

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

**Un algoritmo ALNS para el problema de ruteo de vehículos con drones en la
distribución de última milla**

Jhon Alexander Jiménez Romero, Jeamy Sebastián Tarazona Jiménez

Trabajo de grado para optar el título de Ingeniero Industrial

Directora:

Karin Julieth Aguilar Imitola

M.Sc. en Ingeniería Industrial

Codirector:

Henry Lamos Díaz

PhD. en Física - Matemática

Universidad Industrial de Santander

Facultad de Ingenierías Físico-mecánicas

Escuela de Estudios Industriales y Empresariales

Bucaramanga

2020

Dedicatoria

Quiero comenzar agradeciendo a Dios por haberme permitido estudiar esta carrera, además de darme la sabiduría e inteligencia para cada una de las materias y retos que se fueron presentando; fue primeramente gracias a él que logré llegar hasta aquí.

Agradecer a mis padres y hermanas quienes han sido mi constante apoyo, respaldo y quienes me han brindado de alguna u otra manera la fuerza para continuar y no haber abandonado la carrera, me hubiese sido muy difícil seguir sin su apoyo tanto económico como sentimental.

A mi amada Angela Maritza, quien con sus consejos me daba cada vez más y más fuerza para culminar este proyecto y ampliar mis metas, siempre me ha brindado su amor, comprensión y apoyo incondicional; te doy muchas gracias por darme tu punto de vista y ayudarme tanto.

A mi compañero de proyecto Sebastián, con el que conjuntamente nos apoyamos durante el proceso de formación y más aún en la elaboración de este proyecto de investigación.

A todos aquellos que contribuyeron con sus esfuerzos para que este camino llamado vida fuese más cómodo, a quienes guiaron mis pasos con sus consejos y quienes apoyaron mi proceso formativo.

Jhon Alexander Jimenez Romero

Dedicatoria

A Dios todopoderoso, por poner el querer como el hacer

A mis padres, abuelos, hermanos y toda mi familia por ser siempre ese motivo implacable que me impulsa día a día a conseguir todos los retos que me propongo en la vida.

A una de las personas más geniales que he conocido en los últimos tres años por siempre apoyarme, quererme y comprenderme; además de ser más que una guía en el proyecto.

Al bacán, compañero y amigo Jhon Alexander, ya que gracias a su apoyo logramos superar las dificultades durante el proceso de formación y la culminación de esta carrera.

A todos aquellos que contribuyeron con sus esfuerzos para que este camino llamado Ingeniería Industrial fuese más cómodo, fácil y ameno.

Jeamy Sebastian Tarazona Jiménez

Agradecimientos

Al grupo de investigación OPALO, por proporcionarnos las herramientas para el correcto desarrollo de nuestro proyecto.

A nuestra maestra y directora de proyecto, Karin Julieth Aguilar Imitola, por darnos la confianza para desarrollar este proyecto y también por su exigencia para lograr un resultado favorable con este.

Al profesor Henry Lamos por aportarnos sus conocimientos, darnos las observaciones de manera oportuna y guiarnos en este proyecto.

A nuestra escuela de estudios industriales y empresariales, y a todos aquellos profesores que hicieron parte de nuestra formación.

A nuestra alma mater, la UIS, por todas sus enseñanzas y experiencias vividas que nos permitieron crecer personal y profesionalmente.

A todos aquellos que hicieron posible este proyecto.

Tabla de contenido

Introducción	17
1. Generalidades del proyecto	20
1.1 Planteamiento del problema	20
1.2 Objetivos	24
1.2.1 Objetivo general	24
1.2.2 Objetivos específicos	24
1.3 Metodología	24
1.3.1 Etapa 1. Revisión bibliográfica.	25
1.3.2 Etapa 2. Definición del problema.....	26
1.3.3 Etapa 3. Formulación del modelo matemático.....	26
1.3.4 Etapa 4. Diseñar un algoritmo ALNS al problema planteado.....	26
1.3.5 Etapa 5. Validación del algoritmo ALNS.	27
1.3.6 Etapa 6. Documentación y síntesis de resultados.....	27
2. Revisión de literatura	27
2.1 Análisis preliminar de la literatura	27
3. Marco de antecedentes	37
4. Marco teórico	39
4.1 Optimización Combinatoria	39
4.2 Optimización de Redes	40

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

4.3 Complejidad Computacional.....	41
4.4 Problemas NP y NP-Hard	41
4.5 Vehículos aéreos no tripulados (UAV) o drones	42
4.6 Distribución de última milla.....	43
4.7 Problema de Ruteo de Vehículos (VRP).....	43
4.8 Problema de Ruteo de Vehículos con Drones (VRPD)	44
4.9 Heurísticas de solución para los Problemas de Ruteo.....	47
4.10 Metaheurísticas de solución para los Problemas de Ruteo	48
4.10.1 Algoritmos de Hormigas (<i>Ant Colony Optimization</i> , ACO).....	48
4.10.2 Búsqueda Tabú (<i>Tabu Search</i>).....	49
4.10.3 Búsqueda Tabú para el Problema de Ruteo de Vehículos	49
4.10.4 Algoritmo genético (Genetic Algorithm).....	50
4.10.5 Algoritmos Genéticos para el Problema de Ruteo de Vehículos.	50
4.10.6 Adaptive Large Neighborhood Search (ALNS).....	52
4.10.7 Algoritmo de K-medias (<i>K-means clustering</i>).....	54
5. Formulación del modelo matemático.....	55
5.1 Descripción del modelo.....	55
5.2 Supuestos.....	57
6. Diseño del ALNS para la solución del VRPD	69
6.1 Generación de la solución inicial	72

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM	10
6.1.1 Algoritmo de ahorros de Clarke y Wright.....	74
6.1.2 Inserción de drones	77
6.2 Algoritmos de destrucción	79
6.3 Algoritmos de reparación.....	84
6.4 Parámetros del algoritmo	90
7. Validación del algoritmo.....	94
7.1 Instancias.....	94
7.2 Resultados de la validación del algoritmo ALNS propuesto	96
8. Experimentación y Resultados.....	98
8.1 Aceptación de soluciones.....	99
8.2 Soluciones del ALNS, CWA y VRPD.....	102
8.3 Resistencia de vuelo (UAV).....	106
8.4 Mejores soluciones.....	107
8.5 Velocidad de vuelo (UAV)	108
9. Conclusiones	109
10. Recomendaciones.....	112
Referencias Bibliográficas	114

Lista de tablas

Tabla 1 Cumplimiento de los objetivos del proyecto.....	19
Tabla 2 Elementos de la optimización combinatoria.	39
Tabla 3 Metodología del algoritmo de Clarke y Wright	75
Tabla 4 Pseudocódigo del algoritmo de inserción de los drones.	78
Tabla 5 Pseudocódigo del algoritmo Worst Removal.....	81
Tabla 6 Pseudocódigo del algoritmo Cluster Removal.....	83
Tabla 7 Pseudocódigo de los algoritmos de reparación o inserción	85
Tabla 8 Pseudocódigo del algoritmo Repair method Closest Insertion.	90
Tabla 9 Puntuaciones que toma el parámetro Ψ , según el tipo de solución generada.	92
Tabla 10 Valores de los parámetros utilizados en la configuración del problema.....	96
Tabla 11 Valores de los parámetros del algoritmo ALNS	97

Lista de figuras

Figura 1. Fases metodológicas de un estudio en investigación de operaciones.....	25
Figura 2. Un ejemplo del VRPD.....	45
Figura 3. Clasificación de las heurísticas clásicas	47
Figura 4. Ejemplo base del operador de mutación Shift	52
Figura 5. Pseudocódigo del algoritmo K-means	54
Figura 6. Ejemplo de una ruta entre camiones y drones	56
Figura 7. Metodología de algoritmo ALNS	71
Figura 8. Procedimientos del algoritmo ALNS.....	72
Figura 9. Ejemplo del algoritmo de Clarke y Wright.....	75
Figura 10. Casos particulares de clientes..	77
Figura 11. Ejemplo de inserción de drones	78
Figura 12. Ejemplo de la heurística Random Removal.....	80
Figura 13. Ejemplo del algoritmo Worst Removal	82
Figura 14. Ejemplo de la heurística Cluster Removal.....	84
Figura 15. Ejemplo del algoritmo Greedy Insertion with Noise function (GIN).....	87
Figura 16. Ejemplo del algoritmo Regret-n Insertion (nRI)	89
Figura 17. Configuración de las instancias de 20 clientes usadas para la validación	95
Figura 18. Resultados de las mejores soluciones conocidas para el ALNS, CWA y extendido	98
Figura 19. Soluciones aceptadas en la instancia 20.5.1.	101
Figura 20. Soluciones aceptadas para la instancia 100.10.4.	102

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Figura 21. GAP ALNS para 20 clientes.....	104
Figura 22. GAP del ALNS para 100 clientes.....	105
Figura 23. Utilización promedio de la batería del dron	106
Figura 24. Mejor solución conocida para el VRP y VRPD	107
Figura 25. Resultados tiempos de vuelo del dron al variar su velocidad.....	109

Lista de Apéndices

(Apéndices disponibles en el CD adjunto)

Apéndice A. Análisis bibliométrico	22
Apéndice B. Pseudocódigos traducidos de los algoritmos utilizados	65
Apéndice C. Código del algoritmo ALNS del presente trabajo de investigación	65
Apéndice D. Ejemplo del algoritmo Clarke y Wright	68
Apéndice E. Instancias utilizadas en la validación del algoritmo	90
Apéndice F. Artículo científico de carácter publicable	102

Resumen

Título: Un algoritmo ALNS para el problema de ruteo de vehículos con drones en la distribución de última milla*

Autores: Jiménez Romero, Jhon Alexander; Tarazona Jiménez, Jeamy Sebastián**

Palabras clave: Ruteo de Vehículos con Drones, *Adaptive Large Neighborhood Search*, UAV's, Drones, VRPD.

Descripción: Los vehículos aéreos no tripulados o más conocidos como drones han despertado gran interés en los últimos años, teniendo aplicaciones en operaciones militares y civiles, recientemente se ha investigado acerca de las ventajas de su uso en la distribución de paquetes. En el presente trabajo de investigación se formula un problema similar a el *Flying Sidekick Traveling Salesman Problem* (FSTSP), en referencia a que los drones y camiones pueden trabajar en conjunto; mientras el dron hace una entrega, el camión puede visitar varios clientes. La diferencia del FSTSP es que este modelo considera que los camiones tienen un límite de capacidad y además se tienen restricciones que limitan el tiempo de la ruta, dicha formulación se conoce como el Problema de Ruteo de Vehículos con Drones (VRPD); esta investigación se centra en la distribución de última milla. Para resolver esta formulación se presenta el algoritmo *Adaptive Large Neighborhood Search* (ALNS), el cual es validado usando instancias propuestas en la literatura. Se verifica cómo varía la función objetivo de la solución inicial mediante el uso de heurísticas destrucción y reparación, finalmente se realiza un análisis de sensibilidad para algunos parámetros del algoritmo y características de los drones; realizando conclusiones de los resultados arrojados y efectuando recomendaciones para futuras investigaciones.

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Estudios Industriales y Empresariales. Directora: Karin Aguilar Imitola, M.Sc. en Ingeniería Industrial. Codirector: Henry Lamos Díaz, Ph.D. en Física, Matemática.

Abstract

Title: An ALNS algorithm for vehicle routing problem with drones in the last mile distribution*

Author: Jiménez Romero, Jhon Alexander; Tarazona Jiménez, Jeamy Sebastián**

Keywords: Vehicle routing with Drones, *Adaptive Large Neighborhood Search*, UAV's, Drones, VRPD.

Description: Unmanned aerial vehicles or better known as drones have attracted great interest in recent years, having applications in military and civilian operations, recent researches on transportation logistics, have done investigations with the aim of determining the advantages of using them for the distribution of packages. In this research project, a similar problem to the Flying Sidekick Traveling Salesman Problem (FSTSP) is formulated, in which the drones and truck can work together; while the drone makes a delivery, the truck can visit several customers. The difference between the FSTSP and the present model, is that the later considers that the trucks have a limited capacity as well as restrictions that limit the time of the route, the formulation is known as the Vehicle Routing Problem with Drones (VRPD). This research focuses on the last mile distribution and in order to solve this formulation, the algorithm *Adaptive Large Neighborhood Search* (ALNS) is presented, which is validated using instances proposed previously in the related literature. Additionally, the variations in the initial target function are also verified by means of the application of destruction and repair heuristics, finally a sensitivity analysis is performed to certain parameters of the ALNS and characteristics of the drones; stating conclusion based on the results returned and making recommendations for future researches.

* Bachelor's degree

** Faculty of Physical Mechanical Engineering. School of Industrial and Business Studies. Director: Karin Aguilar Imitola, M.Sc. in Industrial Engineering. Co-director: Henry Lamos Díaz, Ph.D. in Mathematical Physics.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Introducción

En los últimos años la complejidad en la distribución de la última milla ha incrementado significativamente, debido al aumento en las ventas de productos que se entregan puerta a puerta; se estima que los envíos generalmente oscilan de 50 a 150 paradas por día (Brown Eric, 2019), adicionalmente según datos del Estudio Anual de ECommerce 2016 de IAB, para un 52% de los clientes resulta fundamental elegir un comercio electrónico y que el plazo de entrega propuesto sea rápido; un 66% considera que dicho plazo debe ser inferior a 5 días; un 24% estaría dispuesto a pagar por una entrega rápida (la mitad de los días); y para un 55% son clave los gastos de envío gratuitos (Observatorio eCommerce & Transformación Digital Centro de Estudios EY, 2017, pag 18). Dicha actividad requiere un conjunto de acciones propias o ejecutadas por la distribución de última milla, operación que implica el movimiento de la mercancía o producto desde el almacén hasta la entrega al usuario. En este sentido, se comprende que los inconvenientes no solo tienen que ver con el tráfico, dificultad en zonas peatonales, la reglamentación y regulación de dichas descargas, sino que también se despachan productos de tamaño pequeño a clientes individuales; por ello, este proceso tiene un costo considerable y una eficiencia baja. Además, desencadena disputas corporativas, causando temas relacionados con la contaminación ambiental, la conciencia ecológica, circunstancias laborales en los repartidores, etc. Por lo anterior, las empresas y los investigadores han centrado su atención en determinar rutas con el mínimo costo posible que permitan satisfacer a la totalidad de sus clientes; como resultado de distintas investigaciones se han probado diferentes algoritmos con los que es posible, y según los parámetros de cada caso, determinar rutas de entregas con un mínimo tiempo y costo; entre algunos de los algoritmos se destacan el genético, la búsqueda tabú, el recocido simulado, los de colonia de hormigas, el de búsqueda de vecindario y el de búsqueda de vecindario adaptativo. La distribución de última milla

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

se convierte en el punto clave a intervenir para optimizar de manera efectiva el transporte de bienes o productos, esta necesidad surge debido a que los sistemas de transporte terrestre no son suficientes en términos de tiempo y costos; igualmente el transporte aéreo actual se muestra insuficiente, debido a la falta de pilotos entrenados y los elevados costos que estos representan. Por estos motivos, es fundamental plantear la posibilidad de usar flotas mixtas, en este caso una de camiones y vehículos aéreos no tripulados, también conocidos como UAVs.

El uso de Vehículos Aéreos no tripulados o más conocidos como drones, ha ido incrementando considerablemente debido a los avances tecnológicos que han permitido mejorar sus características técnicas, además de tener ventajas sobre los medios de transporte terrestres, una de ellas es el poco impacto negativo que tiene sobre el medio ambiente, la eliminación de atascos de tráfico, entre otras; todo esto ha motivado a realizar una gran variedad de aplicaciones. Entre las primeras se tiene las operaciones militares, donde se han usado en labores de vigilancia, desminado de territorios y bombardeos; otra aplicación ha sido en operaciones de logística humanitaria, entre las que se destacan el mapeo de zonas de desastre, apoyo en operaciones de rescate y la distribución de insumos médicos. Viendo estos avances y las ventajas que podría generar, Jeff Bezos CEO de Amazon, anunció en 2013 el Prime air, un servicio que usa drones multirrotor para realizar la entrega de paquetes desde Amazon hasta sus clientes (Rose, 2013); así mismo, la empresa alemana de paquetería integrada DHL también inició su proyecto Parcelcopter en 2013, este ha transportado medicamentos a la isla de Juist en el mar del norte (Bonn, 2014); en 2014 Google reveló el proyecto “Wing”, en el que se producen drones con mayor capacidad de carga que los prime air y el parcelcopter (Stewart J, 2014). Paralelamente, en 2014 los Emiratos Árabes Unidos anunciaron un plan para usar aviones no tripulados para distribuir documentos oficiales del gobierno, tales como permisos y tarjetas de identificación, entre otras empresas como UPS, FedEx y Domino’s pizza,

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

quienes ven en el uso de estos dispositivos grandes beneficios, como reducción de costos de ruta y menor tiempo para realizar las entregas, por esta razón surgen los FSTSP (*Flying Sidekick Traveling Salesman Problem*) (Murray & Chu, 2015) y posteriormente el VRPD (*Vehicle Routing Problem with Drones*) como ramas de investigación para este problema. Esta investigación pretende contribuir en el estudio de los problemas de ruteo de vehículos con drones, enfocada en la distribución de última milla, un tema relativamente reciente y que cuenta con gran interés por muchas empresa como Amazon (Amazon.com, 2016), DHL (Bonn, 2014) y Google (Stewart J, 2014). Por ello se aplicó la formulación de un modelo matemático del problema de ruteo de vehículos con drones para la distribución de última milla, el modelo propuesto es resuelto mediante el algoritmo *Adaptive Large Neighborhood Search* o ALNS. Posteriormente se realiza la evaluación del desempeño de dicho algoritmo con el uso de instancias encontradas en la literatura y el respectivo análisis de los resultados; el cumplimiento de los objetivos del presente trabajo de investigación se evidencia en la Tabla 1.

Tabla 1

Cumplimiento de los objetivos del proyecto

Objetivos específicos	Numerales relacionados
1. Realizar una revisión de literatura sobre el problema de ruteo de vehículos con drones (VRPD), su aplicación en modelos de distribución de última milla, y su solución mediante el uso de algoritmos basados en estructuras de vecindario.	Capítulo 2
2. Formular un modelo matemático para el problema de ruteo de vehículos con drones en la distribución de última milla.	Capítulo 5
3. Desarrollar en el lenguaje de programación Python un algoritmo Adaptive Large Neighborhood Search (ALNS) para la solución del VRPD planteado.	Capítulo 6 y Apéndice C
4. Validar el algoritmo ALNS propuesto para la solución del VRPD con instancias de prueba encontradas en la literatura.	Capítulo 7 y 8

5. Elaborar un artículo científico de carácter publicable en base a la investigación realizada que contenga los resultados del proyecto de investigación.

Apéndice
F

1. Generalidades del proyecto

1.1 Planteamiento del problema

Recientemente algunos investigadores han definido el proceso y efectos de la distribución de última milla como aquel segmento final en el que llegan las mercancías a su sitio de consumo, al mismo tiempo, puede considerarse el recorrido de inicio donde los artículos se despachan desde su origen; es decir, desde su ubicación inicial, hasta el lugar de destino. La complejidad en la distribución de última milla ha incrementado significativamente, debido al aumento en las ventas de productos que se entregan puerta a puerta, se estima que los envíos, generalmente, oscilan entre 50 y 150 paradas por día (Brown Eric, 2019).

Por lo anterior, existe una creciente preocupación y necesidad de optimizar la distribución de última milla debido a las exigencias planteadas por diversas características como la demanda que enfrentan los operadores de transporte, la efectividad y la gestión en la entrega de mercancías, entre otras; de manera que la trascendencia de este fenómeno está en las intervenciones propuestas por quienes toman las decisiones. En este sentido, es necesario resaltar que la planificación y supervisión en el traslado de mercancías se puede mejorar para evitar molestias en el desarrollo de cada una de las actividades de la cadena, con el fin de eludir los impactos negativos típicos, propios del transporte terrestre como es el flujo del tráfico, los accidentes en las vías, el contraflujo, las emisiones de Dióxido de Carbono (CO₂) y las grandes distancias, entre otros.

En la conferencia “*A GIS Methodology For Estimating The Transport Network Impedance To Last-Mile Delivery*” presentada en la “*State of Australian Cities Conference*” los autores plantean

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

un factor clave que afecta el tramo de última milla en la distribución de productos, y es la planificación de los controles en áreas urbanas, como restricciones de estacionamiento, acceso a las bahías de carga, capacidad restringida de infraestructura y transporte. Dichos factores afectan negativamente a las empresas que no los consideran en sus rutas de entregas, pues a medida que la red avanza hacia el cliente final, se vuelve más poblada y es más probable que se produzcan cuellos de botella, retrasos en los pedidos e ineficiencias (Gevaers, 2013). Hay indicios de que hasta un 28% de los costos de transporte pueden estar relacionados con la entrega de la última milla o la recogida en la primera milla (Goodman, 2005); todo esto apunta a que las empresas no puedan tener en su totalidad entregas de pedidos perfectos en sus operaciones. El porcentaje de pedido perfecto para Colombia es alrededor del 57.8%, el cual puede aumentar de manera significativa si se mejoran las estrategias de transporte y distribución de paquetes (Peñaloza, 2016). Un pedido se considera perfecto cuando se satisfacen dos aspectos vitales, el primero está asociado con la percepción de seguridad y confianza que brinda la plataforma virtual (Tavera, Sánchez, & Ballesteros, 2011), y el segundo está relacionado con la reducción en el tiempo de espera del producto, ya que este aspecto es decisivo para un comprador al momento de escoger dónde hacer su compra. Con el propósito de tener mejoras en las entregas, el problema de distribución de recursos ha sido abordado mediante un enfoque de optimización e implementado a través del problema de ruteo de vehículos (VRP). El VRP, busca establecer las rutas de recorrido de una flota de vehículos que deben visitar distintos puntos de demanda, de manera que se minimicen los costos, o distancias de las rutas; objetivos que son importantes para las empresas a la hora de tener una mejor eficiencia en sus operaciones.

Las nuevas tendencias han incorporado otros desafíos para la distribución de última milla, entre estos tenemos la reducción en la huella de carbono, esto ha ocasionado que su administración sea

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

cada vez más compleja; es por esto por lo que, el uso de drones en la distribución de paquetes presenta una alternativa de acelerada evolución tecnológica con el fin de dar una solución a estos problemas. En abril de 2017, la empresa canadiense GlobalMedic unió sus esfuerzos con Aeryon Labs Inc. para ayudar a los rescatistas de la región del Putumayo colombiano, equipando con un generador, dos Aeryon SkyRanger (drones), una computadora portátil y Pix4Dmapper, el equipo estaba preparado para enfrentar el desafío de operar en una zona de desastre con un terreno difícil y un acceso limitado a la energía y los datos, después de que se presentará una avalancha que causó más de 300 muertos y más de 40.000 desplazados (Pix4D, 2017). En el mismo sentido, en la ciudad de Cali (Colombia) se comenzó a implementar un proyecto en el que se usan drones para transportar suministros médicos desde y hacia la región de Montebello que es una zona de difícil acceso (El tiempo, 2017).

La combinación de camiones con UAVs (*Unmanned aerial vehicles*) o mejor conocidos como drones, usados para la distribución de última milla, ha sido estudiada desde la investigación de operaciones mediante el VRPD (*Vehicle routing problems with drones*) (Z. Wang & Sheu, 2019), que es una extensión del problema de ruteo de vehículos (VRP). Una característica distintiva del VRPD es que un avión no tripulado puede viajar con un camión, despegar de su parada para servir a los clientes, y aterrizar en un centro de servicio para viajar con otro camión siempre que se cumplan las limitaciones de alcance de vuelo y capacidad de carga. El VRPD representa ventajas significativas en los problemas de distribución de corta distancia, debido a que no cuentan con las limitaciones de infraestructura, sus costos de mantenimiento son menores a los de los camiones de reparto tradicionales y se pueden hacer entregas más rápidas; no obstante, para su implementación se deben tener en cuenta los factores como tiempo o rango de vuelo ya que este es limitado a causa de su batería, y otro factor es la capacidad de carga útil de estos dispositivos.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Al igual que el VRP clásico, el VRPD es un problema tipo NP-Hard, es decir, computacionalmente difícil de resolver a medida que aumentan el número de datos del problema. Para dar solución a un problema de esta complejidad, se decide abordarlo mediante distintos métodos de solución, como el que plantean Murray y Chu, quienes proponen la programación lineal entera mixta (MILP) en conjunto con una heurística que adopta una idea de “*Truck First, Drone Second*”, los autores prueban los casos con 10 clientes (Murray y Chu, 2015). En 2016 Ponza extendió el trabajo de Murray y Chu proponiendo una mejora al modelo MILP planteado, resolviéndolo mediante la metaheurística iterativa de recocido simulado y probándola con instancias de prueba adaptadas del problema “*Travelling Salesman Problem*” (TSP) con una transformación de tiempo polinomial (Ponza, 2016). Agatz en 2018 construyó un modelo de programación entera y desarrollaron la heurística “*route-first, cluster-second*” basada en la búsqueda local y la programación dinámica (Agatz et al., 2018). Uno de los algoritmos usados para la solución del VRPD es el “*Adaptive Large Neighborhood Search*” (ALNS), definido como una búsqueda local basada en vecindarios grandes que trata de mejorar la solución inicial, aplicando diferentes heurísticas de destrucción y reparación con éxito variable, el ALNS ha sido aplicado a diversas extensiones del VRP debido a su simplicidad para acoplarse a nuevos problemas. Con base en lo anterior, este trabajo busca proponer una solución al modelo VRPD mediante el algoritmo ALNS, para lo cual se realiza una revisión de literatura sobre el problema VRPD. Posteriormente se formula un modelo matemático sobre el VRPD; así mismo se desarrolla el algoritmo de solución ALNS en el lenguaje de programación Python, finalmente el algoritmo propuesto es validado mediante instancias de la literatura presentadas por (Sacramento et al., 2019), con el fin de evaluar la eficiencia de la metaheurística en el VRPD aplicado a la distribución de última milla.

1.2 Objetivos

1.2.1 Objetivo general

Desarrollar un algoritmo ALNS para la solución del problema de ruteo de vehículos con drones en la distribución de última milla.

1.2.2 Objetivos específicos

- Realizar una revisión de literatura sobre el problema de ruteo de vehículos con drones (VRPD), su aplicación en modelos de distribución de última milla, y su solución mediante el uso de algoritmos basados en estructuras de vecindario.
- Formular un modelo matemático para el problema de ruteo de vehículos con drones en la distribución de última milla.
- Desarrollar en el lenguaje de programación Python un algoritmo Adaptive Large Neighborhood Search (ALNS) para la solución del VRPD planteado.
- Validar el algoritmo ALNS propuesto para la solución del VRPD con instancias de prueba encontradas en la literatura.
- Elaborar un artículo científico de carácter publicable en base a la investigación realizada que contenga los resultados del proyecto de investigación.

