El método de solución propuesto consiste en una heurística híbrida que incorpora una búsqueda de vecindad variable adaptiva (AVNS por sus siglas en inglés) con el algoritmo de

búsqueda tabú. Finalmente, los investigadores realizan experimentos en instancias de prueba a pequeña y gran escala para demostrar la efectividad del algoritmo propuesto.

Entre las más recientes investigaciones sobre el LRPTW se encuentra el trabajo de Ponboon, Qureshi & Taniguchi (2016), el cual propone como método de solución al problema un algoritmo *Branch and Price*. Los investigadores prueban que el algoritmo propuesto puede resolver muchas instancias de manera efectiva. Debido a que las instancias encontradas en la literatura no tienen en cuenta las ventanas de tiempo, los autores agregan ventanas de tiempo “adecuadas” para garantizar que se puede producir una solución factible del LRPTW resultante, proporcionando así una validación del algoritmo propuesto. Finalmente, los resultados computacionales y el efecto de las ventanas de tiempo también se comparan y se discuten.

4.3. Revisión de literatura del LRP difuso

Los parámetros necesarios para la solución de problemas de optimización complejos, como el LRP, en múltiples situaciones se caracterizan por su imprecisión y ambigüedad. “La teoría de conjuntos difusos es una herramienta matemática muy conveniente para tratar la incertidumbre, la subjetividad y la indeterminación de manera apropiada” (Teodorović, 1994, p. 379). A continuación, se presentan algunos trabajos encontrados en la literatura sobre aplicaciones de la lógica difusa a problemas de optimización.

En 1991, Teodorovic & Kikuchi (como se citó en Brito, 2013) presentaron una de las primeras contribuciones a la investigación del VRP con parámetros difusos, en donde para obtener soluciones utilizan una modificación del algoritmo de Clarke y Wright con los tiempos de viaje tratados como números difusos.

Zheng & Liu (2006) trabajan el VRPTW, considerando los tiempos de viaje como variables difusas. Formulan un modelo de optimización difuso y resuelven el problema mediante un algoritmo inteligente híbrido compuesto por una simulación difusa y el algoritmo genético, enfocándose en la minimización de la distancia recorrida por la flota vehicular.

Xu, Yan, & Li (2011) abordan el VRPSTW bajo un entorno difuso, con el objetivo de minimizar el costo de transporte y maximizar el nivel de satisfacción de todos los clientes. Para modelar el problema, utilizan el concepto de valor esperado aleatorio difuso. Aunque en este estudio se consideran las ventanas de tiempo bajo incertidumbre, las demandas de los clientes son fijas, pero los autores reconocen que, en situaciones reales, la demanda suele ser cambiante e incierta por lo que sugieren considerarla como una variable difusa en futuras investigaciones. Con respecto al método de solución, aplican un algoritmo de optimización del enjambre de partículas global-local-vecino con partículas intercambiables (GLNPSO-ep por sus siglas en inglés).

Zarandi, Hemmati, & Davari (2011) abordan el problema de localización-ruteo con múltiples depósitos y limitaciones de capacidad (MDCLRP por sus siglas en inglés). El problema propuesto se define teniendo en cuenta: los costos de apertura de los depósitos, las capacidades limitadas de los vehículos, un conjunto de clientes con demandas conocidas, tiempos de viaje difusos entre nodos, y también ventanas de tiempo. En este sentido, se propone un procedimiento de recocido simulado (SA por sus siglas en inglés) para resolver el problema, donde se estima la credibilidad de las soluciones por medio de un algoritmo de simulación. El método propuesto es validado usando un problema de prueba estándar para analizar y justificar los resultados. Años más tarde, Zarandi, Hemmati, Davari & Turksen (2013) plantean el FLRPTW, en donde tanto los tiempos de viaje como las demandas de los clientes se representan con variables difusas. Por esto, los investigadores diseñan un modelo de programación con restricciones posibilistas (CCP por sus

siglas en inglés) utilizando la teoría de la credibilidad. Además, aplican un algoritmo SA para resolver el problema e inicializan las soluciones empleando un método heurístico basado en el agrupamiento fuzzy *c*-means (FCM por sus siglas en inglés) y el algoritmo de barrido.

Golozari, Jafari, & Amiri (2013) abordan el LRP considerando números difusos para determinar las demandas de los clientes, el tiempo de viaje y el tiempo de visita a los clientes. Además, agregan al problema una restricción máxima de tiempo de viaje. El problema es modelado como una programación lineal difusa, mediante la aplicación del método de función de clasificación difusa; el modelo propuesto se convierte en una programación lineal exacta (LP). Para resolver el modelo LP se utiliza un *Lingo solver*, el cual presenta una gran limitación en resolver ejemplos numéricos medios y de gran tamaño. Por lo anterior, los autores deciden utilizar un algoritmo híbrido que incluye el recosido simulado y el operador de mutación para resolver los ejemplos numéricos.

Ghaffari-Nasab, Ahari & Ghazanfari (2013) consideran el problema de localización y ruteo con demandas difusas (LRPFD) y diseñan una programación con restricciones posibilistas para modelarlo. Además, desarrollan y proponen un método híbrido de SA que incorpora un algoritmo de simulación para resolver el problema. Para manejar mejor la compensación entre la distancia planificada y la distancia adicional debida a fallos de ruta, se utilizó un índice de preferencia del despachador (Cr^*) y ambas distancias se calcularon para diferentes valores del índice. Finalmente, prueban la eficiencia del procedimiento de solución propuesto con los métodos existentes en la literatura utilizando un conjunto estándar de problemas de prueba de diferentes tamaños.

Nadizadeh & Nasab (2014) señalan que la mayoría de la literatura sobre el LRP se centra en un solo período de tiempo y presentan parámetros determinísticos. Por esto, consideran el problema de localización-ruteo dinámico con demandas difusas y limitaciones de capacidad (DCLRP-FD

por sus siglas en inglés), donde tanto los vehículos como los depósitos tienen una capacidad predefinida para atender a los clientes con demandas cambiantes durante un horizonte temporal. Los investigadores diseñan un modelo de programación con restricciones posibilistas. Con respecto al método de solución, aplican un algoritmo heurístico híbrido (HHA por sus siglas en inglés) con cuatro fases, incluyendo una simulación difusa para estimar las distancias adicionales ocasionadas por fallas en la ruta. Así mismo, aplican un método de búsqueda local para evaluar el rendimiento de los depósitos abiertos y mejorar la calidad de la solución. Finalmente, realizaron experimentos numéricos con instancias de prueba estándar del CLRP, aplicando algunos cambios para mostrar el rendimiento del modelo y la eficiencia del algoritmo propuesto.

5. Marco Teórico

5.1. Optimización combinatoria

Cunquero (2003) afirma que la optimización puede definirse como el proceso de tratar de encontrar la mejor solución a un problema determinado, de forma más precisa, los problemas de optimización se pueden expresar como sigue: encontrar el valor de unas variables de decisión (sujeto a unas restricciones) para que se alcance el valor máximo o mínimo de una determinada función objetivo. “A pesar de su nombre, la optimización no necesariamente significa encontrar la solución óptima a un problema, ya que puede ser inviable debido a las características del problema” (Baños, 2006, Summary, párr. 3).

Hay un tipo de problemas de optimización, llamados problemas de optimización combinatorios, los cuales se pueden describir de la siguiente manera: existe un conjunto de objetos con potencial ubicación en distintas posiciones y una familia de lugares en los que se deben ubicar dichos objetos; cada ubicación de los objetos en los lugares, se denomina una configuración; la

combinatoria se dedica al estudio de las configuraciones; la búsqueda de la mejor configuración se denomina un problema de optimización combinatoria y la construcción de una o más funciones de valor sobre el espacio de las configuraciones permite ordenar estas, para saber cuál es la mejor (Sánchez, 2000).

5.1.1. Complejidad computacional. Para determinar la mejor manera de abordar un problema de optimización es importante clasificarlo en función de su complejidad computacional. Esta clasificación es regida por la necesidad en espacio (cantidad de memoria necesaria para resolver el problema) y tiempo computacional (cuantos pasos se necesitan para resolverlo) (Baños, 2006). Los problemas en computación y matemáticas son comúnmente clasificados en tres categorías: P, NP y NP-C (o NP-*Hard*). Los problemas de categoría P, son aquellos para los que existe un algoritmo eficiente que halla la solución en un tiempo menor a cierto valor calculable, conforme el número de variables implicadas en el problema, utilizando una expresión polinómica (tiempo polinómico). A los problemas de clase P puede conferírseles la clasificación de deterministas (se conocen las salidas del algoritmo siempre que se conozcan sus entradas). La clase de problemas para la que existe un algoritmo no determinista, pero puede ser resuelto en tiempo polinomial, se designa como NP. González, Sánchez, Velásquez y Galve (como se citó en Schweickardt, 2009) afirman que la clase NP contiene múltiples problemas de decisión/optimización, en los que no se conoce si determinada solución es única (óptima) o si existen soluciones mejores. Para algunos problemas NP no es posible encontrar el algoritmo polinomial, y resulta más conveniente una búsqueda exhaustiva sobre el espacio de soluciones, lo que conduce a la clase NP-C o Np-*Hard* (Schweickardt, 2009).

El LRP puede considerarse la combinación de dos problemas NP-*Hard*, los cuales son el FLP y el VRP. Debido a que los dos problemas pertenecen a la clase NP-*Hard*, el LRP es también un problema NP-*Hard* (Barreto, Ferreira, Paixao, & Santos, 2007, p. 968).

5.1.2. Métodos de solución

5.1.2.1. Métodos exactos. “Son aquellos que parten de una formulación como modelos de programación lineal (enteros) o similares, y llegan a una solución factible (entera) gracias a algoritmos de acotamiento del conjunto de soluciones factibles” (Lüer, Benavente, Bustos & Venegas, 2009, 4.1. Métodos exactos, párr.1). Los métodos o algoritmos exactos no son eficientes para resolver problemas de categoría NP-*Hard*, en términos de cantidad de tiempo y memoria computacional.

5.1.2.2. Métodos aproximados. Debido a que en ocasiones no se conocen algoritmos eficientes que resuelvan los problemas tipo NP-*Hard*, es necesario buscar algoritmos que den respuestas aproximadas (cercanas a la respuesta óptima) a tales problemas, así surgen los métodos o algoritmos aproximados, los cuales pueden ser específicos para el problema tratado o formar parte de una estrategia general que se puede aplicar a multitud de problemas (Gagliardi, Hernández, Leguizamón & Taranilla, 2007). Las heurísticas y metaheurísticas son métodos aproximados.

5.1.2.3. Algoritmos híbridos. “En ellos se combinan aspectos de varias heurísticas, metaheurísticas o algoritmos exactos para obtener lo mejor de ellos” (Lüer et al., 2009, 4.4. Algoritmos híbridos, párr.1). Los métodos de hibridación de metaheurísticas pueden ser de tres clases: la primera consiste en componentes de un algoritmo dado a otro; la segunda forma se refiere

a intercambios de información entre varios algoritmos y la tercera clase es la integración de métodos aproximados (Blum & Roli, 2003).

5.2. Problema de localización-ruteo (LRP).

El LRP puede definirse como la combinación de dos problemas logísticos bastante tratados en la literatura científica, a saber, el VRP y el problema de localización de instalaciones. Chan, Y. (2005) afirma:

Solo toma un momento de reflexión darse cuenta de que, hablando de forma realista, las decisiones de ruteo no pueden separarse de las decisiones de localización, ya que una ubicación adecuada es aquella a la que se puede acceder. En la medida en que la accesibilidad esté determinada por el ruteo, el ruteo es una parte integral de la localización de la instalación (p.210).

5.2.1. Problema de localización de instalaciones. El problema consta de un conjunto clientes distribuidos espacialmente en una determinada área geográfica y los cuales demandan una cierta cantidad de productos o servicios. La demanda de los clientes debe ser cubierta por una o más instalaciones. Se debe tomar la decisión respecto a la ubicación de las instalaciones en el espacio geográfico, teniendo en cuenta las restricciones presentes en el problema. Cada ubicación que se determine para cada instalación implica un costo asociado al establecimiento de la instalación, y algunos costos operativos. Para determinar la ubicación de las instalaciones se tienen en cuenta aspectos como: la reducción de costos, el suministro de servicios asequibles, el tiempo de respuesta rápida, y otros (Scaparra, 2001).

5.2.2. Problema de ruteo de vehículo (VRP). Según Moghaddam, Ruiz & Sadjadi (2012), el VRP es un problema de optimización combinatoria en el cual se pretende servir a una serie de clientes con una flota de vehículos. En un VRP clásico, se tiene un depósito y varios vehículos que se destinan a entregar bienes o servicios a diferentes clientes; el objetivo principal es minimizar el costo asociado a la distancia de recorrido.

El VRP es uno de los problemas más conocidos y desafiantes en la programación lineal entera, es un problema *NP-Hard*, es decir, el tiempo y esfuerzo computacional requerido para resolver este problema aumenta exponencialmente respecto al tamaño del problema. Para este tipo de problemas es a menudo deseable obtener soluciones aproximadas como las que brindan los algoritmos heurísticos, que han tenido en los últimos años un gran éxito resolviendo problemas *NP-Hard* (Lüer, Benavente, Bustos y Venegas, 2009).

Una de las heurísticas más populares para resolver el VRP es el método de barrido, en el cual se forman grupos de nodos girando una semirrecta con origen en el depósito e incorporando los clientes “barridos” por dicha semirrecta hasta que se viole la restricción de capacidad. El método de barrido también es muy útil resolver la parte del ruteo de vehículos en problemas más complejos que el VRP, como el LRP. Existen otros algoritmos heurísticos populares que se usan en la solución de problemas de ruteo de vehículos, estos son: el algoritmo de ahorros, las heurísticas de inserción secuencial y las de inserción en paralelo. (Olivera, 2004).

5.2.3. Clasificación del LRP. En la literatura científica se abordan diferentes clases de LRP, con el propósito de adecuar el LRP cada vez más a situaciones reales. En la tabla 2 se muestra una clasificación al problema LRP.

Tabla 2.

Clasificación del LRP con respecto a su problemática.

Problemática	
Naturaleza de demandas	Determinista
	Estocástica
	Difusa
Naturaleza de tiempos de viaje	Determinista
	Estocástica
	Difusa
Número de depósitos	Depósito único
	Múltiples depósitos
Tamaño de flota vehicular	Vehículo único
	Múltiples vehículos
Capacidad de vehículos	Vehículo sin capacidad limitada
	Vehículo con capacidad limitada
Capacidad del depósito	Depósito sin capacidad limitada
	Depósito con capacidad limitada
Horizonte de planificación	Periodo único
	Plazos múltiples
Restricción de tiempo	Tiempo no especificado sin un plazo
	Ventanas de tiempo suaves
	Ventanas de tiempo duras
Función objetivo	Objetivo único
	Múltiples objetivos

Nota: Adaptado de Hassanzadeh, Mohseninezhad, Tirdad, Dadgostari & Zolfagharinia (2009).

5.3. Lógica difusa.

Se puede definir la lógica como el conjunto de reglas que se usan para llegar a conclusiones aceptables. La lógica tradicional es la lógica dicotómica o binaria, en la cual, al resolver un problema hay dos posibles respuestas: verdadero-falso (o bien ceros y unos). Sin embargo, la información que obtenemos del mundo real es diferente, es incierta e imprecisa. Es más

conveniente utilizar la lógica difusa cuando la complejidad del problema en cuestión es muy alta y no existen modelos matemáticos precisos, para problemas que no son lineales, y cuando se manejan definiciones y conocimiento sin definición (impreciso o subjetivo) (De Vito, 2006).

5.3.1. Conjuntos difusos. La teoría de conjuntos difusos se soporta en predicados inciertos donde sus elementos se obtienen en forma gradual y no abrupta, por medio de una función de pertenencia. Cabe señalar que un conjunto difuso puede depender tanto del universo sobre el que se considera como de los observadores que lo establecen. Pueden aplicarse varios métodos para formar conjuntos difusos, por ejemplo, uno de los acercamientos más prácticos es confiar en el conocimiento de un experto. Otro método útil es adquirir el conocimiento de múltiples expertos (Kassir, 2015).

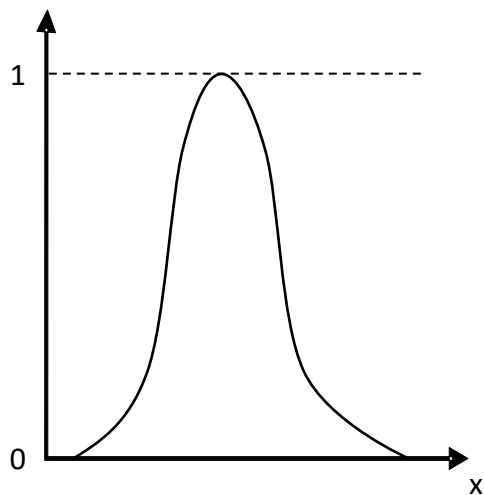


Figura 1. Representación de un conjunto difuso

Un elemento puede pertenecer a un conjunto difuso, dependiendo de su grado de pertenencia, el cual se encontrará entre 0 y 1, como se presenta en la figura 1. “Un conjunto difuso puede

representarse gráficamente como una función, especialmente cuando el universo de discurso (o dominio subyacente) es continuo (no discreto)” (Galindo, 2008, p.6). Mientras más cercano esté el valor a 0 menos podemos asegurar la pertenencia de un elemento a un conjunto. Por el contrario, cuanto más cercano esté el valor a 1 más podemos asegurar la pertenencia del elemento al conjunto (Hernández y Valdés, 2016, p.50).

Entre los diversos tipos de conjuntos difusos, de importancia especial son los conjuntos difusos que se definen en el conjunto \mathbb{R} de números reales. Como dicho conjunto es infinito, el conjunto difuso tendrá infinitos miembros. Además, como en todo conjunto difuso, a cada número real se corresponde un valor de pertenencia que es asignado por una función de pertenencia.

5.3.2. Números difusos. A cada uno de los miembros de un conjunto difuso le corresponde un valor de membresía que es asignado por una función de pertenencia. Esta función tiene claramente un significado cuantitativo y bajo ciertas condiciones, se puede ver al conjunto difuso como un número difuso.

Un número difuso se define como un conjunto normalizado y convexo A , cuya función de pertenencia tiene el valor funcional $A(x) = 1$ justo para un elemento. Es decir, un conjunto difuso puede verse intuitivamente como el conjunto de números cercanos a un determinado valor de x en A (Reina y Moscovitz, 2008). Para ilustrar mejor esta situación, en la figura 2 se presenta un ejemplo de un número difuso que expresa la proporción “cerca de $x=1.3$ ”.

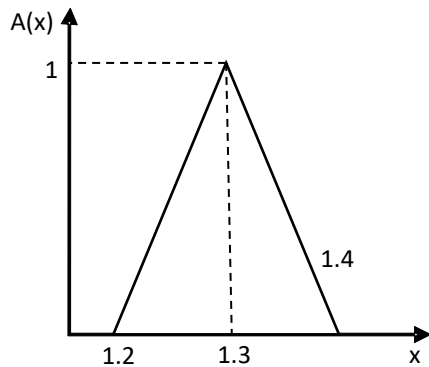


Figura 2. Numero difuso que expresa la proporción “cerca de $x = 1.3$ ”. Adaptado de Reina y Moscovitz (2008).

5.3.3. Medidas difusas. En muchas situaciones reales, cuando hay manipulación de la información es muy frecuente encontrarse con obstáculos debido a la incertidumbre dada por la imposibilidad de utilizar datos exactos. Dicha incerteza no es necesariamente aleatoria; por eso esta incertidumbre no tiene que ser medida a través de la probabilidad.

La probabilidad es un ejemplo muy importante de medida clásica, pero es aplicable solo a ciertos casos especiales de incertidumbre basada en aleatoriedad, la cual no es aplicable a la imprecisión de la información, que son la base del razonamiento humano.

Los conjuntos difusos dan una solución al problema presentado por la medida de probabilidad. Las medidas difusas completan la solución, pues son una forma más adecuada de medir ciertos grados de incertidumbre, valores que dependen únicamente de la subjetividad humana (Díaz & Lamus, 2013).

Las medidas difusas que se han convertido en fundamento para algunas teorías matemáticas, algunos ejemplos de estas son: la teoría de la probabilidad, teoría de la plausibilidad y la teoría de la credibilidad.

Con respecto a la medida de la probabilidad, esta mide la frecuencia con la que se obtiene un resultado al llevar a cabo un experimento aleatorio del cual todos los resultados posibles son conocidos. En relación con la medida de plausibilidad, esta sirve para medir grados de verosimilitud, los cuales representan la máxima creencia en una hipótesis como resultado de una evidencia. Por otro lado, la medida de la credibilidad se centra en el grado de credibilidad que se asigna a un evento para ocurrir, desde el punto de vista y de acuerdo con la experiencia de la persona que toma las decisiones (Ayyub y Klir, 2006).

5.3.4. Teoría de la credibilidad. En el año 1978 surge la teoría de la posibilidad, la cual se ha desarrollado como un fundamento para la teoría de los conjuntos difusos y como una teoría que complementa a la probabilidad (Dubois & Prade, 2012). Con el paso del tiempo surgió la teoría de la credibilidad, la cual es una modificación a la teoría de la posibilidad. A continuación, se presentan algunas definiciones respecto a la teoría de la credibilidad.

Sea Ω un conjunto no vacío, \mathcal{P} sea el conjunto potencia o de partes de Ω (conjunto de todos los subconjuntos de Ω), y Cr una medida de credibilidad. A continuación, el triplete $(\Omega, \mathcal{P}, Cr)$ se llama un espacio de credibilidad (Liu, 2002; Zarandi et al., 2013).

Una variable difusa se puede definir como una función del espacio de credibilidad $(\Omega, \mathcal{P}, Cr)$ al conjunto de números reales (Liu, 2002; Zarandi et al., 2013).

Sea E una variable difusa en el espacio de credibilidad $(\Omega, \mathcal{P}, Cr)$. Entonces su función de pertenencia se deriva de la medida de credibilidad Cr por: $\mu(x) = (2Cr\{E = x\}) \wedge 1, x \in \mathfrak{R}$. Donde \wedge es el operador mínimo (Liu, 2002; Zarandi et al., 2013).

Sea E una variable difusa en un espacio de posibilidad (Ω, P, Cr) . Entonces el conjunto $\mathcal{E}_\alpha = \{E(\theta) | \theta \in \Omega, Pos\{\theta\} \geq \alpha\}$ es llamado el conjunto α -nivel de E (Zheng & Liu, 2006; Zarandi et al., 2013).

Las medidas de posibilidad, necesidad y credibilidad pueden ser utilizadas para comparar cantidades difusas. La posibilidad mide la mejor situación mientras la necesidad mide la peor situación, es decir, la posibilidad representa el optimismo hacia la ocurrencia de los eventos difusos y la necesidad presenta el pesimismo (Brito, 2013). Por último, la credibilidad de un evento difuso se define como el promedio de su posibilidad y necesidad.