1.3 Metodología

En este capítulo se presenta la metodología que se lleva a cabo en el presente trabajo de investigación se siguió de Frederick Hillier y Gerald Lieberman en el libro “Introducción a la Investigación de Operaciones”(Hillier & Lieberman, 2013), la cual está ilustrada en la Figura 1, en esta se describe a detalle cada una de las seis etapas a seguir.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

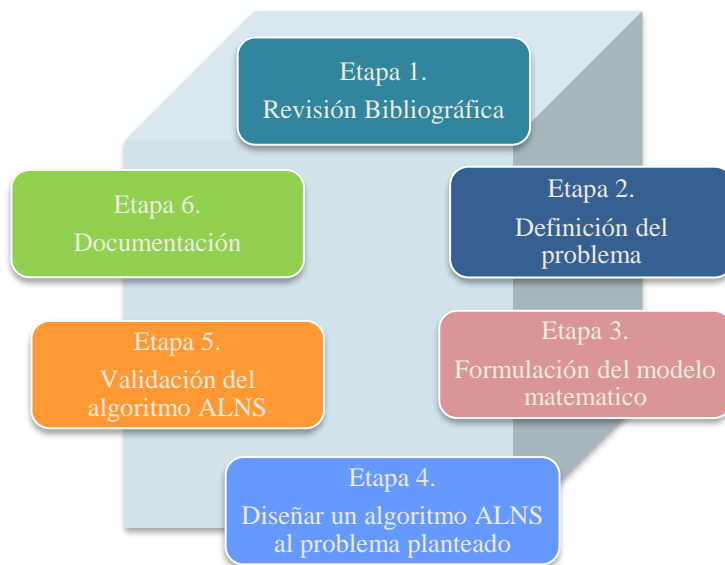


Figura 1. Fases metodológicas de un estudio en investigación de operaciones

1.3.1 Etapa 1. Revisión bibliográfica. En esta etapa se llevará a cabo una revisión y análisis de artículos científicos, relacionados con el problema de ruteo (*Vehicle routing problem with drones*, VRPD) bajo un enfoque multidisciplinario, sin perder de vista su aplicación en la logística de última milla, con el propósito de construir un marco referencial y conceptualizar el tema de investigación. Además, se estudian los principales algoritmos de solución propuestos hasta el momento para resolver el problema.

1.1. Definir las palabras claves para la construcción de la ecuación de búsqueda en las bases de Datos de la Universidad y otras fuentes.

1.2. Revisar la literatura disponible en bases de datos sobre la solución al problema VRPD mediante el uso de heurísticas y metaheurísticas.

1.3. Revisión de la literatura sobre las soluciones del VRPD por medio del *Adaptive Large Neighborhood Search (ALNS)* en las bases de Datos de la Universidad y otras fuentes.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

1.3.2 Etapa 2. Definición del problema.

2.1. Clasificar los aportes más relevantes obtenidos de la revisión bibliográfica, con el propósito de estudiar las formas en las que el problema ha sido abordado y solucionado en la literatura.

2.2. Plantear el problema a resolver, acorde a la teoría existente e incorporando las decisiones tomadas en etapas previas.

1.3.3 Etapa 3. Formulación del modelo matemático. Esta fase consiste en formular el problema de manera adecuada para su análisis. La forma convencional donde OR (*Operation Research*) logra esto, es en la construcción de un modelo matemático que represente la esencia del problema, por ende, se realizarán las siguientes actividades:

3.1 Identificar de la literatura la función objetivo más usada para el problema VRPD, así como las variables de decisión y restricciones más comunes.

3.2 Formular un modelo matemático que represente al problema VRPD.

1.3.4 Etapa 4. Diseñar un algoritmo ALNS al problema planteado. Una vez analizada la información más relevante en torno al problema y formulado el modelo matemático, se da paso al desarrollo de un algoritmo de solución, este trabajo en particular consiste en la metaheurística evolutiva (*Adaptive Large Neighborhood Search, ALNS*). En el cumplimiento de esta etapa se realizan las siguientes actividades:

4.1. Conocer y comprender el lenguaje de programación Python.

4.2. Estudiar las bases teóricas del algoritmo ALNS para su adaptación como método de solución al VRPD.

4.3. Diseñar un algoritmo ALNS de solución al problema sujeto al modelo planteado.

4.4. Construir el algoritmo en lenguaje de programación Python.

4.5. Verificar que el ALNS propuesto genera soluciones factibles y adecuadas al problema.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

1.3.5 Etapa 5. Validación del algoritmo ALNS. En esta etapa se debe comprobar el algoritmo, con el objetivo de determinar las mejoras y/o ajustes que deben ser realizados.

5.1 Identificar o buscar las instancias en la literatura.

5.2 Adecuar y transformar las instancias en un formato idóneo para el algoritmo planteado.

5.3 Validar el algoritmo con instancias adaptadas de la literatura.

1.3.6 Etapa 6. Documentación y síntesis de resultados. En esta última fase se recopila la información y los resultados obtenidos durante el proceso investigativo, con la finalidad de:

6.1. Elaborar el informe académico final en donde se exponen y consolidan los resultados obtenidos en el desarrollo del trabajo de investigación.

6.2. Realizar el artículo de carácter publicable con la información destacada de la investigación y los resultados obtenidos.

2. Revisión de literatura

2.1 Análisis preliminar de la literatura

El problema de ruteo de vehículos (VRP) ha sido ampliamente estudiado a lo largo de los años, dado que el interés de muchas empresas está fijado en tener el menor costo en sus rutas de entrega, dichas rutas involucran el transporte de productos desde los depósitos hasta los usuarios finales, y se estima que los costos de transporte se encuentran entre un 10% y 20% del valor final de los bienes (Toth, P., Vigo, 2000); por esta razón, se han elaborado numerosas investigaciones que buscan alternativas de solución a dichos problemas.

La primera investigación referente al VRP se dio con "*The Truck Dispatching Problem*" (Dantzig & Ramser, 1959), donde se realizó una formulación del problema para la distribución de combustible. En dicho artículo, Dantzig y Ramser obtuvieron una ruta óptima entre una terminal granelera y las terminales de gasolina que atiende esta, mediante el uso de una flota

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

homogénea de camiones repartidores de gasolina. Se estableció la demanda de los productos a entregar y se asignaron las estaciones de abastecimiento de los camiones repartidores, de manera que fuera satisfecha la demanda y que, al mismo tiempo el recorrido transitado por la flota de vehículos fuera el mínimo posible. Luego en 1964, los autores Clarke y Wright, tomando como referencia la investigación de Dantzig y Ramser, publicaron el artículo “*Scheduling of vehicles from a central depot to a number of delivery points*” (Clarke & Wright, 1964), en la formulación matemática eliminaron una restricción de carga y además para resolver el problema propusieron un algoritmo diferente al planteado por Dantzig y Ramser con el que fue posible encontrar mejores resultados; dicho algoritmo actualmente es conocido como algoritmo de ahorros.

A partir del análisis bibliométrico el cual se puede encontrar en el Apéndice A, se pudo observar que ha crecido de manera significativa el interés por parte de los investigadores en abordar el VRP, el cual consiste en minimizar los trayectos y los costos de las rutas de los vehículos, comenzando desde un almacén central para satisfacer la demanda de un determinado conjunto de clientes. Con el paso de los años se han estudiado distintas variantes a este problema, con el fin de aproximarlos más a la realidad, a su vez se han planteado y evaluado distintos tipos de algoritmos, cada vez más robustos y completos, lo que les permite tener soluciones factibles con menor tiempo de procesamiento. Entre algunos de estos algoritmos se encuentran: recocido simulado, colonia de hormigas, Búsqueda de Vecindario adaptativa, metaheurísticas híbridas, heurísticas de Búsqueda de Vecindario Variable (*Variable Neighborhood Search* o VNS), Búsqueda Tabú (*Tabu Search* o TS), entre otros; todos con la finalidad de obtener soluciones eficientes y factibles.

Una de las variantes que se han estudiado es el uso de una flota heterogénea denominado HVRP (Golden et al., 1984), donde se analiza el comportamiento de la respuesta al tener vehículos de diferentes capacidades, en relación a esto, en el artículo “*The fleet size and mix vehicle routing*

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

problem” se realizó una comparación entre distintos algoritmos y finalmente se estableció la combinación óptima en la que se minimizaran los costos totales. Trece años después Crevier, Cordeau, y Laporte, analizaron el problema del VRP con rutas entre puntos (MDVRPI), en esta variante se consideraron depósitos intermedios en los que los vehículos pueden reponerse con mercancías durante el curso de una ruta, los depósitos fueron ubicados en forma central luego de haber agrupado a los clientes. Este problema fue resuelto mediante una heurística que combina el principio de memoria adaptativa y búsqueda tabú, probado con instancias generadas aleatoriamente, propuestas por Cordeau, Gendreau, y Laporte, donde se tenían entre 48 y 216 clientes, de 3 a 6 depósitos y de 4 a 6 vehículos (Cordeau et al., 1997). Posteriormente, algunos investigadores como Homberger y Gehring centraron su atención en el ruteo de vehículos con ventanas de tiempo (VRPTW), donde se busca satisfacer la demanda del cliente en una franja determinada de tiempo y en caso de no realizarse se deben asumir penalizaciones, la función objetivo planteada por Homberger y Gehring era minimizar el número de vehículos (criterio principal) y la distancia total de viaje (criterio secundario), que resolvieron mediante una metaheurística híbrida de dos fases, en la primera fase usaron una estrategia de evolución y en la segunda un algoritmo de búsqueda tabú; el metaheurístico se sometió a una prueba comparativa sobre la base de 356 problemas de la literatura con tamaños que varían de 100 a 1000 clientes (Homberger & Gehring, 2005).

En los últimos años y motivado en gran manera por los cambios climáticos se han incorporado a la flota de camiones, vehículos con combustión alterna como biodiesel, gas natural e híbridos (Erdoğan & Miller-Hooks, 2012), esto con el propósito de investigar las posibilidades de contribuir a mejorar el medio ambiente y reducir la huella de carbono de las rutas actuales; luego de esta contribución surge el problema de ruteo de vehículos eléctricos (EVRP), donde se consideran

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

restricciones como el tiempo de carga de la batería del vehículo y la disponibilidad de la infraestructura de carga a lo largo de la ruta. Derivado del trabajo realizado por Erdoğan y Miller-Hooks “los autores Schneider et al. (2014)”(Schneider et al., 2014) investigaron sobre el problema de ruteo de vehículos eléctricos con ventanas de tiempo y estaciones de recarga (E-VRPTW), que incorpora la posibilidad de recarga en cualquiera de las estaciones disponibles, además consideraron la limitada capacidad de carga que tienen los vehículos, así como los plazos de entrega con los clientes. Como método de solución los autores usaron una heurística híbrida que combina una búsqueda de vecindario variable (VNS) con una heurística de búsqueda tabú, evaluando con instancias propuestas por Solomon en “Algoritmos para los problemas de enrutamiento y programación de vehículos con restricciones de ventana de tiempo” (Solomon, 1987).

Otra de las posibilidades para reducir la huella de carbono y además minimizar los tiempos de ruta, es el uso de camiones en combinación con drones para la entrega de paquetes en la distribución de última milla, este problema es también una variante del VRP; se denomina problema de ruteo de vehículos con drones (VRPD). Murray y Chu en 2015 realizaron un primer acercamiento al problema abordando el denominado FSTSP (*the flying sidekick traveling salesman problem*), el problema consistía en determinar las asignaciones óptimas de clientes para un UAV que trabaja en conjunto con un único camión de reparto; el objetivo del modelo es reducir el tiempo de duración de la ruta de los camiones y drones para dar servicio a todos los clientes y devolver ambos vehículos al depósito. En este problema se contaba con un dron que podía desplazarse sin necesidad de que el camión esperara, el problema fue resuelto mediante un modelo de programación lineal entera mixta y probado con instancias de la literatura (Murray & Chu, 2015).

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Después del aporte de Murray y Chu, Ferrandez et al. (2016) realizaron una investigación en la que se analizaba la efectividad de usar UAVs en las redes de entrega, analizando el tiempo y la energía asociados a una red de camiones y drones en comparación con un camión o dron independiente, también determinaron el número óptimo de sitios y ubicaciones de lanzamiento de los drones por cada camión acorde a los requisitos de entrega, este problema fue resuelto mediante el algoritmo de agrupación de *K-means* con el propósito de encontrar ubicaciones de lanzamiento, luego haciendo uso de un algoritmo genético para resolver la ruta del camión como un problema de vendedor ambulante (TSP), el tiempo total de entrega, la energía total utilizada, los costos totales basados en el costo por kilómetro y el costo por hora fueron criterios de rendimiento probados para analizar las soluciones obtenidas. En ese mismo año T. Wen, Zhang & Wong hicieron un estudio sobre del uso de drones para el problema del suministro de sangre multiobjetivo en situaciones de emergencia, en esta investigación se tuvieron en cuenta aspectos complejos como: temperatura a la que se debe mantener la sangre durante el transporte, programación y planificación de rutas de los UAV en caso de que múltiples sitios soliciten sangre y la capacidad de carga limitada de los UAVs. El foco de atención de los investigadores fue el cambio de temperatura de la sangre debido al ambiente externo, el agente de calentamiento (o refrigerante) y el factor tiempo durante el transporte, el problema fue resuelto mediante el algoritmo evolutivo multiobjetivo basado en descomposición, y comparado mediante 9 instancias tradicionales del CVRP multiobjetivo (Wen et al., 2016). El VRPD fue introducido por Wang et al. en el 2017, en el, se establece un depósito, un grupo de clientes que esperan ser servidos y una flota de vehículos. De forma que cada vehículo tenga una cuadrilla predefinida de drones. El objetivo es minimizar el *makespan*, es decir, el intervalo de tiempo requerido para atender a todos los clientes de modo que la flota dada, es decir, vehículos y drones, cuente con un retorno hacia el depósito final. Wang

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

et al. (2017) introdujeron algunos límites superiores sobre las probables reducciones que tiene el *makespan* mediante la utilización de drones, en comparación con un caso en el que no se administran drones. Los límites superiores se obtienen al investigar la estructura de soluciones óptimas y dependiendo de la velocidad relativa de los drones en comparación con los vehículos y el número de drones transportados por cada vehículo.

En 2017 Coelho et al., (2017), presentaron un problema multiobjetivo de ruteo de vehículos con flota heterogénea de UAV, donde se consideró la autonomía limitada de los vehículos, múltiples estaciones de carga y se respetaron los requisitos operativos, se formuló usando un modelo de programación lineal entera mixta (MILP), resuelto mediante un algoritmo de búsqueda dinámica y validado mediante un caso de estudio de dos capas. Poikonen et al. (2017) refinaron el trabajo de Wang et al. (2016) al considerar el impacto en los límites introducidos previamente al integrar en la función objetivo la vida útil de la batería, diferentes métricas de distancia y gastos operativos de despliegue de drones y vehículos. Además, los autores abordaron la posibilidad de lanzar y recuperar drones desde ubicaciones aleatorias en lugar de limitarse a las posiciones de los clientes y del depósito. Por último, concluyen su trabajo planteando que algunas heurísticas e instancias de referencia deben desarrollarse como el siguiente paso en la investigación del VRPD desde un punto de vista computacional. Schermer et al. (2018), basado en el trabajo de Di Puglia Pugliese y Guerriero (2017), presentaron una formulación MILP para el VRPD. Proporcionaron un enfoque heurístico basado en la Búsqueda de vecindario variable (VNS) y dos operadores heurísticos de inserción de drones para resolver el VRPD. Finalmente, se demuestra que un tándem¹ de drones

¹ Unión de dos personas o dos grupos que realizan una misma actividad en equipo o que combinan sus esfuerzos para hacer algo

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

para vehículos permite una disminución significativa con respecto a costo en comparación con la entrega de vehículos clásicos.

Ercan & Gencer (2018) estudiaron un problema de ruteo de vehículos dinámico con drones, donde consideraron por primera vez una versión dinámica del problema de enrutamiento de UAV en la que la información tridimensional de los objetivos con sus distancias de seguridad especiales, se introdujo también la altura mínima de vuelo para cualquier UAV; para minimizar el costo computacional de la inserción del método, se usó la agrupación difusa y fue resuelto mediante heurísticas dinámicas, posteriormente probado con un caso empírico de dos escenarios. En ese mismo año los investigadores Alotaibi et al., estudiaron para el sector militar el problema de ruteo de múltiples vehículos aéreos no tripulados UAV en presencia de amenazas del enemigo, donde su objetivo principal fue encontrar rutas óptimas que consideren los objetivos visitados, la exposición a ser derribado y el tiempo de viaje, para minimizar el riesgo de ser derribado. Los investigadores propusieron varios métodos para la generación de puntos de referencia, optimizados con la metodología de *Branch and cut and price (BCP)*, posteriormente resolvieron el problema de las rutas mediante dos algoritmos diferentes, primero el algoritmo con el camino simple heurístico, denominado DCG que considera solo la generación de columnas retrasadas; y un segundo algoritmo que considera DCG denominado DCG-MDS, las soluciones obtenidas fueron probadas con instancias donde se consideraban hasta 5 vehículos aéreos no tripulados, 10 objetivos, 30 puntos de amenaza y 55 puntos de referencia generados mediante el método de reducción de amenaza rectangular.

Otro aporte en ese año fue el de Ham, (2018a) donde se amplió el problema al considerar dos tipos diferentes de tareas de drones: soltar y recoger, con el problema de programación multi-camión, multi-dron y multi-depósito restringido por ventanas de tiempo, la sincronización de

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

recogida y visitas múltiples, donde luego que un dron completa una caída, puede volar de regreso al depósito para entregar el o los paquetes o volar directamente a otro cliente para que lo recojan, este problema fue modelado de forma única como una programación de máquina paralela no relacionada (PMS) con configuración dependiente de la secuencia (para la distancia de viaje), relación de precedencia (para recoger y soltar) y reentrante (para múltiples visitas y ventana de tiempo), resuelto mediante una programación de restricciones (CP) y probado con 120 instancias de múltiples camiones, drones, depósitos y cientos de clientes distribuidos en una región cuadrada de 8 millas (Ham, 2018b).

A mediados de 2018 los autores Boysen et al. estudiaron sobre el uso de los drones para la distribución de última milla por considerarlos una idea prometedora para reducir el tráfico excesivo por carretera. Con el fin de superar las dificultades causadas por la corta capacidad relativa de vuelo de los drones, estos vehículos aéreos no tripulados propusieron un concepto innovador al usar los camiones como plataformas móviles de despegue y aterrizaje, partiendo de que la ruta de los camiones ya estuviese dada, el objetivo es optimizar el cronograma de los drones lanzados desde el camión hasta el préstamo del servicio a un conjunto determinado de clientes, resuelto con dos modelos de programación entero y mixto, probado con la generación de tres tipos de instancias.

En Marinelli et al., (2018), los autores permitieron lanzar y recuperar drones desde lugares arbitrarios ubicados en la trayectoria del vehículo. Más exactamente, proponen una variante novedosa del TSP-D en la cual los drones pueden ser lanzados y recuperados en arcos por el vehículo, lo que ellos llaman una operación en ruta. Afirman que este procedimiento puede ser beneficioso para aumentar la utilización de drones y superar su rango limitado de operación. Para resolver el TSP-D extendido, Marinelli et al. (2018) plantean una heurística que consta de tres pasos distintos: Primero, encuentran una ruta para el vehículo utilizando la conocida heurística

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Lin-Kernighan (Lin & Kernighan, 1973). Segundo, dada la ruta del vehículo, construyen una lista de todas las operaciones de drones posibles que producen un costo reducido. Tercero, la lista inicial se mejora mediante la consideración de la operación en ruta y los movimientos se insertan de acuerdo con su costo reducido. A través de su estudio numérico, muestran que las operaciones en ruta pueden producir mejoras de hasta 10% con respecto al *makespan*. (Agatz et al., 2018) consideraron el problema con la variante de que el dron fuese más rápido que el camión y ambos se desplazaran por la misma red de carreteras, este problema fue resuelto mediante un algoritmo de programación dinámica y evaluado con instancias artificiales de distintas características y tamaños (Ferrandez et al., 2016, Wen et al., 2016).

Durante 2019 los autores (Sacramento et al., 2019) consideraron el uso de los drones con camiones para la distribución de paquetes, teniendo dentro de su modelo restricciones de capacidad, carga y tiempo de vuelo de los drones, además los clientes de este modelo se agrupaban. El objetivo de esta investigación fue minimizar el costo de las rutas, para lograrlo emplearon el algoritmo ALNS (*Adaptive Large Neighborhood Search*) del cual se obtuvo buenas soluciones, que posteriormente fueron comparadas y evaluadas mediante 112 instancias de (Sacramento et al., 2018).

Por otra parte, el algoritmo ALNS fue usado por primera vez en “*An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows*” (Ropke & Pisinger, 2006), los autores lo desarrollaron como una extensión de la heurística “*Large Neighborhood Search*” (LNS) de Shaw (1998). El ALNS permitió obtener una mejora de más del 50% de las instancias de la literatura en las que fue probada (Ropke & Pisinger, 2006). En los últimos años el ALNS ha sido aplicado a diferentes variantes del VRP, pues a menudo es fácil adaptarlo a nuevos problemas (Pisinger & Ropke, 2010), una de estas variantes fue investigada

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

por (Keskin & Catay, 2016) donde resolvieron el *Partial Recharge strategies for the Electric Vehicle Routing Problem with Time Windows* (EVRPTW-PR), donde la función objetivo se centró en minimizar la distancia recorrida y fue resuelto mediante el ALNS con ocho heurísticas de remoción *Random, Worst-Distance, Worst-Time, Shaw, Proximity-Based, Demand-Based, Time-Based, Zone Removal* y cuatro heurísticas de inserción usaron *Regret-2, Regret-3, Time Based Insertion* y *Zone Insertion*; evaluado con dos configuraciones de instancias propuestas por Schneider et al. (2014), una configuración fue de 36 instancias pequeñas que tenían entre 5 y 15 clientes; la otra fue de 56 instancias grandes que tenían 100 clientes y 21 estaciones de recarga. Un año más tarde Schiffer & Walther (2017) usaron el ALNS para resolver el *Location Routing Problem With Intra-route Facilities* (LRPIF), en el que plantearon nueve heurísticas para la destrucción, *Swap perfect facility, Drop, worst, Station vicinity, related, Shaw Station vicinity* y *Route*; para la reparación usaron dos heurísticas *Sequential perturbed insertion* y *Sequential insertion*; la función objetivo se enfocó en los costos totales, considerando los costos de inversión para vehículos, así como para ambos tipos de instalaciones en función de los costos fijos y el número respectivo de vehículos e instalaciones utilizados y los costos de conducción; evaluado con las instancias propuestas por Schneider et al. (2014). Otros trabajos en años posteriores como (Li et al., 2019; Phan et al., 2018; Sacramento et al., 2019; Schiffer & Walther, 2017), con las diferentes heurísticas seleccionadas para la destrucción y reparación obtuvieron buenos resultados en sus respectivas investigaciones comparándose con otros métodos de solución de la literatura.

Basados en los hallazgos de la revisión de la literatura, el ruteo de vehículos con drones es un tema relativamente reciente, las últimas investigaciones se han centrado en aproximarlos cada vez más a la realidad, incluyendo nuevas restricciones y ajustando los valores de los parámetros a los avances tecnológicos de los drones, incrementando el tiempo de vuelo y la capacidad de carga;

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

además se encontró que en la configuración del VRPD existen tres posibilidades referente a las entregas en conjunto de vehículos y drones; la primera es que puede realizarse de manera simultánea, mientras el dron entrega un paquete el camión puede hacer varias entregas; la segunda es que el camión espera al dron en el punto que lo lanzó; la tercera es que el dron se puede lanzar en cualquier instante de la ruta y no necesariamente desde un nodo, como ocurre en las dos configuraciones anteriores. En cuanto al ALNS se encontró que las heurísticas más usadas para la destrucción han sido *Worst removal* y *Random removal*, y en las heurísticas de reparación o inserción se encontró que las más utilizadas han sido las de *Regret-n insertion* y *Greedy insertion*. Es por esto que, se formulará un problema de ruteo de vehículos con drones en el que se considere que los camiones y drones trabajan de manera simultánea en la que el dron solo puede realizar una entrega al tiempo que el camión, proponiendo una formulación matemática con base a las de Murray & Chu, (2015); Sacramento et al., (2019), usando como método de solución la metaheurística ALNS, que como se encontró en la revisión de literatura se ha usado en varias investigaciones obteniendo buenos resultados.

3. Marco de antecedentes

En la Universidad Industrial de Santander hasta la fecha se han realizado dieciséis trabajos en modalidad de proyecto de investigación referentes al tema de ruteo de vehículos, luego de un análisis a las diferentes variantes del problema, se encontró una investigación de particular interés para el presente trabajo titulada “Solución del problema de ruteo de vehículos con ventanas de tiempo (VRPTW) mediante métodos heurísticos” (Lozada & Cadena, 2012), en el que los autores usaron tres heurísticas para dar solución al VRPTW: Algoritmo de Inserción de Solomon, el Algoritmo de Ahorros de Clarke y Wright y el Algoritmo del Vecino más Cercano, seleccionadas con el propósito de dar respuesta a la función objetivo planteada que era minimizar el tiempo total

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

del recorrido, además en el contenido del trabajo se realiza una descripción general sobre los métodos heurísticos de solución que buscan hacer un mejoramiento de las rutas inicialmente encontradas, dicho concepto es base de la búsqueda de vecindario que se usara en el presente trabajo.

En el trabajo titulado “Desarrollo de un algoritmo híbrido para la resolución del problema de ruteo de vehículos con entrega y recogida simultaneas (VRPSPD)” (Cruz Jimenez & Gonzalez Villamizar, 2013), los autores investigaron sobre el ruteo de vehículos con entrega y recolección simultanea (VRPSPD) la función objetivo minimiza los costos de transporte teniendo en cuenta los costos fijos y variables asociados a los vehículos, resuelto con el método de “optimización de enjambre de partículas” que al igual que el anterior contiene una heurística de mejoramiento de búsqueda local y posteriormente validado con 18 instancias de la literatura adaptadas por Tang y Galvao de las propuestas por Solomon (Solomon, 1987).