Sea (Ω, P, Pos) un espacio de posibilidad, y A sea un conjunto en $P(\Omega)$. Entonces, la medida de credibilidad de A es definida por $Cr\{A\} = \frac{1}{2} (Pos\{A\} + Nec\{A\})$. Si la función de pertenencia de E se da como μ (u es un evento) (Zheng & Liu, 2006; Zarandi et al., 2013). Entonces la posibilidad, la necesidad y la credibilidad del evento borroso $\{E \geq r\}$ pueden ser representadas por:

$$Pos\{E \geq r\} = \sup_{u \geq r} \mu(u)$$

$$Nec\{E \geq r\} = 1 - \sup_{u < r} \mu(u)$$

$$Cr\{E \geq r\} = \frac{1}{2} (Pos\{E \geq r\} + Nec\{E \geq r\})$$

Un evento difuso puede fracasar, aunque su posibilidad sea 1, y permanecer, aunque su necesidad sea 0. Sin embargo, un evento difuso debe permanecer si su credibilidad es 1, y fracasar si su credibilidad es 0. Ahora se presenta un ejemplo de la posibilidad, necesidad y

credibilidad del evento difuso $\{E \geq r\}$ considerando una variable triangular difusa $E = (r_1, r_2, r_3)$ presentada en la figura 3 (Zarandi et al., 2013).

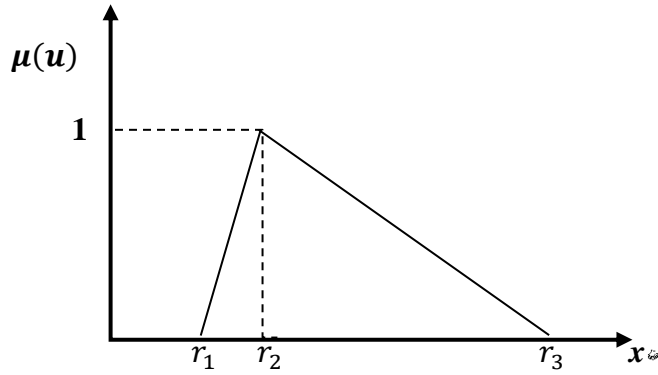


Figura 3. Una variable triangular difusa. Adaptado de Zarandi et al. (2013).

De las definiciones de posibilidad, necesidad y credibilidad, es fácil obtener:

$$Pos\{E \geq r\} = \begin{cases} 1 & \text{si } r \leq r_2 \\ \frac{r_3 - r}{r_3 - r_2} & \text{si } r_2 \leq r \leq r_3 \\ 0 & \text{si } r \geq r_3 \end{cases}$$

$$Nec\{E \geq r\} = \begin{cases} 1 & \text{si } r \leq r_1 \\ \frac{r_2 - r}{r_2 - r_1} & \text{si } r_1 \leq r \leq r_2 \\ 0 & \text{si } r \geq r_2 \end{cases}$$

$$Cr\{E \geq r\} = \begin{cases} 1 & \text{si } r \leq r_1 \\ \frac{2r_2 - r_1 - r}{2(r_2 - r_1)} & \text{si } r_2 \leq r \leq r_3 \\ \frac{r_3 - r}{2(r_3 - r_2)} & \text{si } r_2 \leq r \leq r_3 \\ 0 & \text{si } r \geq r_3 \end{cases}$$

5.3.5. Programación lineal difusa. “Entre todos los problemas de optimización, los modelos que han recibido más atención son los modelos de Programación Lineal (PL), caracterizados porque los objetivos y las funciones utilizadas en las restricciones son funciones lineales de las variables de decisión” (Brito, 2013, p.67). El objetivo es encontrar los valores mínimos o máximos de una función objetivo bajo restricciones representadas por funciones lineales (Klir & Yuan, 1995).

Un problema de programación lineal puede ser definido de la siguiente manera:

$$\text{Min } z \text{ (o Max } z) = \mathbf{c}\mathbf{x}$$

$$\text{s. a. } \mathbf{A}\mathbf{x} \leq \mathbf{b}$$

$$\mathbf{x} \geq 0$$

Donde $\mathbf{x} = \langle x_1, x_2, \dots, x_n \rangle^T$ es un vector de variables, llamadas variables de decisión. Los números denominados por c son llamados coeficientes de costo y la matriz $A = [a_{ij}]$, donde $i, j \in \mathbb{N}$, es llamado matriz de restricciones (Klir & Yuan, 1995).

Los problemas generales de PL en los que algunas de sus componentes son difusas, se denominan problemas de Programación Lineal Difusa (PLD). En un modelo con noción de factibilidad difusa, no es necesario que una solución difusa satisfaga completamente y de manera precisa las restricciones (ejemplo: en el LRP, que se cumpla con todas las demandas de los clientes o la capacidad del vehículo) (Brito, 2013).

Un problema de programación lineal difusa puede ser definido de la siguiente manera:

$$\sum_{j=1}^n C_j X_j$$

$$s. a. \sum_{j=1}^n A_{ij} X_j \leq B_i \quad (i \in \mathbb{N})$$

$$X_j \geq 0 \quad (j \in \mathbb{N})$$

Donde A_{ij}, B_i, C_j son números difusos y X_j son variables cuyos estados son números difusos. Las operaciones de multiplicación y suma son operaciones aritméticas para números difusos (operaciones de aritmética difusa). No todos los parámetros en un modelo de programación lineal difusa son números difusos, puede ocurrir que algunos parámetros sean “*crisp*” y otros sean difusos (Klir & Yuan, 1995).

5.3.6. Agrupamiento difuso. El análisis de agrupamiento difuso es una variante de los métodos de partición que tiene una ventaja sobre la teoría de conjuntos convencional (un elemento está o no incluido dentro de un grupo) con el hecho que un elemento puede pertenecer simultáneamente a varios grupos, de acuerdo con cierto grado de pertenencia. Los objetos que se encuentran en las fronteras de los grupos difusos no necesariamente pueden ser forzados a pertenecer a uno de ellos. A cada elemento se le asigna un grado de pertenencia a los grupos en el intervalo $[0,1]$, resolviendo así el problema de la frontera en estos grupos (García y Palomares, 2012).

Los algoritmos de agrupamiento tienen por objetivo formar grupos de elementos, de manera que los pertenecientes a un mismo grupo sean parecidos entre sí y distintos a los miembros de los restantes grupos. El análisis de agrupamiento es clasificado en dos categorías: jerárquicos y de partición. Los métodos jerárquicos entregan una estructura de grupos a diferentes niveles de

granularidad, según con su nivel de similitud entre los elementos, de acuerdo con un índice de distancia elegido; los de partición, en cambio, dividen el conjunto muestral en grupos internamente homogéneos (Soto, Ponzoni y Vázquez, 2006).

El algoritmo FCM es el método más utilizado en el agrupamiento difuso, y tiene sus ventajas: permite encontrar un conjunto de prototipos representativos de cada grupo y se identifican grados de pertenencia de cada elemento a cada grupo (García y Palomares, 2012).

5.3.6.1. Algoritmo fuzzy *c*-means (FCM). En muchas situaciones cotidianas ocurre el caso que un elemento está lo suficientemente cerca de dos grupos de tal manera que es muy difícil colocarlo en uno o en otro (ver figura 4), esto se presenta por la relativa frecuencia con la cual un elemento contiene características que pertenecen a grupos diferentes y como consecuencia no es fácil clasificarlo; el algoritmo fuzzy *c*-means (FCM) es una alternativa desarrollada para solucionar este inconveniente.

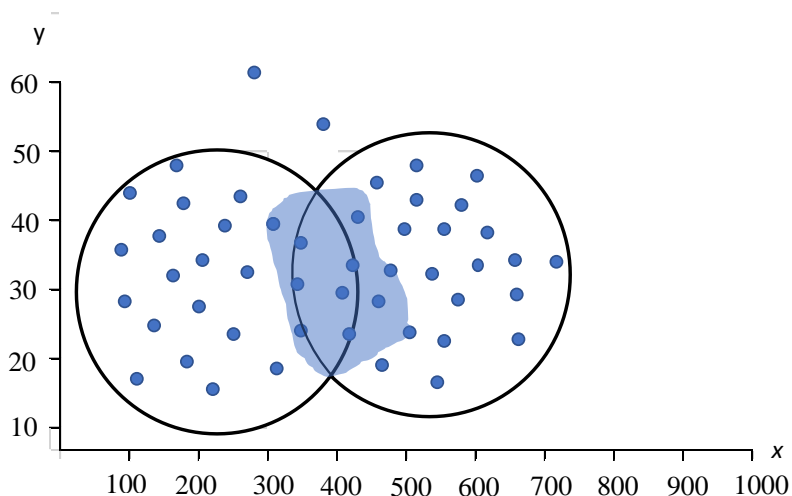


Figura 4. Puntos agrupados por FCM. Adaptado de Cox (2005).

El algoritmo FCM asigna a cada elemento un valor de pertenencia dentro de cada grupo y de esta forma un elemento específico puede pertenecer parcialmente a más de un grupo. Este algoritmo es una versión difusa de el algoritmo *c*-means clásico que trabaja con una partición dura del conjunto de datos, a diferencia con el FCM que realiza una partición suave (Porrás, Laverde & Díaz, 2008).

El FCM se basa en la minimización de la siguiente función objetivo:

$$J_m = \sum_{i=1}^D \sum_{j=1}^N \mu_{ij}^m \|x_i - c_j\|^2$$

donde:

- *D* es el número de puntos de datos a agrupar.
- *N* es el número de grupos.
- *m* es un peso exponente en la función de pertenencia para controlar el grado de superposición difusa, con $m > 1$ (generalmente $m = 2.0$). La superposición difusa se refiere a qué tan difusos son los límites entre los grupos, es decir, el número de puntos de datos que tienen una membresía significativa en más de un grupo.
- x_i es el *i*-ésimo punto de datos.
- c_j es el centro del *j*-ésimo grupo.
- μ_{ij} es el grado de membresía de x_i en el *j*-ésimo grupo. Para un punto de datos dado x_i , la suma de los valores de membresía para todos los grupos es uno.

FCM realiza los siguientes pasos para la agrupación de los puntos de datos:

1. Iniciar aleatoriamente los valores de membresía del grupo μ_{ij} .
2. Calcular los centros de grupo:

$$c_j = \frac{\sum_{i=1}^D \mu_{ij}^m x_i}{\sum_{i=1}^D \mu_{ij}^m}$$

3. Actualizar μ_{ij} de acuerdo con lo siguiente:

$$\mu_{ij} = \frac{1}{\sum_{k=1}^N \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

4. Calcule la función objetivo J_m
5. Repetir los pasos del 2 – 4 hasta que J_m mejore en menos de un umbral mínimo especificado (generalmente una mejora de $1e-5$) o hasta después de un número máximo especificado de iteraciones (generalmente 100 iteraciones).

5.4. Algoritmo genético

El algoritmo genético (GA por sus siglas en inglés) es un método metaheurístico basado en los procesos genéticos de los seres vivos. Una población de individuos evoluciona a lo largo de las generaciones de acuerdo con los principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin (1859). Para imitar este proceso, el GA crea soluciones para

problemas del mundo real. La codificación de dichas soluciones influye mucho en la evolución de estas hacia valores óptimos del problema.

El algoritmo genético usa una analogía directa con el comportamiento natural. Trabaja con una población de individuos, cada uno de los cuales representa una solución a un problema dado. A cada individuo se le asigna un valor relacionado con la bondad de dicha solución (función de adaptación). Cuanto mayor sea la adaptación de un individuo al problema, mayor será la probabilidad de que el mismo sea seleccionado para reproducirse, cruzando su material genético con otro individuo seleccionado de la misma forma. El cruce producirá nuevos individuos (descendientes o hijos), los cuales comparten algunas de las características de sus padres. También, a algunos descendientes de los padres se les aplica un operador de mutación y consiste en la alteración aleatoria (normalmente con probabilidad pequeña) de una pequeña parte que compone la solución. Cuanto menor sea la adaptación de un individuo, menor será la probabilidad de que dicho individuo sea seleccionado para la producción, y por tanto de que su material genético se propague en sucesivas generaciones. De esta manera, se produce una nueva población de posibles soluciones la cual reemplaza a la anterior y se va explorando las áreas más prometedoras del espacio de búsqueda (Moujahid, Inza & Larrañaga, 2008).

El algoritmo genético es una técnica robusta, por lo cual tiene gran poder para tratar una gran cantidad de problemas provenientes de diferentes áreas; aunque no se garantiza que se encuentre la solución óptima del problema, sí encuentra soluciones aceptables en un tiempo competitivo con algunos otros algoritmos de optimización combinatoria (Pérez, 2015).

Es importante tener en cuenta algunos conceptos importantes del algoritmo genético: Una población está compuesta por un conjunto de cromosomas; los cromosomas representan una posible solución al problema; una característica particular del cromosoma está representada por un

genoma. En muchos casos, un cromosoma puede consistir en un solo genoma, pero para problemas multiobjetivos y de restricción múltiple, un cromosoma puede constar de varios genomas. El valor particular en el genoma se llama alelo; en una representación binaria, este será uno o un cero, pero en una representación de números reales, el alelo será un número entero. Finalmente, la estructura completa de un cromosoma a menudo se denomina genotipo, es una forma práctica de referirse a todos los genomas (Cox, 2005).

5.4.1. Cromosoma. Cada elemento de una población compuesta de posibles soluciones a un problema determinado se denomina “cromosoma”, es decir, un cromosoma es el representante de una posible solución al problema dentro del algoritmo genético (Yolis, Britos, Perichisky & García, 2003). En la figura 5 se presenta un ejemplo de la representación de un cromosoma en un algoritmo genético, el cual está formado por dos partes o subcadenas, una que corresponde a un primer grupo (A) y la otra correspondiente al segundo grupo (B); ambas partes tienen la misma dimensión y cada gen representa un individuo de los N posibles a seleccionar (Pérez, 2015).

A ₁	A ₂	A ₃	...	A _N	B ₁	B ₂	B ₃	...	B _N
----------------	----------------	----------------	-----	----------------	----------------	----------------	----------------	-----	----------------

Figura 5. Cromosoma. Adaptado de Pérez (2015, p.29).

5.4.2. Tamaño de la población. Al definir este parámetro se indica el número de cromosomas que se tienen en la población para cada generación. Al tener un pequeño tamaño de las poblaciones se corre el riesgo de no cubrir adecuadamente el espacio de búsqueda, mientras que el trabajar con poblaciones de gran tamaño puede acarrear problemas relacionados con el alto costo computacional. (Moujahid et al., 2008). “De hecho estudios revelan que hay un límite a partir del cual es ineficiente elevar el tamaño de la población puesto que no se consigue una mayor

velocidad en la resolución del problema” (Arranz y Parra, 2007, párr. 11).

5.4.3. Criterio de selección. La selección es un procedimiento que se efectúa al azar sobre los individuos mejor adaptados de la población, cada individuo es seleccionado de acuerdo con el valor de su función de adaptación. En la revisión de literatura se encontraron los principales criterios de selección: selección por ruleta y selección por torneo.

5.4.3.1. Selección por ruleta. En el método de selección por ruleta a cada uno de los individuos de la población se le asigna una parte proporcional de su ajuste de una ruleta, donde la suma de los porcentajes es la unidad. Los individuos mejor adaptados recibirán una mayor porción de la ruleta. Para seleccionar un individuo es necesario generar un número aleatorio entre el intervalo $[0, 1]$ y devolver el individuo situado en esa posición de la ruleta. Esta posición se obtiene recorriendo los individuos de la población y acumulando sus proporciones en la ruleta hasta que la suma sobrepase el valor obtenido. Es un método sencillo, pero presenta dos inconvenientes: es ineficiente a medida que aumenta población y el peor individuo puede ser seleccionado más de una vez (Pose, 2000).

5.4.3.2. Selección por torneo. Este es un método de selección muy utilizado en los algoritmos evolutivos que funciona bien para resolver una amplia variedad de problemas. Consiste en realizar la selección en base a comparaciones directas entre individuos.

Existen dos tipos de selección por torneo: determinista y probabilista. En el primero se selecciona al azar un número p de individuos (generalmente $p = 2$). Después, entre los individuos seleccionados se selecciona el de mejor valor en la función de adaptación para pasarlo a la siguiente generación. En el segundo tipo de selección, únicamente se diferencia del determinista en el paso

de selección del ganador del torneo; en vez de elegir siempre el mejor individuo, se genera un número aleatorio entre el intervalo $[0, 1]$, si es mayor que un parámetro p (generalmente $0,5 < p \leq 1$) se escoge el individuo más apto y en caso contrario el menos apto (Pose, 2000).

5.4.4. Operadores de cruce y mutación. La población inicialmente evoluciona sucesivamente hacia mejores regiones del espacio de búsqueda a través de procesos probabilísticos llamados operadores genéticos. El primer operador es el de selección, con el cual se busca seleccionar aquellos individuos más adaptados en la población para dejar descendencia; entre los métodos de selección existentes están los siguientes: proporcional, ranking y torneo (Valencia, 1997). El segundo operador genético es el de cruce, el cual consiste en generar individuos “hijos” a partir de padres aptos (cada hijo y cada padre se representa por un cromosoma), lo cual se logra combinando de alguna manera los cromosomas de dichos padres (Moreno, Rivera y Ceballos, 2011). El último operador genético es el de mutación, el cual se aplica de manera individual a cada hijo, y consiste en la alteración aleatoria (normalmente con probabilidad pequeña) de los genes que componen el cromosoma (Moujahid et al., 2008).

En el algoritmo genético hay que definir 2 parámetros importantes: la probabilidad de cruce y la probabilidad de mutación. La primera probabilidad indica la frecuencia con la que se producen cruces entre los cromosomas padre, es decir, que haya probabilidad de reproducción entre ellos, y en el caso de que no exista probabilidad de reproducción, los hijos serán copias exactas de los padres, pero, en caso de haberla, los hijos tendrán partes de los cromosomas de los padres. La segunda probabilidad indica la frecuencia con la que los genes de un cromosoma son mutados, si

no se aplica mutación, los hijos son los mismos que había tras el cruzamiento, pero, en caso de que haya mutaciones, parte del cromosoma hijo es modificado (Arranz y Parra, 2007).

5.4.4.1. Cruce. Después de seleccionar los individuos, entre estos se realizan cruces para producir los descendientes que se insertaran en la siguiente generación. Este operador genético es muy importante para la transición entre generaciones, puesto que las probabilidades de cruce con las que se suele trabajar rondan el 90%. Entre los tipos de operadores de cruce más utilizados están los siguientes: cruce de un punto, cruce de dos puntos y cruce uniforme.

5.4.4.1.1. *Cruce de un punto.* Consiste en que una vez seleccionados dos individuos, se cortan sus cromosomas por un punto elegido aleatoriamente para generar dos segmentos diferenciados: la cabeza y la cola. Luego, se intercambian las colas entre los dos individuos para generar los nuevos descendientes de los padres. En la literatura se le conoce a este método con el nombre de SPX (*Single Point Crossover*) (Pose, 2000). Para ilustrar mejor el procedimiento del cruce de un punto, se presenta la figura 6.

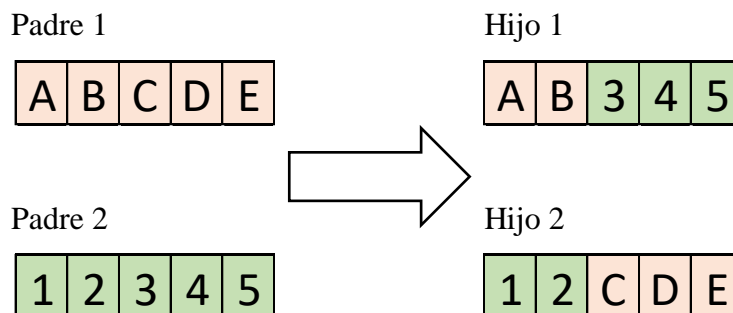


Figura 6. Cruce de un punto. Adaptado de Pose (2000).

5.4.4.1.2. *Cruce de dos puntos.* En este método se corta el cromosoma de los padres por dos puntos diferentes a los extremos del cromosoma. Para formar el descendiente se escoge el segmento central del primer padre y los segmentos laterales del segundo padre. En la literatura se le conoce a este método con el nombre de DPX (*Double Point Crossover*) (Pose, 2000). Para ilustrar mejor el procedimiento del cruce de dos puntos, se presenta la figura 7.

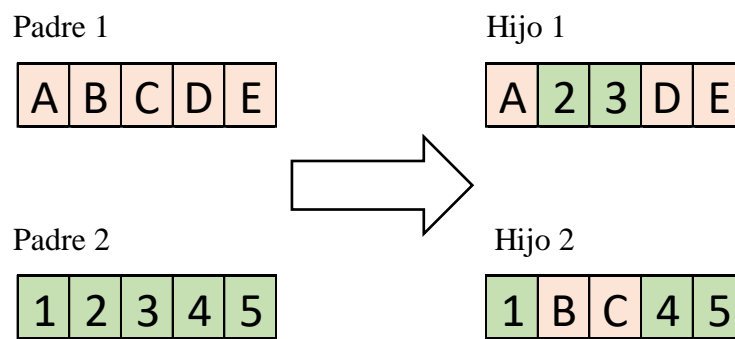


Figura 7. Cruce de dos puntos. Adaptado de Pose (2000).

5.4.4.1.3. *Cruce uniforme.* El cruce uniforme es una técnica muy diferente del cruce de un punto y del cruce de dos puntos. En este método cada gen de la descendencia tiene las mismas probabilidades de pertenecer a uno u otro padre.

Hay diferentes formas de implementar el cruce uniforme, sin embargo, la técnica implica la generación de una máscara de cruce con valores binarios. Si en alguna de las posiciones de la máscara hay un 1, el gen situado en esa posición en uno de los descendientes se copia del primer padre. Si por el contrario hay un 0, el gen se copia del segundo padre. Para producir el segundo descendiente se intercambia el papel de los padres, o también se intercambia la interpretación de los unos y de los ceros de la máscara de cruce (Pose, 2000). Para ilustrar mejor el procedimiento de cruce uniforme, se presenta la figura 8.

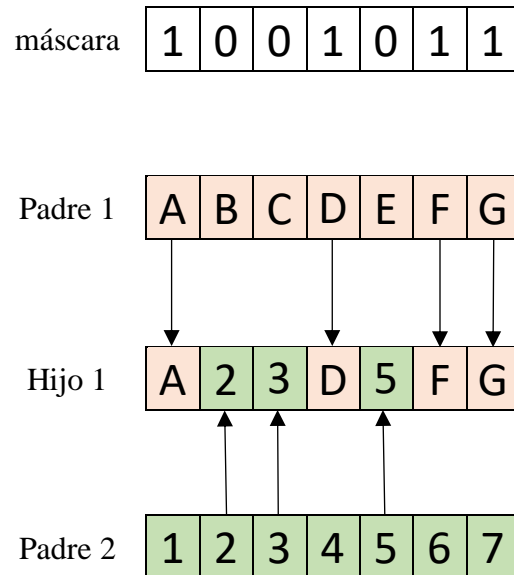


Figura 8. Cruce uniforme Adaptado de Pose (2000).