Por otro lado, es posible consultar en el repositorio de la Universidad de los Andes trabajos relacionados con drones, entre los que se destaca una tesis de maestría titulada “*The Colombian emergency response platform (PCRE) - design and testing of a sdi-based system to facilitate disaster management in Colombia*” (Mesa et al., 2016), en la que los drones son usados para realizar la captura de videos posterior a una emergencia, la cual es transmitida mediante una aplicación incorporada a las brigadas de rescate, esto con la fin de facilitar la labor de los rescatistas. Permitiéndoles contar con más información de datos especializados y tener mayor eficiencia a los requerimientos de los sitios afectados.

Como se mencionó en el artículo anterior apenas hasta el año 2016 comenzaron a utilizar los drones en el país, pero en el área de desastres naturales, lo que lamentablemente implica que los proyectos en el campo investigativo sobre el uso de drones sean escasos en el país y en la

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

exhaustiva búsqueda de antecedentes, además tampoco existen instancias de la literatura nacional que contemplen las restricciones adicionales generadas por la geografía colombiana y los recursos que una empresa nacional disponen son limitados para iniciar operaciones con este tipo de vehículos.

4. Marco teórico

A continuación, se presentan los conceptos más relevantes relacionados con la distribución de última milla, el problema de ruteo de vehículos, sus variaciones y métodos de solución para la realización del trabajo.

4.1 Optimización Combinatoria

Es un campo de la optimización en matemáticas aplicadas y la ciencia de la computación a través del cual se estudian problemas que se caracterizan por presentar una cantidad finita de soluciones factibles y trabajar con variables discretas. En términos generales, un problema de optimización consiste en encontrar el valor de un número específico de variables de decisión con el fin de lograr maximizar o minimizar una determinada función objetivo, sujeta a una serie de restricciones. Comúnmente, lo que se pretende maximizar hace alusión a unos beneficios, ganancias o utilidades y cuando se intenta minimizar, se habla de pérdidas, egresos o costos. Un ejemplo claro, corresponde al problema de ruteo de vehículos con drones (*Vehicle routing problem with drones, VRPD*). Los elementos que puede definir un Problema de Optimización Combinatoria pueden evidenciarse en la Tabla 2.

Tabla 2

Elementos de la optimización combinatoria. Adaptado de (Blum & Roli, 2003)

ELEMENTO	EXPRESIÓN
Un conjunto de variables.	$X = (X_0, X_1, X_2, X_3, \dots, X_k)$
Un dominio de las variables.	$D_0, D_1, D_2, D_3, \dots, D_k$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Restricciones entre variables. $X_0, \dots, X_k > a; X_0, \dots, X_k < b; X_0, \dots, X_k = c$

Una o múltiples funciones objetivo con criterios de maximización o minimización. Donde $Z = f(X_0, X_1, X_2, X_3, \dots, X_k)$ y en el que su dominio $D_f \in R$

En donde el conjunto de todas las posibles asignaciones factibles es:

$S = \{s: (X_1, V_1), \dots, (X_k, V_k)\}$, donde $X_k \in D_k$, y s satisface todas las restricciones.

Resolver un problema de optimización combinatoria consiste en encontrar una solución $s^* \in S$ tal que, $f(s^*) \leq f(s)$ para cualquier $s \in S$ si se minimiza o $f(s^*) \geq f(s)$ si se maximiza (Blum & Roli, 2003).

4.2 Optimización de Redes

Los problemas de optimización se ejemplifican mediante gráficos de redes, los cuales tienen en cuenta las distintas uniones que existen dentro de estos. El sistema se basa en un conjunto de elementos que logran el entendimiento del tipo de situación que se está representando, esto con el fin de que el problema se entienda de una manera sencilla (Hillier & Lieberman, 2013). Comprender un problema de optimización de redes es vital, los términos que se utilizan ampliamente en la literatura son: una red la cual está compuesta por puntos que reciben el nombre de nodos o vértices; líneas de conexión entre estos puntos, también llamadas arcos o ramas: por ejemplo, si A y B son dos nodos en una red, entonces el arco que conecta estos puntos se llamará AB. Sin embargo, dependiendo de la dirección del flujo en el arco, si se dirige de B hacia A, el arco recibirá el nombre de BA (Hillier & Lieberman, 2013). Los arcos se dividen en dos grupos: dirigidos y no dirigidos, el primer grupo son aquellos en el que el flujo se permite en una sola dirección, por el contrario, si va entre uno y otro de sus dos nodos, son denominados arcos no dirigidos. Dependiendo de los tipos de arcos que la componen, una red puede ser dirigida o no dirigida; en el caso de que contemple los dos grupos de arcos está se convierte en una red mixta,

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

siendo posible transformarla en una dirigida mediante la sustitución de cada arco no dirigido en dos dirigidos con direcciones opuestas (Hillier & Lieberman, 2013), se define como trayectoria a una serie de arcos diferentes que conectan dos nodos, también puede ser dirigida o no dirigida. Así mismo, se denomina ciclo a aquel recorrido cuyo nodo inicial y final son el mismo.

4.3 Complejidad Computacional

Un problema desde el punto de vista computacional está conformado por “un conjunto de datos de entrada, un conjunto de datos de salida y una función que asigna a cada dato del problema, una salida correcta del mismo” (Pérez-Jiménez & Sancho-Caparrini, 2003). Clasificar correctamente el problema de optimización que se pretende resolver en función de la teoría de la complejidad algorítmica es de vital importancia. Estos criterios de complejidad algorítmica son el espacio, es decir cuánta memoria y cuántos pasos se requieren para resolver el problema (Baños Navarro, 2006). Con base a esta idea, los problemas matemáticamente pueden clasificarse por su grado de dificultad, de esta manera los problemas pueden ser NP y NP-hard.

4.4 Problemas NP y NP-Hard

En la teoría de la complejidad computacional la clase NP es aquella que comprende todos los problemas computacionales que pueden ser resueltos en tiempo polinómico no determinístico de tal forma que el problema de decisión correspondiente sea resuelto con ayuda de una máquina de Turing no determinística.

Los problemas NP-hard son en parte semejantes, pero más difíciles de resolver que los problemas NP-completo. Dichos problemas pueden definirse alternativamente como la intersección entre clase NP y NP-hard, pero todos los problemas de la clase NP pueden reducirse a ellos. Con frecuencia, los problemas NP-hard son problemas de decisión que como mínimo son tan difíciles que los NP (pero no necesariamente en NP), así mismo requieren de más tiempo

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

exponencial y toma mayor tiempo computacional para darles solución. Los problemas NP-completo son un subconjunto de problemas NP-hard, y es por eso por lo que los problemas NP-completo a veces se llaman NP-hard (W. Hämmäläinen, 2006).

4.5 Vehículos aéreos no tripulados (UAV) o drones

Los UAS por sus siglas en inglés (Sistemas Aéreos No Tripulados), son aparatos capaces de desarrollar una función de forma completamente independiente, sin intervención humana de ningún tipo. Por lo tanto, los vehículos aéreos no tripulados (UAV) son dispositivos que no requieren un piloto a bordo, comenzaron a usarse en operaciones militares, específicamente para la captura de imágenes de territorios enemigos y barridos topográficos de zonas con misiles; los últimos desarrollos tecnológicos han permitido tener una mayor variedad de propósitos, incluyendo entregas de restaurantes como ¡YO! Sushi (Megan Willett, 2013) y Domino's (Reid, 2016), por agencias de transporte (Overly, 2017), y compañías de construcción (Balfour Beatty, 2017) y aerolíneas (Manalo, 2015). Un ejemplo de uso se presenta en Góez Sánchez (2016), donde los UAV implementan un sistema de planificación de ruta, que permite determinar la dirección a un objetivo. Evalúan los sistemas basados en microcontroladores integrados que ejecutan un planificador de ruta en tiempo real, que utiliza el algoritmo de optimización de PSO.

En general, se podría suponer que un avión no tripulado puede moverse más rápido que un vehículo entre dos puntos, debido a que no están sujetos a la restricción de movilizarse en la red de carreteras, ni tampoco afectarse por la congestión de estas. Además de esto son ligeros, aspecto que le permite consumir menor energía para moverse entre dos puntos. Sin embargo, su capacidad de carga es más restringida que la de un vehículo. Por otro lado, los UAV dependen de baterías relativamente pequeñas para la alimentación de su vuelo, el rango de operación es más limitado que el de un vehículo impulsado por un combustible fósil o por una batería mucho más grande (en

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

el caso de los transportes eléctricos). Por ende, las características complementarias de los vehículos y aviones no tripulados podrían tener un papel añadido en la entrega de última milla. Clásicamente, se supone que comúnmente el lanzamiento y la recuperación de aviones no tripulados se limitan a las ubicaciones de los clientes y el depósito (Murray & Chu, 2015).

4.6 Distribución de última milla

Definida como el segmento final en el proceso de entrega de pedidos, iniciando desde el estante de un almacén, hasta que el paquete finalmente llega a la puerta del comprador. Esto se lleva a cabo al recorrer en uno o distintos medios de transporte, un tramo con un paquete que sale de la última posición de distribución (centro de distribución, establecimiento, depósito, tienda, etc.) con el motivo de llegar a la ubicación de determinado(s) cliente(s) (Grupo Valora, 2016). Esto a su vez es un factor que incide significativamente en la satisfacción del consumidor, además de ser la parte más lenta y costosa de los procesos de envíos. Debido a que cuando se presenta una zona de acceso limitado y en la que no es posible que los vehículos de gran tonelaje puedan acceder a esta, se utilizan embarcaciones de poca dimensión con el fin de optimizar el tiempo de ruta (ECN, 2018).

4.7 Problema de Ruteo de Vehículos (VRP)

Enunciado por primera vez por Dantzig y Ramser en 1959, llamado entonces “*The Truck Dispatching Problem*” y enfocado al abastecimiento de estaciones de gasolina (Daneshzand, 2011). El VRP surge como una ampliación del problema del agente viajero y consiste en la determinación del conjunto de rutas a través de las cuales se sirve la demanda de los clientes al menor costo, definiendo así los requerimientos por parte de los clientes a atender según cada vehículo. (Toth & Vigo, 2014). De forma general, El VRP puede ser planteado de la siguiente manera: Sea $G = (V, E)$ un grafo donde E es el conjunto de arcos que deben recorrerse para realizar la distribución y $V = \{V_0, V_1, V_2 \dots V_n\}$ es el conjunto de vértices asociados a la ubicación de cada uno de los clientes,

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

excepto V_0 que es el vértice asociado a la ubicación del depósito de una flota de m vehículos disponibles (Laporte et al., 2014). Cada cliente tiene una demanda r y la suma de las demandas de los clientes que deben ser visitados por un solo vehículo debe ser menor a la capacidad Q del vehículo en cuestión. El objetivo del VRP es minimizar la suma de los costos de viaje, por ello cada arco $(i, j) \in E$ siendo $i \neq j$, tiene un costo asociado C_{ij} del cual se pueden fragmentar dos subproblemas. Si C_{ij} es igual a C_{ji} se afirma que la matriz de costos asociada a los arcos es simétrica; si por el contrario $C_{ij} \neq C_{ji}$, la matriz será asimétrica y los costos de recorrer los arcos dependerán del sentido en que dichos arcos sean recorridos.

Las características básicas del problema principal del VRP toman las siguientes asunciones:

- La demanda es determinística.
- Los clientes no pueden ser visitados más de una vez, esta restricción implica el hecho de que la demanda de un cliente no puede ser superior a la capacidad del vehículo destinado a atender su demanda. Por ende, cada cliente solo puede ser visitado por un vehículo.
- Los vehículos inician y retornan a un depósito o punto de partida fijo.
- Los costos de ir de un punto i a un punto j son los mismos.

4.8 Problema de Ruteo de Vehículos con Drones (VRPD)

El problema de ruteo de vehículos con drones (VRPD) es una extensión del clásico VRP, en el que hay dos tipos de entidades, un dron o varios drones que pueden tener múltiples tiempos de vuelo y aterrizaje, cada uno puede estar asociado con un camión diferente; y un vehículo que puede lanzar y recoger distintos drones en diferentes horas y ubicaciones. Tanto vehículos como drones se usan para entregar paquetes a los clientes. En el VRPD un dron puede viajar con un camión, salir desde el para servir a los clientes, aterrizar en un centro de servicio o distribución para viajar

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

con otro camión, siempre y cuando las restricciones de rango de vuelo y capacidad de carga no sean violadas (Z. Wang & Sheu, 2019).

Las principales decisiones del problema contemplan determinar el orden de servicio al cliente cumplido por los dos tipos de transporte, las ubicaciones de lanzamiento y recogida de los drones, así como las rutas recorridas por los vehículos; buscando minimizar la duración de las rutas. La dificultad del problema aumenta si se consideran restricciones de capacidad, tiempo de carga y rango de vuelo (Wang et al., 2019). La Figura 2 muestra un VRPD simple y en su solución se emplean dos camiones y dos drones para la entrega. Como se ilustra, el dron dos se toma inicialmente en el camión dos desde el depósito, luego vuela para servir a dos clientes, aterriza en un centro de acoplamiento para el servicio subsiguiente, y finalmente vuela de vuelta al depósito.

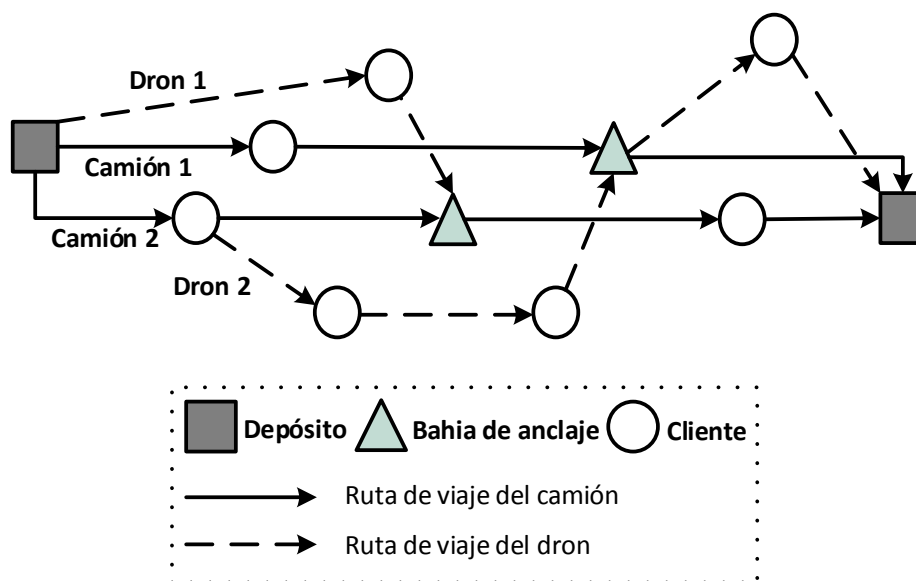


Figura 2. Un ejemplo del VRPD. Adaptado de (Z. Wang & Sheu, 2019)

X. Wang, Poikonen, & Golden, (2017) describe el problema VRPD así: Se asume que hay n clientes por ser atendidos por una flota homogénea de m camiones, cada uno llevando k drones. Cada cliente demanda un paquete que puede ser entregado por un camión o un dron. No hay tiempo de servicio para una entrega. Un camión tiene una capacidad para C paquetes.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Considerando las siguientes suposiciones sobre el comportamiento de los drones:

- Un dron puede transportar como máximo un paquete cuando está en el aire.
- Un dron tiene una duración de la batería de U unidades de tiempo.
- Los camiones y drones siguen la misma distancia métrica. Se supone que el camión y el dron deben viajar desde a hasta b a lo largo de la red vial. Restringir los drones para viajar a lo largo de la red es una suposición inicial. Además, al seguir la red vial, los drones evitarán obstáculos y espacio aéreo privado.
- Al regresar al camión, el tiempo necesario para preparar el dron para otro lanzamiento con un nuevo paquete y una batería cargada es insignificante.
- La velocidad del dron es α veces la velocidad del camión.

Así mismo, se consideran los siguientes supuestos sobre la coordinación entre el camión y los drones:

- Un dron lanzado desde un camión debe ser recogido por el mismo camión.
- El camión puede despachar y recoger un dron únicamente en un nodo; es decir, la ubicación del depósito o del cliente. El camión puede seguir sirviendo a los clientes después de que un dron es lanzado y recoger el dron en, posiblemente, un nodo diferente.
- El vehículo (camión o dron) que llega al nodo de recolección primero, tiene que esperar al otro.
- El objetivo es minimizar el tiempo total de la ruta; es decir, desde el tiempo que los camiones son despachados desde el depósito con los drones hasta que el último camión o dron regresa al depósito. Se hace alusión a este nuevo problema de ruteo de vehículos como el Problema de ruteo de vehículos con drones, denotado por VRPD m, α, k , donde m es el número de camiones en la flota, k el número de drones y α la relación entre la velocidad del dron y la velocidad del camión.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

La notación del VRPD más utilizada se enumera a continuación (X. Wang et al., 2017):

$Z(P)$: El valor óptimo de la función objetivo del problema P (por ejemplo, VRPD m, α, k)

$Z^f(P)$: El valor de la función objetivo de una solución factible f al problema P

L_r : La longitud de la ruta r .

T_r^{veh} : El tiempo de viaje en vehículo veh ($veh = trc$ para el camión y $veh = drn$ para dron) en la ruta r .

W_r^{veh} : El tiempo de espera por vehículo veh en la ruta r .

$D_r^{veh} = T_r^{veh} + W_r^{veh}$: La duración de la ruta r por vehículo veh .

4.9 Heurísticas de solución para los Problemas de Ruteo

Los algoritmos heurísticos se pueden agrupar de la siguiente manera: Constructivos, de descomposición y división, de reducción y de búsqueda de vecindad (Gallego Rendón et al., 2006).

Los métodos constructivos más usados son “El Algoritmo de Ahorros de Clarke & Wright”, las variantes de éste, y “Las heurísticas de inserción secuencial” (Figura 3).

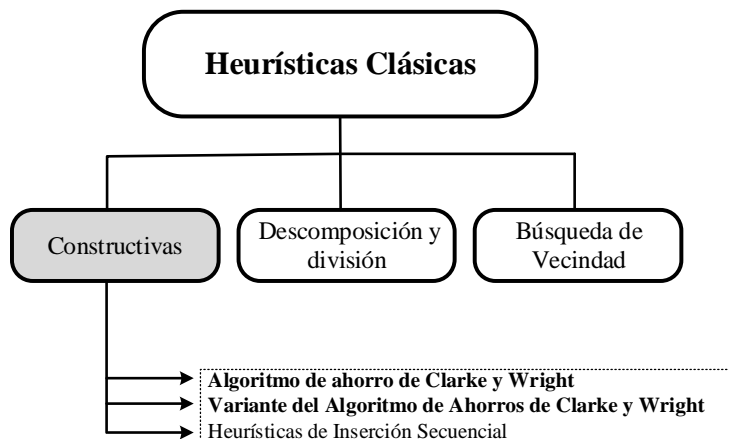


Figura 3. Clasificación de las heurísticas clásicas. Adaptado de (Benito Quintanilla, 2015)

En respuesta a la enorme complejidad de los problemas NP-Hard, los investigadores han creado métodos que proporcionan buenas respuestas en tiempos considerablemente cortos. Cabe resaltar

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

que dichos métodos están ideados para encontrar óptimos locales y que algunos toman ejemplos de comportamientos observados en la vida real.

4.10 Metaheurísticas de solución para los Problemas de Ruteo

En los años setenta, el término metaheurística se antepone al de heurística. El concepto del sufijo *meta* es “más allá” o “un nivel superior”, estos a su vez tienen distintas interpretaciones de lo que es llegar a resolver un problema de manera inteligente. Las metaheurísticas surgieron para mejorar procedimientos heurísticos generales con un gran desempeño de tiempo computacional favorable, la idea básica era dar solución a un tipo de problema de optimización general, usando parámetros dados por los usuarios, con el fin de dar lugar a la combinación de diversos métodos heurísticos y así conseguir una exploración del espacio de búsqueda de forma más eficiente y efectiva (Sánchez & Penín, 2003).

Estas técnicas se denominan metaheurísticas. Se pueden considerar, generalmente, los siguientes tipos de metaheurísticas para aplicar al VRP: metaheurísticas de relajación, constructivas, de búsqueda y evolutivas; entre estos grandes grupos podemos encontrar el recocido simulado, los algoritmos genéticos, las redes neuronales, la búsqueda tabú, colonia de hormigas y ALNS. En este documento, se explica a detalle la última de ellas.

4.10.1 Algoritmos de Hormigas (*Ant Colony Optimization, ACO*). percibe dicha sustancia, decide con alta probabilidad moverse por los trayectos con mayor frecuencia, lo que a su vez provocará un aumento de la feromona depositada en esa zona.

De este proceso emerge un comportamiento denominado auto catalítico: cuantas más hormigas sigan cierto trayecto, más atractivo se vuelve para ellas. En los Algoritmos de Hormigas se simula el comportamiento de una colonia de estos animales. Cada hormiga construye una solución combinando un criterio ávido que indica cuán acertada resulta cierta decisión, y la información

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

histórica (bajo la forma de feromona). El primer problema al que se aplicó esa metodología fue el TSP (Traveling Salesman Problem).

4.10.2 Búsqueda Tabú (Tabu Search). El algoritmo de Búsqueda Tabú (TS) fue planteado por Glover (Glover, 1986) este se basa en una búsqueda loca no aleatoria que acepta soluciones que incrementan el costo. En este procedimiento de selección de vecindario se cuenta con una iteración t en la que el algoritmo arranca con una solución S_t y llega a una S_{t+1} , la cual es la mejor de un subconjunto de sus soluciones vecinas $N(t)$, al pasar del estado S_t al S_{t+1} se llama movida. No necesariamente siendo está un valor inferior en costos que S_t , por ende, se debe utilizarse un criterio para que en la iteración siguiente no se regrese a S_t . Una manera de hacer es guardar todas las soluciones que se han realizado, pero como consecuencia de esto se va a crear un almacenamiento excesivo de estos.

En lugar de eso se emplea una memoria de corto plazo que registra ciertos atributos de las soluciones ya revisadas y lo que se busca es evitar que durante cierta cantidad de iteraciones p_u , se tomen en consideración soluciones que posean dichos atributos. El conjunto de soluciones prohibidas se les denomina soluciones tabúes y las movidas que llevan hacia soluciones tabú se llaman movidas tabúes. Usualmente se utiliza un criterio aspiración, para aceptar soluciones aun cuando sean tabú, por ejemplo, si estas minimizan el costo de la mejor solución encontrada hasta el momento. Se les llama soluciones admisibles a aquellas que no son soluciones tabúes y a las que pasan el criterio de aspiración (aun si son tabú). Esta búsqueda se realiza sobre las soluciones admisibles de la vecindad.

4.10.3 Búsqueda Tabú para el Problema de Ruteo de Vehículos

El Algoritmo de Osman. El Algoritmo de Osman (Osman, 1993). Aquellas soluciones vecinas se obtienen a través de un intercambio de clientes entre pares de rutas. Si rp y rq son dos rutas

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

distintas en una solución y se define un λ -intercambio de lo sumo λ clientes de rp a rq . Cuando una ruta envía a otra una cantidad de clientes no necesariamente debe ser la misma, pero ambas deben ser menores que el λ de la vecindad de una solución s , $N\lambda(s)$, consiste en todas las soluciones que se pueden obtener aplicando esta operación a cualquier par de rutas diferentes de s .

4.10.4 Algoritmo genético (*Genetic Algorithm*). El algoritmo voraz, también conocido como devorador, emplea una metodología de exploración, que se rige mediante una heurística compuesta por seleccionar una alternativa óptima en cada zancada local, esto con el propósito de llegar a una solución óptima. Este conjunto de operaciones obedece en gran parte al grupo de los algoritmos constructivos. El proceso llevado a cabo por esta heurística consiste en la construcción por partes de la solución. El algoritmo selecciona en cada iteración la entidad que ofrece mayor beneficio, es decir, prioriza los elementos hasta encontrar una configuración factible (Gallego Rendón et al., 2006). En caso de los problemas de ruteo, el algoritmo seleccionaría el nodo cliente más cercano a la ubicación del vehículo en determinado punto.

4.10.5 Algoritmos Genéticos para el Problema de Ruteo de Vehículos. *Genetic Vehicle Representation* (GVR) (Pereira et al., 2002) presentan un esquema de representación de dos niveles diseñado para tratar de manera efectiva con toda la información que las soluciones candidatas deben codificar y trabajar directamente sobre las soluciones.

Para cruzar dos soluciones p_1 y p_2 , se toma una sub-ruta $r = (v_1, \dots, v_k)$ de p_1 (al tratarse de una sub-ruta, no necesariamente se cumple que $v_1 = 0$ y $v_k = 0$) y se determina el cliente w_j más cercano a v_1 que no está en r . Si la ruta a la que pertenece w en la solución p_2 es $r' = (0, w_1, \dots, w_j, w_j + 1, \dots, 0)$ entonces esta, se reemplaza por $(0, w_1, \dots, w_j, v_1, \dots, v_k, w_j +$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

$1, \dots, 0$). En otras palabras, se inserta r en r' a continuación de w . Si esta ruta no fuera factible, se fracciona en tantas rutas factibles como sea necesario.

Esto genera un hijo, el otro hijo es una copia de p_1 . El cual utilizan cuatro operadores de mutación: el primero es el intercambio de la posición en la que se encuentran actualmente dos clientes en una ruta, el segundo es invertir el orden de una ruta, el tercero es insertar un cliente en una ruta distinta a la que corresponde y último es seleccionar una sub-ruta e insertarla en otro lugar de la solución. Los dos últimos operadores pueden generar nuevas rutas o eliminar rutas, mientras que los dos primeros mantienen la cantidad de rutas de la solución.

Operador de mutación Shift. El operador por desplazamiento (SHIFT mutación) toma un elemento de una posición, lo elimina de esta y lo posiciona en otra, un ejemplo de esto se presenta en la Figura 4. La metodología de este operador empieza generando un número aleatorio que va desde 1 hasta n , donde este representa el tamaño de la población. Dicho número aleatorio proporciona la identificación del cliente que se va a mutar, luego se determina otro número al azar entre 0 y 1, esto con el fin de establecer la probabilidad de mutación, es decir la probabilidad de que el cliente seleccionado sea sometido a la mutación. De esta forma, si la probabilidad asociada al cliente es mayor o igual a la probabilidad de mutación, el cliente es mutado (Ruiz García, 2004).

Las dos fases por las cuales se rige este operador son la siguientes: la fase número uno es la delimitación, en la que se determinan tres números aleatorios con valores entre 0 y n (R1, R2 Y R3). Siendo R1 y R2 el comienzo y el final de bloque de cromosomas a mover, R1 tiene $n - 4$ grados de libertad ya que este está entre $(2, n - 2)$; así mismo, R2 cuenta con $n - 2$ grados de libertad y se encuentra entre valores de $(1, n - 1)$. Por otra parte, en cuanto a R3 este es un número que determina la posición en la cual se realiza el desplazamiento y se encuentra entre $(0, n)$, contando con n grados de libertad. Además, este no está incluido entre R1 y R2. En la fase se

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

presentan dos casos, uno de ellos se da cuando $R3$ sea menor que $R1$ el cual consiste en trasladar el bloque hasta la posición $R3$ y ajustar cada cliente del cromosoma en su lugar relativo; por otro lado, tenemos $R3$ mayor que $R2$, el cual sigue el mismo proceso que el anterior de manera inversa.

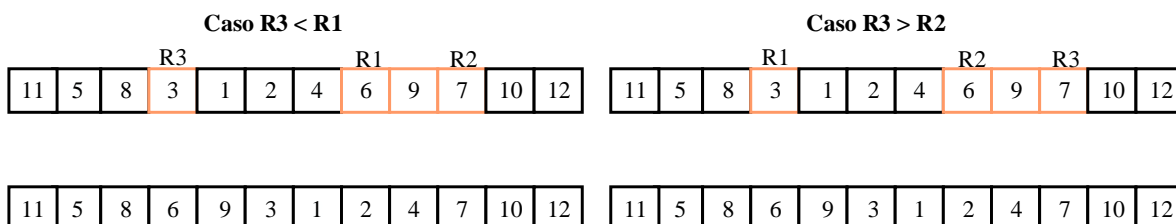


Figura 4. Ejemplo base del operador de mutación Shift para 12 clientes en los que $R3$ es mayor que $R1$ y $R3$ es mayor que $R2$, dando lugar a que se llegue a la misma mutación en ambos casos.

4.10.6 Adaptive Large Neighborhood Search (ALNS). Los problemas de optimización combinatoria como ya se mencionó previamente cuenta con un impedimento, que es la imposibilidad de analizar todas las soluciones factibles en un espacio de tiempo razonable, dado que se cuenta con una enorme cantidad de estas; para resolver esta dificultad se han propuesto los algoritmos de búsqueda de vecindario (*Neighborhood Search* o NS).

Con la finalidad de comprender cómo operan los algoritmos de búsqueda de vecindario, y por consiguiente el ALNS, es indispensable saber a qué hace alusión el concepto de vecindario. Inicialmente se define un vecindario $N(s) \subseteq S(I)$ para cada solución $s \in S(I)$, que contiene resultados similares entre sí y s , el NS permite analizar el vecindario de cada configuración con el objetivo de encontrar elementos que mejoren la función final entre los vecinos cercanos.

Un factor fundamental de la búsqueda de vecindario es el tamaño del vecindario. Aunque resulta más lento buscar en un vecindario grande se podrían conseguir mejores resultados, puesto que, evita que los algoritmos se estancuen en los óptimos locales. Con la finalidad de generar mejores resultados de solución se desarrolló una búsqueda de vecindario amplia (Large Neighborhood Search o LNS) que parte de una solución parcial s , compuesta por unos elementos (nodos y arcos),

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

una vez se encuentra una mejor solución se reemplaza la anterior por esta. En el LNS se emplea en su totalidad dos funciones; una de destrucción para remover determinados elementos y otra de reparación para reemplazarlos por otros que hagan parte del vecindario de la solución s , de manera que se genera una nueva solución s' . Cada vez que el algoritmo encuentra una configuración superior s_{\min} , s debe ser reemplazada. Sin embargo, cuando se acepta una solución s' que no representa una mejora de la función objetivo, entonces s toma la nueva configuración s' y s_{\min} permanece igual. En este algoritmo se presenta un condicional, el cual se basa en decidir si se toma o no una solución alternativa, para ello es obligatorio seleccionar unos criterios de aceptación, algunos de ellos pueden ser: admitir todas las nuevas soluciones, llevarlo a cabo sólo en el caso de que implique una disminución en los costos; el número o cantidad de iteraciones consecutivas, este se elige para permitir la convergencia; definir un límite de permisibilidad que sea adaptable con cada iteración y que restrinja o admita determinada solución s' , entre otros.

El ALNS surge como una variante del LNS, cuya traducción aproximada sería Búsqueda de Vecindario Amplia y Adaptativa (Adaptive Large Neighborhood Search o ALNS), que amplía el marco de búsqueda y difiere del LNS en la cantidad de heurísticas de destrucción y reparación que requiere. En el LNS se seleccionan dos funciones, una para la etapa de destrucción y otra para la fase de reparación, dichas funciones son escogidas por el investigador. La idea básica es buscar una mejor solución en cada iteración destruyendo una parte de la solución actual y reconstruyéndola de una manera diferente. La particularidad de la extensión adaptativa consiste en que en cada iteración se selecciona una heurística de cada set y dependiendo de su desempeño, se ajusta a una función de peso que, a su vez, también afectará las probabilidades de que esa heurística sea elegida nuevamente en las siguientes iteraciones (Lutz, 2014).

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

4.10.7 Algoritmo de K-medias (*K-means clustering*). Gran parte de los algoritmos de partición emplean su fundamento en la optimización de una función, la cual se denota como F , esta a su vez es una variable dependiente de un conjunto de datos $\{C_1, C_2, C_3, \dots, C_K\}$, es decir:

$$F: P_k(X) \rightarrow \mathbb{R}$$

El método de agrupamiento de K-medias emplea la partición de un conjunto de n elementos $X = (x_1, x_2, x_3, \dots, x_n)$, donde cada observación es un vector real de dimensión d del conjunto de datos, la metodología de este, establece k grupos no vacíos, constituidos por elementos cuya media aritmética es la más cercana, siendo $k \leq n$, el propósito de esto es minimizar la suma de los cuadrados (error cuadrático) dentro de cada grupo: $S = \{s_1, s_2, s_3, \dots, s_k\}$, la agrupación se representa mediante un espacio de datos en celdas de Voronoi, el pseudocódigo de este se presenta en la Figura 5. La forma estándar del algoritmo utiliza una técnica de refinamiento iterativo, utilizando como criterio F , esto asume que dado un conjunto inicial de k grupos con centroides $(C_1, C_2, C_3, \dots, C_k)$, p_i el número de objetos en el grupo i , x_{ij} como el j –ésimo objeto de i –ésimo y \bar{C}_i el valor medio de los centroides, el algoritmo puede expresarse de la siguiente manera:

$$F(C_1, C_2, C_3, \dots, C_K) = \sum_{i=1}^K \sum_{j=1}^{p_i} \|x_{ij} - \bar{C}_i\|^2$$

Paso 1. Seleccionar los K centroides iniciales $\{C_1, C_2, C_3, \dots, C_K\}$
 Paso 2. Asignar los objetos x_i del conjunto de datos X , a su centroide más cercano
 Paso 3. Recalcular los nuevos centros, regresar al paso anterior, hasta que el algoritmo converja

Figura 5. Pseudocódigo del algoritmo K-means. Adaptado de (Lara et al., 2016)

5. Formulación del modelo matemático

5.1 Descripción del modelo

El problema de distribución propuesto consta de dos subproblemas a resolver sugerido por Murray y Chu (2015). El primero, consiste en la necesidad de trazar una ruta que optimizara el uso de una flota heterogénea que llevara a cabo la repartición de bienes y servicios encaminados a la distribución de última milla; el segundo, en determinar cuál ruta reduce los costos operacionales y cumple con restricciones como el tiempo máximo de la ruta.

Para el primer caso, en el trazado de la ruta, es necesario considerar las particularidades de una flota que está compuesta por un conjunto de vehículos y UAVs, entre ellas se tiene que el camión de reparto funciona bajo el mando de un conductor y debe estar equipado por un dron, ambos con la misma misión de entregar paquetes a los clientes, los cuales son servidos una sola vez, ya sea por un camión o por un dron; además, cada camión transporta su correspondiente dron y este debe partir y regresar a un único depósito, el retorno puede ser en tándem o de manera independiente. En los momentos en los que el dron no realiza ninguna entrega, permanecerá en el camión para ahorrar energía de la batería, en cambio cuando el dron está en operación es lanzado por el vehículo desde un nodo i y recogido por este en varios nodos k a lo largo del recorrido hecho por cada uno de los arcos de la ruta trazada. En el modelo se considera que el dron solo puede atender un cliente a la vez, debido a que la capacidad de carga útil del dron no es ilimitada y que el peso de determinados paquetes solo puede ser transportado por los vehículos, puesto que los drones no cuentan con la fuerza idónea para llevar productos que excedan su capacidad de elevación. Así mismo, se tiene un tiempo asociado al lanzamiento y recuperación del dron; igualmente, se considera un tiempo de servicio para entregar los paquetes a los clientes. Por otra parte, los vehículos tienen una restricción vinculada, a fin de que no se exceda un límite de tiempo de

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

operación por día. Una representación visual de una solución al problema se presenta en la Figura

6.

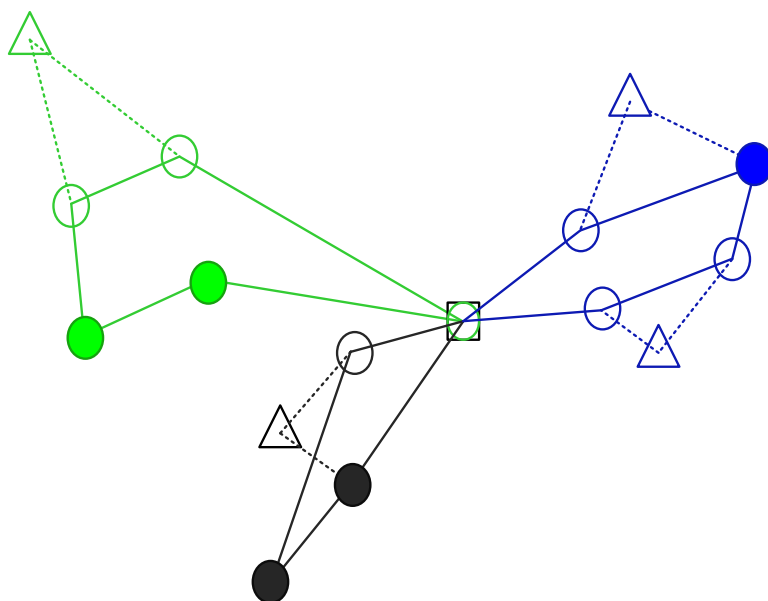


Figura 6. Ejemplo de una ruta entre camiones y drones, donde los círculos hacen referencia a sitios que visita el vehículo y los triángulos son clientes que pueden ser servidos por drones o vehículos. Adaptado de Sacramento et al.(2019)

El segundo problema consiste en determinar cuál ruta reduce los costos operacionales y cumple con las restricciones del problema; se consideran los siguientes aspectos: el tiempo máximo de la ruta, que hace referencia al límite de horas laborales por día estipulado en el contrato de los trabajadores; el costo operativo del vehículo, el cual hace alusión al consumo de combustible usado para atravesar los arcos de la ruta trazada; el costo total de operación de los drones, que se compone por el costo de lanzamiento y recuperación del dron, ambos generados por el tiempo que debe esperar el vehículo para que el dron despegue y aterrice. Como último aspecto se tiene en cuenta la velocidad del dron, en la cual se considera la influencia del viento y masa del paquete, esto permite que se pueda calcular el tiempo necesario para desplazarse de un tramo a otro. Adicionalmente, para la velocidad del dron se tienen en cuenta factores como la variación que

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

puede tener este a la hora de estar cargado o descargado y cómo influye el viento en su desplazamiento. Los factores previamente mencionados se consideran mediante el planteamiento expuesto en las ecuaciones de velocidad del dron.

5.2 Supuestos

Se suponen las siguientes condiciones de funcionamiento para el modelo matemático propuesto en (Murray & Chu, 2015):

- Flota homogénea ilimitada de vehículos de reparto, asumiendo que no hay distinciones específicas entre ellos.
- Aunque el UAV solo puede visitar un cliente por salida, el camión puede visitar a varios clientes mientras el UAV está en vuelo.
- Se supone que el UAV permanece en vuelo constante mientras está en una salida, excepto para entregar el paquete a un cliente. Por lo tanto, al coordinar el regreso al camión, el UAV no puede aterrizar temporalmente mientras está en ruta para conservar la energía de la batería, en caso de que el UAV llegue antes que el camión.
- Si el vehículo recolecta el UAV en algún nodo del cliente i , el UAV se puede lanzar nuevamente desde i . Sin embargo, si el UAV se inicia desde i es posible que no regrese al camión en el nodo i .
- Si el tramo final de una salida de los UAVs implica una recogida por parte del vehículo, esto debe tener lugar en la ubicación k de un cliente atendido por el vehículo; el UAV no puede reconectarse con el vehículo en alguna ubicación intermedia. Además, el vehículo no puede regresar sobre su ruta a visitar ningún nodo de cliente para recuperar el UAV.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Ni el UAV ni el vehículo pueden visitar un nodo que no sea un nodo cliente (aparte del depósito, por supuesto). Además, ninguno de los vehículos puede volver a visitar a un cliente en el recorrido trazado.
- La velocidad del viento se tomará como un promedio de las velocidades en un intervalo de tiempo de un mes, este dato será suministrado por medio de una fuente secundaria IDEAM (Instituto de Hidrología, Meteorología y Estudios Ambientales).
- La velocidad de la ficha del dron está determinada por las especificaciones dadas por el fabricante.
- No está permitido que un UAV opere de manera independiente del camión, es decir, comience la operación desde el depósito, entregue los productos al cliente y luego regrese al depósito.
- El vehículo de reparto no puede esperar al dron en la misma ubicación desde la que lo lanzó.
- Debido a que el UAV no requiere llegar de manera simultánea con el vehículo de reparto, en el caso de que un recorrido del UAV termine en el depósito, este queda fuera de servicio, es decir, no puede lanzarse desde el depósito.
- No se tienen en cuenta los costos cesantes por daños o pérdidas en mercancías y drones, tampoco los costos asociados a penalizaciones por la no entrega de los paquetes, ni los costos generados por las personas que operan los vehículos.

La siguiente formulación matemática es una extensión a la formulación del modelo matemático propuesto por (Sacramento et al., 2018).

Índices

i = Nodo de partida, $i = 1, 2, 3 \dots, n$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

j = Nodo de llegada, $j = 1, 2, 3 \dots, n$

k = Nodo de recuperación, $k = 1, 2, 3 \dots, n$

n = Número de nodos

v = Número de vehículos

d = Número de drones

s = Número de tuplas u ordenes en las que el dron puede salir (i, j, k)

Listado de conjuntos

C = Conjunto de clientes

$c \{1, 2, 3 \dots, c\}$

$C' \subseteq C$ = Subconjunto de clientes que pueden ser visitados por los drones

$\{1, 2, 3 \dots c\}$

V = Conjunto de vehículos

$v \{1, 2, 3 \dots, V\}$

D = Conjunto de drones "UAV's"

$d \{1, 2, 3 \dots, D\}$

N = Conjunto de todos los nodos incluyendo el depósito cuando sea 0 $\{1, 2, 3 \dots, c, c + 1\}$

N_0 = Conjunto de nodos desde el cual un vehículo puede partir

$\{1, 2, 3 \dots, c\}$

N_* = Conjunto de nodos al cual un vehículo puede llegar $\{1, 2, 3 \dots, c, c + 1\}$

$\Delta^+(i)$ = El conjunto de nodos que pueden ser alcanzados desde el nodo $i \in N_0 = N_*$ $\{i\}$

$\Delta^-(i)$ = El conjunto de nodos que pueden ser usados para alcanzar el nodo $i \in N_* = N_0$ $\{i\}$

A = Conjunto de todos los arcos $\{(i, j): i \in N_0, j \in$

$\Delta^+(i)\}$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

La formulación matemática del VRPD requiere varios parámetros que describen las características y los tiempos de operación relacionados con el vehículo y el UAV, se buscan valores cercanos a la realidad y que sean aplicable a este modelo. Los parámetros requeridos para la formulación matemática se exponen a continuación:

Parámetros determinísticos

q = Capacidad del vehículo

[Kg]

d_i = Demanda del cliente $i \in C$

[Kg]

W = Capacidad de carga del dron

[Kg]

e = Capacidad de la batería dada en tiempo de vuelo del dron

[min]

A_{ij} = Distancia requerida para ir desde el nodo $i \in N_0$ hasta el nodo $j \in N_*$

[Km]

t_{ij}^v = Tiempo requerido por un vehículo para viajar desde $i \in N_0$ a $j \in N_*$

[H]

t_{ij}^d = Tiempo requerido por un "UAV" para viajar desde $i \in N_0$ a $j \in N_*$

[H]

C_{ij}^v = Costo de operar un vehículo para viajar desde $i \in N_0$ a $j \in N_*$

[\$]

C_{ij}^d = Costo de operar un "UAV" para viajar desde $i \in N_0$ a $j \in N_*$

[\$]

M = Un número suficientemente grande

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Se_i^v = Tiempo de servicio del vehiculo al cliente $i \in C$

[*min*]

Se_i^d = Tiempo de servicio para el UAV al cliente $i \in C'$

[*min*]

SL = Tiempo requerido para lanzar el UAV

[*min*]

SR = Tiempo requerido para recuperar el UAV

[*min*]

T_{max} = Tiempo máximo de duración de una ruta

[*min*]

T_{max}^d = Tiempo máximo de duración de una ruta para el dron

[*min*]

El C_{ij}^v se definió como el producto entre la distancia recorrida, el consumo y el precio de combustible. El costo del dron se estableció como un factor α del costo del vehículo. Adicionalmente, fue necesaria una nomenclatura complementaria para identificar los posibles puntos de partida o salida de un dron. Una salida del dron se compone por tres nodos de la ruta desde los cuales se puede operar, siendo P el conjunto de posibles salidas, representadas por las tuplas $\langle i, j, k \rangle$. En donde el primer nodo (i) representa la posición de lanzamiento del UAV, el segundo nodo (j) representa al cliente que es visitado por el UAV y, finalmente, el tercer nodo (k) representa la posición de recuperación del UAV por parte del vehículo. Un elemento $\langle i, j, k \rangle$, pertenece al conjunto P si se obedecen las siguientes condiciones:

- Posición del nodo de lanzamiento $i \in N_0$ de una tupla: es la ubicación desde la cual se puede lanzar un UAV y puede salir un camión.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Posición del nodo de entrega $j \in \{C': j \neq i\}$ de una tupla: es el conjunto de clientes que pueden ser atendidos por el UAV diferente de la posición de lanzamiento i .
- Posición del nodo de encuentro o de recuperación $k \in \{N_* : k \neq i, k \neq j, SL + SR + \tau_{ij}^D + \tau_{jk}^D + Se_j^D \leq e\}$ de una tupla: es la ubicación en la que el camión puede recuperar el UAV respetando la duración de la batería.

Conjuntamente, las tuplas $\langle 0, i, c + 1 \rangle$ se encuentran excluidas del conjunto P para todo $i \in C'$.

Dichas tuplas hacen referencia a las entregas que deben realizar los drones, donde sus posiciones de lanzamiento y recuperación son en el depósito. Para una salida $s = \langle i, j, k \rangle$, se define el costo como $C_s^D = C_{ij}^D + C_{jk}^D$. Adicionalmente, se establece a P_i^+ como todas las salidas de P lanzadas desde el nodo $i \in N_0$, P_k^- como todas las salidas de P con un nodo de recuperación $k \in N_*$ y P_j como todas las salidas de P que se entregan al cliente $j \in C'$.

Variables

u_i^v : Indica la posición de la visita $i \in N$ en la ruta del vehículo $v \in V$ [x, y]

τ_i^v : Indica el tiempo en la ruta del vehículo $v \in V$ llegando a la ubicación $i \in N$

[*min*]

τ_i^D : Indica el tiempo de un dron desde el vehículo $v \in V$ llegando a la ubicación $i \in N$

[*min*]

p_{ij}^v : Indica si un cliente $j \in C$ se visita después de la ubicación $i \in N_0$ en la ruta del camión

$x_{ij}^v : \begin{cases} 1, & \text{si el vehículo } v \in V \text{ viaja desde } i \in N_0 \text{ a } j \in N_* \\ 0, & \text{en otros casos} \end{cases}$

$y_s^v : \begin{cases} 1, & \text{si la salida } s \in P \text{ es usada en la ruta del vehículo } v \in V \\ 0, & \text{en otros casos} \end{cases}$

$z_{ij}^v : \begin{cases} 1, & \text{si la salida } j \in C \text{ es visitada después de la ubicación } i \in N_0 \text{ en la ruta del vehículo } v \in V \\ 0, & \text{en otros casos} \end{cases}$

Función objetivo

$$\text{Costo operacional de los vehículos} = \left\{ \sum c_{ij}^v x_{ij}^v \right\} \quad [1]$$

$$\text{Costo operacional de los drones "UAV's"} = \left\{ \sum c_s^D = \sum c_{ij}^D + \sum c_{jk}^D \right\} \quad [2]$$

$$1. \quad F.O. = \text{Min} \{ \text{Costo operacional de los vehículos} + \text{Costo operacional de los drones} \}$$

$$\text{Min } C = \sum_{v \in V} \left(\sum_{(i,j) \in A} c_{ij}^v x_{ij}^v + \sum_{s \in P} c_s^D y_s^v \right) \quad (1)$$

La Función objetivo (1) minimiza el costo operativo de la visita a los clientes, esta se compone de dos partes: el costo de operación de la flota de vehículos [1] y el costo de operación de la flota de drones [2], utilizados para la distribución de los recursos. Para el caso de los drones se consideran dos costos: el primero, es el costo de operación desde la ubicación i , en que se puede lanzar un UAV hasta la ubicación de entrega de un cliente j . El segundo, es el costo de operación desde la ubicación del cliente j hasta el nodo de recolección k del UAV. La suma de los dos costos anteriores representa el costo de operación total de los drones y, a su vez, con la suma de los costos [1] y [2] se obtiene el costo total del recorrido de vehículos y drones trazado en la ruta.

Es importante destacar que, a diferencia de lo presentado en el modelo propuesto por Sacramento et al. (2018) donde la velocidad del dron es un valor fijo, en la presente investigación se propone una ecuación para la velocidad, la cual considera el factor de carga del dron y la velocidad promedio del viento. Así mismo, se consultó al piloto de drones Santiago Cruz, adscrito al grupo de investigación SEPRO de la Universidad Nacional de Colombia (Sede Bogotá) el cual mencionó que, en vuelo real, la batería del dron no debe estar por debajo del 30% ya que puede ocasionar caídas repentinas o deterioro en el funcionamiento del dron, por esta razón se incorpora

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

una restricción sobre la carga energética del dron, que limita su funcionamiento si la capacidad de su batería (dada en tiempo de vuelo), se encuentra por debajo del 30% de la capacidad total. A continuación, se presentan las ecuaciones de velocidad del dron cuando lleva un paquete y cuando no lo está haciendo.

Ecuaciones de velocidad del dron:

$$V_{dc} = c_m(V_{fd}) - \bar{v}_v \quad (2)$$

$$V_{dd} = (V_{fd}) - \bar{v}_v \quad (3)$$

Donde:

$$V_{dc} = \text{Velocidad del dron cuando lleva un paquete o carga} \quad [m/s]$$

$$V_{dd} = \text{Velocidad del dron cuando no lleva un paquete o carga} \quad [m/s]$$

$$c_m = \text{Factor de carga, } c_m = \frac{m_t + m_b}{m_t + m_b + m_l} < 1$$

$$m_t = \text{Masa del dron sin bateria} \quad [Kg]$$

$$m_b = \text{Masa de la bateria del dron} \quad [Kg]$$

$$m_l = \text{Masa de la carga o paquete en el dron} \quad [Kg]$$

$$V_{fd} = \text{Velocidad de la ficha del dron} \quad [m/s]$$

$$\bar{v}_v = \text{Velocidad promedio del viento, } \bar{v}_v = \frac{\sum V_v}{\# \text{ velocidades}} \quad [m/s]$$

El factor de carga está en función de la masa del dron sin batería, la masa de la batería del dron y la masa de la carga (paquete) que se lleva desde un nodo de lanzamiento i hasta un nodo cliente j , dicha ecuación fue tomada de (Figliozzi, 2017).