5.4.4.2. Mutación. La mutación consiste en alterar un individuo de forma aleatoria. Lo anterior permite no solo la exploración del espacio de búsqueda, sino también el aumento de la diversidad en la población.

Con la mutación se imita el comportamiento que se da en la naturaleza, pues cuando se generan los descendientes casi siempre se produce algún tipo de error, generalmente sin mayor trascendencia, en el paso de la carga genética de padres a hijos (Pose, 2000).

Por lo general las mutaciones son desfavorables, incluso letales, para el organismo mutante. Pero ocasionalmente pueden no serlo y conferirle a dicho organismo alguna ventaja que le permita sobrevivir más fácilmente en su medio. Esta característica será transmitida a sus

descendientes y un pequeño paso evolutivo se habrá dado (Kuri & Galaviz, 2002, Un poco de Biología, parr.11).

6. Problema de localización-ruteo con ventanas de tiempo (LRPTW) en condiciones de incertidumbre

6.4. Descripción del problema

El LRP, siendo una extensión del problema VRP, involucra decisiones de localización de instalaciones y ruteo de vehículos. Adicionalmente, el LRP con ventanas de tiempo (LRPTW) considera un rango de tiempo dentro del cual cada cliente tiene que ser servido por un vehículo. Sin embargo, en situaciones reales, este problema suele tener una naturaleza compleja que impone un alto grado de incertidumbre en las decisiones necesarias para solucionarlo (Klibi, Martel & Guitouni, 2010). Por lo anterior, en la literatura se recomienda utilizar la lógica difusa, principalmente la teoría de la credibilidad, para abordar de mejor manera la incertidumbre presente en el LRPTW.

El uso de la lógica difusa en el problema LRP surge siempre que algunos elementos del problema son inciertos. Por ejemplo, tanto la demanda de un cliente como el tiempo para viajar entre nodos son componentes del problema que pueden ser imprecisos hasta cierto punto. El presente trabajo contribuye al estudio del LRPTW bajo incertidumbre, el cual ha sido abordado muy poco en la literatura. En primer lugar, se utiliza un modelo de programación con restricciones posibilistas (CCP por sus siglas en inglés) propuesto por (Zarandi et al, 2013) para abordar el LRPTW con demandas difusas y tiempos de viaje difusos (FLRPTW). En segundo lugar, se propone un algoritmo genético para resolver el FLRPTW. Además, se presenta un método de

inicialización basado en agrupamientos difusos, para proporcionar al algoritmo buenas soluciones incorporadas en la población inicial.

6.4.1. La credibilidad y el índice de preferencia del despachador. Debido a la incertidumbre presente en las demandas de los clientes, durante la construcción de las rutas, se debe tomar la decisión de enviar un vehículo para servir al siguiente cliente de la secuencia o regresar al depósito y enviar otro vehículo para comenzar a servir la demanda del nuevo cliente. El uso de la teoría de la credibilidad es de gran utilidad en estos casos. El valor de la credibilidad, denotado como $Cr \in [0, 1]$, muestra el nivel de certeza sobre la capacidad de un vehículo para atender a un nuevo cliente. Cuando $Cr = 0$, el vehículo no es capaz de ninguna forma de atender al siguiente cliente y debe regresar al depósito del cual inicio la ruta correspondiente. Por otro lado, cuando $Cr = 1$, el vehículo definitivamente tendrá la capacidad de servir al próximo cliente de la secuencia. Es necesario aclarar que los valores de Cr entre 0 y 1 no representan una completa certeza sobre la capacidad de un vehículo para servir a un nuevo cliente.

Un método para tomar la decisión de enviar o no un vehículo al próximo cliente de la secuencia es definir un umbral por encima del cual se prefiere el envío de un nuevo vehículo. Este umbral es el índice de preferencia del despachador, denotado como $Cr^* \in [0,1]$. Siempre que $Cr \geq Cr^*$, un vehículo puede ser enviado a la ubicación del nuevo cliente y atenderlo. De otra forma, el vehículo tiene que regresar al depósito y comienza una nueva ruta. El índice de preferencia del despachador puede afectar considerablemente la solución. Si el despachador desea asumir un riesgo, se prefieren los valores inferiores de Cr^* , pues el despachador intenta utilizar la capacidad del vehículo tanto como sea posible. Sin embargo, esto puede ocasionar algunos problemas cuando el vehículo llega al siguiente cliente y no puede atenderlo debido a la insuficiente capacidad disponible. Por otra parte, si el despachador no es arriesgado, elige un mayor valor de Cr^* para

enviar el vehículo al siguiente cliente con mayor certidumbre, pero la capacidad de los vehículos puede ser utilizada de manera menos eficiente (Zarandi et al, 2013).

6.4.2. Fallo de ruta. Cuando un vehículo llega a la ubicación de un cliente y la capacidad del vehículo es menor a la cantidad demandada por el cliente, se presenta un “fallo de ruta”, por lo cual el vehículo tiene que volver al depósito para abastecerse y, después, atender al cliente no satisfecho (Gelves, Mora & Lamos, 2016). En este trabajo se considera un costo para la distancia adicional cuando el vehículo necesita viajar debido al fallo de ruta que se produce en algunos clientes al evaluar la ruta planificada. Como ya se ha indicado, el Cr^* determinado subjetivamente, puede tener un importante efecto tanto en la longitud de las rutas planificadas como en la distancia adicional a recorrer por “fallos de ruta”. Aunque los valores más altos de Cr^* resultan en distancias planificadas más largas, los valores más bajos aumentan el número de situaciones en las que los vehículos llegan a la ubicación del cliente y no pueden servirlos. Además, los valores más altos de Cr^* se caracterizan por un menor aprovechamiento de la capacidad del vehículo y una menor distancia adicional a cubrir por fallos de ruta. El problema consiste en determinar el valor apropiado de índice de preferencia del despachador para la construcción de las rutas, denotado como CrV^* , que da como resultado la suma mínima total de las longitudes de distancia planificada y la distancia adicional recorrida. En este trabajo de investigación, se utiliza un algoritmo de simulación para evaluar la distancia adicional ocasionada por fallos de ruta.

6.5. Formulación del problema

Entre los modelos matemáticos propuestos en la literatura para representar el problema LRPTW en condiciones de incertidumbre, aquel que más se ajusta es el modelo de programación con

restricciones posibilistas (CCP) propuesto por (Zarandi et al, 2013), en el cual se asume lo siguiente:

- La capacidad de cada vehículo es limitada y denotada como C .
- Un vehículo debe ser asignado a una única ruta.
- Cada nodo de demanda debe ser servido por un único vehículo.
- Cada ruta debe comenzar su recorrido desde un único depósito y terminarlo en el mismo depósito.
- Cada depósito potencial tiene su capacidad distinta QD_i
- El número de depósitos a localizar y el número de vehículos a utilizar son variables.
- Cada depósito y vehículo tiene un costo de apertura fijo, que se indican como ψ y η , respectivamente. Además, hay un costo unitario de transporte que se muestra como θ .
- La demanda de los clientes puede ser estimada como números triangulares difusos $d_i = (d_{1i}, d_{2i}, d_{3i})$
- El tiempo de viaje del vehículo para llegar al nodo del cliente i se indica como $f_i(x, y, z)$.

En el modelo matemático del problema se utilizan los siguientes índices, parámetros y variables:

- $i = 1, 2, 3, \dots, d$ representan los índices de los depósitos.
- $i = d + 1, d + 2, \dots, d + n$ representan los índices de los clientes.
- $k = 1, 2, \dots, m$ los índices de los vehículos.
- La capacidad de los vehículos y los depósitos se denotan como C y QD_i respectivamente.

- El parámetro S_i representa el tiempo de descarga en cada cliente.
- La distancia de recorrido se denota como D_{ij}
- La ventana de tiempo para el nodo i es representada por $[a_i, b_i]$, donde a_i y b_i son los límites inferior y superior de la ventana de tiempo.
- La demanda de cada cliente se denota como d_i
- El tiempo de viaje difuso entre el nodo i y el nodo j se denota como T_{ij}
- CrV^* es el índice de preferencia del despachador con el cual se determina si un vehículo continúa en la ruta o regresa al depósito.
- CrD^* es el nivel de confianza con el cual se determinará si las rutas son visitadas dentro de la capacidad del depósito.
- α es el nivel de confianza con el cual se determinará si los clientes son visitados dentro de la ventana de tiempo.
- La variable binaria que asigna un vehículo a un depósito es t_{kj}

$$t_{ki} = \begin{cases} 1 & \text{si el vehiculo } k \text{ es asignado a el deposito } i \\ 0 & \text{de otra forma} \end{cases}$$

- La variable binaria que indica si un depósito es usado es U_i

$$U_i = \begin{cases} 1 & \text{si el deposito } k \text{ es usado} \\ 0 & \text{de otra forma} \end{cases}$$

La capacidad disponible del vehículo después de servir a r clientes será igual a:

$$Q_r = C - \sum_{i=1}^r d_i$$

Debido a las reglas de la aritmética difusa, Q_r es un número difuso triangular, donde:

$$Q_r = \left(C - \sum_{i=1}^r d_{3i}, C - \sum_{i=1}^r d_{2i}, C - \sum_{i=1}^r d_{1i}, \right) = (q_{1,r}, q_{2,r}, q_{3,r})$$

Por lo cual, después de haber atendido r clientes, la credibilidad que la demanda del siguiente cliente no sea mayor a la capacidad restante del vehículo es como sigue:

$$Cr = Cr\{d_{r+1} \leq Q_r\} = Cr\{(d_{1,r+1} - q_{3,r}, d_{2,r+1} - q_{2,r}, d_{3,r+1} - q_{1,r}) \leq 0\} =$$

$$\begin{cases} 0 & \text{si } d_{1,r+1} \geq q_{3,r} \\ \frac{q_{3,r} - d_{1,r+1}}{2 * (q_{3,r} - d_{1,r+1} + d_{2,r+1} - q_{2,r})} & \text{si } d_{1,r+1} \leq q_{3,r}, d_{2,r+1} \geq q_{2,r} \\ \frac{d_{3,r+1} - q_{1,r} - 2 * (d_{2,r+1} - q_{2,r})}{2 * (q_{2,r} - d_{2,r+1} + d_{3,r+1} - q_{1,r})} & \text{si } d_{2,r+1} \leq q_{2,r}, d_{3,r+1} \geq q_{1,r} \\ 1 & \text{si } d_{3,r+1} \geq q_{1,r} \end{cases}$$

Entre mayor sea la capacidad restante del vehículo y menor sea la demanda del próximo cliente, la posibilidad que el vehículo pueda servir al siguiente cliente es mayor

6.5.1. Vectores de decisión. En el modelamiento del problema se incorporan tres vectores de decisión. El primero denotado por $\mathbf{x} = (x_1, x_2, \dots, x_n)$ representa a n clientes con organización $\{1, 2, \dots, n\}$ donde $1 \leq x_i \leq n$, $x_i \neq x_j$ ($i \neq j$), $i, j = 1, 2, \dots, n$. El vector x no contiene números duplicados y cambia constantemente buscando varias secuencias de nodos de clientes. El segundo vector denotado por $\mathbf{y} = (y_1, y_2, \dots, y_m)$ se emplea para delimitar los nodos de los clientes que son atendidos en una misma ruta, donde $y_0 \equiv 0 \leq y_1 \leq y_2 \leq \dots \leq y_{m-1} \leq n \equiv y_m$. Finalmente, el tercer vector denotado por $\mathbf{z} = (z_1, z_2, \dots, z_m)$ donde $1 \leq z_k \leq d$, $k = 1, 2, \dots, m$, contiene los índices de los depósitos que se habilitan para el servicio y puede tener valores repetidos.

6.5.2. Función objetivo. El objetivo del modelo es minimizar el costo de operación representado por $g(\mathbf{x}, \mathbf{y}, \mathbf{z})$, este costo total comprende: el costo de operación de los vehículos, el costo de desplazamiento y el costo de localización de depósitos. Entonces, se tiene:

$$g(\mathbf{x}, \mathbf{y}, \mathbf{z}) = (m * \eta) + \left(\theta * \sum_{k=1}^m g_k(\mathbf{x}, \mathbf{y}, \mathbf{z}) \right) + \left(\sum_{i=1}^d (U_i * \psi_i) \right)$$

Donde $g_k(\mathbf{x}, \mathbf{y}, \mathbf{z})$ es el costo de desplazamiento para el vehículo k .

$$g_k(\mathbf{x}, \mathbf{y}, \mathbf{z}) = D_{z_k, x_{y_{k-1}+1}} + \sum_{j=y_{k-1}+1}^{y_k-1} D_{d+x_j, d+x_{j+1}} + D_{d+x_{y_k}, z_k}$$

Además, se debe incorporar una función objetivo c' que busca minimizar la distancia adicional debida a fallas en la ruta. La distancia adicional se encuentra usando el algoritmo de simulación propuesto.

6.5.3. Restricciones

$$Cr\{f_i(x, y, z) \in [a_i, b_i]\} \geq \alpha \quad i = 1, 2, \dots, d + n \quad (1)$$

$$Cr\left\{\sum_{j=y_{k-1}+1}^{y_k} d_{x_j} \geq C\right\} \geq CrV^* \quad k = 1, 2, \dots, m \quad (2)$$

$$Cr\left\{\sum_{k=1}^m \sum_{j=y_{k-1}+1}^{y_k} d_{x_j} t_{ki} \leq QD_i\right\} \geq CrD^* \quad i = 1, 2, \dots, d \quad (3)$$

$$1 \leq x_i \leq n \quad i = 1, 2, \dots, n \quad (4)$$

$$x_i \neq x_j; i \neq j; i, j = 1, 2, \dots, n \quad (5)$$

$$y_0 = 0 \leq y_1 \leq y_2 \leq \dots \leq y_{m-1} \leq n = y_m \quad (6)$$

$$1 \leq z_k \leq d \quad k = 1, 2, \dots, m \quad (7)$$

$$\sum_{i=1}^d t_{ki} = 1 \quad k = 1, 2, \dots, m \quad (8)$$

$$t_{ki} \leq U_i \quad i = 1, 2, \dots, d; k = 1, 2, \dots, m \quad (9)$$

$$x_i, y_j, z_k \quad \text{son enteros} \quad i = 1, 2, \dots, n; j = 1, 2, \dots, m; k = 1, 2, \dots, m \quad (10)$$

$$t_{ki}, U_i \in \{0, 1\} \quad (11)$$

Teniendo en cuenta que los tiempos de viaje no se conocen con precisión y por tanto se presentan como variables difusas, la restricción (1) hace referencia a la restricción que estima la credibilidad que los clientes sean entendidos dentro de sus ventanas de tiempo con un nivel de confianza predeterminado α . Siendo $f_i(x, y, z)$ el tiempo de viaje del vehículo para llegar al nodo del cliente i , se hace el supuesto que si un vehículo llega antes del inicio de la ventana de tiempo debe esperar hasta que comience la ventana de tiempo. Por el contrario, el servicio inicia inmediatamente si un vehículo llega dentro de la ventana de tiempo del cliente. Para cada vehículo k , se tiene lo siguiente:

$$\begin{cases} f_{d+x_{y_{k-1}+1}}(x, y, z) = T_{z_k, d+x_{y_{k-1}+1}} \\ f_{d+x_{y_{k-1}+j}}(x, y, z) = f_{d+x_{y_{k-1}+j-1}}(x, y, z) \vee a_{d+x_{y_{k-1}+j-1}} + S_{x_{y_{k-1}+j-1}} + T_{d+x_{y_{k-1}+j-1}, d+x_{y_{k-1}+j}} \\ f_{z_k}(x, y, z) = f_{d+x_{y_k}}(x, y, z) \vee a_{d+x_{y_k}} + S_{x_{y_k}} + T_{d+x_{y_k}, z_k} \end{cases}$$

Donde \vee es el operador máximo. Dado que los tiempos de viaje entre nodos T_{ij} son considerados como variables difusas, $f_i(x, y, z)$ es también una variable difusa.

Las restricciones (2) y (3) son dos restricciones del modelo en donde también se usa la teoría de la credibilidad. La restricción (2) asegura que todos los clientes sean visitados dentro de la capacidad del vehículo con cierto nivel de confianza dado por CrV^* . De igual modo, la restricción (3) asegura que todas las rutas sean visitadas dentro de la capacidad de un depósito con un nivel de confianza determinado por CrD^* . Debido a que la localización de instalaciones es una decisión de tipo estratégico, se considera importante tener seguridad que la capacidad de un depósito es suficiente para servir una ruta, por tanto, se hace el supuesto que CrD^* es igual a 1.

Las restricciones (4) – (7) definen la constitución de los tres vectores de decisión. La restricción (8) indica que cada vehículo puede ser asignado a un único depósito. La restricción (9) señala que un vehículo es asignado a un depósito, si y solo si ese depósito está habilitado. La restricción (10) define como números enteros a los componentes de los tres vectores de decisión. Finalmente, la restricción (11) indica las variables binarias.

7. Descripción del algoritmo genético

7.1. Representación de la solución

Una representación de la solución debe determinar la secuencia de clientes a ser servidos por un vehículo específico que comienza y termina en un depósito, los clientes asignados a cada vehículo y los depósitos que se establecerán. Entre las diferentes opciones de codificación para el LRPTW, se ha utilizado una adaptación a la representación propuesta por Zarandi et al. (2013). para el problema. En la representación se consideran n clientes, m vehículos y d ubicaciones candidatas para depósitos, una representación que incorpora tres secciones de igual dimensión. La primera

sección de cada solución muestra la secuencia de clientes a ser servidos por los vehículos. La segunda sección de cada solución se utiliza para determinar los índices de los clientes a ser servidos por un vehículo. Finalmente, la última sección exhibe los depósitos que son habilitados para cada vehículo. La figura 9 muestra cómo cada sección de la representación corresponde a uno de los vectores de decisión del modelo matemático descritos en la sección 6.5.1. En efecto, mientras los valores de la primera sección corresponden a los mismos valores del vector de decisión x ; solo aquellos valores mayores a cero, en la segunda y tercera sección, corresponden a los valores de los vectores de decisión y y z respectivamente.

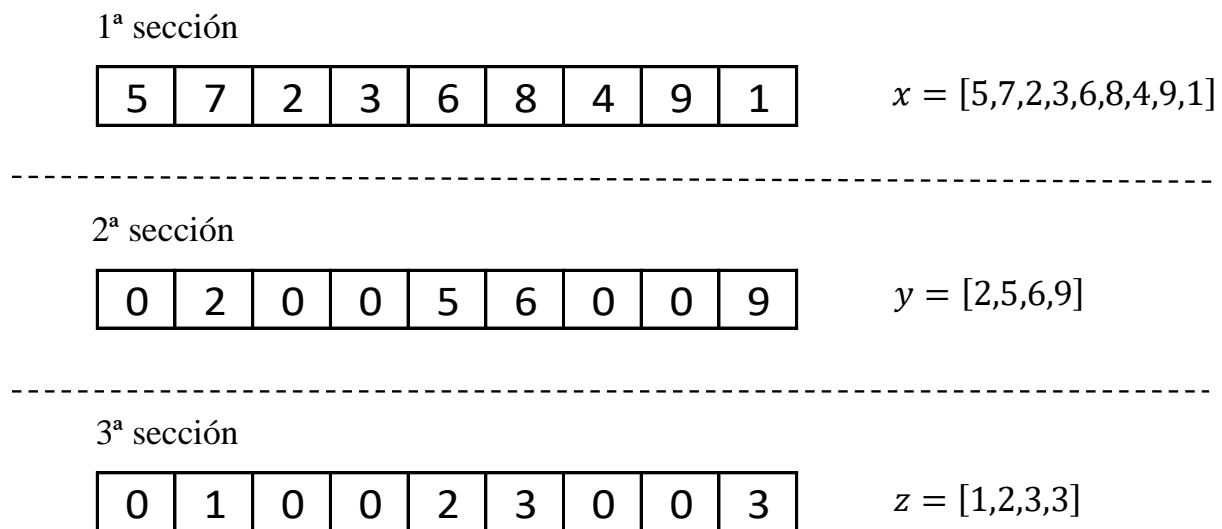


Figura 9. Ajuste entre las secciones de la representación y los vectores de decisión del modelo matemático.

Para aclarar la representación de una solución, se presenta un ejemplo simple de nueve clientes, cuatro vehículos y tres ubicaciones candidatas para depósitos. La representación y su contraparte se muestran en la figura 10. La primera sección de la solución muestra que los clientes deben ser servidos de acuerdo con el orden $[5\ 7\ 2\ 3\ 6\ 8\ 4\ 9\ 1]$. Como ya se indicó, la segunda sección

codificada determina qué clientes son servidos por cada vehículo. Dado que hay cuatro vehículos en este ejemplo, la segunda sección contiene cuatro números enteros mayores a cero que representan la posición, en la primera sección, del último cliente atendido por cada vehículo. De acuerdo con lo anterior, es claro que el último elemento de la segunda sección debe ser igual a n (número de clientes). En el ejemplo, los clientes 5 y 7 (primera y segunda posición en la primera sección) son atendidos por un solo vehículo, los clientes 2, 3 y 6 (tercera, cuarta y quinta posición en la primera sección) son servidos por un vehículo diferente, el cliente 8 (sexta posición en la primera sección) es atendido por otro vehículo, y finalmente los clientes 4, 9 y 1 (séptima, octava y novena posición en la primera sección) son atendidos por el cuarto vehículo. Por otra parte, la tercera sección muestra que el primer vehículo comienza su recorrido desde el depósito 1, el segundo vehículo sale del depósito 2 y los últimos dos vehículos comienzan su servicio desde el depósito 3.

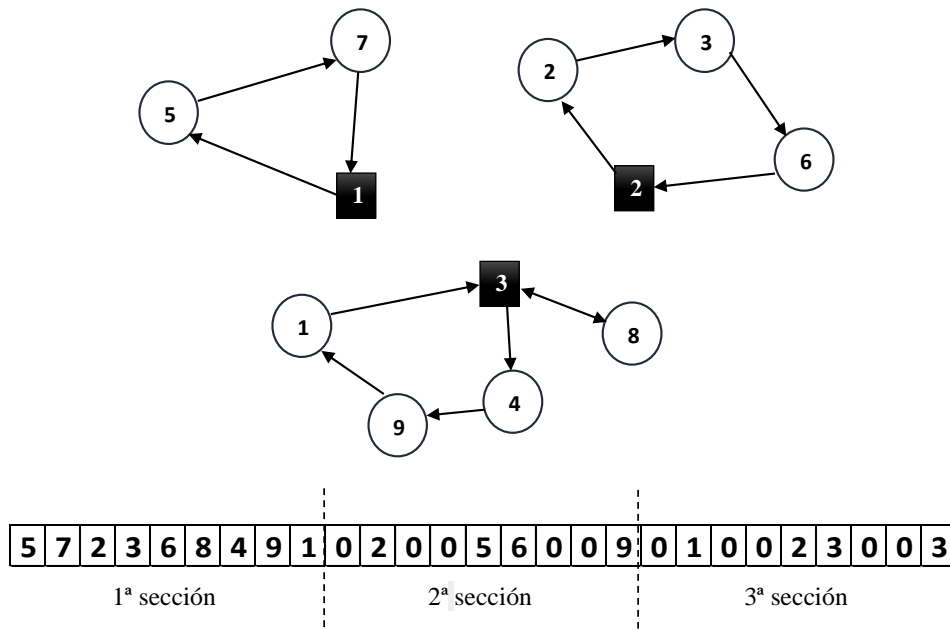


Figura 10. Representación de una solución con nueve clientes, cuatro vehículos y tres depósitos.

7.2. Población inicial

Después de definir una codificación adecuada para los individuos, el siguiente paso en un algoritmo genético es la creación de una población inicial de soluciones del problema. Generalmente, se fija un tamaño de población y se generan aleatoriamente los individuos para conseguir una población inicial amplia y representativa del espacio de búsqueda. Sin embargo, en los últimos tiempos, se ha ido incluyendo métodos heurísticos en la creación de la población inicial, con el fin de generar individuos con un buen valor en la función de adaptación (Vega, 2003).

En este trabajo se propone un método heurístico para generar buenas soluciones que hagan parte de la población inicial. En primer lugar, se emplea un algoritmo de agrupación difusa FCM para agrupar a los clientes, y que estos puedan ser asignados al depósito más cercano al centro del grupo. Luego, se utiliza el algoritmo de barrido para generar una secuencia en los clientes considerando la restricción de capacidad en los vehículos.

El método heurístico propuesto tiene una limitación en la cantidad de soluciones diferentes que puede generar. Si se desea incluir más de una solución creada por la heurística FCM en la población inicial, una alternativa es variar el número de agrupaciones o *clusters*. Sin embargo, según Ren, Liu, Wang & Yi (2016) la cantidad máxima de grupos c_{max} recomendable es \sqrt{n} ($c_{max} = \sqrt{n}$), siendo n el número de puntos de datos a agrupar. Teniendo en cuenta que mínimo se deben formar 2 grupos de datos, para un problema con n clientes, la población inicial incluye $\sqrt{n} - 1$ soluciones generadas por el método heurístico propuesto.

Cabe señalar que utilizar el método de agrupación difusa FCM, tiene dos grandes ventajas frente a la utilización de métodos de agrupación dura. En primera instancia, puede aumentar la diversificación en las soluciones, pues el FCM es un algoritmo iterativo que tiene un comportamiento aleatorio. En segundo lugar, si se utilizan métodos de agrupación dura, pueden generarse soluciones que no son factibles, pues es posible que no se logre localizar ningún depósito para uno o varios clientes (Zarandi et al., 2013).

Por otro lado, el algoritmo de barrido es una buena alternativa para generar un acomodamiento de los clientes después de la aplicación del FCM. Este método de “asignar primero rutear después” (*cluster first - route second*), consiste en formar subgrupos de clientes girando una semirrecta con origen en el depósito, incorporando los clientes “barridos” por dicha semirrecta hasta que se viole la restricción de capacidad en los vehículos (Olivera, 2004).

7.3. Criterio de selección.

Para seleccionar los mejores individuos de la población se utilizó el criterio de selección por torneo determinístico. El individuo es seleccionado al azar entre la población y se compara directamente con otros individuos seleccionados de la misma forma. La comparación entre los individuos es una competencia entre los que tengan mejor función de adaptación. La selección por torneo determinístico se describe a continuación:

1. Escoger el tamaño del torneo k (se utiliza $k=2$).
2. Elegir aleatoriamente a k padres entre la población.
3. Comparar la función de adaptación de los k y seleccionar el mejor.
4. Agregar un descendiente (copia del padre) al padre escogido en el paso anterior.
5. Repetir hasta que el número de descendientes sea igual al tamaño de la población.

7.4. Operador de cruce

El objetivo del operador de cruce es generar nuevos individuos o descendientes, los cuales comparten algunas características de sus padres. De acuerdo con el valor de la función de adaptación de cada individuo, se selecciona para que se propague parte de su material genético en sucesivas generaciones. Así, a lo largo de las generaciones se crea una nueva población con mejores características que reemplaza a la anterior. Hay una mejor exploración de las áreas más prometedoras del espacio de búsqueda a través del cruce de los individuos más adaptados (Moujahid et al., 2008).

7.4.1. Operador de cruce de un punto. En este trabajo el operador de cruce se aplica en la primera sección del cromosoma, el vector de secuencia de clientes. Se realiza un cruce de un punto, eligiendo aleatoriamente un punto de cruce cp seleccionado entre 1 y el número de clientes n . El procedimiento de cruce utilizado está basado en Murata & Ishibuchi (1994). Después de seleccionar el punto de cruce cp , se copia la permutación desde el punto cero hasta el punto de cruce del primer padre, luego se agrega esta permutación al segundo padre y se eliminan los clientes repetidos para formar la descendencia. Para mayor claridad, en la figura 11 se muestra un ejemplo con 9 clientes y un punto de cruce $cp = 3$.

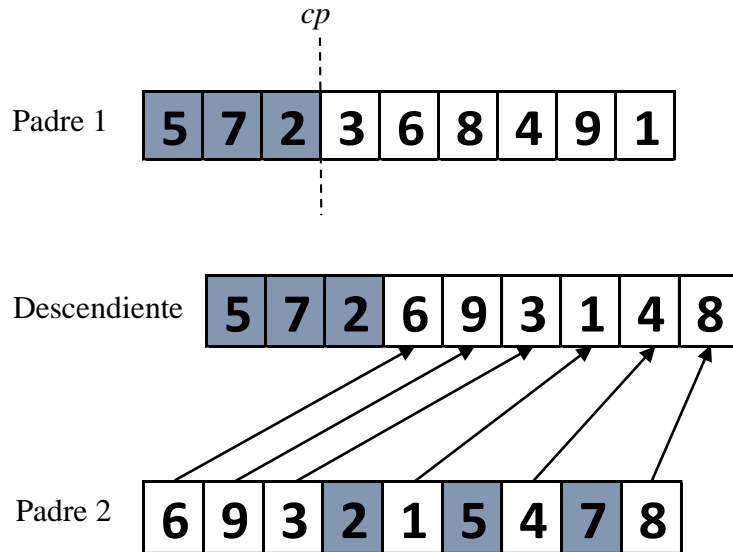


Figura 11. Un ejemplo de aplicación del operador de cruce de un punto.

7.5. Operador de mutación

El operador de mutación consiste en la alteración aleatoria de los genes que componen a los cromosomas de un hijo. Este operador garantiza que ningún espacio de búsqueda tenga una probabilidad nula de ser explorado, y es muy importante para asegurar la convergencia del algoritmo genético (Moujahid et al., 2008).

El operador de mutación utilizado selecciona un gen de la segunda sección de manera aleatoria, luego el gen cambia de valor, mientras el gen de la misma posición en la tercera sección también cambia de valor, de la siguiente forma:

- Un número aleatorio entre 1 y d (número de depósitos), si el gen de la misma posición en la segunda sección es mayor a cero.
- Cero, si el gen de la misma posición en la segunda sección es igual a cero.

Para mayor claridad, en la figura 12 se muestra un ejemplo de la aplicación del operador de mutación propuesto.

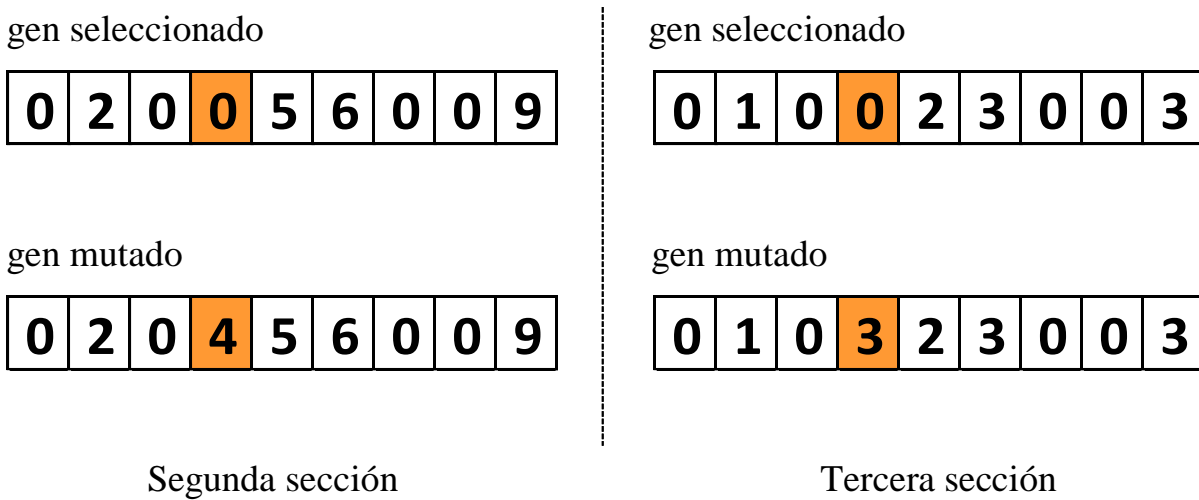


Figura 12. Un ejemplo de aplicación del operador de mutación.

7.6. Función de adaptación

Según lo afirman Derbel, Jarboui, Hanafi & Chabchoub (2012), definir la función de adaptación es importante para cualquier algoritmo genético. En este trabajo la función de adaptación para evaluar un individuo se define como la suma de dos componentes, el costo y la penalización. El primer componente corresponde a la respuesta de la función objetivo del problema y el segundo componente, la penalización, es un “castigo” a la infracción de las restricciones de capacidad y tiempo en el modelo matemático. La función de adaptación utilizada tiene la siguiente formulación:

$$\text{Funcion de Adaptación} = FO(x) + (\alpha_1 \cdot NIR(x)) + (\alpha_2 \cdot NID(x)) + (\alpha_3 \cdot NIT(x))$$

donde:

- $FO(x)$ es el valor de la función de objetivo en la solución x .
- α_1 es un coeficiente de penalización para las infracciones en la restricción de capacidad de los vehículos.
- α_2 es un coeficiente de penalización para las infracciones en la restricción de capacidad de los depósitos.
- α_3 es un coeficiente de penalización para las infracciones en la restricción de las ventanas de tiempo.
- $NIR(x)$ es el número de infracciones de la restricción de capacidad de los vehículos en la solución x .
- $NID(x)$ es el número de infracciones de la restricción de capacidad de los depósitos en la solución x .
- $NIT(x)$ es el número de infracciones de la restricción de las ventanas de tiempo en la solución x .
-

Los coeficientes de penalización pueden ser dinámicos, es decir, que el número de generaciones influye en la penalización de la función objetivo. Generalmente la penalización es mayor conforme avanza la ejecución del algoritmo (Jaramillo, 2007). El coeficiente de penalización debe ser pequeño en las primeras iteraciones, para que las soluciones que infrinjan las restricciones tengan oportunidad de generar descendientes.

7.7. Diagrama de flujo del algoritmo genético

La estructura del algoritmo genético propuesto se describe de la siguiente manera:

1. Población inicial: La población inicial es generada mediante la combinación del método heurístico propuesto (FCM + barrido) y de manera aleatoria.
2. Función de adaptación: La función de adaptación descrita en la sección 7.6 permite valorar la aptitud de los individuos en cada generación.
3. Selección: El método de selección por torneo se utiliza para seleccionar los individuos en cada generación.
4. Cruce: Después de la selección de los individuos, la operación de cruce intenta intercambiar partes de dos padres para generar nuevos descendientes. Se utiliza el cruce de un punto propuesto por Murata & Ishibuchi (1994).
5. Mutación: La mutación se utiliza para preservar la diversificación de la población y escapar de óptimos locales. El operador de mutación utilizado se describe en la sección 7.5.
6. Criterio de parada: Numero de generaciones.

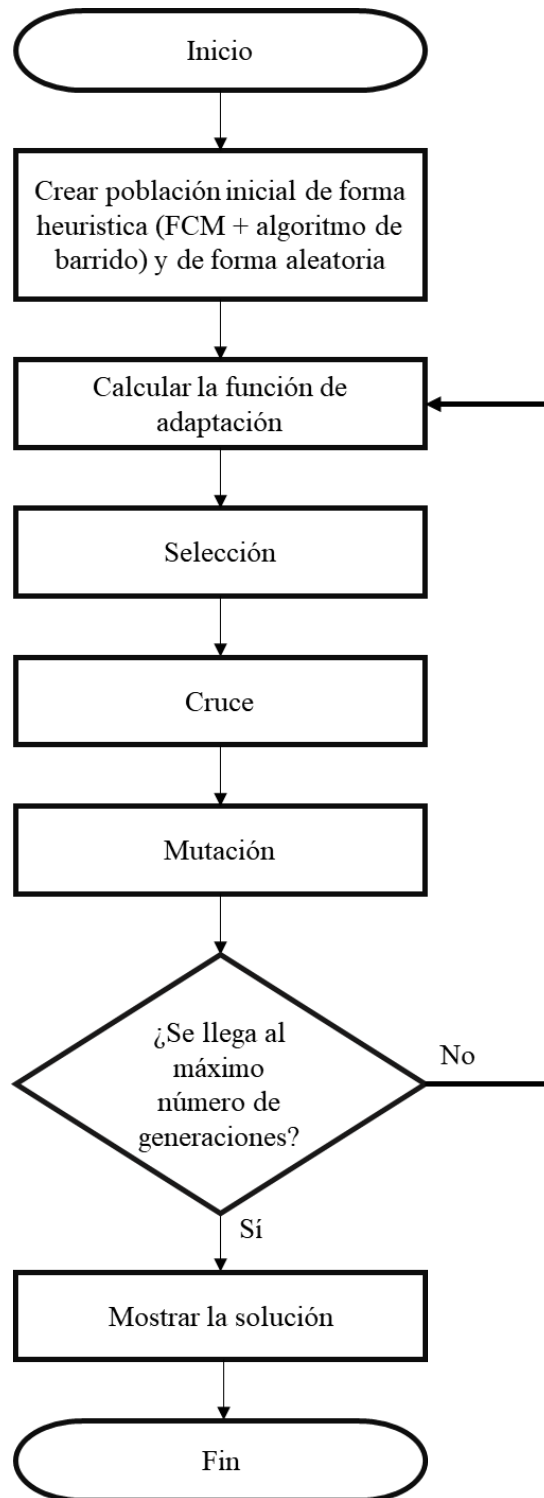


Figura 13. Diagrama de flujo del algoritmo genético.

8. Algoritmo de simulación propuesto

En muchas situaciones reales, la demanda de un cliente solo se conoce hasta que el vehículo llega al cliente, es decir, que no debe considerarse una demanda determinística al momento de definir las rutas. Además, el tiempo de viaje entre nodos puede ser impreciso, debido a la congestión de tráfico o el estado de las vías (Brito, 2013). Por lo anterior, en este trabajo se asume que la demanda de cada cliente y los tiempos de viaje son números triangulares difusos.

Para poder determinar un valor aproximado de las distancias adicionales debido a “fallos de ruta” y también definir la credibilidad de la solución en el cumplimiento de las ventanas de tiempo, se propone utilizar un algoritmo de simulación. Este algoritmo es propuesto por (Zarandi et al, 2013) y se describe a continuación:

Para cada cliente, se genera un número real x en el intervalo entre los límites izquierdo y derecho del número triangular difuso que representa la demanda del cliente y se calcula su valor de pertenencia ρ . Después, se genera un número aleatorio α en el intervalo $[0,1]$. Si $\alpha \leq \rho$, entonces la demanda real es igual a x . Por lo contrario, los números x y ρ se generan una y otra vez hasta que $\alpha \leq \rho$. Finalmente, este proceso se repite hasta se tenga una demanda simulada para cada cliente.

Por otra parte, estableciendo $T = \{T_{ij}, i, j = 0, 1, \dots, d + n\}$ y T_{ij} como el tiempo de viaje entre los nodos i y j . Entonces, se generan N números aleatorios T_{ij}^l a partir del conjunto α -nivel de T_{ij} donde l es el índice del número de la iteración. Después, se establece $T^l = \{T_{ij}^l, i, j = 0, 1, \dots, d + n\}$ y $\mu(T^l) = \mu_{11}(T_{11}^l) \wedge \mu_{12}(T_{12}^l) \wedge \dots \wedge \mu_{(n-1)n}(T_{(n-1)n}^l) \wedge \mu_{nn}(T_{nn}^l)$ donde μ_{ij} es el grado de pertenencia o membresía de $T_{ij}, i, j = 0, 1, \dots, d + n$. Finalmente, la credibilidad de servir a los

clientes entre los límites de las ventanas de tiempo puede ser estimada de acuerdo con la siguiente ecuación:

$$Cr([a, b]) = \frac{1}{2} \left(\max_{1 \leq l \leq N} \{ \mu(T^l) | f(x, y, z)(T^l) \in [a, b] \} \right. \\ \left. + \min_{1 \leq l \leq N} \{ 1 - \mu(T^l) | f(x, y, z)(T^l) \notin [a, b] \} \right)$$

La distancia adicional recorrida por los vehículos puede ser estimada obteniendo la distancia extra promedio en N iteraciones.

9. Validación del algoritmo genético propuesto

Para analizar el comportamiento del algoritmo genético propuesto para resolver el problema de localización-ruteo se llevó a cabo un experimento 2^k . El algoritmo genético fue desarrollado en el software matemático Matlab versión 2017a y el diseño factorial se realizó en el software estadístico Minitab 18.

Debido a que durante el desarrollo de la investigación no se encontraron instancias en la literatura sobre el LRP con ventanas de tiempo, para validar el algoritmo, se utilizaron las instancias de Barreto (2004) que resuelven el problema LRP clásico.

Tabla 3.

Instancias de Barreto (2004)

	<i>n x d</i>	Nombre de instancia
Clase 1	21 x 5	Gaskell67-21 x 5
	22 x 5	Gaskell67-22 x 5
	32 x 5	Gaskell67-32x 5
Clase 2	50 x 5	Christofides69-50x5
	75 x 10	Christofides69-75x10
Clase 3	100 x 10	Christofides69-100x10
	150 x 10	Daskin95-150x10

El diseño experimental fue un 2^2 , con 10 réplicas, los factores y niveles se muestran en la tabla 4.

Tabla 4.

Factores y niveles del diseño experimental 2^2

Factores	Nivel inferior	Nivel superior
POB: Tamaño de población	100	500
GEN: Número de generaciones	400	1000

9.1. Diseño experimental para la instancia Gaskell67-21 x 5

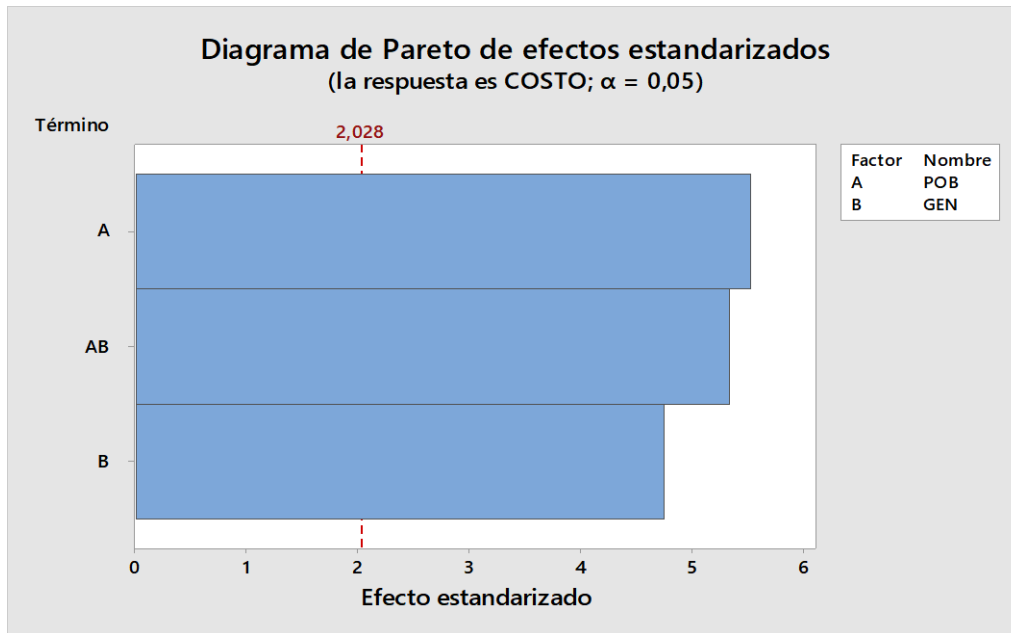


Figura 14 . Diagrama de Pareto de efectos estandarizados de la Gaskell67-21 x 5.

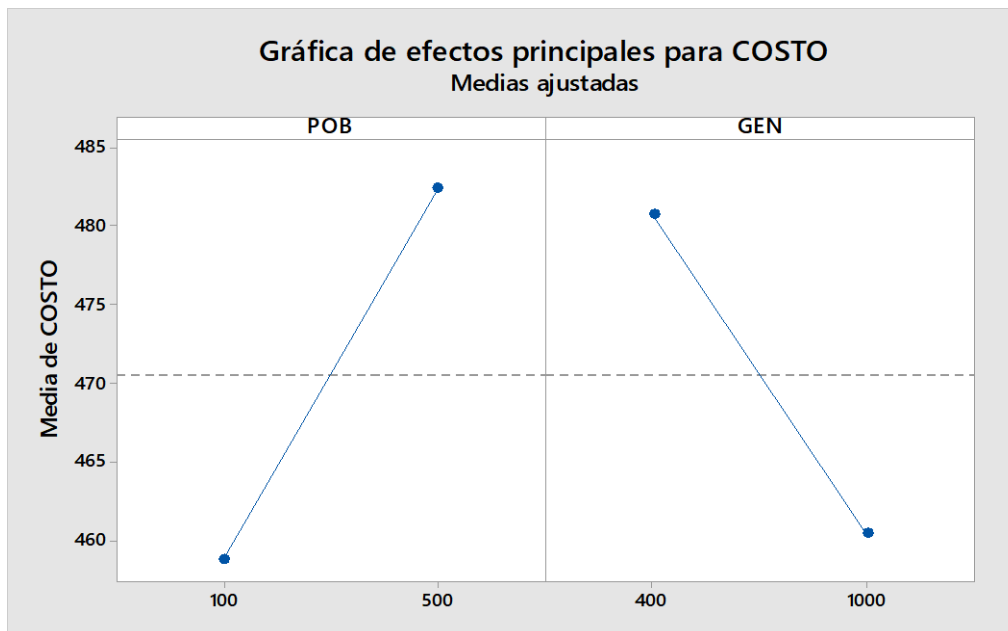


Figura 15. Gráfica de Efectos principales para la instancia Gaskell67-21 x 5

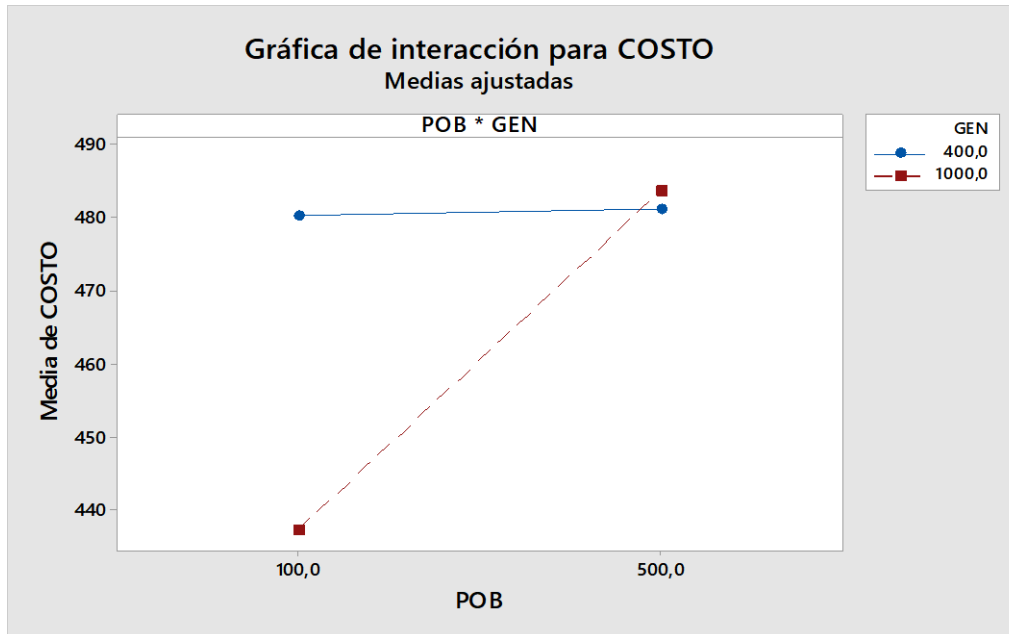


Figura 16. Gráfica de interacción para Costo de la instancia Gaskell67-21 x 5

Se presenta el diagrama de Pareto de efectos estandarizados de la instancia Gaskell6721x5 en la figura 14. Este diagrama de Pareto se utiliza para determinar la magnitud y la importancia de los efectos. Las barras de los factores que cruzan la línea de referencia y por eso son factores estadísticamente significativos, son los siguientes: tamaño de la población y la interacción entre tamaño de población y número de generaciones (POB * GEN).