Las restricciones a las que se encuentra sujetas la función objetivo del problema son las siguientes:

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

$$\sum_{v \in V} \sum_{i \in \Delta^-(j)} x_{ij}^v + \sum_{v \in V} \sum_{s \in P_j} y_s^v = 1 \quad j \in C \quad (4)$$

$$\sum_{j \in N_*} x_{0,j}^v \leq 1 \quad v \in V \quad (5)$$

$$\sum_{i \in N_0} x_{i,c+1}^v \leq 1 \quad v \in V \quad (6)$$

$$x_{0,c+1}^v = 0 \quad v \in V \quad (7)$$

$$\sum_{i \in \Delta^-(j)} x_{ij}^v = \sum_{k \in \Delta^+(j)} x_{jk}^v \quad v \in V, j \in C \quad (8)$$

$$u_i^v + 1 \leq u_j^v + M(1 - x_{ij}^v) \quad v \in V, (i, j) \in A \quad (9)$$

$$u_j^v \leq M \left(\sum_{i \in \Delta^-(j)} x_{ij}^v \right) \quad v \in V, j \in N_* \quad (10)$$

$$\sum_{j \in C} \left(\sum_{k \in \Delta^+(j)} d_j x_{jk}^v + \sum_{s \in P_j} d_j y_s^v \right) \leq q \quad v \in V \quad (11)$$

$$\sum_{s \in P_i^+} y_s^v \leq 1 \quad v \in V, i \in N_0 \quad (12)$$

$$\sum_{s \in P_k^-} y_s^v \leq 1 \quad v \in V, k \in N_* \quad (13)$$

$$2y_s^v \leq \sum_{h \in \Delta^+(i)} x_{ih}^v + \sum_{l \in \Delta^-(k)} x_{lk}^v \quad v \in V, s = \langle i, j, k \rangle \in P \quad (14)$$

$$\tau_i^v = 0 \quad v \in V \quad (15)$$

$$\tau_i^D = 0 \quad v \in V \quad (16)$$

$$\tau_{c+1}^v \leq T_{max} \sum_{i \in N_0} x_{i,c+1}^v \quad v \in V \quad (17)$$

$$\tau_{c+1}^D \leq T_{max} \sum_{s \in P_{c+1}^-} y_s^v \quad v \in V \quad (18)$$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

$$\tau_h^v + t_{hk}^v + Se_h^T + SL \sum_{s \in P^+(h)} y_s^v + SR \sum_{s \in P^-(k)} y_s^v \leq \tau_k^v + T_{max}(1 - x_{hk}^v) \quad (19)$$

$$v \in V(h, k \in A)$$

$$\tau_i^v + t_{ij}^d + SL - T_{max} \left(1 - \sum_{s \in P_i^+ \cap P_j} y_s^v \right) \leq t_j^d \quad v \in V, (i, j) \in A \quad (20)$$

$$\tau_i^d + t_{ij}^d + Se_j^d + SR - T_{max} \left(1 - \sum_{s \in P_j \cap P_k^-} y_s^v \right) \leq t_k^d \quad v \in V, j \in C', k \in \Delta^+(j) \quad (21)$$

$$\tau_i^v - T_{max} \left(1 - \sum_{s \in \cap P_i^+} y_s^v \right) \leq \tau_i^D \quad v \in V, i \in N_0 \quad (22)$$

$$\tau_i^v + T_{max} \left(1 - \sum_{s \in \cap P_i^+} y_s^v \right) \geq \tau_i^D \quad v \in V, i \in N_0 \quad (23)$$

$$\tau_k^v - T_{max} \left(1 - \sum_{s \in P_k^-} y_s^v \right) \leq \tau_k^D \quad v \in V, k \in C \quad (24)$$

$$\tau_k^v + T_{max} \left(1 - \sum_{s \in P_k^-} y_s^v \right) \geq \tau_k^D \quad v \in V, k \in C \quad (25)$$

$$T_{max}^d \leq 0,7 (e) \quad (26)$$

$$e + T_{max} \left(1 - \sum_{s \in P_i^+ \cap P_k^-} y_s^v \right) \geq \tau_k^D - \tau_i^D \quad v \in V, i \in N_0, k \in N_* \quad (27)$$

$$(u_j^v - u_i^v) \leq M (p_{ij}^v) \quad v \in V, i \in N_0, j \in C \setminus \{i\} \quad (28)$$

$$(u_j^v - u_i^v) \geq M (p_{ij}^v - 1) + 1 \quad v \in V, i \in N_0, j \in C \setminus \{i\} \quad (29)$$

$$\tau_k^D - T_{max} \left(3 - \sum_{s \in (P_i^+ \cap P_k^-)} y_s^v - \sum_{s \in P_b^+} y_s^v - p_{ib}^v \right) \leq \tau_b^D \quad (30)$$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

$$v \in V, i \in N_0, k \in N_*, b \in C \setminus \{b\}$$

$$x_{ij}^v \in \{0,1\} \quad v \in V, (i,j) \in A \quad (31)$$

$$y_s^v \in \{0,1\} \quad v \in V, s \in P \quad (32)$$

$$u_i^v, \tau_i^v, \tau_i^D \geq 0 \quad v \in V, i \in N \quad (33)$$

$$p_{ij}^v \in \{0,1\} \quad v \in V, i \in N_0, j \in C \setminus \{i\} \quad (34)$$

La restricción (4) asegura que cada cliente sea visitado exactamente una vez, ya sea por un vehículo o por un avión no tripulado. La restricción (5) asegura que todos los camiones deben salir del depósito como máximo una vez. De manera similar, en (6) se garantiza que todos los camiones deben regresar al depósito como máximo una vez. Además, está prohibido viajar entre depósitos según lo dado por (7). Las restricciones de conservación de flujo para el vehículo se definen en (8). Las restricciones (9) y (10), se refieren a la eliminación de recorridos secundarios para el vehículo en relación con la posición de un cliente visitado en la ruta del vehículo. También, la restricción de capacidad de los vehículos y drones se da en (11). Igualmente, el UAV puede ser lanzado y recuperado como máximo una vez desde cada nodo, como se indica en las restricciones (12) y (13). La restricción (14) asegura que si el UAV se lanza y recupera en la ubicación $i \in N_0$ y $k \in N_*$, el vehículo visita las mismas ubicaciones, respectivamente. La inicialización de los tiempos para el vehículo y el UAV al comienzo de cada ruta se dan en (15) y (16). Así mismo, la duración máxima de una ruta se establece imponiendo un límite al tiempo de regreso al depósito por los vehículos, como se indica en (17) y (18). Las limitaciones de tiempo para el movimiento del vehículo (19), definen el momento en que el vehículo llega a la ubicación con respecto a las acciones correspondientes que pueden ocurrir en el medio. Las limitaciones de tiempo para el movimiento del dron establecen el tiempo en que un UAV visita un cliente según la posición del

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

vehículo estas se definen en (20) y (21). La restricción (20) asegura que, si se lanza un UAV desde la ubicación $i \in N_0$ al cliente $j \in C'$, entonces el tiempo de llegada del UAV al cliente j debe ser mayor que el tiempo de llegada del vehículo a la ubicación i más el tiempo de viaje del UAV entre la ubicación y el tiempo de lanzamiento del UAV. De manera similar para las operaciones de recuperación, la restricción (21) asegura que si un UAV termina de atender a un cliente $j \in C'$ y vuela de regreso al vehículo en la ubicación $k \in N_*$, la suma de los tiempos de llegada del UAV a la ubicación j , el tiempo de viaje entre las ubicaciones $j - k$, el de servicio del UAV en el nodo cliente j y el tiempo necesario en la recuperación de este; debe ser menor que el tiempo de llegada del UAV a la ubicación k .

Las restricciones de sincronización de tiempo para el vehículo y el UAV se definen en (22)-(25). Estas restricciones imponen que el lanzamiento y las operaciones de recuperación estén sincronizadas en el tiempo. Teniendo en cuenta que la sincronización no es necesaria cuando el vehículo y el UAV están regresando por separado al depósito, por ende, las restricciones (24) y (25) se definen para el conjunto de clientes C en lugar de N_* . La restricción de resistencia para la batería del UAV viene dada por las variables de lanzamiento y recuperación. El tiempo máximo de vuelo de un dron en una ruta, no puede estar por debajo del 30% de la capacidad de su batería, esto se asegura con la restricción (26). La restricción (27) asegura que si se lanza un UAV desde la posición $i \in N_0$ al cliente $j \in C$ para recuperarlo en la posición $k \in N_*$, entonces la diferencia de tiempo entre las operaciones debe satisfacer el tiempo de la duración de la batería del dron.

La variable binaria p_{ij}^v define el orden en que el vehículo realiza las visitas, estableciendo que un cliente es visitado antes que otro en la ruta. El valor de esta variable se asigna de acuerdo con la diferencia de posiciones entre la ubicación del cliente y el punto de partida del dron como se indica en (28) y (29). Suponiendo la ubicación de salida $i \in N_0$ y el cliente $j \in C$, si el vehículo

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

visita al cliente j después de estar en la ubicación i , entonces la diferencia entre las variables de posición u_i^v será positiva, imponiendo el valor uno a p_{ij}^v . Por el contrario, si la diferencia es negativa, p_{ij}^v se impone el valor de cero. Cuando un vehículo no visita a un cliente en la ruta, establecido en las restricciones (9) y (10) se define la posición de la visita como cero. Por lo tanto, en la restricción (29) se agrega un 1 en el lado derecho para garantizar que p_{ij}^v permanezca en cero cuando tanto u_i^v como u_j^v tomen el valor de cero. La restricción (30) fue definida para evitar que ocurrieran nuevos lanzamientos mientras el UAV estaba volando en la ruta. La hora de llegada de un UAV a otra ubicación del cliente $b \in C$ desde el cual el UAV puede ser lanzado nuevamente tiene que ser mayor que la hora de llegada a la ubicación $k \in N_*$, sí y solo si un dron se recupera en la ubicación k . Esto se tiene en cuenta en el orden en que el vehículo visita los lugares i y b (p_{ij}^v), si se produce un lanzamiento desde la posición b y si un avión aéreo no tripulado se recupera en la ubicación k . Finalmente, el dominio de las variables se define en las restricciones (31) - (34).

6. Diseño del ALNS para la solución del VRPD

El modelo matemático formulado se resolverá mediante el uso del algoritmo *Adaptive Large Neighborhood Search*, el cual está basado en los mecanismos de búsqueda de vecindario, introducidos por primera vez en 1998 por Shaw con el LNS, el procedimiento que este sigue se presenta en la Figura 7. Este algoritmo consiste en mejorar progresivamente una solución inicial destruyendo y reparando reiterativamente la solución parcial (Shaw, 1998).

En este trabajo se utilizó “*The Adaptive Large Neighborhood Search*” (ALNS) cuya traducción aproximada al español sería Búsqueda Adaptativa de Vecindario Extenso, presentado por Ropke y Pisinger (2006), siendo esta una extensión de la búsqueda de gran vecindario (LNS). A diferencia del LNS, el ALNS usa múltiples heurísticas en lugar de solo una, siendo una de las características

UN ALGORITMO ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

a destacar de este algoritmo. Este comienza su proceso con una solución inicial S y en cada iteración se elige la mejor solución basada estadísticamente en el éxito pasado de la heurística, luego se aplica una heurística de destrucción y finalmente una de inserción o reparación. Los métodos de destrucción se encargan de eliminar una fracción de la solución actual, mientras que los de reparación reconstruyen la solución parcial (Lutz, 2014). Por lo general, los métodos de destrucción tienen un componente aleatorio con el que son capaces de destruir diversas partes de la solución y de esta forma diversificar la búsqueda de nuevas soluciones. Por otro lado, los métodos de reparación pueden ser no determinísticos, evitando construir una solución idéntica si lo que se busca es no encontrarse con la misma solución parcial varias veces, otorgando la posibilidad de desarrollar cambios sustanciales y no quedarse estancado en los óptimos locales a lo largo de la búsqueda. Un parámetro muy importante para el algoritmo es la tasa de destrucción, esta debe seleccionarse de forma adecuada, puesto que si es muy pequeña se dificulta escapar de mínimos locales, por el contrario, si esta es muy grande puede dificultar la etapa de reparación para construir una buena solución. Para una mejor comprensión de la metodología del algoritmo ALNS, los Pseudocódigos presentados en este capítulo se encuentran disponibles en español en el Apéndice B. Por otra parte, en el Apéndice C se encuentra el código de Python empleado en el desarrollo del algoritmo presentando a continuación.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

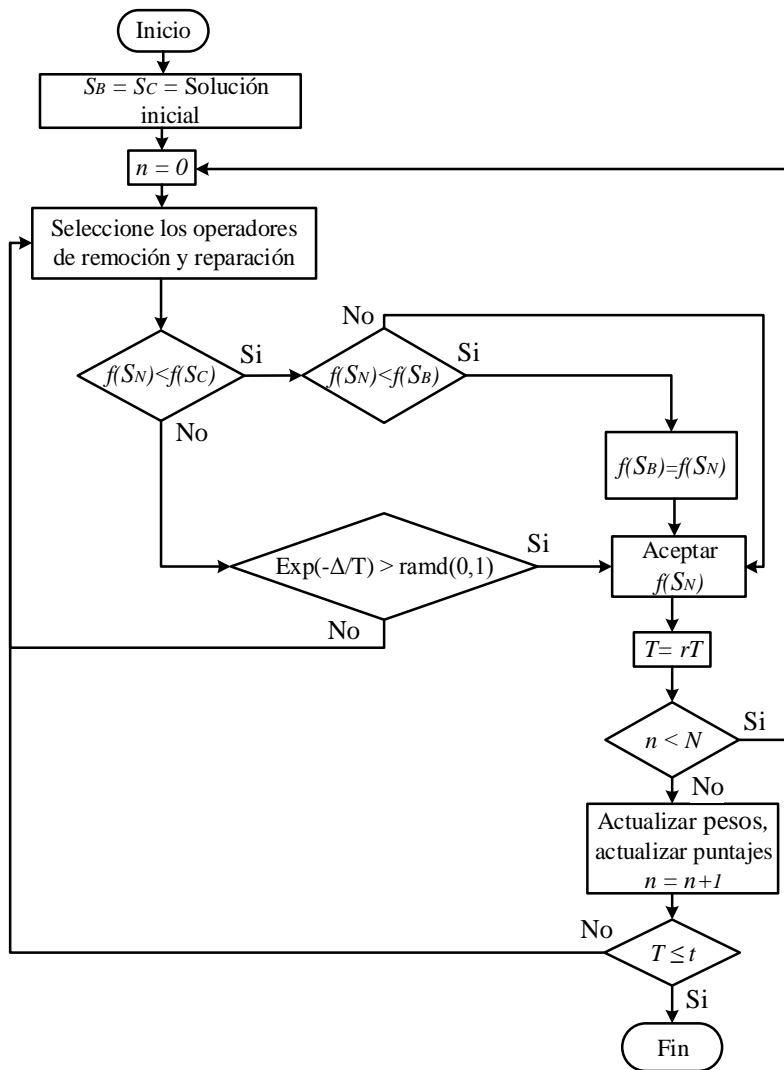


Figura 7. Metodología de algoritmo ALNS tomando como criterio de aceptación el recocido simulado. Las notaciones establecidas son las siguientes: t la temperatura más baja; r la velocidad de enfriamiento; N el número de búsquedas a una temperatura determinada; $f(S_B)$, $f(S_C)$, $f(S_N)$ son los costos de la mejor, la inicial y la parcial solución, respectivamente. Adaptado de (Li et al., 2019)

En comparación con muchas búsquedas locales mediante heurísticas que solo aplican cambios muy pequeños y poco significativos a una solución, el ALNS trabaja con un mayor espacio de búsqueda, el llamado vecindario de la solución actual. Dentro de una iteración, el algoritmo puede modificar entre el 30 - 40% de una solución (Lutz, 2014). Naturalmente, las diferentes heurísticas con sus distintos éxitos manejan diversas instancias de problemas e incluso diferentes soluciones para el mismo problema. Con frecuencia puede resultar complejo predecir cuál heurística será la

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

más adecuada, por esta razón, el ALNS le da la posibilidad al investigador de seleccionar tantas heurísticas como este lo requiera; una vez se haga esto, el algoritmo asignará un peso que refleja el éxito de cada heurística con relación a la función objetivo, al iniciar el proceso del ALNS estos pesos son iguales para todas las heurísticas y los pesos pueden ser ajustados en cada iteración o periódicamente, un período de actualización está compuesto de iteraciones p_u ; de las heurísticas disponibles se seleccionan dos, una para hacer la destrucción y otra para la reparación, el algoritmo termina su ejecución con el criterio de parada. En la figura 8 se expone una representación visual de los procedimientos que sigue el algoritmo ALNS.

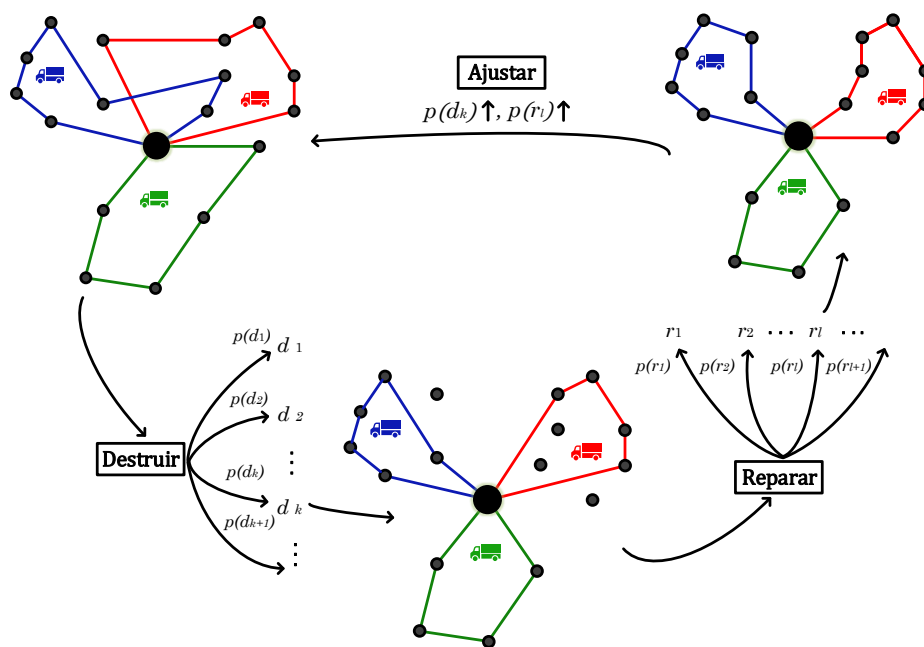


Figura 8. Procedimientos del algoritmo ALNS. Una solución inicial atraviesa por un proceso de destrucción y reparación. Tanto la heurística de destrucción como la de reparación se eligen de un conjunto de heurísticas. Las probabilidades de selección se basan en su éxito después de varias iteraciones, las probabilidades se ajustan según el éxito de la heurística. Adaptado de Lutz (2014).

6.1 Generación de la solución inicial

El problema de la gestión y repartición de productos inicia desde el momento en el que se envían paquetes desde un depósito y culmina cuando llega a manos de sus consumidores finales. Esto es un desafío de gran envergadura y desempeña un rol importante en la administración de los sistemas

UN ALGORITMO ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

logísticos, su idónea planificación puede significar considerables ahorros en los costos de cualquier compañía (Toth, P., Vigo, 2000). Dichos ahorros son potenciales y justifican en gran medida llevar a cabo técnicas de Investigación Operativa para facilitar su planificación, debido a que se estima que los costos del transporte representan un porcentaje importante del costo final de los bienes. En ese sentido, se ha visto un enorme esfuerzo en las últimas cuatro décadas por resolver estos problemas, empezando en 1959 con Dantzig y Ramser (1959) quienes materializaron por primera vez una formulación matemática del problema con una aplicación en la distribución de combustible. Luego, cinco años más tarde, Clarke y Wright (1964) presentaron el primer algoritmo que resultó efectivo para su resolución denominado algoritmo de ahorros. A partir de estos trabajos, el área de Ruteo de Vehículos ha ascendido de manera explosiva; puesto que, se buscan modelos que incorporen más características de la realidad y, por otro lado, la búsqueda de algoritmos que permitan resolver de manera eficiente dichos modelos.

El algoritmo ALNS comienza su funcionamiento determinando una solución inicial “s” por medio de heurísticas que buscan soluciones primarias, que se pueden destacar debido a la facilidad para generar soluciones factibles. Entre las más usadas en el ALNS se encuentran la de construcción voraz (Emeç et al., 2016), la de vecino más cercano (Sacramento et al., 2019) y la de Clarke y Wright (Schermer et al., 2019). En esta investigación la solución inicial se dividió en dos pasos, el primero enfocado a la creación de las rutas para los vehículos haciendo uso de la heurística de Clarke y Wright en su forma básica, y de la extensión propuesta por Paessens (1988) en la que se incluyen los parámetros de λ y μ ; adicionalmente, a esta última se le aplica un operador por desplazamiento (SHIFT mutación) con el fin de intentar mejorar la solución inicial. En el segundo paso para la construcción de la solución inicial, se realiza la inserción de los drones a las

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

rutas creadas mediante el algoritmo de inserción de drones, un ejemplo en el cual se desarrolla el Algoritmo Clarke y Wright en su versión normal y extendida se presenta en el Apéndice D.

6.1.1 Algoritmo de ahorros de Clarke y Wright

El algoritmo de Ahorros se ha empleado en un sinnúmero de aplicaciones reales, enfocado siempre en resolver los Problemas de Ruteo de Vehículos Capacitados. En el artículo de Faulin (2003) "*Applying MIXALG procedure in a routing problem to optimize food product delivery*", presenta su aplicación tomando como ejemplo un caso logístico real de distribución de productos vegetales en Navarra, España. De la misma manera, el artículo de Rand (2009) "*The life and times of the Savings Method for Vehicle Routing Problems*" se dan a conocer las aplicaciones reales y softwares comerciales que se han desarrollado basándose en este método. Por esta razón se decidió utilizar dicho algoritmo.

El Algoritmo de ahorros de Clarke y Wright (CWA) desarrolla un procedimiento iterativo que permite encontrar una ruta parcial buena o casi óptima de manera rápida o en un tiempo de cálculo moderado. Este se preocupa por el enrutamiento inicial de suministros, los cuales se llevan a cabo desde un depósito central a una gran cantidad de puntos de entrega que, a través de una búsqueda en un espacio delimitado, logra encontrar los ahorros más representativos entre dos puntos. Dichas entregas suelen realizarse con una flota de camiones; sin embargo, estas también pueden ser despachadas con drones o en conjunto (camiones y drones). El proceso que sigue el algoritmo se basa en hallar los ahorros de distancia entre cada par de clientes, seguidamente se procede a ordenarlos de mayor a menor, posteriormente se va trazando la ruta entre dos puntos en el sistema, respetando siempre que la suma de las cargas asignadas sea por lo menos igual a la capacidad del vehículo transportador de mercancía.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

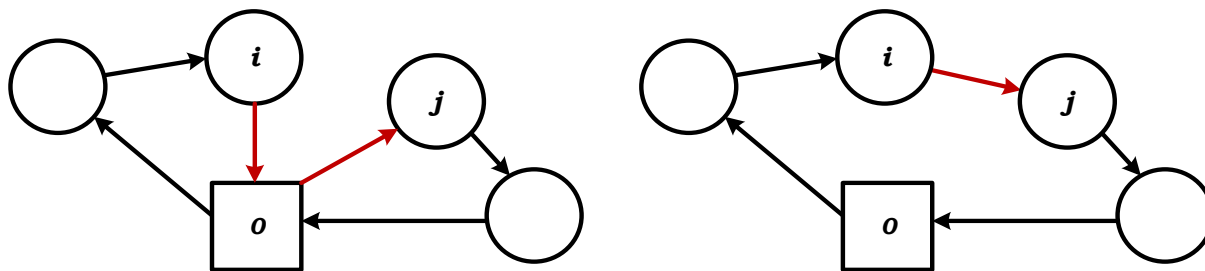


Figura 9. Ejemplo del algoritmo de Clarke y Wright. Dos rutas que van hacia el depósito, antes y después de ser unidas. Adaptado de Benito Quintanilla (2015)

Este algoritmo consiste en evaluar dos rutas diferentes $(0, \dots, i, 0)$ y $(0, j, \dots, 0)$ que pueden ser combinadas formando una nueva ruta $(0, \dots, i, j, \dots, 0)$ como se muestra en la Figura 9, la nueva ruta se crea sí y solo si el ahorro (en distancia) obtenido por dicha unión es mayor que cero. El ahorro para cada conjunto de rutas r es calculado mediante la siguiente expresión:

$$s_{ij} = d_{i0} + d_{0j} - d_{ij} \quad (35)$$

En la nueva solución los arcos $(i, 0)$ y $(0, j)$ no serán utilizados y se agregará el arco (i, j) . En este algoritmo se realizan las uniones que generan mayores ahorros, siempre y cuando no se violen las restricciones de capacidad del problema. El algoritmo de ahorros tiene una versión paralela en la que se trabaja sobre todas las rutas simultáneamente, y otra secuencial que construye las rutas una a la vez. En este trabajo de investigación solo se hizo énfasis en la segunda versión, los pasos a seguir están descritos en la Tabla 3.

Tabla 3

Metodología del algoritmo de Clarke y Wright. Elaboración propia

Paso 1 (Cálculo de distancias): Hallar las distancias entre los nodos con el depósito, y entre estos.

Paso 2 (Cálculo de ahorros): Elaborar a partir de la expresión (35) una matriz de ahorros para todos los clientes.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Paso 3 (Lista de ahorros): Ordenar de mayor a menor los ahorros obtenidos en el paso 2 en una lista de ahorros (LA).

Paso 4 (Identificación de arcos): Seleccionar el par de nodos que conforman el mayor ahorro de la lista (LA) creada en el paso anterior, además comprobar si el par de nodos ya están asignados a una ruta, en caso de estarlo tomar el siguiente par de nodos con el mayor ahorro.

Paso 5 (Verificación de capacidad): Verificar que la demanda de los nodos a unir no sobrepase la capacidad de los vehículos; en caso de que se exceda comenzar desde el paso anterior con el siguiente par de nodos que forman el mayor ahorro.

Paso 6 (Ampliación de ruta): Comparar el par de nodos seleccionado con la ruta creada y verificar si es posible la ampliación de la ruta, esto se hace mediante tres criterios. El primero es que del par de nodos seleccionado solo uno se repita con la ruta creada; el segundo es que ese nodo que se repite en la ruta tenga menos de dos vecinos; el tercero es que la suma de las demandas de esa posible ruta incluyendo el nodo que no se repite, no exceda la capacidad del vehículo; en caso que los tres criterios se cumplan es posible ampliar la ruta y se elimina el par de nodos de la lista de ahorros (LA), si alguno de los criterios no se cumple se deja el par de nodos como otra ruta y el par de nodos es eliminado de la lista de ahorros.

Paso 7 (Iterar): Repetir los pasos 4, 5 y 6 hasta que todos los clientes sean asignados a una ruta.

Como se mencionó previamente, la extensión del algoritmo de ahorros presenta los parámetros de peso λ y μ . El parámetro λ es usado para evitar que se generen rutas circulares, impidiendo así dar lugar a la unión de puntos lejanos. Gaskell (1967) y Yellow (1970), formularon este parámetro entre $0,1 \leq \lambda \leq 2$. Adicionalmente se añadió el parámetro μ el cual tiene en cuenta la asimetría en términos de distancia desde el depósito hasta los nodos que se están evaluando, por ello Paessens (1988) añadió en la expresión (35) un nuevo termino que estaba entre $0 \leq \mu \leq 2$. Debido a esto la ecuación de ahorros sufre unas leves modificaciones, las cuales se presentan a continuación.

$$S_{ij} = d_{i0} + d_{0j} - \lambda d_{ij} - \mu |d_{0i} - d_{0j}|$$

(36)

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

En la aplicación del algoritmo de ahorros se pueden presentar casos particulares (Figura 10), como que existan clientes que no pueden ser incluidos a una ruta como es el caso de nueve y once, dado que la capacidad del vehículo llegó a su máximo, en estos casos se deben realizar las rutas de forma independiente, es decir, crear un recorrido exclusivo de ida y vuelta solo para esos clientes.