En la gráfica de efectos principales presentada en la figura 15. se puede evidenciar que los niveles de los factores que minimizan el costo (función objetivo del problema) son: tamaño de población 100 y número de generaciones 1000

En la gráfica de interacción para el costo presentada en la figura 16 se puede evidenciar que para POB * GEN con 100 tamaño de población y 1000 generaciones contra el resto de las combinaciones tienen efectos muy diferentes.

Teniendo en cuenta el efecto de los factores y la interacción entre los factores POB*GEN, la mejor combinación es: 100 tamaño de población y 1000 generaciones.

9.2. Diseño experimental para la instancia Gaskell67-22 x 5

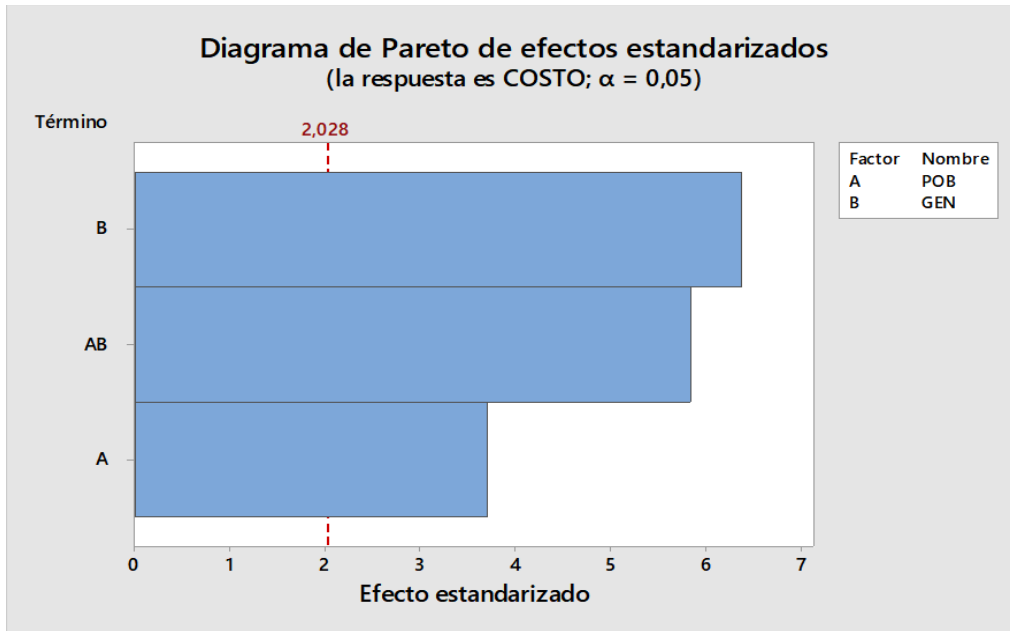


Figura 17. Diagrama de Pareto de efectos estandarizados de la Gaskell67-22x5.

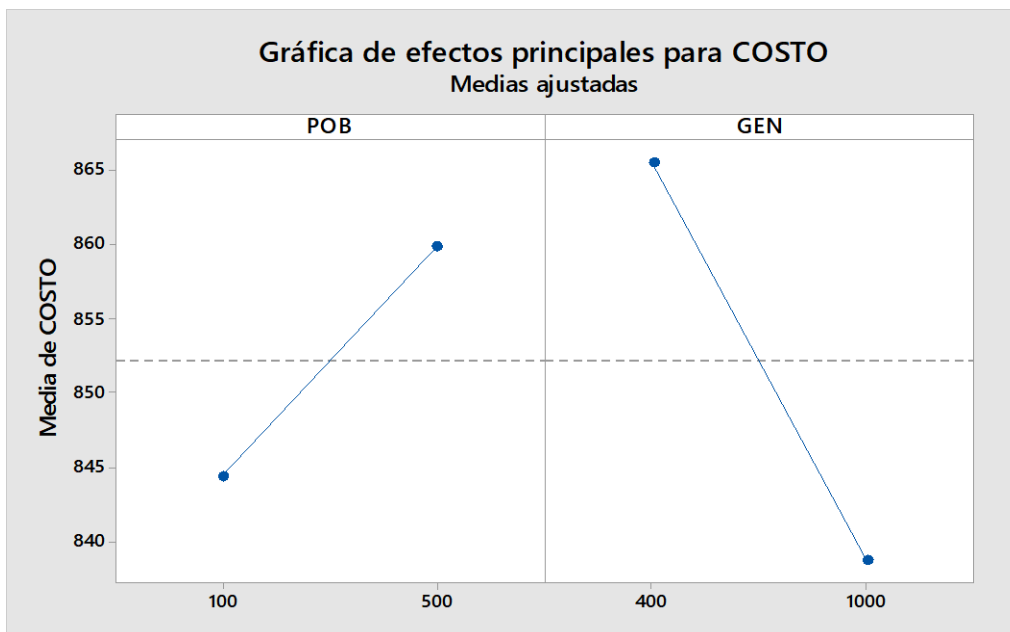


Figura 18. Gráfica de Efectos principales para la instancia Gaskell67-22 x 5.

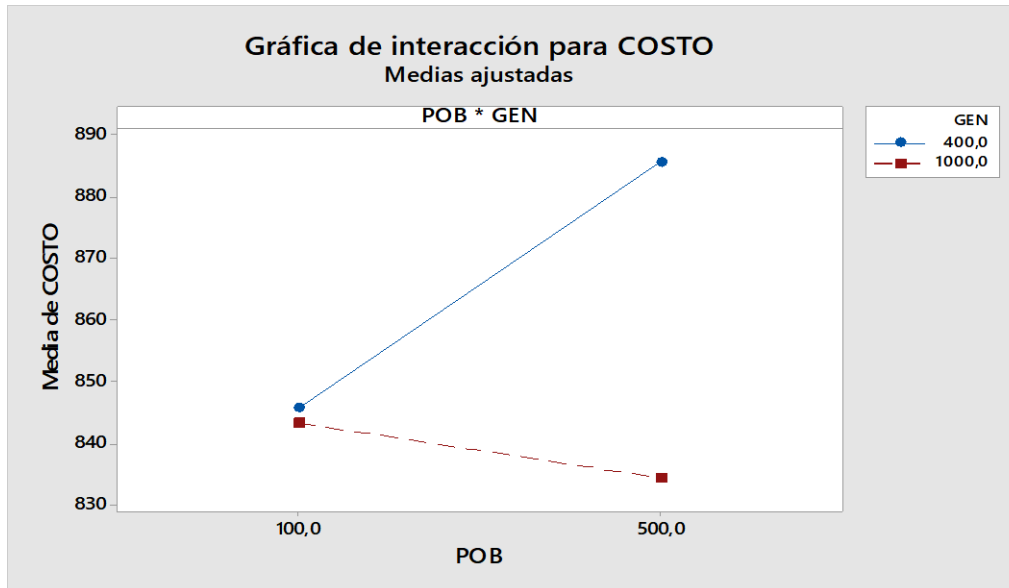


Figura 19. Gráfica de interacción para Costo de la instancia Gaskell67-22 x 5.

Se presenta el diagrama de Pareto de efectos estandarizados de la instancia Gaskell6722x5 en la figura 17. Este diagrama de Pareto se utiliza para determinar la magnitud y la importancia de los efectos. Las barras de los factores que cruzan la línea de referencia y por eso son factores estadísticamente significativos, son los siguientes: número de generaciones, la interacción entre tamaño de población y número de generaciones (POB * GEN) y, finalmente, tamaño de población. En la gráfica de efectos principales presentada en la figura 18 se puede evidenciar que los niveles de los factores que minimizan el costo (función objetivo del problema) son: tamaño de población 100 y número de generaciones 1000

En la gráfica de interacción para el costo presentada en la figura 19 se puede evidenciar que para POB * GEN con 100 tamaño de población y 400 generaciones contra 500 tamaño de población y 1000 generaciones tienen efectos muy similares.

Teniendo en cuenta el efecto de los factores y la interacción entre los factores POB*GEN, la mejor combinación es: 500 tamaño de población y 1000 generaciones. Aunque también resultan buenas soluciones con la combinación: 100 tamaño de población y 1000 generaciones.

9.3. Diseño experimental para la instancia Gaskell67-32 x 5

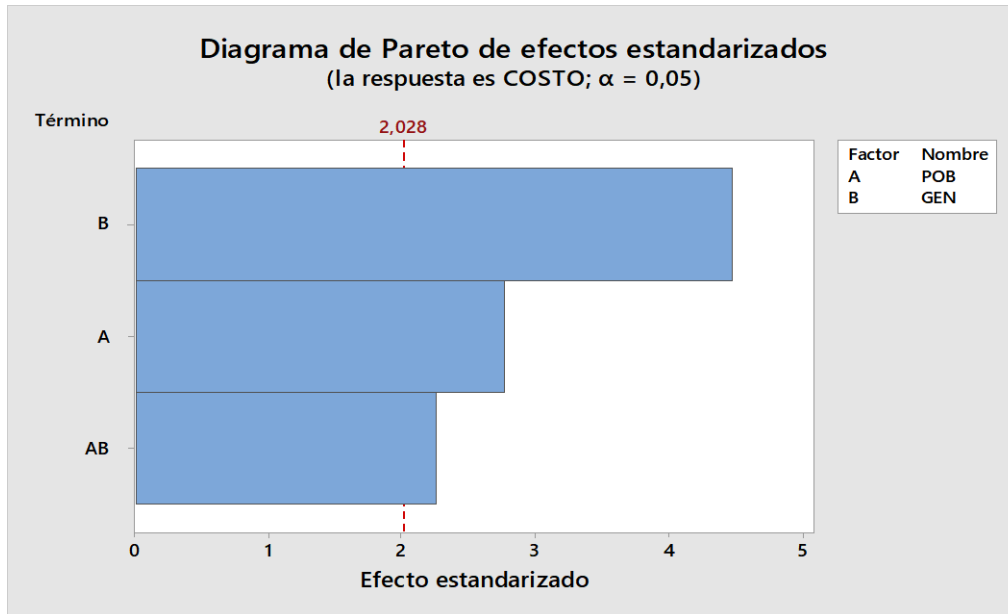


Figura 20. Diagrama de Pareto de efectos estandarizados de la Gaskell67-32x5.

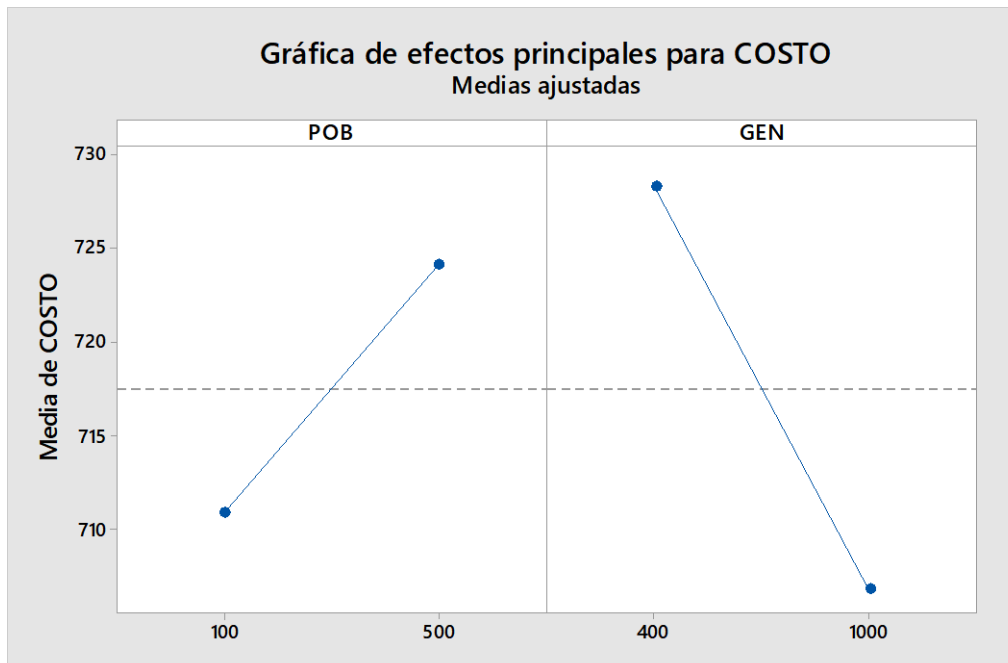


Figura 21. Gráfica de Efectos principales para la instancia Gaskell67-32 x 5.

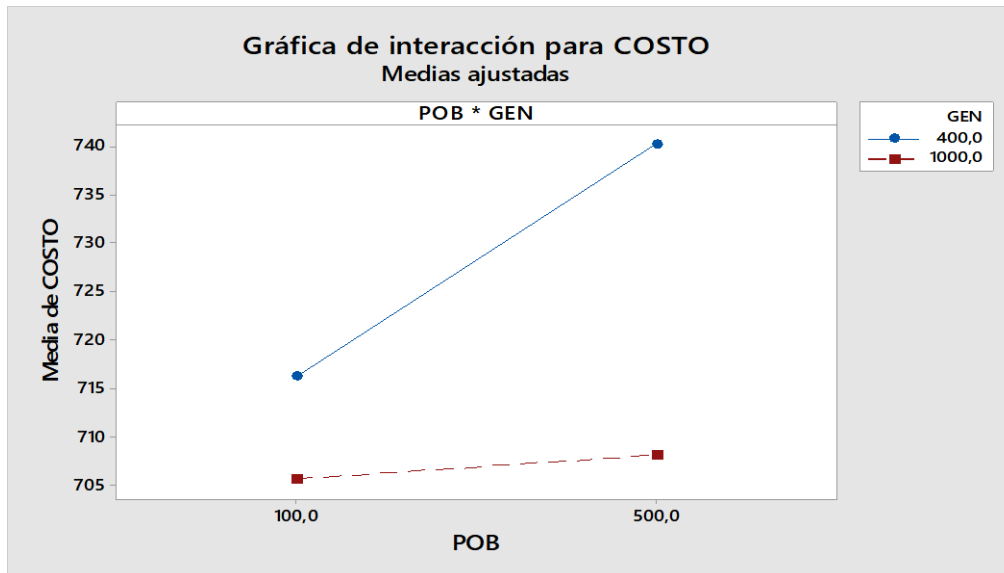


Figura 22. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.

Se presenta el diagrama de Pareto de efectos estandarizados de la instancia Gaskell6722x5 en la figura 20. Este diagrama de Pareto se utiliza para determinar la magnitud y la importancia de los efectos. Las barras de los factores que cruzan la línea de referencia y por eso son factores estadísticamente significativos, son los siguientes: número de generaciones, tamaño de la población y, por último, la interacción entre tamaño de población y número de generaciones (POB * GEN).

En la gráfica de efectos principales presentada en la figura 21. se puede evidenciar que los niveles de los factores que minimizan el costo (función objetivo del problema) son: tamaño de población 100 y número de generaciones 1000.

En la gráfica de interacción para el costo presentada en la figura 22 se puede evidenciar que para POB * GEN con 100 tamaño de población y 1000 generaciones contra 500 tamaño de población y 1000 generaciones tienen efectos similares.

Teniendo en cuenta el efecto de los factores y la interacción entre los factores POB*GEN, la mejor combinación es: 100 tamaño de población y 1000 generaciones.

9.4. Diseño experimental para la instancia Christofides69-50x5

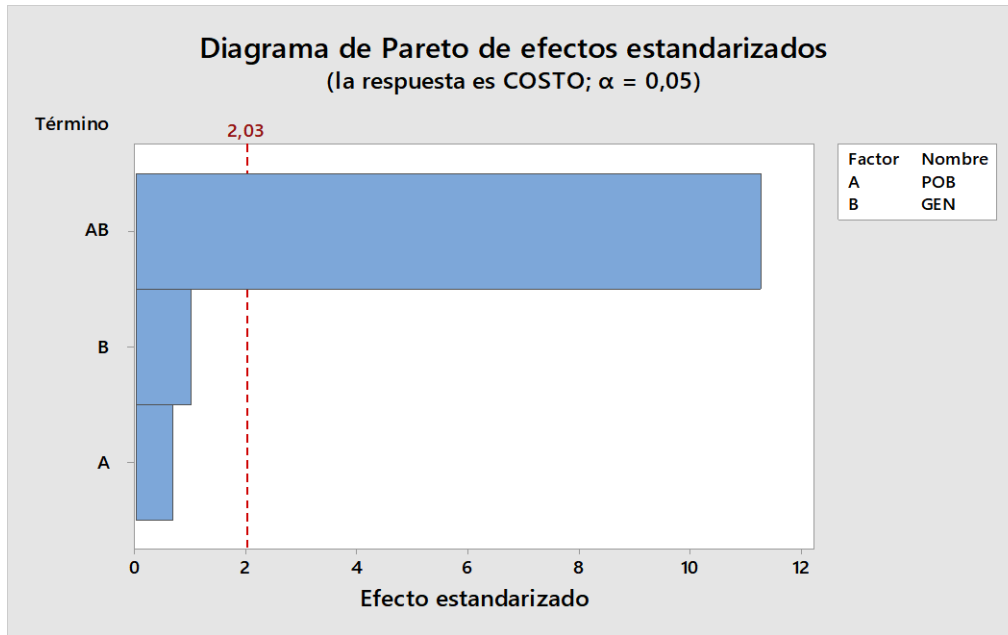


Figura 23. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5.

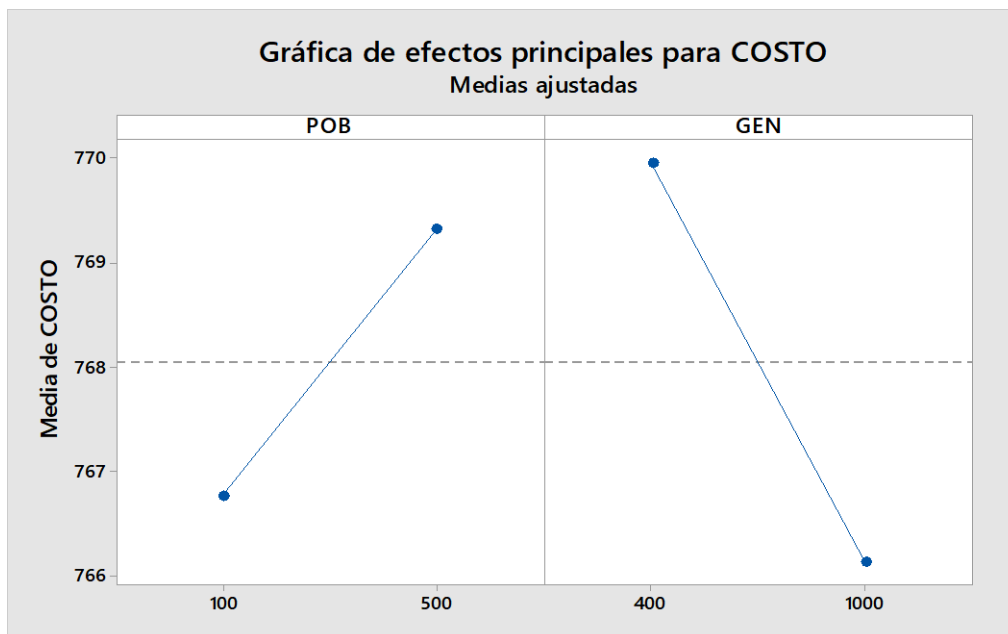


Figura 24. Gráfica de Efectos principales para la instancia Christofides69-50x5.

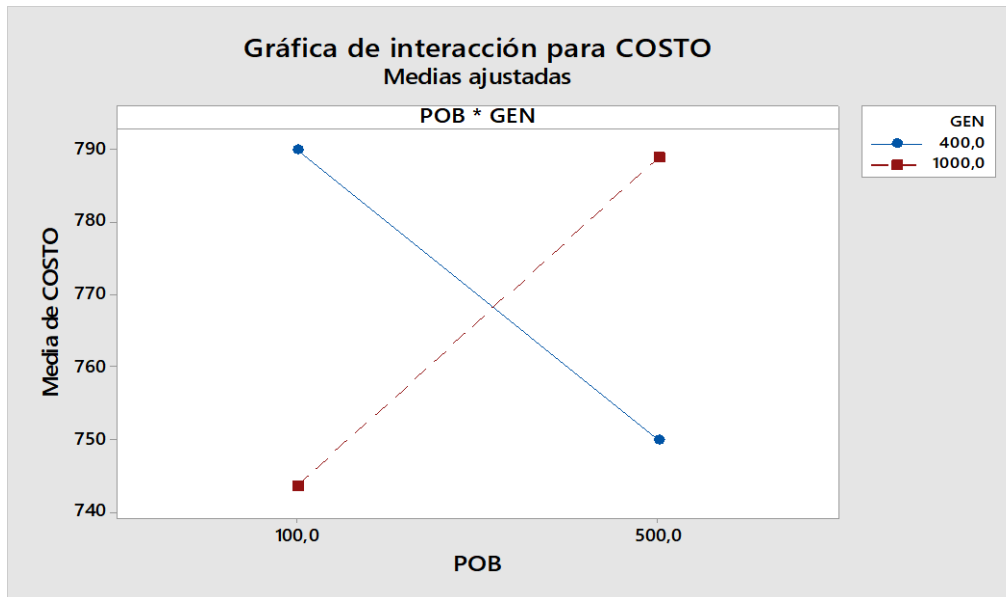


Figura 25. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.

Se presenta el diagrama de Pareto de efectos estandarizados de la instancia Christofides69-50x5 en la figura 23. Este diagrama de Pareto se utiliza para determinar la magnitud y la importancia de los efectos. La barra del factor que cruza la línea de referencia y por eso es un factor estadísticamente significativo, es el siguiente: la interacción entre tamaño de población y número de generaciones (POB * GEN).

En la gráfica de efectos principales presentada en la figura 24. se puede evidenciar que los niveles de los factores que minimizan el costo (función objetivo del problema) son: tamaño de población 100 y número de generaciones 1000.

En la gráfica de interacción para el costo presentada en la figura 25 se puede evidenciar que para POB * GEN con 100 tamaño de población y 400 generaciones contra 100 tamaño de población y 1000 generaciones tienen efectos muy diferentes.

Teniendo en cuenta el efecto de los factores y la interacción entre los factores POB*GEN, la mejor combinación es: 100 tamaño de población y 1000 generaciones. Aunque también resultan buenas soluciones con la combinación: 500 tamaño de población y 400 generaciones

9.5. Diseño experimental para la instancia Christofides69-75x10

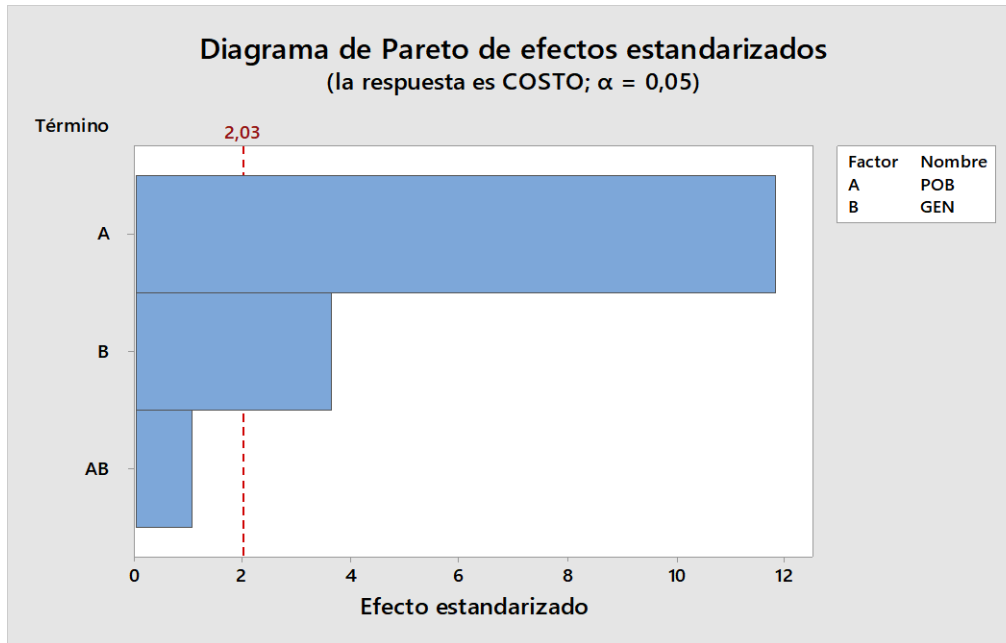


Figura 26. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5.

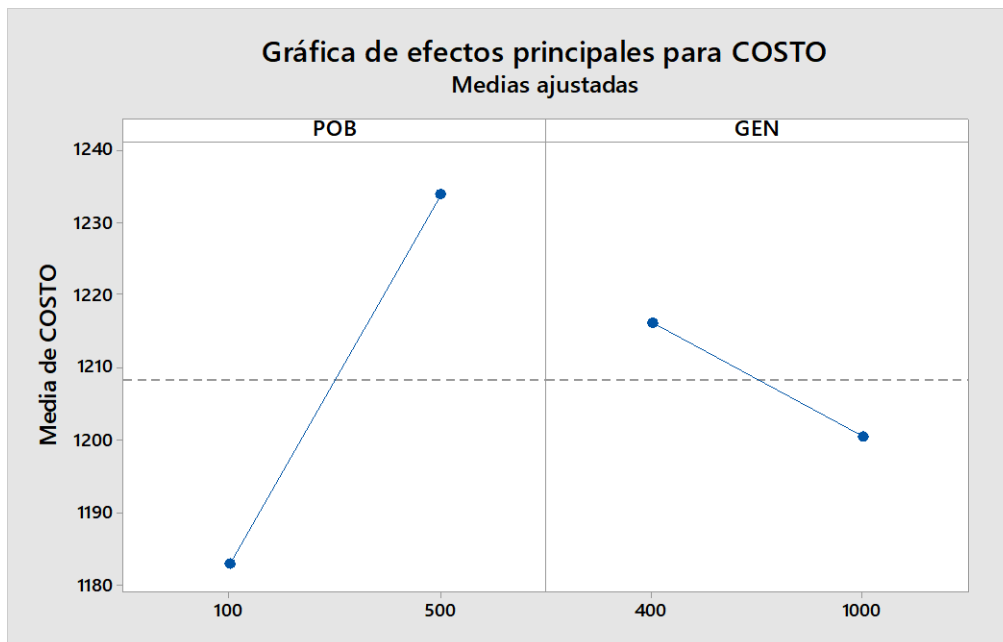


Figura 27. Gráfica de Efectos principales para la instancia Christofides69-50x5.

Se presenta el diagrama de Pareto de efectos estandarizados de la instancia Christofides69-75x10 en la figura 26. Este diagrama de Pareto se utiliza para determinar la magnitud y la importancia

de los efectos. Las barras de los factores que cruzan la línea de referencia y por eso son factores estadísticamente significativos, son los siguientes: tamaño de la población y número de generaciones.

En la gráfica de efectos principales presentada en la figura 27 se puede evidenciar que los niveles de los factores que minimizan el costo (función objetivo del problema) son: tamaño de población 100 y número de generaciones 1000.

Como la interacción entre tamaño de población y número de generaciones (POB * GEN) no es un factor estadísticamente significativo, no se analiza la gráfica de interacción para el costo.

Teniendo en cuenta el efecto de los factores, la mejor combinación es: 100 tamaño de población y 1000 generaciones.

9.6. Diseño experimental para la instancia Christofides69-100x10

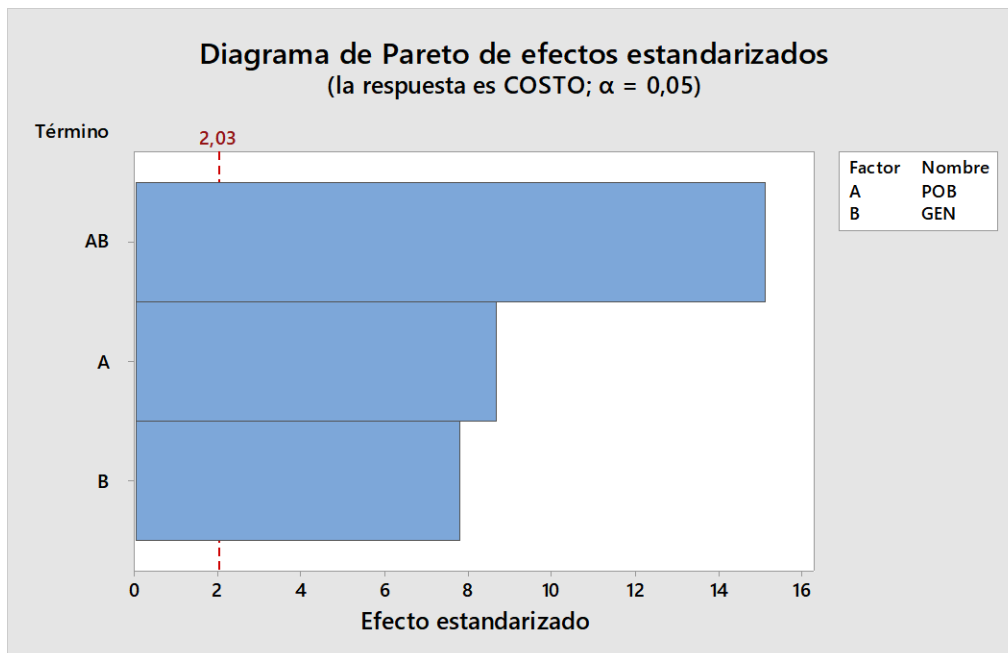


Figura 28. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5.

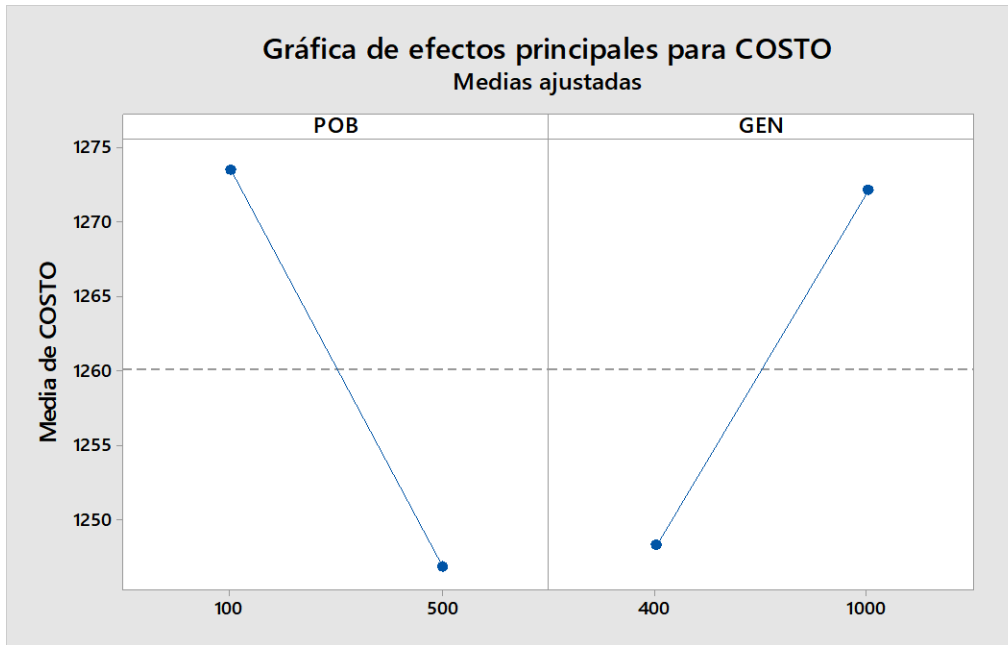


Figura 29. Gráfica de Efectos principales para la instancia Christofides69-50x5.

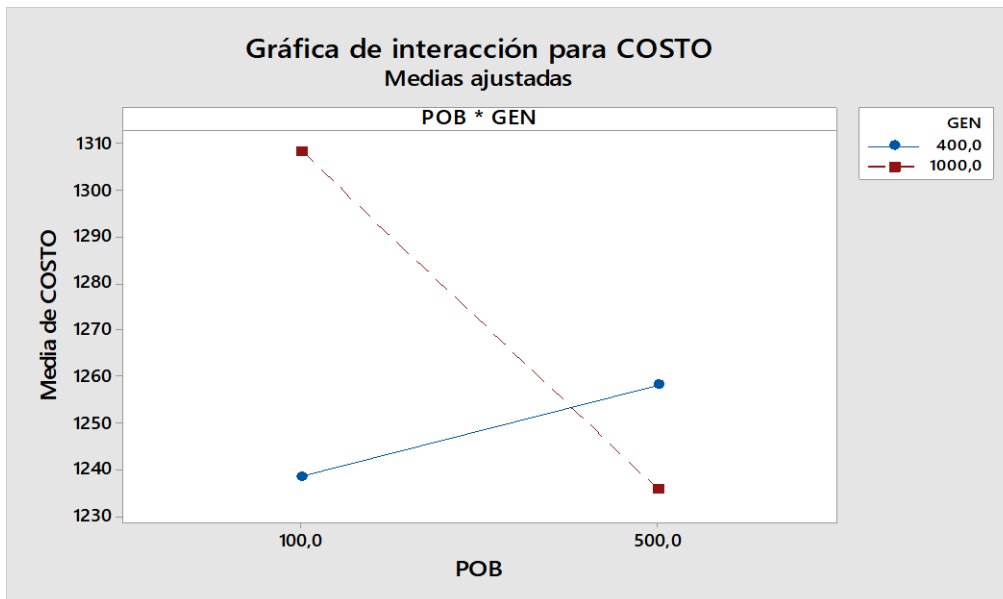


Figura 30. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.

Se presenta el diagrama de Pareto de efectos estandarizados de la instancia Christofides69-100x10 en la figura 28. Este diagrama de Pareto se utiliza para determinar la magnitud y la importancia de los efectos. Las barras de los factores que cruzan la línea de referencia y por eso son factores

estadísticamente significativos, son los siguientes: la interacción entre tamaño de población y número de generaciones (POB * GEN), tamaño de la población y, por último, número de generaciones.

En la gráfica de efectos principales presentada en la figura 29 se puede evidenciar que los niveles de los factores que minimizan el costo (función objetivo del problema) son: tamaño de población 500 y número de generaciones 400.

En la gráfica de interacción para el costo presentada en la figura 30 se puede evidenciar que para POB * GEN con 500 tamaño de población y 1000 generaciones contra 100 tamaño de población y 400 generaciones tienen efectos similares.

Teniendo en cuenta el efecto de los factores y la interacción entre los factores POB*GEN, la mejor combinación es: 500 tamaño de población y 1000 generaciones. Aunque también resultan buenas soluciones con la combinación: 100 tamaño de población y 400 generaciones.

9.7. Diseño experimental para la instancia Daskin95-150x10

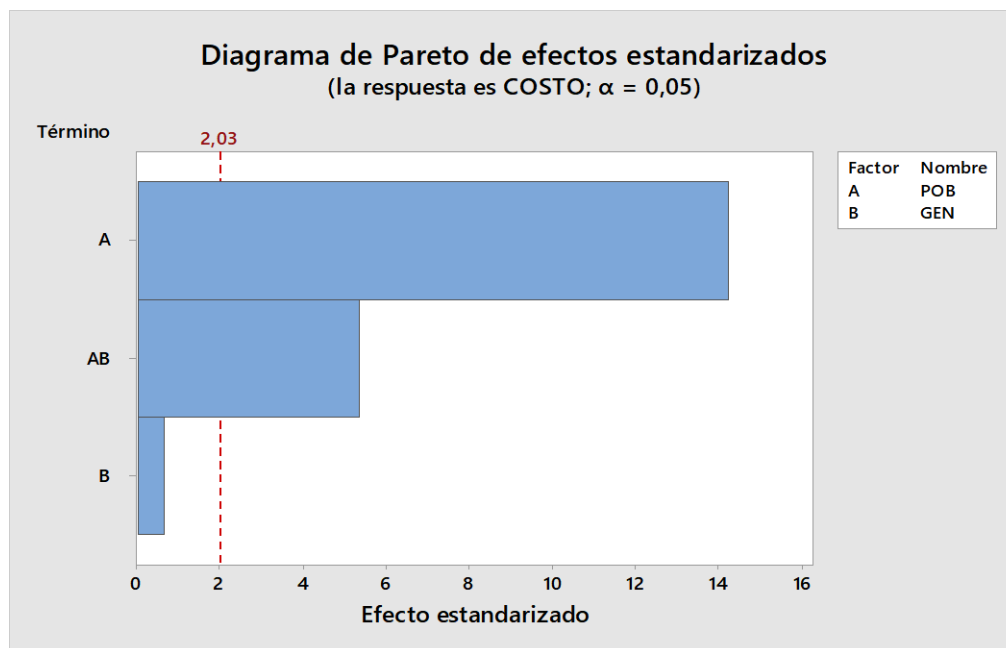


Figura 31. Diagrama de Pareto de efectos estandarizados de la Christofides69-50x5

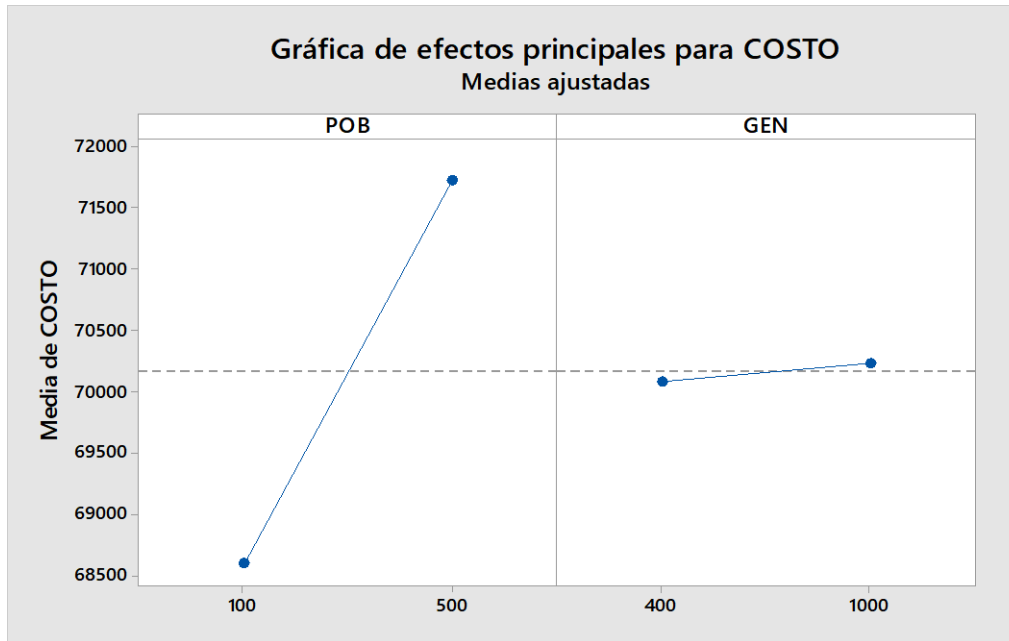


Figura 32. Gráfica de Efectos principales para la instancia Christofides69-50x5.

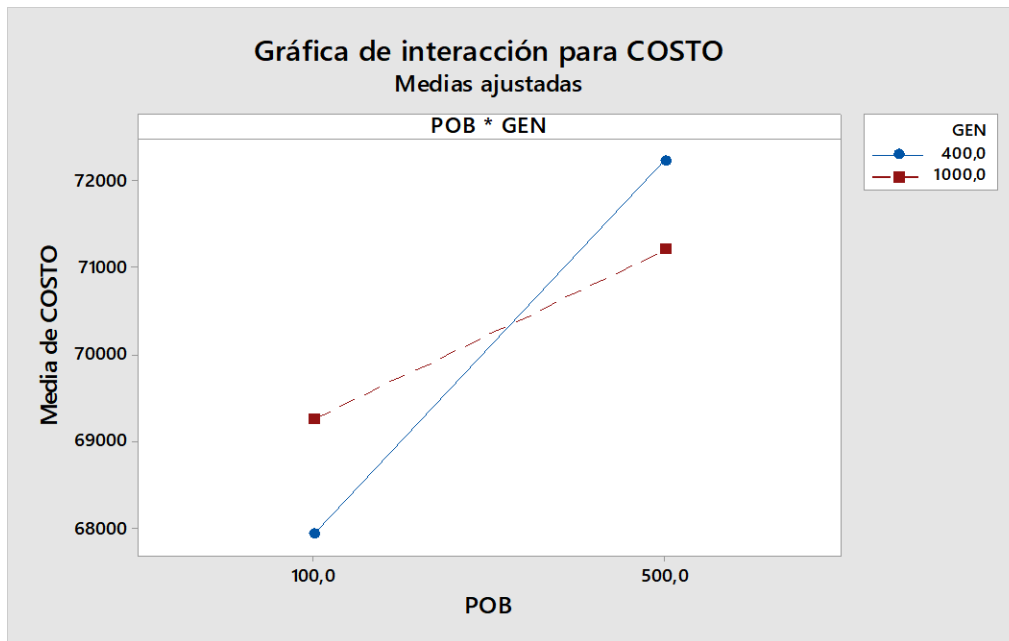


Figura 33. Gráfica de interacción para Costo de la instancia Gaskell67-32 x 5.

Se presenta el diagrama de Pareto de efectos estandarizados de la instancia Daskin95-150x10 en la figura 31. Este diagrama de Pareto se utiliza para determinar la magnitud y la importancia de

los efectos. Las barras de los factores que cruzan la línea de referencia y por eso son factores estadísticamente significativos, son los siguientes: tamaño de la población y la interacción entre tamaño de población y número de generaciones (POB * GEN).

En la gráfica de efectos principales presentada en la figura 32 se puede evidenciar que los niveles de los factores que minimizan el costo (función objetivo del problema) son: tamaño de población 100 y número de generaciones 400.

En la gráfica de interacción para el costo presentada en la figura 33 se puede evidenciar que para POB * GEN con 100 tamaño de población y 400 generaciones contra 100 tamaño de población y 1000 generaciones tienen efectos similares.

Teniendo en cuenta el efecto de los factores y la interacción entre los factores POB*GEN, la mejor combinación es: 100 tamaño de población y 400 generaciones. Aunque también resultan buenas soluciones con la combinación: 100 tamaño de población y 1000 generaciones.

9.8. Resultados de los diseños experimentales

De acuerdo con los resultados arrojados por los experimentos numéricos realizados, en la tabla 5 se presentan el ajuste de los parámetros que se utilizan en el algoritmo genético para cada una de las instancias.

Generalmente en el algoritmo genético la probabilidad de mutación es muy baja (cerca del 10%). Sin embargo, como se puede observar en la Tabla 5, para el algoritmo genético propuesto se utiliza un parámetro de probabilidad de mutación alto ($P_m=70\%$), lo anterior se hace para generar mayor diversidad en la población y garantizar que ningún punto en el espacio de búsqueda tenga una probabilidad nula de ser examinado. Para comprobar si el efecto de este parámetro es significativo sobre el resultado, se decidió comparar el *gap* cuando el algoritmo tiene $P_m = 0.7$ con el algoritmo utilizando $P_m = 0.1$.

Tabla 5.

Parámetros utilizados en el algoritmo genético para cada clase de instancia.

Instancia	n x m	Parámetros			
		POB	GEN	Pm	Pc
Gaskell67-21 x 5	21 x 5	100	1000	0.7	0.9
Gaskell67-22 x 5	22 x 5	100	1000	0.7	0.9
Gaskell67-32x 5	32 x 5	100	1000	0.7	0.9
Christofides69-50x5	50 x 5	100	1000	0.7	0.9
Christofides69-75x10	75 x 10	100	1000	0.7	0.9
Christofides69-100x10	100 x 10	100	400	0.7	0.9
Daskin95-150x10	150 x 10	100	400	0.7	0.9

Nota: POB: Tamaño de población, GEN: Número de generaciones, Pm: probabilidad de mutación, Pc: Probabilidad de cruce.

Tabla 6.