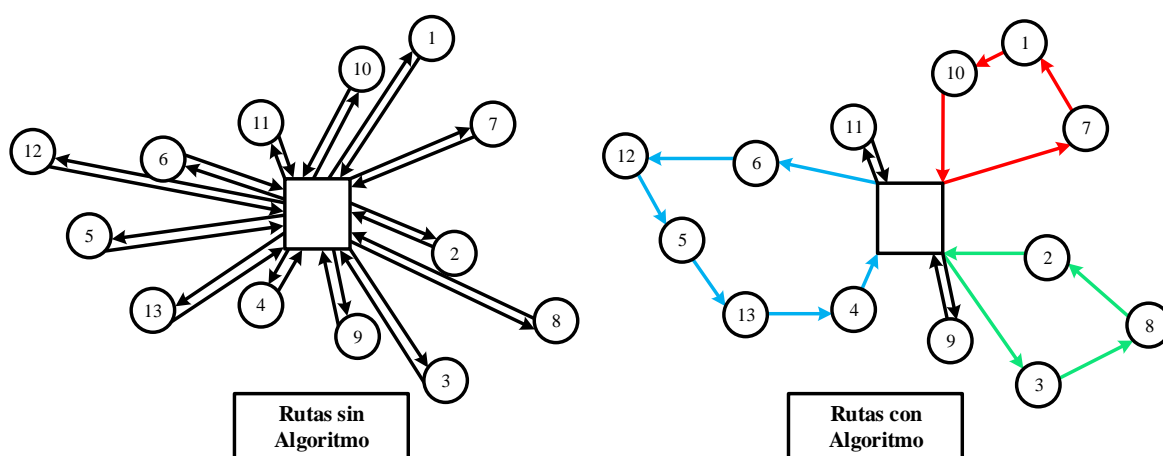


Figura 10. Casos particulares de clientes. Un ejemplo del caso en particular de rutas excluidas aplicando el algoritmo de ahorros de Clarke y Wright para 13 clientes.

6.1.2 Inserción de drones

Luego de la creación de la ruta inicial donde solo se consideran los vehículos, se procede a aplicar el algoritmo de inserción para incluir a los drones en las rutas creadas. Este algoritmo tiene en cuenta las restricciones de capacidad de vuelo y de carga del dron; el proceso inicia identificando los clientes que pueden ser atendidos por los drones, luego de esto se evalúa en la ruta del vehículo las posibles tuplas que se pueden crear para atender a dichos clientes trazando la que permita reducir los costos totales de la ruta. Para mayor claridad en la Tabla 4 se puede observar el pseudocódigo de este algoritmo adaptado de (Sacramento et al., 2019).

El algoritmo analiza cada ruta contenida en el grafo (Figura 11), si la capacidad del dron cumple con la demanda del cliente a visitar se puede efectuar la visita. Una vez realizado esto se dispone

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

a revisar los mejores nodos de lanzamiento, entrega y recuperación del dron. Por último, si el nodo de entrega cumple con las restricciones de tiempos y costos, se crea la tupla.

Tabla 4

Pseudocódigo del algoritmo de inserción de los drones. (Sacramento et al., 2019)

Algoritmo 1: Inserción de drones

Input: *Partial solution: s,*

customer to insert as drone-customer: c'

threshold cost: η ,

1. *BestSortie* = \emptyset ;

2. **For Each** Route in *s do*

3. **If** Capacity (Route) + $q_c < Q$ **then**

4. **For** Pair positions (i, k) in Route where $i < k$ **do**

5. Construct sortie $p = \langle i, c, k \rangle$ with launch-position i, delivery position c and recovery position k;

6. Check feasibility for sortie p;

7. **If** $SL + SR + \tau_{ij}^D + \tau_{jk}^D + Se_j^D < e$ AND $f(s) + Costsortie(p) <$

η **then**

8. *BestSortie* $\leftarrow p$;

9. **Update** η ;

10. **return** *BestSortie*;

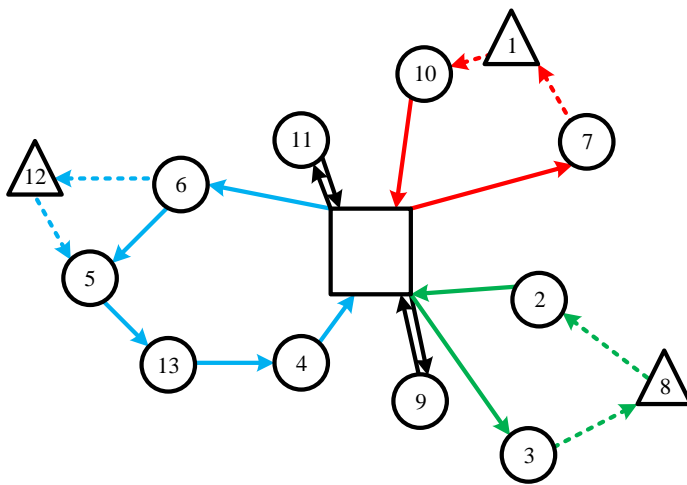


Figura 11. Ejemplo de inserción de drones para 13 clientes, con nodos clientes 1, 8 y 12 visitados drones, siendo los nodos circulares aquellos que son atendidos por un camión y los triangulares por un dron

6.2 Algoritmos de destrucción

La ejecución de los métodos de remoción se da a partir de la solución inicial, el algoritmo sigue su procedimiento aplicando una destrucción a dicha solución s , la cual consiste en remover un número de nodos de la solución, rompiendo los arcos que tiene con los demás nodos; la cantidad de nodos que se quitan depende de la tasa de destrucción y esta es definida previamente a la hora de seleccionar la heurística que va a ser aplicada en cada fase. La tasa es un número entero determinado aleatoriamente entre un intervalo, dicho intervalo se define con un valor mínimo (d_{min}) y uno máximo (d_{max}), en función del número de clientes totales.

Múltiples experimentos de implementación han demostrado que el grado de la destrucción no debe ser demasiado grande, puesto que el tiempo de ejecución de la heurística de reparación crece dramáticamente; además, las heurísticas de reparación no están diseñadas para construir una solución completa desde cero. Por esta razón, la tasa de destrucción debe estar por debajo del 0,5 de n y se sugiere que esta sea entre el 10% y 40% cuando se trate de instancias pequeñas (50 a 100 clientes) y entre 30 y 60 clientes cuando el número de nodos exceda dicho intervalo (Pisinger & Ropke, 2007), todo esto con la finalidad de evitar los problemas mencionados previamente. Luego de definir la tasa de destrucción es importante tener la asignación de pesos para todas y cada una de las heurísticas a usar en el algoritmo, esta asignación modifica los pesos de las heurísticas de destrucción en cada iteración y dependiendo de los criterios de aceptación de la solución el valor de la asignación es distinto. Las normas por las cuales se rige la elección de los clientes dependen de las heurísticas empleadas; las heurísticas usadas en la remoción de los clientes fueron, *Random Removal*, *Worst Removal* y *Cluster Removal*; a continuación, se hace una explicación del funcionamiento de cada una de ellas.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

La eliminación aleatoria o *Random Removal* se encarga de retirar de manera uniforme y al azar γ clientes de la solución inicial como se observa en la Figura 12, dando lugar a rutas parciales, siendo esto visto como una estrategia de destrucción desfavorable a primera vista, porque podría eliminar las partes ya adecuadas de una solución aceptable; aun así, esto mantiene un efecto favorable y es que proporciona diversificación, lo que lleva a explorar un gran espacio de búsqueda.

La heurística de destrucción *Worst Removal*, *Critical Removal* o peor eliminación, basa su mecanismo en ordenar los clientes servidos de una solución s de manera decreciente de acuerdo con sus costos, estos son guardados en una lista L ; aplicando como criterio la diferencia de costos $\Delta C_{-i}(s)$ de cada cliente en la ruta, este se calcula a partir de la diferencia entre las funciones $[C(s) - C_{-i}(s)]$, la cual se compone de los valores de la solución s incluyendo y excluyendo al cliente i . El propósito central de esta heurística es eliminar al cliente de mayor costo, el pseudocódigo de esta se presenta en la Tabla 5.

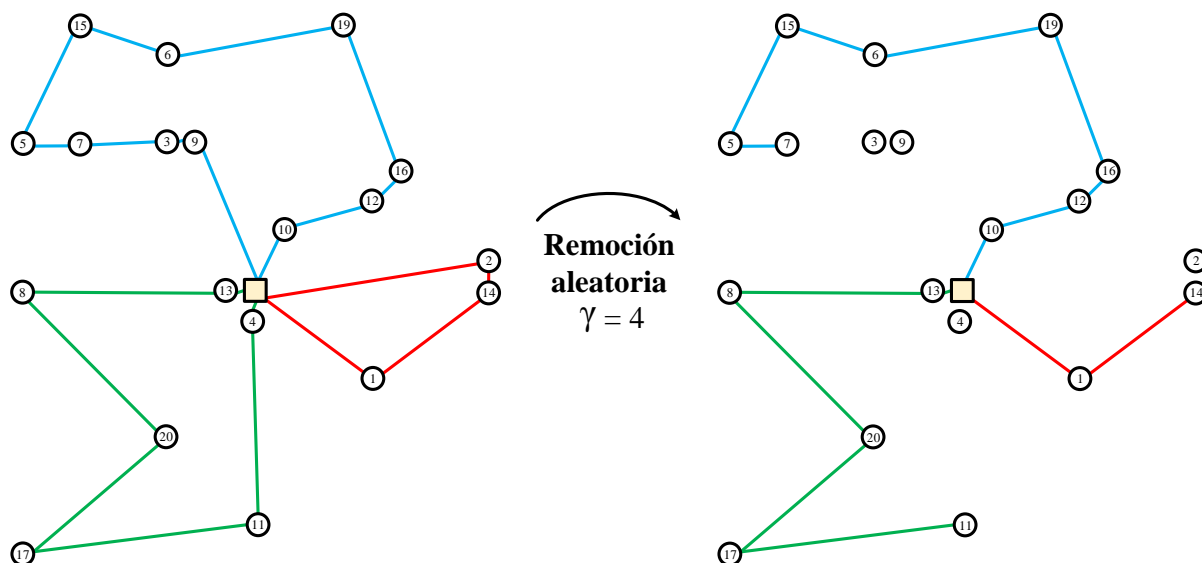


Figura 12. Ejemplo de la heurística Random Removal para cuatro clientes seleccionados de manera aleatoria en una solución de tres rutas compuestas para un total de 20 clientes.

Tabla 5

Pseudocódigo del algoritmo Worst Removal. Adaptado de (Ropke & Pisinger, 2006)

Algoritmo 2: Worst Removal

1. **Function** *WorstRemoval* ($s \in \{\text{solution}\}$, $\gamma \in \mathbb{N}$, $m \in \mathbb{R}_+$)
 2. **While** $\gamma > 0$ **do**
 3. Array: $L = \text{All planned requests } i$, sorted by descending $\text{cost}(i, s)$;
 4. Choose a random number ϑ in the interval $\{0, 1\}$;
 5. Request: $r = L[\vartheta^m | L|]$;
 6. Remove r from solution s ;
 7. $\gamma = \gamma - 1$;
 8. **End while**
-

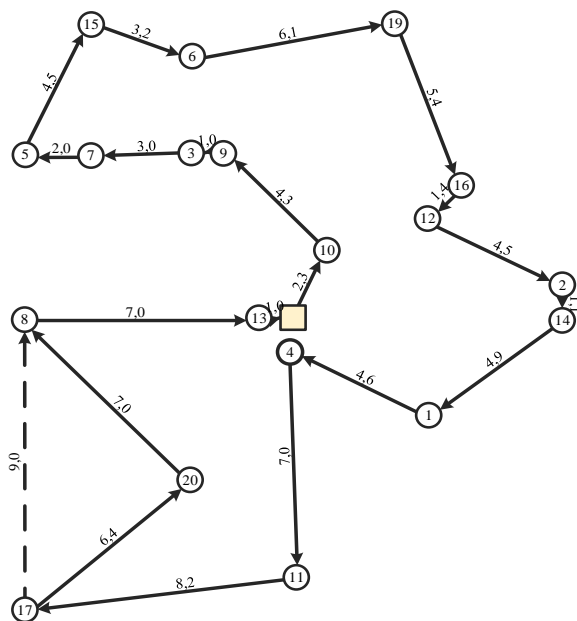
La eliminación en cada iteración se hace mediante el operador $[\vartheta^m]$ (Ropke & Pisinger, 2006), con este se obtiene la posición del cliente a ser retirado. La forma en la que se realiza esto es eligiendo uno a uno los γ clientes, generando así un número aleatorio $\vartheta \in [0, 1]$ y un factor $m \geq 1$ con el que se logra la diversificación de la heurística y controla su aleatoriedad, cuando $m = 1$ la remoción de los clientes en la solución s es hecha totalmente aleatoria.

Se ha demostrado que este criterio se ha empleado en otras heurísticas como lo es el caso de *Shaw Removal* y tiene un desempeño aceptable para valores entre 3 y 5 (Shaw, 1997).

Por ejemplo, se puede considerar una ruta con 20 clientes organizados en la lista L de acuerdo con sus costos, dichos costos se presentan en la Figura 13, para hallar la posición r se genera un número aleatorio $\vartheta = 0,6752$ y un factor $m = 5$, luego dicho número aleatorio es elevado a este factor y el valor que resulta de esto se multiplica con la magnitud de la lista L , dando así la ubicación del cliente a remover ($r = 3$), los cálculos para llegar a esto se presentan a continuación:

$$r = [\vartheta^m | L|] = [(0,6752^5) * (20)] \cong 3$$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM



Posición en L	Nodo	Costo
1	N17	10,2
2	N4	5,2
3	N20	4,4
4	N11	4
5	N8	3,9
6	N19	2,5
7	N5	2,5
8	N15	1,9
9	N12	1,6
10	N1	1,3
11	N10	1,1
12	N16	0,7
13	N2	0,5
14	N6	0,4
15	N14	0,3
16	N9	0,3
17	N7	0
18	N3	0
19	N18	0
20	N13	-1

$$C_{-20}(s) = 84,4 - 7 - 6,4 + 9 = 80,4$$

$$\Delta C_{-20}(s) = C(s) - C_{-20}(s) = 84,4 - 80,4 = 4,4$$

Figura 13. Ejemplo del algoritmo Worst Removal para una ruta compuesta por 20 clientes cuyos costos de recorrido asociados son los arcos que van hacia cada par de nodos, siendo el cuadrado el depósito, los círculos los clientes, las líneas continuas el costo de viajar entre clientes y la línea discontinua el costo que representa ir del nodo 17 al 8 (dado que el costo del cliente 20 se remueve de la solución inicial s)

Finalmente, la heurística de eliminación *Cluster Removal* inspira sus mecanismos en los resultados observados a partir del comportamiento de la remoción orientada a la distancia (Lutz, 2014), para hacerlo divide el conjunto de nodos de una ruta en k subconjuntos disjuntos, siendo $k \geq 2$, agrupando los nodos que se encuentren de alguna manera “ceranos entre sí”. El algoritmo *Cluster Removal* comienza su procedimiento con la tasa de destrucción que es un valor aleatorio entre de 10% y 30% del total de clientes, luego se procede a determinar el número de k subconjuntos a conformar, este se calcula aproximando hacia abajo la división de los clientes totales y el número de clientes a remover establecidos por la tasa.

Los k subconjuntos se generan mediante el método de *K-means*, que tiene en cuenta los centroides con el fin de calcular los clientes que se encuentre más cercanos entre sí y así formar

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

los clúster, este proceso se repite iterativamente hasta lograr obtener clusters de forma uniforme en la solución, finalmente se tienen dos casos: el primero es en el que el número de clientes a remover es mayor que el número de clientes pertenecientes al cluster, para este escenario se remueven todos los clientes que lo componen, la selección de este se hace al azar, luego los clientes faltantes para cumplir la tasa de destrucción son elegidos al azar de otro subconjunto de k clusters creados. Por otro lado, tenemos el caso en que la cantidad de nodos a remover es menor al número de clientes del subconjunto seleccionado, estos son removidos en su totalidad; estos procesos se repiten hasta que se completa con un número de clientes definidos por la tasa de destrucción, el pseudocódigo se representa en la Tabla 6. Una ventaja de este método es que contempla distintos clientes de distintas rutas y logra agrupar nodos de instancias de gran tamaño, debido a que el recorrido empieza y finaliza en la posición central y visita varios grupos, estos subconjuntos se implementan para mejorar notablemente la distancia d entre las rutas creadas y de esta manera mejorar los costos de solución, la heurística de destrucción trata de eliminar un grupo completo y asignarlo a un recorrido distinto, por ello, lo que se busca es evitar una larga distancia entre la unión de ellos (Ropke, 2006).

Tabla 6
Pseudocódigo del algoritmo Cluster Removal

Algoritmo 3: Cluster Removal

1. **Input:** current solution s , set of unassigned requests \mathcal{L} , degree of destruction γ ,
 2. $n_{removed} = 0$
 3. determine k clusters from the requests in s
 4. **while** $n_{removed} < \gamma$ **do**
 5. select one of the clusters $k_{selected}$ randomly
 6. remove all requests in $k_{selected}$ from s
 7. $\mathcal{L} = \mathcal{L} \cup k_{selected}$
 8. update $n_{removed}$
 9. **return** s
-

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Un ejemplo de esta heurística se presenta en la Figura 14. De acuerdo con esta se toman en consideración dos rutas (Ruta 1 y Ruta 2) de las cuales se selecciona una aleatoriamente (Ruta 2) y a partir de esta se establecen grupos de clientes, eligiéndolos nuevamente cada uno de estos al azar y formando así el agrupamiento de cuatro clientes, para este caso se quita el clúster compuesto por los clientes 8,9,10 y 11 respectivamente, una vez retirados de esta ruta son almacenados y pueden llegar a ser insertados en otra (Ruta 1) que mejore el costo de la solución actual.

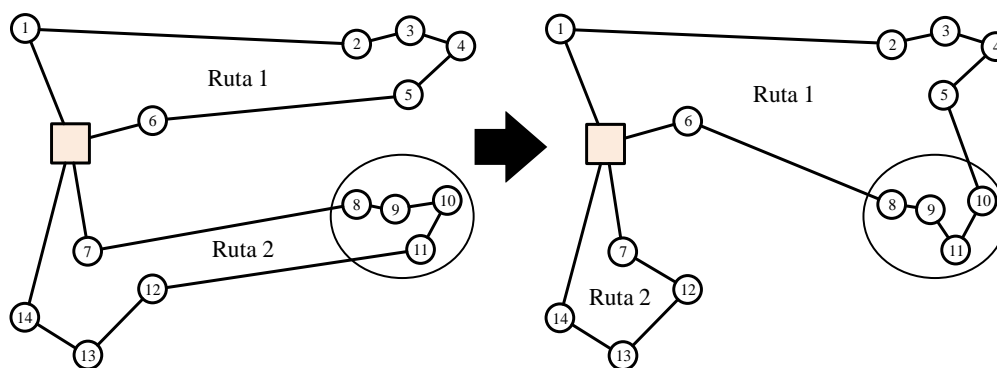


Figura 14. Ejemplo de la heurística Cluster Removal para cuatro clientes removidos de la Ruta 1 e insertados en la Ruta 2. Adaptado de (Ropke, 2006)

6.3 Algoritmos de reparación

El proceso de reparación o inserción es la siguiente fase del algoritmo ALNS, durante esta etapa el algoritmo procede a reconstruir la solución parcial, debido a que en el proceso de destrucción se han removido un conjunto de clientes que deben reinsertarse, para hacerlo existen algoritmos basados en los costos que se conocen como voraces o codiciosos (Lutz, 2014), el más básico de estos funciona netamente con costos, existen otros que combinan la parte voraz con un componente aleatorio, luego existe uno que tiene en cuenta las n primeras posiciones para la inserción de los nodos, otro algoritmo que ha sido usado en la reparación es el *Branch and Bound*; el pseudocódigo de la fase de reparación se presenta en la Tabla 7.

Las heurísticas voraces reparan la solución mediante el cálculo de los costos asociados a todas las posibles inserciones de los clientes que se encuentran almacenados en una lista \mathcal{L} de nodos

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

removidos, y cada vez que se inserta un cliente, este se elimina de dicha lista, de modo que facilita el mecanismo de reconstrucción de las rutas ubicando a los nodos en la posición más prometedora de estas. Las heurísticas de tipo aleatorio se basan en las voraces, incluyendo parámetros que hacen que los costos de las posibles inserciones se diversifiquen, con la finalidad de evitar los problemas al intentar insertar los últimos nodos. El uso de la heurística seleccionada para insertar los clientes debe contemplar que no se construyan soluciones infactibles, si el cliente elegido no se le puede encontrar una inserción factible en dicha solución, este se mantiene para posteriores inserciones o abrir una nueva ruta para este. Adicionalmente, cabe recalcar que los clientes deben ser atendidos en su totalidad en el conjunto de rutas que se establezcan.

Tabla 7

Pseudocódigo de los algoritmos de reparación o inserción (Lutz, 2014)

Algoritmo 4: Métodos de reparación

1. **Input** Current solution s , set of unassigned requests \mathcal{L}
 2. Calculate insertion costs of request r into the tour of vehicle $v, \forall r \in \mathcal{L}, v \in V$
 3. **While** further insertions are possible **do**
 4. Rank request-vehicle pairs (r, v) according to insertion costs
 5. Determine top-ranked request-vehicle pair (r, v)
 6. Insert r into the tour of v in s
 7. $\mathcal{L} = \mathcal{L} \setminus \{r\}$
 8. Update insertion costs of $(r', v) \forall r' \in \mathcal{L}$
 9. **End while**
 10. **Return** s
-

Los algoritmos usados para la fase de reparación son de tipo voraz, como ya se mencionó previamente en la sección 4.16. , dos de este tipo son los algoritmos *Greedy Insertion with Noise function (GIN)* y *Regret-n Insertion (nRI)*, debido a que reparan la solución parcial tomando a los clientes removidos e insertándolos mediante el cálculo del costo de inserción de los clientes que

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

aportan una ganancia a la solución parcial, en cambio la tercera heurística usada contempla las distancias para realizar la inserción, toma al nodo más cercano que pertenece a la solución parcial actual y lleva el nombre de *Repair Closest Insertion*.

La primera heurística usada en este proyecto se conoce como inserción codiciosa con función de ruido o *Greedy Insertion with Noise function (GIN)*, esta es una versión extendida del GI, la cual se diferencia de esta última porque contempla un grado de libertad a la hora de elegir el mejor lugar en el cual se va a posicionar un nodo, con el que se obtiene un nuevo costo (Phan et al., 2018), este se utiliza para localizar la ubicación en la cual se va a reinsertar el nodo. Esta extensión incorpora \bar{d} , α y ε al costo actual de la ruta, en donde \bar{d} equivale a la distancia más larga que recorre el camión entre los nodos si considera insertar a un nodo entrega del camión; de la misma manera, si se contempla la opción de insertar un cliente como nodo de dron, \bar{d} será la distancia máxima que recorre un dron. Asimismo α representa un parámetro de ruido que es empleado para variar la selección de los nodos a escoger, este toma el valor de 0,1. Adicionalmente se tiene ε el cual es un número aleatorio entre [-1,1]. El costo actual se compone por los costos de los camiones y drones. Siendo así como el *GIN* halla la posición y lograr reinsertar un nodo con un mínimo costo, el cálculo mediante el cual se obtiene dicho costo es el siguiente:

$$\text{NuevoCosto} = \text{CostoActual} + \bar{d} * \alpha * \varepsilon \quad (37)$$

Un ejemplo de este algoritmo, que contempla tres vehículos, cada uno atendiendo rutas diferentes se presenta en la Figura 15, este inicia tomando la solución parcial y la lista \mathcal{L} compuesta por aquellos clientes que fueron removidos de las rutas mediante las heurísticas de remoción. Luego el *GIN* calcula para cada cliente i de la lista, el costo de incluirlo entre dos nodos j y k . Las heurísticas de remoción dejan nodos factibles con el fin de construir una ruta con estos, de forma que, hay ciertas conexiones entre estos que son realizables sin tener que reformar el sentido de la

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

ruta, ya sea asignando el nodo insertado a una ruta que no sea de ese vehículo o incluir conexiones entre los nodos que no fueron removidos previamente, es decir, el costo de conectar N10-N12 y N12-N2 dado que, los nodos N10 y N2 forman parte de rutas distintas, originando así uniones no factibles. Los arcos $(j, i) = (N4, N9)$ y $(i, k) = (N9, N0)$ crean la ruta en la cual se inserta el nodo nueve entre el nodo cuatro y el nodo cero (deposito), respectivamente. Entre las posibles inserciones del nodo nueve, este recorrido es el que tiene el menor costo, haciendo así que esta sea la ruta idónea para insertar dicho nodo.

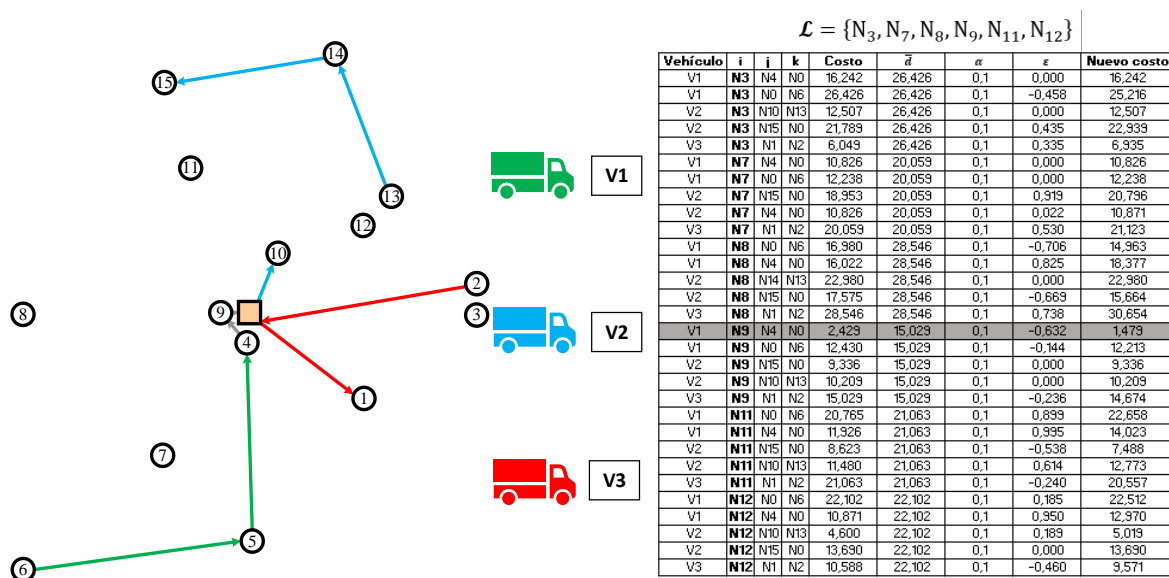


Figura 15. Ejemplo del algoritmo Greedy Insertion with Noise function (GIN) para 15 clientes con la inserción del nodo nueve entre los nodos cuatro y cero, formando así el recorrido gris que a su vez se agrega a la ruta que realiza el vehículo 1 (V1).