Comparación de costo utilizando Pm=0.7 vs Pm=0.1 para las instancias de la literatura

Nombre de la instancia	M.S.	T.C.	M.S.	T.C.	Gap (%)
	(Pm = 0.7)	(Pm = 0.7)	(Pm = 0.1)	(Pm = 0.1)	
Gaskell67-21 x 5	432.9	18.78	451.6	19.00	4.32
Gaskell67-22 x 5	836.8	19.21	874.3	19.19	4.48
Gaskell67-32x 5	697.0	21.58	721.4	22.26	3.51
Christofides69-50x5	728.5	27.00	793.8	30.16	8.97
Christofides69-75x10	1157.6	38.00	1169.9	42.70	1.07
Christofides69-100x10	1210.7	19.05	1254.7	19.06	3.63
Daskin95-150x10	67326.4	27.86	70944.8	29.82	5.37

Nota: M.S. (Pm=0.7) Mejor solución de 10 iteraciones del algoritmo con 70% de probabilidad de mutación, T.C (Pm=0.7) Tiempo computacional de 10 iteraciones del algoritmo con 70% de probabilidad de mutación, M.S. (Pm=0.1) Mejor solución de 10 iteraciones del algoritmo con 10% de probabilidad de mutación, T.C (Pm=0.1) Tiempo computacional de 10 iteraciones del algoritmo con 10% de probabilidad de mutación.

De acuerdo con los resultados presentados en la tabla 6, se puede evidenciar que el algoritmo cuando tiene $P_m=0.7$ arroja mejores soluciones que cuando tiene $P_m=0.1$, para todas las instancias de la literatura analizadas, siendo 8.97% el porcentaje de diferencia más alto y 1.07% el más bajo. Sin embargo, en los dos escenarios la diferencia en tiempo computacional no es significativa.

Tabla 7.

Brecha con la mejor respuesta de la literatura y tiempo computacional en segundos empleado en conseguir la mejor respuesta con el algoritmo propuesto.

Nombre de la instancia	BKS	GA	TC (s)	Gap (%)
Gaskell67-21 x 5	424.9	432.9	18.78	1.89
Gaskell67-22 x 5	585.1	785.0	19.21	34.17
Gaskell67-32x 5	571.7	697.0	21.58	21.91
Christofides69-50x5	565.6	728.5	27.00	28.80
Christofides69-75x10	861.6	1157.6	38.00	34.35
Christofides69-100x10	842.9	1210.7	19.05	43.63
Daskin95-150x10	44011.7	67326.4	27.86	52.97

Nota: TC= Tiempo computacional en segundo, BKS= mejor solución conocida de la literatura

En la tabla 7 se presenta una comparación de las mejores soluciones obtenidas utilizando el método propuesto, contra la mejor solución encontrada en la literatura de las instancias de Barreto (2004). Estas instancias se resolvieron utilizando dos métodos de inicialización, uno totalmente aleatorio y otro que incorpora los métodos heurísticos propuestos: algoritmo de barrido y FCM; los resultados obtenidos a través de este análisis se evidencian en la tabla 8, en donde se puede observar que el método de inicialización propuesto es mucho más efectivo que un método de inicialización totalmente aleatorio, para todas las instancias de la literatura resueltas.

En las instancias de Barreto (2004) resueltas en la presente investigación, el algoritmo genético propuesto logra un buen resultado en el costo (función objetivo) para el problema Gaskell67-21 x 5, pues este está muy cercano a la mejor solución conocida de la literatura con una brecha (*gap*) de 1.89%.

Para los problemas Gaskell67-22x5, Gaskell67-32x5 y Christofides69-50x5 los resultados arrojados por el algoritmo genético tienen valores aceptables con respecto a las mejores soluciones conocidas de la literatura con un *gap* menor al 35%.

Para los demás problemas de la literatura, el algoritmo genético arroja resultados de baja calidad, en comparación con las mejores soluciones conocidas de la literatura, con brechas mayores al 35%.

Tabla 8.

Comparación de resultados con población inicial usando heurísticas vs. población inicial aleatoria

Nombre de la instancia	Población inicial con heurísticas		Población Inicial aleatoria		
	Min	Max	Min	Max	Gap (%)
Gaskell67-21 x 5	432.9	444.5	735.3	843.6	69.9
Gaskell67-22 x 5	785.0	851.5	1191.0	1397.3	51.7
Gaskell67-32x 5	697.0	709.3	1235.6	1350.4	77.3
Christofides69-50x5	728.5	759.1	1631.4	1837.2	123.9
Christofides69-75x10	1157.6	1186.2	2832.2	2996.0	144.7
Christofides69-100x10	1210.7	1252.5	3397.2	3604.6	180.6
Daskin95-150x10	67326.4	68549.2	196692.7	204847.6	192.1

Nota: Gap= brecha en porcentaje del mejor resultado usando una inicialización aleatoria de la población contra el mejor resultado usando inicialización con heurísticas.

9.9. Ejemplos numéricos

En cuanto a la evaluación computacional de problemas de prueba, se presentan ejemplos numéricos de cómo se lleva a cabo el algoritmo genético propuesto para resolver el LRPTW en condiciones de incertidumbre. Para generar los ejemplos numéricos, se toman todos los parámetros del problema “20-5-1a”, instancia de Prins, Prodhon & Calvo (2006). Además, se añaden tiempos de viaje difusos y ventanas de tiempo de acuerdo con la modificación hecha por Zarandi et al. (2011) a la instancia. Los tiempos de viaje se presentan en el apéndice F y las ventanas de tiempo se muestran la tabla 9.

Con respecto a las demandas de los clientes que se muestran en la tabla 10, se generaron variables aleatorias triangulares en el rango $[0-M]$, siendo M el máximo valor de las demandas determinísticas ya establecidas en la instancia. En cuanto al parámetro CrD^* , su valor es igual a 1.

Teniendo en cuenta los costos de operación del problema, los coeficientes de penalización α_1, α_2 y α_3 en la función de adaptación son iguales a 1; además, estos factores se aumentan en diez unidades por cada iteración del algoritmo genético. Lo anterior es necesario para que, en las primeras generaciones, las soluciones que infrinjan las restricciones tengan oportunidad de generar descendientes. Finalmente, se realizan 1000 iteraciones en el algoritmo de simulación propuesto.

9.9.1. Problema de prueba 1

Tabla 9.

Ventanas de tiempo en los nodos (Problema de prueba 1).

<i>i</i>	$[a_i \ b_i]$	<i>i</i>	$[a_i \ b_i]$
D1	[8:30, 13:00]	8	[8:30, 11:30]
D2	[8:30, 13:00]	9	[8:30, 11:30]
D3	[8:30, 13:00]	10	[8:30, 11:30]
D4	[8:30, 13:00]	11	[8:30, 11:30]
D5	[8:30, 13:00]	12	[8:30, 11:30]
		13	[8:30, 11:30]
1	[8:30, 11:30]	14	[8:30, 11:30]
2	[8:30, 11:30]	15	[8:30, 11:30]
3	[8:30, 11:30]	16	[8:30, 11:30]
4	[8:30, 11:30]	17	[8:30, 11:30]
5	[8:30, 11:30]	18	[8:30, 11:30]
6	[8:30, 11:30]	19	[8:30, 11:30]
7	[8:30, 11:30]	20	[8:30, 11:30]

Nota: Adaptado de Zarandi, M. H. F., Hemmati, A., & Davari, S. (2011). The multi-depot capacitated location-routing problem with fuzzy travel times. *Expert Systems with Applications*, 38(8), 10075–10084. <https://doi.org/10.1016/j.eswa.2011.02.006>

Tabla 10.

Demandas de los nodos de clientes.

<i>i</i>	d_1	d_2	d_3
1	2	4	19
2	10	19	20
3	10	13	20
4	3	8	19
5	3	10	15
6	3	4	7
7	8	15	16
8	4	4	17
9	5	9	18
10	10	12	13
11	6	11	15
12	7	7	12
13	7	12	12
14	5	17	17
15	7	10	12
16	1	7	15
17	2	10	19
18	10	12	19
19	4	6	17
20	4	10	17

Tabla 11.

Resultados con diferentes valores de CrV^* (Problema de prueba 1).

CrV^*	TC (s)	M.S.	Distancia planeada	Distancia adicional	Distancia total	Credibilidad (TW)
0	21.90	29866.56	708.56	0	708.56	0.98
0.1	22.30	32893.78	735.78	0	735.78	0.98
0.2	23.00	32820.50	662.50	0	662.50	0.98
0.3	22.59	31888.50	730.50	0	730.50	0.98
0.4	22.21	33030.70	601.70	0	601.70	1.00
0.5	22.54	31866.52	708.52	0	708.52	1.00
0.6	23.21	29786.35	628.35	0	628.35	0.91
0.7	22.54	32882.75	724.75	0	724.75	0.99
0.8	22.49	33015.55	857.55	0	857.55	1.00
0.9	22.49	34857.15	699.15	0	699.15	0.97
1	22.56	32874.64	716.64	0	716.64	0.96

Nota: T.C= Tiempo computacional de 30 iteraciones del algoritmo, M.S.=Mejor solución de 30 iteraciones del algoritmo. Credibilidad (TW)= Credibilidad de la solución respecto a las ventanas de tiempo.

Sobre los resultados del problema de prueba, se hace un análisis usando diferentes valores para del índice de preferencia del despachador CrV^* , cambiando su valor en el intervalo [0, 1] por pasos de longitud 0.1. Para cada valor de CrV^* , el algoritmo se ejecutó 30 veces y las mejores soluciones de estas ejecuciones se muestran en la tabla 11. Los resultados computacionales muestran claramente que la distancia total de recorrido no tiene una tendencia clara con el cambio del parámetro CrV^* , tal como se evidencia en la figura 34. Hay que mencionar, además, que en ninguno de los escenarios existe una distancia adicional por recorrer ocasionada por los fallos de ruta. Lo anterior se debe a que las estrechas ventanas de tiempo hacen que las rutas sean muy cortas

y, por tanto, no se permite una visualización del efecto de cambio en el parámetro CrV^* sobre la distancia de recorrido.

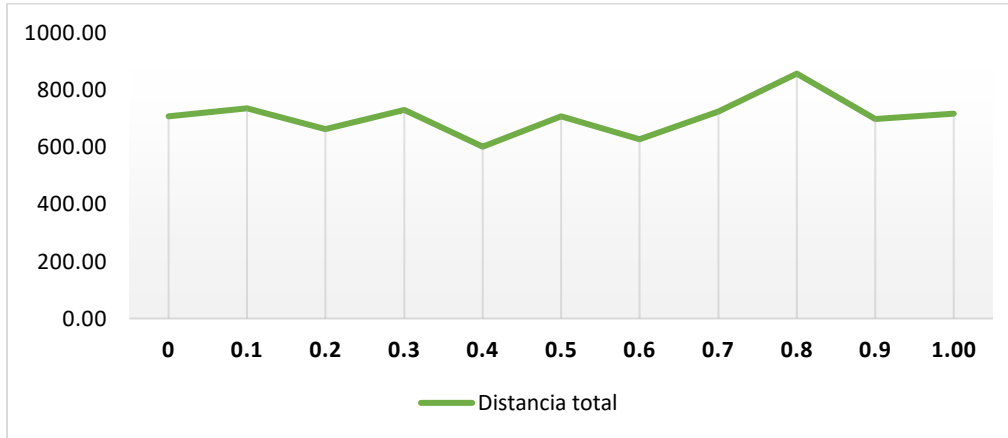


Figura 34. Distancia total en el problema con diferentes valores del CrV^* (Problema de prueba 1)

Por otra parte, se realiza un análisis de sensibilidad de las soluciones frente al valor del nivel de confianza α . En la figura 35 se muestran los resultados de este análisis, donde se compara el valor de la función objetivo (costo) frente al cambio del parámetro α , en el intervalo [0.9-0.99], por pasos de longitud 0.1. De acuerdo con los resultados del análisis, se evidencia que entre mayor sea el nivel de confianza, hay una mayor inversión en la operación.

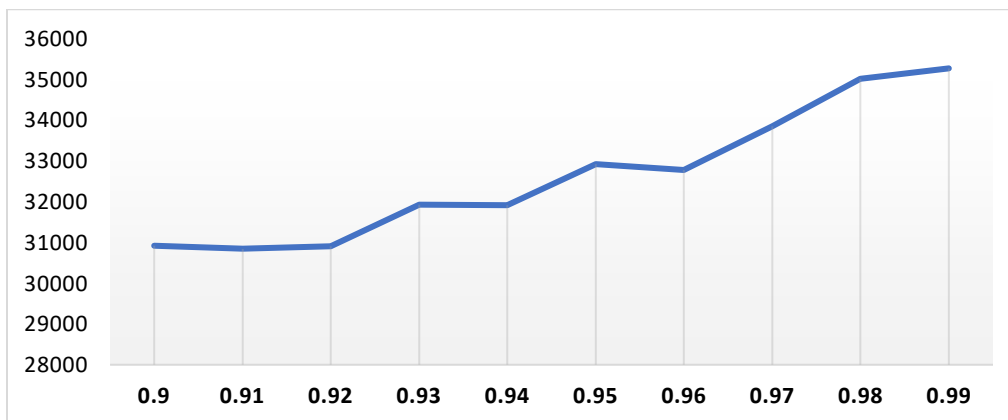


Figura 35. Sensibilidad de la solución con respecto al valor del nivel de confianza α (Problema de prueba 1)

9.9.2. Problema de prueba 2

Tabla 12.

Las ventanas de tiempo en los nodos (Problema de prueba 2).

<i>i</i>	$[a_i \ b_i]$	<i>i</i>	$[a_i \ b_i]$
D1	[8:30, 18:00]	8	[8:30, 16:30]
D2	[8:30, 18:00]	9	[8:30, 16:30]
D3	[8:30, 18:00]	10	[8:30, 16:30]
D4	[8:30, 18:00]	11	[8:30, 16:30]
D5	[8:30, 18:00]	12	[8:30, 16:30]
		13	[8:30, 16:30]
1	[8:30, 16:30]	14	[8:30, 16:30]
2	[8:30, 16:30]	15	[8:30, 16:30]
3	[8:30, 16:30]	16	[8:30, 16:30]
4	[8:30, 16:30]	17	[8:30, 16:30]
5	[8:30, 16:30]	18	[8:30, 16:30]
6	[8:30, 16:30]	19	[8:30, 16:30]
7	[8:30, 16:30]	20	[8:30, 16:30]

En el anterior problema de prueba, las ventanas de tiempo interfirieron en el análisis de las distancias adicionales por fallos de ruta; por consiguiente, se propone un segundo problema de

prueba, en donde se expanden las ventanas de tiempo a 5 horas más, tal como se muestra en la tabla 12.

Tabla 13.

Resultados con diferentes valores de CrV^ (Problema de prueba 2).*

CrV^*	TC (s)	M.S.	Distancia planeada	Distancia adicional	Distancia total	Credibilidad (TW)
0	21.36	24522.08	364.08	172.27	536.35	0.97
0.1	22.20	24629.08	471.08	69.12	540.20	1.00
0.2	22.15	25667.91	509.91	68.38	578.29	0.99
0.3	22.07	24696.07	538.07	60.15	598.23	0.96
0.4	22.15	25686.16	528.16	26.66	554.81	0.98
0.5	21.92	25679.87	521.87	12.71	534.59	0.96
0.6	21.85	26698.13	540.13	0	540.13	0.97
0.7	22.21	26689.88	531.88	0	531.88	0.99
0.8	22.91	26754.20	596.20	0	596.20	1.00
0.9	22.33	26717.21	559.21	0	559.21	0.94
1	21.38	28747.59	589.59	0	589.59	1.00

Nota: T.C= Tiempo computacional de 30 iteraciones del algoritmo, M.S.=Mejor solución de 30 iteraciones del algoritmo. Credibilidad (TW)= Credibilidad de la solución respecto a las ventanas de tiempo.

En este segundo problema, para cada valor de CrV^* , el algoritmo se ejecutó 30 veces y las mejores soluciones de estas ejecuciones se muestran en la tabla 13. Así mismo, la figura 36 representa la tendencia de cambio con respecto a la distancia planificada, la distancia adicional y la distancia total; para diferentes valores del índice de preferencia del despachador CrV^* . Los resultados muestran que, cuando $CrV^* = 0.7$, la distancia total de recorrido es la menor en todos los escenarios. En definitiva, se puede concluir que a medida que disminuye el valor de CrV^* , la

distancia planeada presenta una tendencia descendente, mientras la distancia adicional presenta una tendencia ascendente; sin embargo, la distancia total no muestra una tendencia clara.

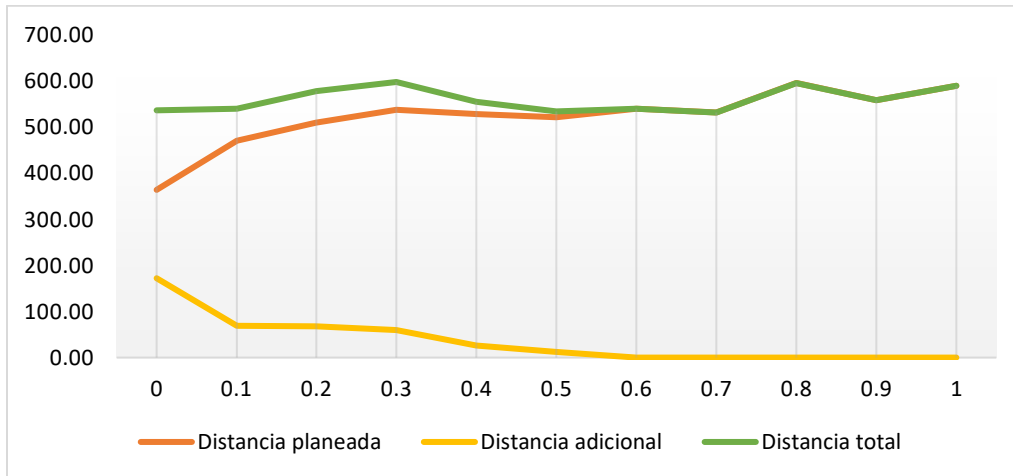


Figura 36. Distancias en el problema con diferentes valores del CrV^* (Problema de prueba 2)

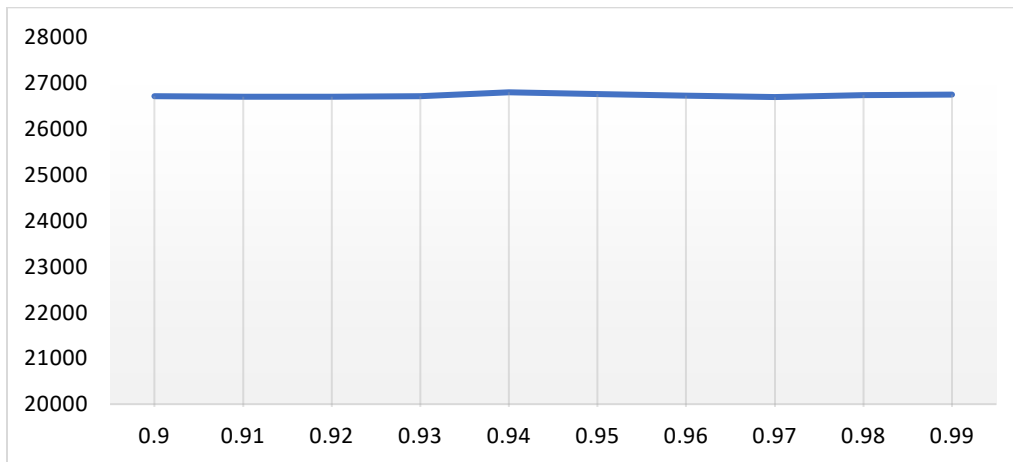


Figura 37. Sensibilidad de la solución con respecto al valor del nivel de confianza α (Problema de prueba 2).

Respecto al segundo problema, también se realiza un análisis de sensibilidad de las soluciones frente a diferentes valores del nivel de confianza predeterminado α . De acuerdo con los resultados del análisis, presentados en la figura 37, se evidencia que no hay una tendencia clara en el costo total frente al cambio del nivel de confianza. Lo anterior se debe a que las ventanas de tiempo son

extensas y su efecto sobre el costo de la operación es opacado por las demás restricciones del problema. Para concluir, las pruebas realizadas anteriormente pueden ser consideradas no solo como un análisis de sensibilidad sino también como una validación del método de solución propuesto.

10. Conclusiones

Según la revisión de literatura, son pocos los trabajos de investigación que han abordado la versión difusa del problema de localización-ruteo con ventanas de tiempo (LRPTW), por lo cual este proyecto puede ser considerado como una referencia para futuras investigaciones sobre el problema.

De acuerdo con los resultados de los diseños experimentales realizados para la validación del algoritmo propuesto, al utilizar una combinación entre la heurística fuzzy *c*-means (FCM) y el método de barrido para generar parte de la población inicial, el algoritmo tiene un mejor desempeño en todas las instancias del análisis.

El algoritmo genético propuesto presenta buenas soluciones para las pequeñas y medianas instancias de Barreto (2004); con un *gap* de 1.89% por encima de la mejor solución conocida en la literatura para la instancia Gaskell67-21 x 5 y un diferencial entre 20% y 35% con la mejor solución conocida en la literatura para las instancias: Gaskell67-22x 5, Gaskell67-32x 5, Christofides69-50x5 y Christofides69-75x10. Sin embargo, para las instancias de mayor tamaño, el algoritmo presenta resultados alejados a la mejor solución conocida en la literatura, con un *gap* de 43.63% y 52.97% para las instancias Christofides69-100x10 y Daskin95-150x10 respectivamente.

Los ejemplos numéricos presentados en este trabajo permiten afirmar que el método de solución propuesto es eficiente para resolver el LRPTW con demandas difusas y tiempos de viaje difusos en poco tiempo computacional.

11. Recomendaciones

Para futuras investigaciones, hay muchas oportunidades de extender este trabajo en algunos aspectos. Primero, considerar variables difusas a otros parámetros, por ejemplo, las ventanas de tiempo. Segundo, abordar otras variantes del LRP, tales como el LRP con entregas y recogidas simultáneas. Finalmente, otra variación interesante del problema sería utilizar vehículos con capacidad heterogénea.

Existe la posibilidad de utilizar otras heurísticas o metaheurísticas para resolver el problema, tales como: optimización de enjambre de partículas (PSO), búsqueda tabú y algoritmos basados en colonias de hormigas (AC). Por otra parte, se podrían emplear algoritmos híbridos, y así aprovechar los beneficios de la combinación de diferentes métodos de solución. Por último, otra opción viable, es utilizar un algoritmo genético auto-adaptivo con el fin de mejorar la diversidad de la población e impedir la convergencia en óptimos locales.

Fomentar en los estudiantes de ingeniería industrial el interés por el estudio de la lógica difusa como herramienta importante para resolver problemas de investigación de operaciones, y así incentivar el desarrollo de este tipo de proyectos.

Referencias Bibliográfica

- Aguilar, K. (2017). *Un modelo multiobjetivo de localización-ruteo para la planeación logística en la fase de preparación a sismos en Bucaramanga*. (Tesis de maestría). Universidad Industrial de Santander, Bucaramanga. Recuperado de <http://tangara.uis.edu.co/biblioweb/tesis/2017/165948.pdf>
- Arenas Díaz, G., & Ramírez Lamus, É. R. (2013). Medidas difusas e integrales difusas. *Universitas Scientiarum*, 18(1). Recuperado de <http://www.redalyc.org/html/499/49925681005/>
- Arranz de la Peña, J. y Parra Truyol, A. (2005). *Algoritmos genéticos*. Recuperado de <http://www.it.uc3m.es/jvillena/irc/practicas/06-07/05.pdf>
- Ayyub, B. M., & Klir, G. J. (2006). Uncertainty modeling and analysis in engineering and the sciences. *Chapman and Hall/CRC*. Recuperado de https://www.researchgate.net/profile/Bilal_Ayyub/publication/265068692_Uncertainty_Modeling_and_Analysis_in_Engineering_and_the_Sciences/links/544643210cf22b3c14de17b6/Uncertainty-Modeling-and-Analysis-in-Engineering-and-the-Sciences.pdf
- Bae, H., & Moon, I. (2016). Multi-depot vehicle routing problem with time windows considering delivery and installation vehicles. *Applied Mathematical Modelling*, 40, 6536–6549. <https://doi.org/10.1016/j.apm.2016.01.059>
- Bahri, O., Amor, N. Ben, & Talbi, E.-G. (2016). Robust Routes for the Fuzzy Multi-objective Vehicle Routing Problem. *IFAC-PapersOnLine*, 49(12), 769–774. <https://doi.org/10.1016/j.ifacol.2016.07.867>
- Baldacci, R., Hadjiconstantinou, E., & Mingozzi, A. (2004). An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. *Operations research*, 52(5), 723-738.
- Ballou, R. H. (2004). *Logística: Administración de la cadena de suministro*. Pearson Educación. Recuperado de https://books.google.com.co/books?hl=es&lr=&id=ii5xqLQ5VLgC&oi=fnd&pg=PA1&dq=cadena+se+suministro&ots=u3-yojLp8l&sig=XG8AAX8cP7A-CmJ60ejRskMX0Hs&redir_esc=y#v=onepage&q&f=false

- Baños Navarro, L. (2006). *Metaheurísticas Híbridas para Optimización Mono-objetivo y Multi-objetivo. Paralelización y Aplicaciones* (Tesis doctoral). Universidad de Almería, Almería, España.
- Barkaoui, M., & Gendreau, M. (2013). An adaptive evolutionary approach for real-time vehicle routing and dispatching. *Computers and Operations Research*, 40(7), 1766–1776. <https://doi.org/10.1016/j.cor.2013.01.022>
- Barreto, S. D. S. (2004). *Análise e Modelização de Problemas de localização-distribuição*. (tesis doctoral), University of Aveiro, Portugal, 2004. Recuperado de <https://ria.ua.pt/bitstream/10773/1850/1/2009001061.pdf>
- Barreto, S., Ferreira, C., Paixao, J., & Santos, B. S. (2007). Using clustering analysis in a capacitated location-routing problem. *European Journal of Operational Research*, 179(3), 968-977.
- Barreto, M., y Niño, P. (2016). *Un algoritmo memético para el problema de localización-ruteo con ventanas de tiempo para la atención de desastres sísmicos en la ciudad de Bucaramanga* (Tesis de pregrado). Universidad Industrial de Santander, Bucaramanga. Recuperado de <http://tangara.uis.edu.co/biblioweb/tesis/2016/165444.pdf>
- Boventer, E. (1961). The relationship between transportation costs and location rent in transportation problems. *Journal of Regional Science*, 3(2), 27–40. <https://doi.org/10.1111/j.1467-9787.1961.tb01276.x>
- Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys (CSUR)*, 35(3), 268-308. Recuperado de <https://goo.gl/6EGQuU>
- Brito Santana, J. A. (2013). *Optimización de rutas de distribución con información y restricciones difusas* (Tesis doctoral). Universidad de La Laguna, Servicio de Publicaciones. Recuperado de <https://riull.ull.es/xmlui/bitstream/handle/915/3370/Julio%20Antonio%20Brito%20Santana.pdf?sequence=1>
- Chan, Y. (2005). *Location, transport and land-use: modelling spatial-temporal information*. Little Rock, Estados Unidos: Springer Science & Business Media. Recuperado de <https://goo.gl/Ue8mj6>

- Chopra, S., & Meindl, P. (2007). Supply chain management. Strategy, planning & operation. *Das summa summarum des management*. Gabler, 265-275. Recuperado de <http://lean.mty.itesm.mx/resumenes/SupplyChainManagementStrategyPlanningandOperationsSecondEdition.pdf>
- Cox, E. (2005). *Fuzzy modeling and genetic algorithms for data mining and exploration*. Recuperado de <https://goo.gl/UK4NMZ>
- Cunquero, R. M. (2003). Algoritmos heurísticos en optimización combinatoria. *Universidad de Valencia, Facultad de Ciencias Matemáticas*. Recuperado de <http://yalma.fime.uanl.mx/~roger/work/teaching/mecbs5122/1-Introduction/Intro-by-Rafa%20Marti.pdf>
- Darwin, C. (1859). *On the origins of species by means of natural selection*. Londres: Murray.
- De Vito, E. L. (2006). Introducción al razonamiento aproximado: lógica difusa. *RAMR*, 3, 126-36. Recuperado de www.redalyc.org/html/3821/382138367007/
- Derbel, H., Jarboui, B., Hanafi, S., & Chabchoub, H. (2012). Genetic algorithm with iterated local search for solving a location-routing problem. *Expert Systems with Applications*, 39(3), 2865-2871.
- Dubois, D., & Prade, H. (2012). Possibility theory: an approach to computerized processing of uncertainty. *Springer Science & Business Media*. Recuperado de <https://goo.gl/ZLzZ6B>
- Escobar, J. W., Linfati, R., & Adarme Jaimes, W. (2015). Problema de Localización y Ruteo con Restricciones de Capacidad: Revisión de la Literatura. *Facultad de Ingeniería*, 24(39), 85-98. Recuperado de <http://www.scielo.org.co/pdf/rfing/v24n39/v24n39a08.pdf>
- Gagliardi, E. O., Hernández, G., Leguizamón, G., & Taranilla, M. T. (2007). Algoritmo genético para la Descomposición de Minkowski de poligonos convexos. Recuperado de <https://www.infor.uva.es/egc07/articulos/13.pdf>
- Galindo Gómez, J. (2008). Conjuntos y Sistemas Difusos (Lógica Difusa y Aplicaciones). *Departamento de Lenguajes y Ciencias de la Computación Universidad de Málaga*. Recuperado de <http://www.lcc.uma.es/~ppgg/FSS/FSSindex.pdf>
- Gallego, L. A., y Escobar, A. (2006). Planeamiento de la Expansión de sistemas de transmisión de energía eléctrica considerando contingencias. *Scientia et Technica*, 12(30). Recuperado de <http://www.redalyc.org/html/849/84920491047/>

- Gandy, C. D. T., & Dohrn, P. J. (1973). Depot location with van salesmen - a practical approach. *Omega*, 1(3), 321-329. Recuperado de <https://www.sciencedirect.com/science/article/pii/0305048373901084>
- Gagliardi, E. O., Hernández, G., Leguizamón, G., y Taranilla, M. T. (2007). Algoritmo genético para la Descomposición de Minkowski de poligonos convexos. Recuperado de <https://www.infor.uva.es/egc07/articulos/13.pdf>
- García, M. S. (2000). Optimización combinatoria. *Números*, (43), 115-120. Recuperado de <http://www.sinewton.org/numeros/numeros/43-44/Articulo22.pdf>
- García-Aracil, A., & Palomares-Montero, D. (2012). Agrupación alternativa para la evaluación de las universidades públicas españolas. *Estudios Económicos Regionales y Sectoriales*, 12(2),177-192. Recuperado de <http://roderic.uv.es/bitstream/handle/10550/31198/084957.pdf?sequence=1>
- Gelves-Tello, N. A., Mora-Moreno, R. A., & Lamos-Díaz, H. (2016). Solución del problema de ruteo de vehículos con demandas estocásticas mediante la optimización por espiral. *Facultad de Ingeniería*, 25(42), 7-19. Recuperado de <http://www.scielo.org.co/pdf/rfing/v25n42/v25n42a02.pdf>
- Ghaffari-Nasab, N., Ahari, S. G., & Ghazanfari, M. (2013). A hybrid simulated annealing based heuristic for solving the location-routing problem with fuzzy demands. *Scientia Iranica*, 20(3), 919-930.
- Ghannadpour, S. F., Noori, S., Tavakkoli-Moghaddam, R., & Ghoseiri, K. (2014). A multi-objective dynamic vehicle routing problem with fuzzy time windows: Model, solution and application. *Applied Soft Computing*, 14, 504-527. <https://doi.org/10.1016/j.asoc.2013.08.015>
- Golozari, F., Jafari, A., & Amiri, M. (2013). Application of a hybrid simulated annealing-mutation operator to solve fuzzy capacitated location-routing problem. *The International Journal of Advanced Manufacturing Technology*, 67(5-8), 1791-1807.
- González, J., Sánchez, A., Velásquez, J., Galve, J. (1993)- Algorítmica. Diseño y Análisis de algoritmos Funcionales e Imperativos. Addison Wesley iberoamericana.
- Hassanzadeh, A., Mohseninezhad, L., Tirdad, A., Dadgostari, F., & Zolfagharinia, H. (2009). Location-routing problem. In *Facility location: concepts, models, algorithms and case studies*. <https://doi.org/10.1007/978-3-7908-2151-2>

- Hernández, L. R. B., & Valdés, O. R. (2016). Lógica Difusa Basada en la Experiencia del Usuario para Medir la Usabilidad. *Revista Latinoamericana de Ingeniería de Software*, 4(1), 48-54.
- Jaramillo Garzón, J. A. *Metodología de optimización de los parámetros de control de un algoritmo genético= Optimization Methodology for Control Parameters of Genetic Algorithms* (Tesis de maestría). Universidad Nacional de Colombia-Sede Manizales.
- Kassir, E. E. (2015). Sistemas De Control Difuso. *Eslava Zuluaga*, AF. Recuperado de http://www.unicauca.edu.co/matematicas/eventos/log&co/MEMORIAS/SxCtrl_Difuso.pdf.
- Klir, G., & Yuan, B. (1995). *Fuzzy sets and fuzzy logic (Vol. 4)*. New Jersey: Prentice hall.
- Klibi, W., Martel, A., & Guitouni, A. (2010). The design of robust value-creating supply chain networks: a critical review. *European Journal of Operational Research*, 203(2), 283-293.
- Kuri, Á., & Galaviz, J. (2002). Algoritmos genéticos (No. Sirsi) Recuperado de <http://cursos.itam.mx/akuri/PUBLICA.CNS/2000/AGS.PDF>.
- Laporte, G., Nobert, Y., & Arpin, D. (1986). An exact algorithm for solving a capacitated location-routing problem. *Annals of Operations Research*, 6(9), 291-310. <https://doi.org/10.1007/BF02023807>
- Leclerc, F., Potvin, J. Y., & Potvin, J. Y. (1997). Genetic algorithms for vehicle dispatching. *International Transactions in Operational Research*, 4(5-6), 391-400. [https://doi.org/10.1016/S0969-6016\(97\)00028-2](https://doi.org/10.1016/S0969-6016(97)00028-2)
- Liu, B. (2009). *Theory and practice of uncertain programming (Vol. 239)*. Berlin: Springer. Recuperado de <https://bibliotecavirtual.uis.edu.co:2142/content/pdf/10.1007%2F978-3-7908-1781-2.pdf>
- Ren, M., Liu, P., Wang, Z., & Yi, J. (2016). A self-adaptive fuzzy c-means algorithm for determining the optimal number of clusters. *Computational intelligence and neuroscience*, 2016.
- Li-ying, W., & Yuan-bin, S. (2015). Multiple Charging Station Location-Routing Problem with Time Window of Electric Vehicle. *Journal of Engineering Science & Technology Review*, 8(5).
- Lüer, A., Benavente, M., Bustos, J., & Venegas, B. (2009, December). El Problema de Rutas de Vehículos: Extensiones y Métodos de Resolución, estado del Arte. *In EIG*. Recuperado de <https://goo.gl/t2WdzD>

- Medina, L. B. R., La Rota, E. C. G., & Castro, J. A. O. (2011). Una revisión al estado del arte del problema de ruteo de vehículos: Evolución histórica y métodos de solución. *Ingeniería*, 16(2), 35-55.
- Minh, T. T., Van Hoai, T., & Nguyet, T. T. N. (2013, June). A memetic algorithm for waste collection vehicle routing problem with time windows and conflicts. In *International Conference on Computational Science and Its Applications* (pp. 485-499). Springer, Berlin, Heidelberg. Recuperado de <https://goo.gl/mFTMzu>
- Moreno, J., Rivera, J. C., & Ceballos, Y. F. (2011). Agrupamiento homogéneo de elementos con múltiples atributos mediante algoritmos genéticos. *Dyna*, 78(165), 246-254. Recuperado de <http://www.scielo.org.co/pdf/dyna/v78n165/a26v78n165.pdf>
- Moujahid, A., Inza, I., y Larrañaga, P. (2008). Tema 2. Algoritmos Genéticos. *Departamento de Ciencias de la Computación e Inteligencia Artificial Universidad del País Vasco*. Recuperado de www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t2geneticos.pdf
- Murata, T., & Ishibuchi, H. (1994, June). Performance evaluation of genetic algorithms for flowshop scheduling problems. In *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference*, p.p. 812-817.
- Nadizadeh, A., & Hosseini Nasab, H. (2014). Solving the dynamic capacitated location-routing problem with fuzzy demands by hybrid heuristic algorithm. *European Journal of Operational Research*, 238(2), 458–470. <https://doi.org/10.1016/j.ejor.2014.04.012>
- Nier, A., y Niño, J. (2015). *Desarrollo de un modelo híbrido para resolver el problema de localización-ruteo (LRP)* (Tesis de pregrado). Universidad Industrial de Santander, Bucaramanga. Recuperado de <http://tangara.uis.edu.co/biblioweb/tesis/2015/157662.pdf>
- Olivera, A. (2004). Heurísticas para problemas de ruteo de vehículos. *Reportes Técnicos* 04-08. Recuperado de <https://goo.gl/pFP249>
- Or, I., & Pierskalla, W. P. (1979). A transportation location-allocation model for regional blood banking. *AIIE transactions*, 11(2), 86-95. Recuperado de www.anderson.ucla.edu/faculty/william.pierskalla/Health/54_Box_14_3.pdf
- Oviedo, D. y Valdiviezo, K. (2016). *Un algoritmo híbrido para el problema de ruteo de vehículos con tiempos de viaje estocásticos y ventanas de tiempo suave*. (Tesis de pregrado). Universidad Industrial de Santander, Bucaramanga.

- Pishvaei, M. S., Rabbani, M., & Torabi, S. A. (2011). A robust optimization approach to closed-loop supply chain network design under uncertainty. *Applied Mathematical Modelling*, 35(2), 637-649. Recuperado de <https://goo.gl/UhfCor>
- Pappis, C. P., & Mamdani, E. H. (1977). A fuzzy logic controller for a traffic junction. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(10), 707-717.
- Pose, M. G. (2000). Introducción a los algoritmos genéticos. Departamento de Tecnologías de la Información y las Comunicaciones Universidad de Coruña. Recuperado de https://www.researchgate.net/profile/Marcos_Gestal/publication/237812449_Introduccion_a_los_Algoritmos_Geneticos/links/02e7e520a93bdade8f000000.pdf
- Prodhon, C. (2014). Instancias de Prodhon para el LRP. Francia. Recuperado de <http://prodhonc.free.fr/>
- Prodhon, C., & Prins, C. (2014). A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1), 1-17. <https://doi.org/10.1016/j.ejor.2014.01.005>
- Parisi, A., Parisi, F., y Cornejo, E. (2004). Algoritmos genéticos y modelos multivariados recursivos en la predicción de índices bursátiles de América del Norte: IPC, TSE, Nasdaq y DJI. *El Trimestre Económico*, 789-809. Recuperado de <http://www.redalyc.org/pdf/313/31328402.pdf>
- Pérez Lugo, L. J. (2015). *Método para la agregación de rankings a partir de dos grupos con intereses contrapuestos* (Tesis doctoral) Universidad Central "Marta Abreu" de Las Villas, Santa Clara, Cuba. Recuperado de http://dspace.uclv.edu.cu/bitstream/handle/123456789/7315/LazaroJ_Perez.pdf?sequence=1&isAllowed=y
- Ponboon, S., Qureshi, A. G., & Taniguchi, E. (2016). Branch-and-price algorithm for the location-routing problem with time windows. *Transportation Research Part E: Logistics and Transportation Review*, 86, 1-19.
- Porrás, J. C. C., Laverde, R. M., & Díaz, J. R. (2008). Técnicas de lógica difusa aplicadas a la minería de datos. *Scientia et Technica*, 3(40), 1-6.
- Prins, C., Prodhon, C., Ruiz, A., Soriano, P., & Calvo, R. W. (2005). A cooperative lagrangean relaxation-granular tabu search heuristic for the capacitated location routing problem. *MIC'05*.

- Prins, C., Prodhon, C., & Calvo, R. W. (2006a). Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking. *4OR*, 4(3), 221-238.
- Prins, C., Prodhon, C., & Calvo, R. W. (2006b). A memetic algorithm with population management (MA| PM) for the capacitated location-routing problem. *In European Conference on Evolutionary Computation in Combinatorial Optimization*. Springer, Berlin, Heidelberg. 183-194.
- Pullen, H. G. M., & Webb, M. H. J. (1967). A computer application to a transport scheduling problem. *The Computer Journal*, 10(1), 10-13.
- Rabbani, M., Heidari, R., Farrokhi-Asl, H., y Rahimi, N. (2018). Usar algoritmos metaheurísticos para resolver un problema de enrutamiento de ubicación de residuos peligrosos industriales multiobjetivo considerando tipos de desechos incompatibles. *Journal of Cleaner Production*, 170, 227-241. Recuperado de <https://reader.elsevier.com/reader/sd/265D9E623631CB2E4964A65D9E1FC987839CD6842E5CFC2B81B7F99200A25AA27C35EFD91CB63B2B36B5292FAFAB2CD0>
- Reina, D., & Moscovitz, L. J. (2008). *Fundamentos de matemática difusa* (Tesis doctoral). Fundación Universitaria Konrad Lorenz. Facultad de Matemáticas. Búsqueda realizada 18-19-10. Recuperado de http://www.konradlorenz.edu.co/images/stories/suma_digital_matematicas/EDICION_09_0I/trabajo_de_grado_daniel_reina.pdf.
- Ren, M., Liu, P., Wang, Z., & Yi, J. (2016). A self-adaptive fuzzy c-means algorithm for determining the optimal number of clusters. *Computational intelligence and neuroscience*, 2016.
- Rocha Medina, L. B., González La Rota, E. C., & Orjuela Castro, J. A. (2011). Una revisión al estado del arte del problema de ruteo de vehículos: Evolución histórica y métodos de solución. *Ingeniería*, 16(2). Recuperado de <http://www.redalyc.org/html/4988/498850173004/>
- Scaparra, M. P., & Scutellà, M. G. (2001). *Facilities, Locations, Customers: Building Blocks of Location Models*. A Survey. Technical Report del Dipartimento di Informatica . Università di Pisa, Pisa, IT. Recuperado de <http://eprints.adm.unipi.it/2061/>

- Schweickardt, G., (2009). Metaheurística FPSO Multiobjetivo. Una aplicación para la planificación de la expansión de mediano/largo plazo de un sistema de distribución eléctrica. *Energética*-, 42, 73-88. Recuperado de <http://www.bdigital.unal.edu.co/26526/1/24129-84462-1-PB.pdf>
- Soto, A. J., Ponzoni, I. y Vázquez, G. E. (2006). Análisis numérico de diferentes criterios de similitud en algoritmos de clustering. *Mecánica Computacional*. XXV: 993-1011. Recuperado de <https://goo.gl/REH1xv>
- Taş, D., Jabali, O., & Van Woensel, T. (2014). A vehicle routing problem with flexible time windows. *Computers & Operations Research*, 52, 39-54.
- Teodorović, D. (1994). Fuzzy sets theory applications in traffic and transportation. *European Journal of Operational Research*, 74(3), 379-390.
- Teodorovic, D., & Kikuchi, S. (1991). Application of fuzzy sets theory to the saving based vehicle routing algorithm. *Civil Engineering Systems*, 8(2), 87-93.
- Ting, C. J., & Chen, C. H. (2013). A multiple ant colony optimization algorithm for the capacitated location routing problem. *International Journal of Production Economics*, 141(1), 34-44.
- Tuzun, D., & Burke, L. I. (1999). A two-phase tabu search approach to the location routing problem. *European journal of operational research*, 116(1), 87-99.
- Valenzuela Díaz, L. V. (2016). *Optimización simultánea del problema de localización y ruteo con flota heterogénea* (Tesis de maestría) Escuela Colombiana De Ingeniería Julio Garavito, Bogotá, Colombia.
- Vega, F. D. (2003). *Algoritmos Evolutivos para la Resolución de Sistemas de "Word Equations"*. Universidad de Oviedo, España. Recuperado de <http://www.aic.uniovi.es/Tc/papers/Trabajo.pdf>
- Valencia, E. (1997, August). *Optimización mediante algoritmos genéticos*. In Anales del Instituto de Ingenieros de Chile 109 (2), 83-92. Recuperado de https://www.researchgate.net/profile/Pablo_Estevez/publication/228708779_Optimizaci_n_Mediante_Algoritmos_Geneticos/links/0912f5111f82b2a61000000.pdf
- Xu, J., Yan, F., & Li, S. (2011). Vehicle routing optimization with soft time windows in a fuzzy random environment. *Transportation Research Part E: Logistics and Transportation Review*, 47(6), 1075–1091. <https://doi.org/10.1016/j.tre.2011.04.002>

- Yücenur, G. N., & Demirel, N. C. (2011). A new geometric shape-based genetic clustering algorithm for the multi-depot vehicle routing problem. *Expert Systems with Applications*, 38(9), 11859–11865. <https://doi.org/10.1016/j.eswa.2011.03.077>
- Zarandi, M. H. F., Hemmati, A., & Davari, S. (2011). The multi-depot capacitated location-routing problem with fuzzy travel times. *Expert Systems with Applications*, 38(8), 10075–10084. <https://doi.org/10.1016/j.eswa.2011.02.006>
- Zarandi, M. H. F., Hemmati, A., Davari, S., & Turksen, I. B. (2013). Capacitated location routing problem with time windows under uncertainty. *SciVerse ScienceDirect*. Recuperado de [file:///C:/Users/EEIE/Downloads/articulo ruteo ELSEVIER.pdf](file:///C:/Users/EEIE/Downloads/articulo%20ruteo%20ELSEVIER.pdf)
- Zheng, Y., & Liu, B. (2006). Fuzzy vehicle routing model with credibility measure and its hybrid intelligent algorithm. *Applied Mathematics and Computation*, 176(2), 673–683. <https://doi.org/10.1016/j.amc.2005.10.013>