La segunda heurística usada es *Regret-n Insertion (nRI)*, al igual que la anterior basa su mecanismo en los costos; sin embargo, en esta heurística no solo se tiene en cuenta la mejor posición de inserción, sino que se evalúan las n mejores posiciones para cada cliente removido buscando con esto evitar los problemas presentados en las heurísticas netamente voraces, uno de ellos es que del conjunto de clientes removidos comienzan a ser insertados en la ruta los clientes con mayores ahorros, pero al evaluar los últimos nodos pueden resultar difíciles de insertar

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

llegando a generar soluciones no factibles. La ecuación (37) usada en esta heurística calcula la diferencia de valor entre la n -ésima y su mejor posición, gracias a esto en las primeras iteraciones se puede identificar aquellos clientes críticos y los va insertando en las rutas.

$$\arg \max_{r \in \mathcal{L}} \left\{ \sum_{i=2}^n [c(s_i(r)) - c(s_1(r))] \right\} \quad (38)$$

En esta heurística es importante la definición de n , debido a que si se le asigna un valor muy grande se beneficia en que en menor cantidad de iteraciones se puede saber que un cliente no tiene muchas posibilidades de inserción, pero del mismo modo se incrementa el tiempo de computación (Lutz, 2014). Debido a esto, en este trabajo de investigación se trabajó esta heurística para un valor de $n = 3$, dando lugar a que se tuvieran en cuenta las tres primeras posiciones con el menor costo de aquellos nodos removidos anteriormente. Como se muestra en la Figura 16, la inserción del nodo tres se da a partir de los costos presentados en el ejemplo previamente mencionado. Las ubicaciones (j, k) con los menores costos asociados para insertar entre estas al nodo tres son, N1-N2 de la ruta 3, N10-N13 de la ruta 2 y N4-N0 de la ruta 1. Una vez se obtienen los costos de las distintas posiciones se procede a calcular la diferencia entre ellas; este proceso se realiza para todos los elementos contenidos en \mathcal{L} . Por último, el algoritmo *nRI* selecciona el nodo cliente cuya sumatoria sea de mayor valor e insertar a este en la mejor posición de acuerdo con este criterio.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

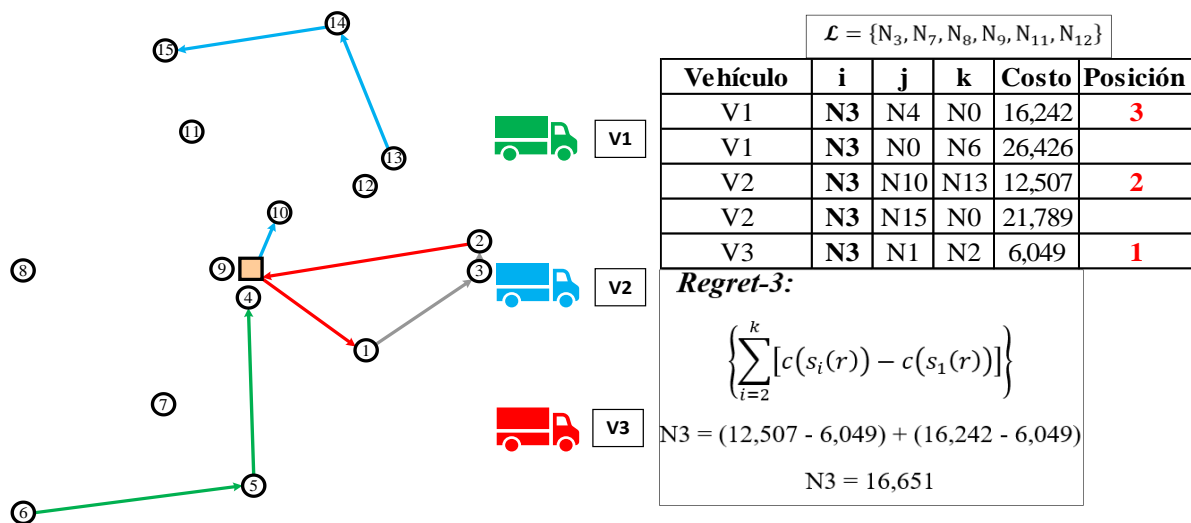


Figura 16. Ejemplo del algoritmo Regret-n Insertion (nRI) para 15 clientes realizando la inserción del nodo tres mediante la sumatoria de mayor valor, utilizando los tres costos de menor valor, creando así el recorrido gris, el cual logra la mejor posición de inserción para dicho nodo entre los nodos uno y dos.

La metodología de la tercera heurística de reparación *Repair Closest Insertion (RCI)*, basa su funcionamiento en las distancias, debido a que trata de insertar a los nodos removidos mediante el cálculo de distancias, seleccionando el nodo más cercano que pertenece a la solución parcial actual. El procedimiento que hace esta heurística comienza seleccionando aleatoriamente a un cliente de la lista \mathcal{L} de clientes removidos, luego calcula la mejor inserción de este mediante el criterio del nodo más cercano que pertenece a la ruta parcial, siempre y cuando se cumplan las restricciones de capacidad del vehículo y duración de la ruta, además que el nodo de la ruta parcial no tenga más de dos vecinos en la ruta; dentro de esta evaluación también determina si es menos costosa la inserción de un dron mediante el algoritmo de inserción de drones presentado en la Tabla 4 de la sección 6.1.2, el nodo insertado es removido de la lista \mathcal{L} y se continúa con la selección aleatoria de otro nodo. En el caso de que el nodo seleccionado de la lista \mathcal{L} no pueda ser insertado, este nodo es agregado al conjunto D_N de clientes faltantes por insertar, los cuales se agregan a la solución parcial mediante el algoritmo de *Greedy Insertion with Noise function (GIN)*, para mayor claridad el pseudocódigo de esta heurística se presenta en la Tabla 8.

Tabla 8

Pseudocódigo del algoritmo Repair method Closest Insertion. (Sacramento et al., 2018)

Algoritmo 5: Repair method Close Insertion

1. **Data:** Current solution, Customers D
 2. **Result:** New Solution for the trucks and UAVs
 - $D_N = D;$
 4. **while** D is NOT empty **do**
 5. $c \leftarrow \text{RandomCustomer}(D);$
 6. $D = D \setminus \{c\};$
 7. $nn \leftarrow \text{NearestCustomer}(c, \text{Solution});$
 8. $r \leftarrow \text{RouteOf}(nn);$
 9. Insert_flag = false;
 10. Sortie_flag = false;
 11. BestInsertion(c, r);
 12. **if** Insertion Found **then**
 13. Insert_flag = true;
 14. **if** Demand(c) < MaxCapacityDrone **then**
 15. Find Best Sortie containing c as a drone customer;
 16. **if** Sortie Found **then**
 17. Insert_flag = false;
 18. Sortie_flag = false;
 19. **if** Insert_flag **then**
 20. $D_N = D_N \setminus \{c\}$
 21. Add customer c through Best Insertion to the Truck's route in Solution;
 22. **if** Sortie_flag **then**
 23. $D_N = D_N \setminus \{c\}$
 24. Add Best Sortie to the route in Solution;
 25. **if** D_N is NOT empty **then**
 26. Repair TruckMethod ($D_N, \text{Solution}$) (Algoritmo GI);
-

6.4 Parámetros del algoritmo

Para el adecuado funcionamiento del algoritmo ALNS fue necesario definir cinco parámetros importantes los cuales son: el periodo de actualización de los pesos, los pesos a asignarse a cada heurística tanto de destrucción como de reparación, basados en el éxito de cada heurística con relación a la función objetivo, el intervalo de la tasa de destrucción usado en estas heurísticas, el

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

criterio de aceptación, el cual admite una solución factible peor que la actual y finalmente un factor que disminuye la temperatura en cada iteración y funciona como criterio de parada en el algoritmo.

El primer parámetro es el periodo de actualización de los pesos, según Ropke and Pisinger (2006) la actualización de los pesos en las heurísticas debe realizarse cada cierto número de iteraciones p_u , esto se conoce como periodo de actualización o segmento u , los resultado del trabajo investigativo recomiendan ajustar razonablemente dicho periodo en base a la cantidad de heurísticas a usar en el ALNS, ya que si este es un valor pequeño, puede suceder que algunas de las heurísticas usadas no logren ejecutarse desde el principio, esto ocasiona que sus pesos se reduzcan, dicha reducción puede deberse a la falta de pruebas de la heurística y no a un mal desempeño de esta; con el fin de evitar estos problemas y además para reducir el esfuerzo computacional que resultaría hacerlo luego que se produce una modificación, Ropke y Pisinger sugieren que el reajuste de los pesos suceda cada 100 iteraciones cuando se tengan más diez heurísticas en total (Ropke & Pisinger, 2006).

El segundo parámetro son los pesos adaptativos de las heurísticas, que varían según la calidad de la solución alternativa generada, en cada iteración los pesos son modificados (Keskin & Catay, 2016). Como se mencionó en el párrafo anterior, los pesos de las heurísticas son actualizados en un periodo p_u de iteraciones, estos pesos fueron definidos como $w(\theta_d)$ y $w(\theta_r)$, donde $\theta_d \{d = 1,2,3\}$ es el conjunto de heurísticas de destrucción y $\theta_r \{r = 1,2,3\}$ el conjunto de heurísticas de reparación; la elección de la heurística se realiza usando *The roulette Wheel selection principle*, donde en cada iteración la probabilidad de seleccionar una heurística se define como:

$$P(\theta_d) = \frac{w(\theta_d)}{\sum_{l=1}^3 w(\theta_l)} \quad y \quad P(\theta_r) = \frac{w(\theta_r)}{\sum_{l=1}^3 w(\theta_l)} \quad \text{respectivamente} \quad (39)$$

Inicialmente la probabilidad de selección es igual para todas las heurísticas, esta se va ajustando en cada periodo de actualización mediante un factor de reacción $0 \leq \rho \leq 1$, la puntuación Ψ que

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

depende únicamente del tipo de solución obtenida con las heurísticas usadas, al final de cada iteración esta se va sumando en π_d para cada heurística de destrucción y en π_r en las de reparación, estos comienzan con valores en cero; las puntuaciones de Ψ a asignar se definen en la Tabla 9, y finalmente con θ_d y θ_r que cuentan el número de veces que ha sido usada la heurística. Para hacer la actualización de los pesos se usa para las heurísticas de destrucción y reparación respectivamente:

$$w(\theta_d) = \begin{cases} w(\theta_d)(1 - \rho) + \rho \frac{\pi_d}{\theta_d}, & \text{si } \theta_d > 0 \\ w(\theta_d)(1 - \rho), & \text{si } \theta_d = 0 \end{cases}$$

$$w(\theta_r) = \begin{cases} w(\theta_r)(1 - \rho) + \rho \frac{\pi_r}{\theta_r}, & \text{si } \theta_r > 0 \\ w(\theta_r)(1 - \rho), & \text{si } \theta_r = 0 \end{cases}$$

Tabla 9
Puntuaciones que toma el parámetro Ψ , según el tipo de solución generada. (Sacramento et al., 2019)

Parámetro Ψ	Descripción
σ_1	La nueva solución es una mejor solución global.
σ_2	La nueva solución es aceptada y su costo es mejor que la actual.
σ_3	La nueva solución es aceptada pero con un costo peor que el actual.
σ_4	La nueva solución es rechazada.

El tercer parámetro del algoritmo es el intervalo de la tasa de destrucción de estas heurísticas, este se establece que debe estar por debajo del 0,5 de n , se sugiere que esta sea entre el 10% y 40% cuando se trate de instancias pequeñas (50 a 100 clientes) y entre 30 y 60 clientes cuando el número de nodos exceda dicho intervalo (Pisinger & Ropke, 2007), esto con la finalidad de ampliar el espacio de búsqueda; sin embargo, si se fija un valor demasiado alto se pueden presentar

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

inconvenientes en la fase de reparación, debido a que las heurísticas de inserción no están diseñadas para reparar completamente una solución.

El cuarto parámetro es el criterio de aceptación, la función de este parámetro es controlar y aceptar las soluciones creadas en cada iteración del algoritmo; para definirlo se usó la heurística de Recocido Simulado (Černý, 1985; Kirkpatrick et al., 1983), en la que el algoritmo utiliza un parámetro de temperatura T que regula la probabilidad de aceptación. Si la solución S_N generada por las fases de destrucción y reparación da como resultado una solución con menor valor objetivo que la solución actual S_C , entonces se acepta; sin embargo, si la solución S_N tiene un valor objetivo más alto, se hace uso del algoritmo de recocido simulado, con el que se calcula una probabilidad de aceptación $Prob(S_N)$, la cual es comparada con un número generado aleatoriamente entre cero y uno; la solución únicamente es aceptada si el valor de dicha probabilidad es mayor al número aleatorio $Random(0,1) < Prob(S_N)$. El cálculo de la probabilidad se realiza con la diferencia entre el costo de las dos soluciones a razón de una temperatura que disminuye dependiendo del tiempo máximo y el tiempo empleado iterativamente, a continuación, se presenta la fórmula para el cálculo de la probabilidad:

$$Prob(S_N) = e^{\frac{f(S_C) - f(S_N)}{T}} \quad (40)$$

Donde $f(S_C)$ y $f(S_N)$ son los valores de la función objetivo evaluados con S_C y S_N respectivamente, además es necesario definir una temperatura inicial T_{in} para T que se reduzca linealmente hasta hacerse cero; esto con la finalidad de que también se establezca como un criterio de parada del algoritmo, para esto se controla la temperatura usando el tiempo transcurrido t^{pass} desde el inicio del algoritmo y se toma t^{max} como el límite de tiempo impuesto en el algoritmo. Para calcular la nueva temperatura se emplea la siguiente fórmula:

$$T = T_{in} * \left(1 - \frac{t^{pass}}{t^{max}}\right) \quad (41)$$

7. Validación del algoritmo

La validación de los resultados de un algoritmo desarrollado, con frecuencia se hace para comparar con resultados publicados en la literatura, con la finalidad de evaluar las soluciones obtenidas por medio de los métodos para optimización planteados en el proyecto.

7.1 Instancias

Varias de las instancias presentadas en distintos artículos no pueden ser usadas en el problema de ruteo de vehículos con drones, debido a que este contempla parámetros como la capacidad de carga del dron, los tiempos asociados al dron, la capacidad de las rutas, el tiempo de vuelo de la batería del dron, entre otros, esto le da un trato especial a este problema. Por lo tanto, evaluar dicho problema con instancias que no contemplen este tipo de parámetros lleva a un escenario que no refleja correctamente las propiedades reales del problema.

El presente trabajo de investigación propone un algoritmo ALNS, se evaluó su desempeño mediante las instancias propuestas por (Sacramento et al., 2018). Estas instancias presentan una distribución uniforme $U(-d, d)$, las instancias son generadas al azar y cuentan con una acomodación en la que solo se contempla un depósito central ubicado en las coordenadas (0,0), un ejemplo de estas se presenta en la Figura 17. Los clientes se ubican en una cuadrícula de dimensiones $2d$ por $2d$ alrededor del depósito, cada instancia sigue la nomenclatura $n. m. t$, donde n es el número de clientes para el escenario, m es la dimensión de la cuadrícula y t es el nombre genérico del escenario. La dimensión de las cuadrículas se establece en gran medida por los drones, dado que estos pueden operar un recorrido de ida y vuelta con un radio de alrededor de 10 millas desde el centro de distribución (Rose, 2013); por esta razón disponer de un tamaño por encima de ese podría causar implicaciones y escenarios en los que el dron no pueda funcionar de manera

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

independiente, por ende, se obliga a que el camión interactúe con el dron. Paralelamente, Amazon expone que sus entregas corresponden a artículos que tienen un peso menor a 5 lb (2.27 kg) (Allain, 2013), aunque hay compañías como Workhorse que son capaces de transportar hasta 10 lb (4,54 kg) (Trop, 2016) y que así mismo el camión impone una restricción en los paquetes entregados. Según UPS, el paquete transportado no debe exceder las 150 lb (68 kg) (UPS, 2017); siguiendo lo anterior, los clientes que se generan tienen una demanda que viene dada por estas limitantes.

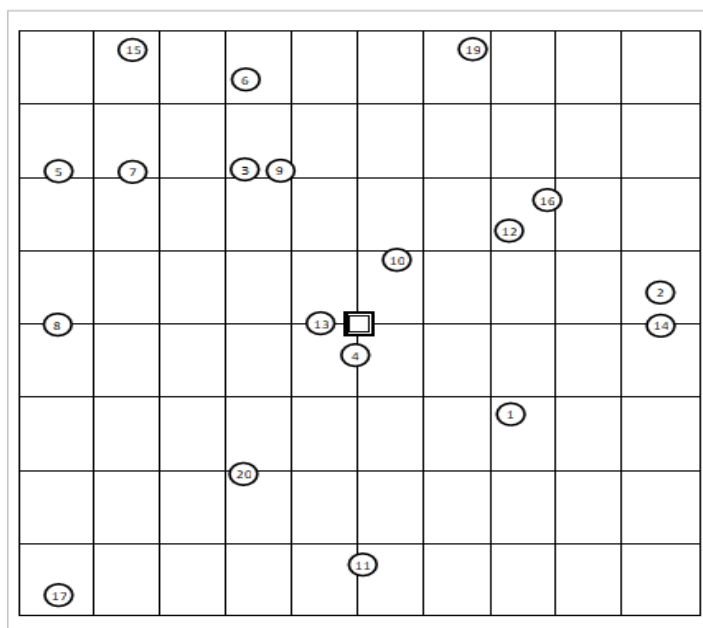


Figura 17. Configuración de las instancias de 20 clientes usadas para la validación. Adaptado de (Sacramento et al., 2019)

Las instancias planteadas por los autores tienen cinco tamaños de cuadrícula, 5, 10, 20, 30 y 40 millas respectivamente; se distribuyen 6, 10, 12 y 20 clientes en los tamaños de cuadrícula 5, 10 y 20, denominadas instancias pequeños, se tienen cuatro configuraciones para cada número de clientes y cuadrícula, siendo 48 problemas en total. Así mismo, se cuentan con 50, 100, 150 y 200 clientes distribuidos en cuadrículas de 10, 20, 30 y 40 con cuadros de diferentes configuraciones siendo 64, consideradas instancias grandes; obteniendo así un total de 112 instancias para realizar

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

la validación del algoritmo. Inicialmente se realizaron pruebas con instancias de 100 y 200 clientes, encontrando que el tiempo computacional era bastante alto; por lo tanto, se decide tomar una instancia de cada configuración probando el algoritmo en 28 instancias en total. Cada instancia se encuentra compuesta por tres columnas: coordenada en X, coordenada en Y demanda de cada cliente. Por último, se tiene un archivo denominado *RouteData*, que proporciona la información de los parámetros del modelo usados para la ejecución del algoritmo descritos en la sección 5, sus respectivos valores se encuentran en la Tabla 10. Cada instancia se corre 10 veces, el algoritmo fue realizado y ejecutado en un equipo con procesador AMD A9 – 9410 RADEON R5, 5 COMPUTE CORES 2C + 3G y 8 GB de RAM instalada, las instancias empleadas en la validación del algoritmo se pueden encontrar consultando el Apéndice E.

Tabla 10
Valores de los parámetros utilizados en la configuración del problema. (Sacramento et al., 2019)

Notación	Parámetro	Valor
SL	Tiempo requerido para lanzar el UAV	1 [min]
SR	Tiempo requerido para recuperar el UAV	1 [min]
V^v	Velocidad del vehículo	56.35 [Km/h]
V^d	Velocidad del dron	80.5 [Km/h]
e	Capacidad de vuelo del dron	30 [min]
q	Capacidad del vehículo	1300 [Kg]
W	Capacidad de carga del dron	5 [Kg]
P_c	Precio combustible	1.13 [€/l]
C_c	Consumo combustible	0.07 [l/Km]
α	Factor de costo del dron	10%
T_{max}	Tiempo máximo de duración de la ruta	480 [min]

7.2 Resultados de la validación del algoritmo ALNS propuesto

Para la validación de los resultados además de los parámetros del modelo y de las instancias, fue necesario establecer los parámetros propios del ALNS, para esto se emplearon los presentados

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

por (Sacramento et al., 2019), con excepción del valor de σ_4 , el cual tiene un valor de 0, pero en la presente investigación se propone usar un valor de 1 con el fin de contemplar una probabilidad para este parámetro

Tabla 11
Valores de los parámetros del algoritmo ALNS

Símbolo	Parámetro	Valor
σ_1	Puntaje de la mejor solución	33
σ_2	Puntaje de mejorar la solución actual	9
σ_3	Puntaje de aceptar una solución peor	13
σ_4	Puntaje por crear una solución no factible	1
T_{in}	Temperatura inicial del algoritmo Recocido Simulado	42
T_f	Temperatura final del algoritmo Recocido Simulado	0,004
ω	Tasa de enfriamiento del algoritmo Recocido Simulado	0,9919
ρ	Factor de reacción	0,9
m	Factor de diversificación del algoritmo <i>Worst Removal</i>	3
p_u	Número de iteraciones para actualizar pesos y probabilidades	100
d_{min}	Tasa mínima de destrucción	10%
d_{max}	Tasa máxima de destrucción	30%
i_{max}	Numero de iteraciones máximo sin mejorar solución para instancias pequeñas	1000

El criterio de evaluación para las instancias será el mejor costo global para hacer la distribución a los clientes. En la Figura 18 se presentan los resultados del experimento para 28 instancias, en esta se incluyen los mejores Z conocidos por el ALNS, las soluciones iniciales obtenidas mediante el Clarke y Wright (CWA), CWA extendido, la mejora porcentual del ALNS se calcula respecto a las soluciones iniciales dadas por medio del CWA y el CWA extendido. Se observa que en las heurísticas más pequeñas el CWA extendido genera soluciones que el ALNS no puede mejorar o lo hace de forma mínima; por su parte, el CWA básico presenta porcentajes de mejora menores

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

que su versión extendida, esto se debe a que en la mayoría de los casos el CWA genera soluciones iniciales con costos inferiores, lo que le permite al ALNS partir de buenas soluciones.

Instancia	Mejor Z conocido	CWA	Mejora %	Mejor Z conocido	CWA Extendido	Mejora %
6.5.1	1.209	1.312	7.831%	1.128	1.128	0.000%
6.10.2	1.894	1.894	0.000%	1.894	1.894	0.000%
6.20.3	4.904	5.722	14.291%	5.163	5.163	0.000%
10.5.2	1.384	1.476	6.202%	1.384	1.462	5.323%
10.10.1	2.681	2.973	9.824%	2.681	2.790	3.904%
10.20.4	5.986	6.363	5.937%	5.986	6.802	12.005%
12.5.4	1.511	1.598	5.420%	1.511	1.718	12.039%
12.10.3	2.473	3.035	18.525%	2.473	3.083	19.799%
12.20.1	5.777	6.436	10.237%	5.777	6.655	13.199%
20.5.1	1.705	2.176	21.620%	1.655	2.208	25.053%
20.10.2	3.576	4.301	16.848%	3.462	4.085	15.259%
20.20.4	6.951	6.951	0.000%	7.270	9.895	26.530%
50.10.1	6.079	6.316	3.761%	7.350	9.702	24.244%
50.20.2	11.223	11.223	0.000%	13.891	15.000	7.390%
50.30.3	20.183	20.183	0.000%	22.837	22.950	0.490%
50.40.4	24.474	25.624	4.488%	29.033	30.952	6.199%
100.10.4	8.256	8.823	6.430%	12.983	14.246	8.868%
100.20.1	17.070	17.070	0.000%	24.010	26.034	7.773%
100.30.2	24.296	25.382	4.280%	36.796	38.711	4.947%
100.40.4	35.408	35.408	0.000%	46.165	49.330	6.416%
150.10.1	10.948	11.576	5.420%	15.196	17.151	11.394%
150.20.2	23.089	23.089	0.000%	30.646	35.395	13.418%
150.30.1	30.783	33.357	7.717%	45.981	50.738	9.376%
150.40.3	43.039	48.816	11.834%	69.664	72.756	4.250%
200.10.1	11.692	12.969	9.844%	17.886	20.086	10.955%
200.20.4	24.353	24.353	0.000%	40.842	40.842	0.000%
200.30.3	41.398	41.398	0.000%	60.507	61.167	1.078%
200.40.4	50.524	50.524	0.000%	73.358	73.358	0.000%

Figura 18. Resultados de las mejores soluciones conocidas para el ALNS, CWA y CWA extendido para 28 instancias

8. Experimentación y Resultados

La experimentación y el análisis de los resultados se presentan en esta sección, se estudiará el rendimiento del algoritmo para el VRPD empleado en esta investigación.

8.1 Aceptación de soluciones

El análisis del criterio de aceptación se hace mediante el recocido simulado, por esto se grafican las soluciones que son aceptadas en una prueba con 1000 iteraciones para las instancias 20.5.1 y 500 iteraciones para la instancia 100.10.4, utilizando los dos métodos de generación de la solución inicial, el CWA y el CWA extendido. En la Figura 19A cuya solución inicial se genera con el CWA, la solución óptima se encuentra en las primeras iteraciones del algoritmo, en dónde, la línea roja se forma de los puntos graficados correspondientes a la mejor solución encontrada dado un número de iteraciones, presentado así un comportamiento plano, los puntos azules que se registran en el espacio de solución son aquellas soluciones que fueron aceptadas de acuerdo al criterio del recocido simulado, las cuales tienen costos peores que las mejores actuales y tiene un cambio agresivo conforme transcurren las iteraciones, es decir, disminuye la dispersión entre ellos a medida que las corridas se acercan al criterio de parada. En la Figura 19B, de acuerdo con la extensión del CWA, el mecanismo del recocido simulado acepta propuestas de costos poco favorables al inicio de la búsqueda respecto a la mejor solución alcanzada en las primeras iteraciones, pero presentando una menor dispersión de los datos en relación con el comportamiento de la Figura 19A, a medida que disminuye la temperatura, las variaciones van desapareciendo moderadamente y se llega a un punto en el que solo se aceptan aquellas soluciones ligeramente menores.

Mientras tanto las instancias de gran tamaño como es el caso de la instancia 100.10.4 que se presenta en la Figura 20, se observa en la parte superior correspondiente a la Figura 20A las soluciones aceptadas por el recocido simulado a partir de la solución inicial con CWA, los puntos rojos de la parte inferior de la gráfica son la mejor solución conocida, los puntos azules superiores son las soluciones que aceptó el algoritmo de recocido simulado, se puede observar que en esta

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

configuración, las instancias presentan fluctuaciones amplias, además que el algoritmo acepta al inicio de la iteraciones, algunas soluciones que resultan ser mejor a la solución inicial. En la Figura 20B, se observa que es mayor el número de soluciones aceptadas que son mejores a la solución inicial en las primeras iteraciones, además se tiene que existen algunos valores iguales a la mejor solución, sin embargo, comparado la solución Inicial realizada por CWA extendida con el CWA básico se presenta un mayor valor de costo en la función objetivo de las soluciones aceptadas, por tanto, resulta más conveniente aplicar el CWA.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

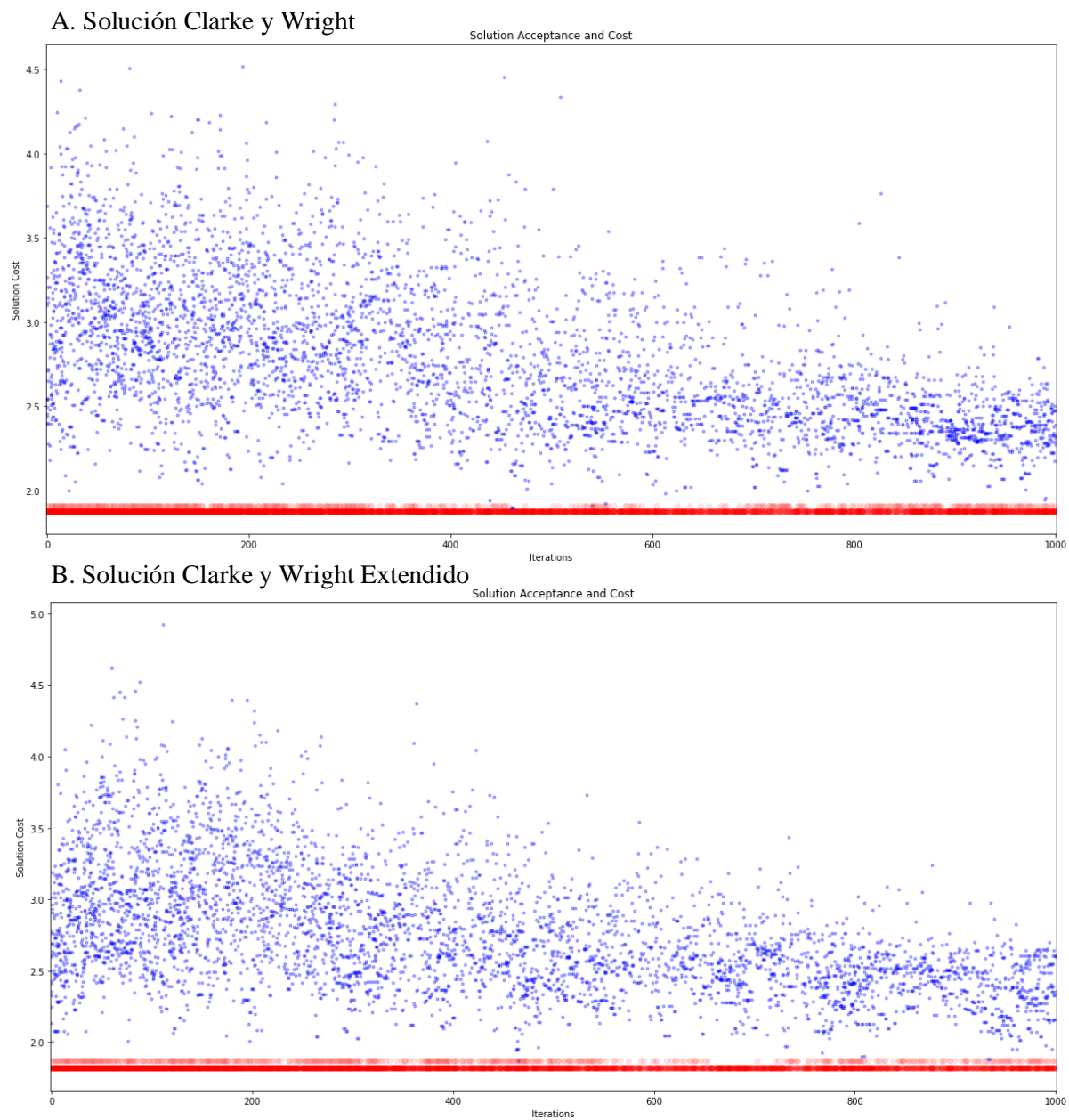


Figura 19. Soluciones Aceptadas en la instancia 20.5.1. La figura A. presenta los resultados de la instancia 20.5.1 de 20 clientes para el Clarke y Wright y la B son los resultados de esta misma para el Clarke y Wright extendido. Las gráficas muestran los costos de las soluciones aceptadas y las mejores soluciones en función del recuento de iteraciones.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

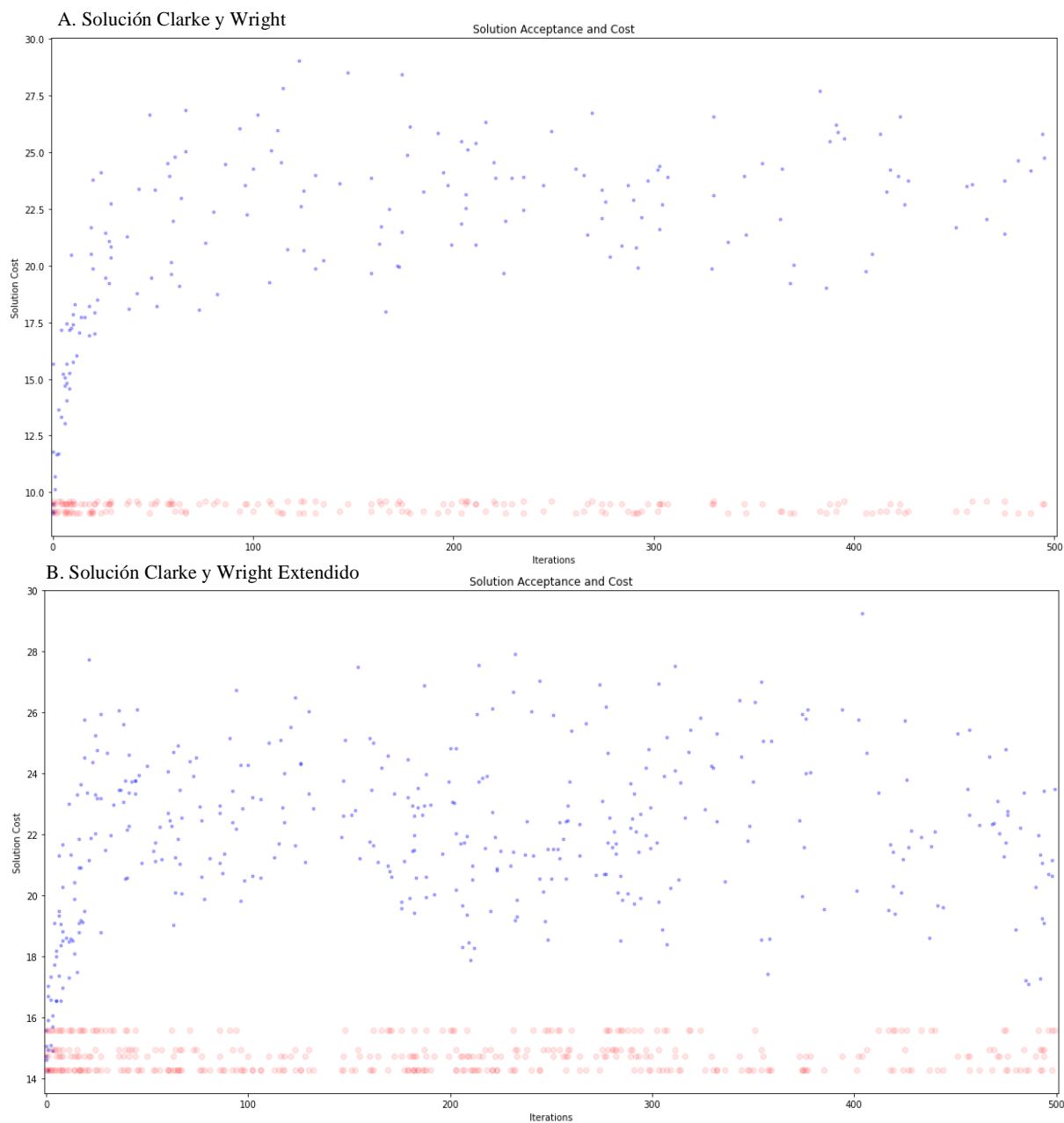


Figura 20. Soluciones aceptadas para la instancia 100.10.4. La figura A. presenta los resultados de la instancia 100.10.4 de 100 clientes para el Clarke y Wright y la B son los resultados de esta misma para el Clarke y Wright extendido. Las gráficas muestran los costos de las soluciones aceptadas y las mejores soluciones en función del recuento de iteraciones.

8.2 Soluciones del ALNS, CWA y VRPD

La Figura 21, presenta los ahorros de la solución obtenida por el ALNS para el VRPD respecto a la solución inicial generada por Clarke y Wright y su versión extendida, en 10 corridas del

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

algoritmo para cada una de las instancias de 20 clientes, así como los ahorros obtenidos para el problema usando solo camiones, bajo las mismas condiciones mencionadas. En la mayoría de los casos, se observa cómo la metaheurística ALNS mejora la solución inicial entre un 5% a 25%, demostrando así la efectividad del algoritmo en relación con salir de los óptimos locales permitiendo obtener mejores soluciones.

Para instancias pequeñas, el gráfico muestra dispersión respecto al ahorro en la solución inicial, esto podría deberse a que ambos métodos de generación de la solución inicial, en ocasiones hallan soluciones casi óptimas, presentando una brecha pequeña en cuanto a la mejor solución generada por el ALNS. Los ahorros bajo el enfoque VRP, es decir, solo camión, suelen estar entre el 50-70% y, a veces, incluso mayor. Es notorio que las instancias más pequeñas tienen mayor variación en el ahorro en relación con la solución del VRP mediante el ALNS. Esto puede deberse al hecho de que al ubicar uno o dos clientes en una instancia pequeña puede tener un gran impacto en el ahorro potencial. Si uno o varios nodos se posicionan distantes del depósito y ambos deben ser atendidos por el camión (debido a entregas pesadas), puede ser imposible obtener un ahorro significativo para esa instancia en comparación con una instancia en la que dichos clientes están menos dispersos en la cuadrícula.

Los GAP con respecto a la solución inicial y en el enfoque de solo camiones se calculan a partir de la siguiente expresión:

$$SVRP = 1 - \frac{Z^{ALNS}}{Z^{VRP}}, \quad SI = 1 - \frac{Z^{ALNS}}{Z^{Inicial}} \quad (42)$$

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

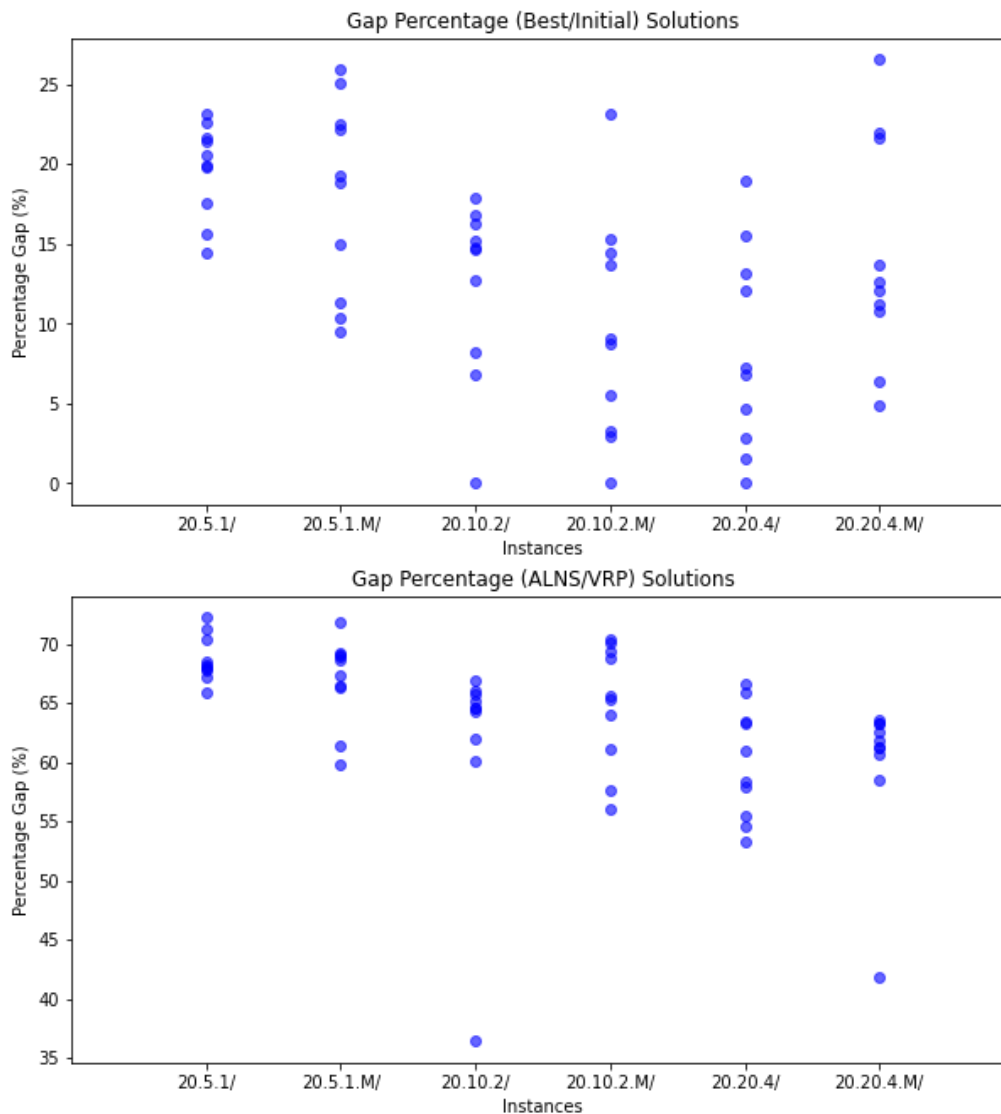


Figura 21. GAP ALNS para 20 clientes, respecto a la solución inicial y VRP para instancias de 20 clientes con CWA y CWA extendido. Arriba: el ahorro de la solución del ALNS para el VRPD respecto a la solución inicial. Abajo: el ahorro de la solución del ALNS respecto a la solución inicial con solo camiones.

Por otro lado, las instancias de gran tamaño presentan una menor diferencia de ahorro entre las mejores soluciones encontrada con el ALNS y las soluciones iniciales (Figura 22). Esto puede deberse a que el CWA y su extensión se basan en los ahorros de costos entre nodos, los cuales se organizan de manera descendente, predeterminando una agrupación de estos, generando algunos nodos en escenarios de difícil inserción; por lo cual, los clientes que se remueven en la sección 6.2 vuelvan a ser insertados en la misma posición en la que se encontraban, ya que los algoritmos de

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

inserción con los que se trabajó tienen como criterio el costo. Con base en lo anterior, se puede considerar que en tamaños de cuadrículas donde los nodos clientes tiene una mayor posibilidad de quedar dispersos entre sí, los algoritmos de inserción tienen pocas opciones diferentes a la inicial de poder incluir uno o dos nodos de manera distinta y generar nuevas rutas.

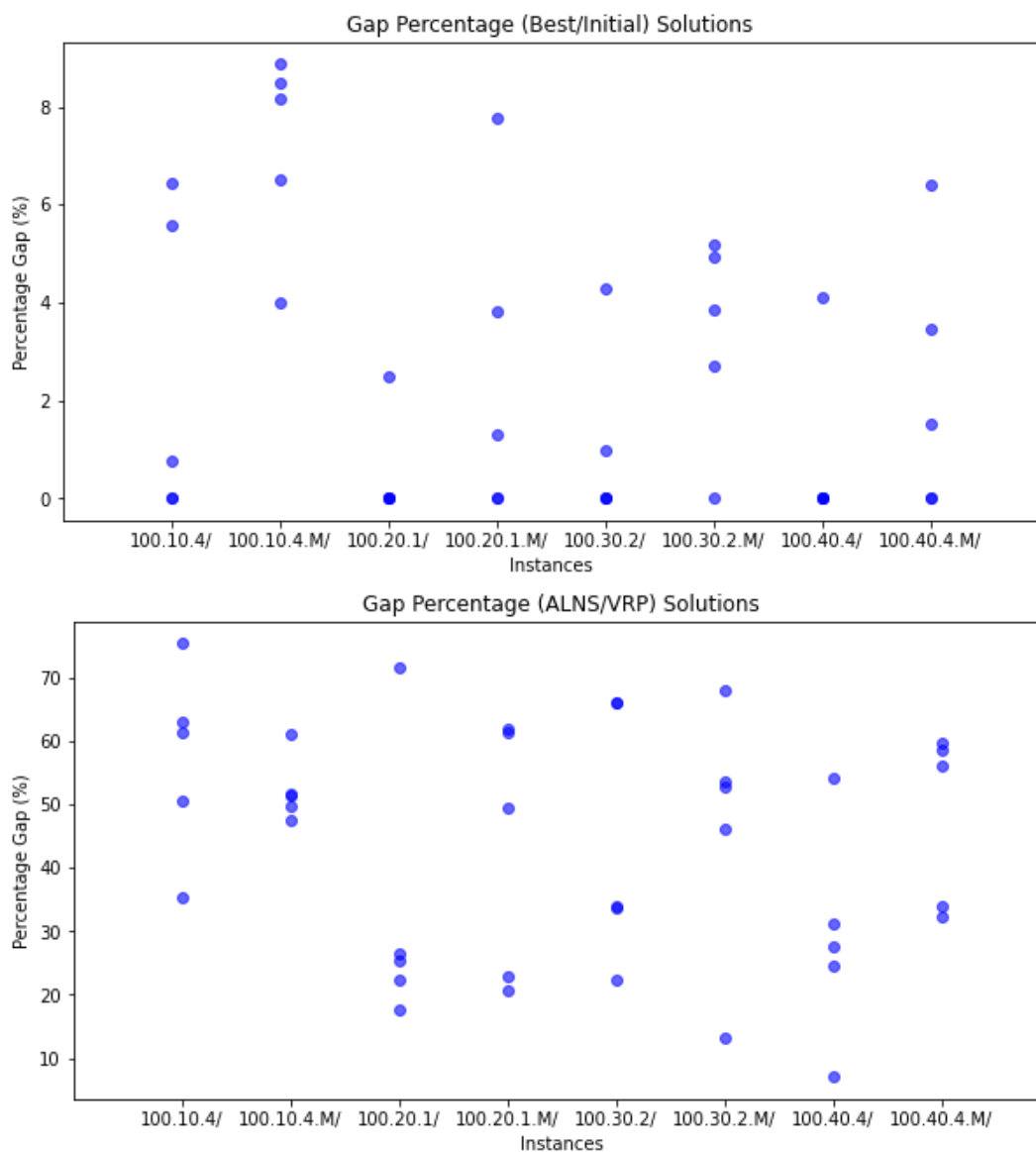


Figura 22. GAP del ALNS para 100 clientes, respecto a la solución inicial y VRP para instancias de 100 clientes con CWA y CWA extendido. Arriba: el ahorro de la solución del ALNS para el VRPD respecto a la solución inicial. Abajo: el ahorro de la solución del ALNS respecto a la solución inicial con solo camiones.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

8.3 Resistencia de vuelo (UAV)

La Figura 23 presenta el porcentaje promedio del uso de la batería del dron, la cual tiende a aumentar a medida que incrementa la dimensión de la cuadrícula que contiene a los clientes, se debe agregar que este aumento está presente en todos los escenarios. Usualmente, para 20 clientes el consumo promedio de la batería del dron no supera el 45%, sin embargo, para 50 clientes el consumo de la batería llega hasta un 90%, además se puede observar que para las instancias de 20 y 50 clientes no se tienen variaciones significativas entre la solución inicial generada por el Clarke y Wright normal y el extendido.

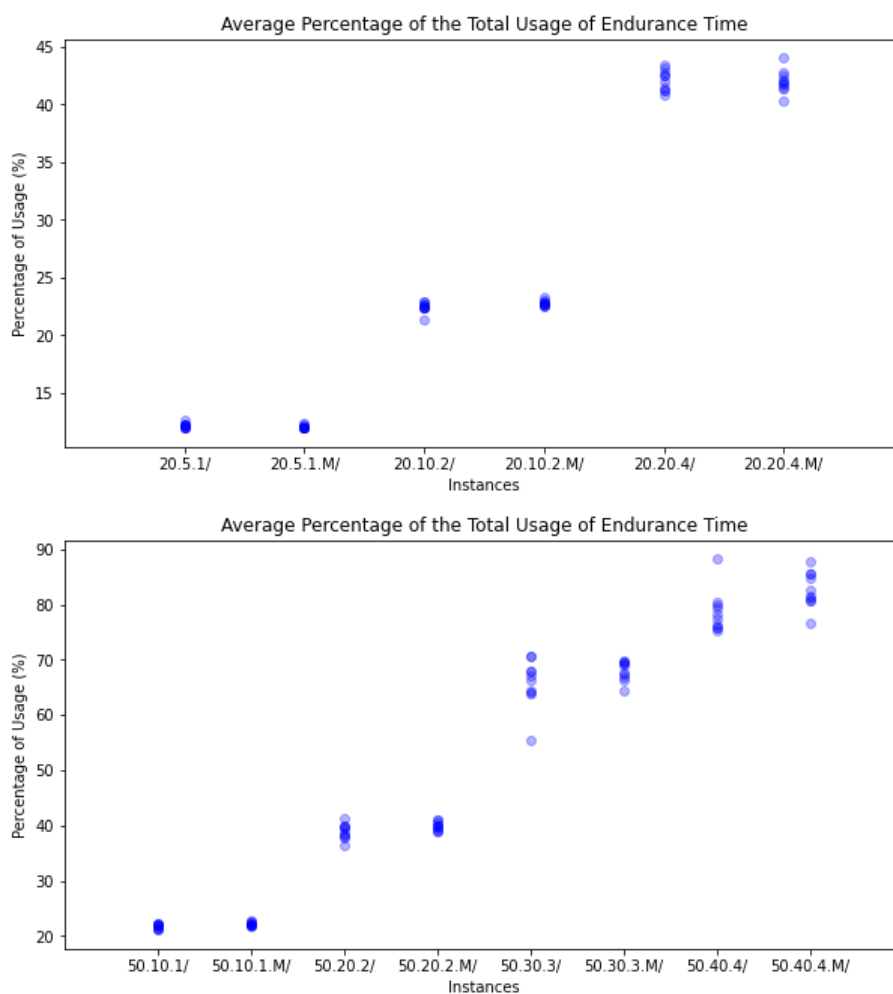


Figura 23. Utilización promedio de la batería del dron. arriba para instancias de 20 clientes, abajo para instancias de 50 clientes.

8.4 Mejores soluciones

En la Figura 24 se representan gráficamente las mejores soluciones encontradas por la metaheurística ALNS para el VRPD y VRP en un escenario con 20 y 100 clientes respectivamente. Cabe destacar que el escenario final para los camiones se ve afectado por la adición de drones en el problema, logrando así que las rutas de los drones tiendan a acortar significativamente los recorridos del camión, enviando al dron a aquellos clientes difíciles de alcanzar.

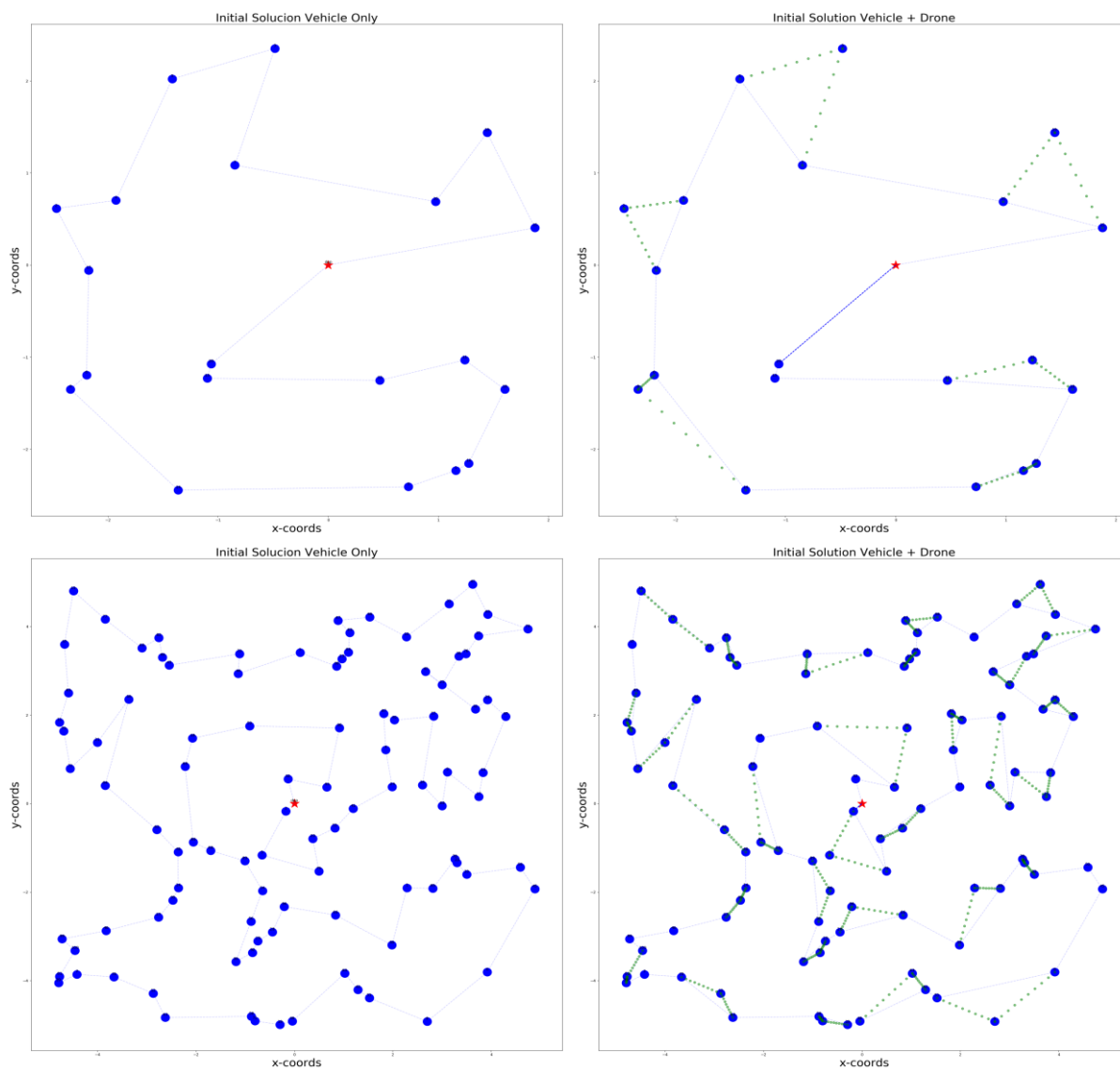


Figura 24. Mejor solución conocida para el VRP y VRPD, para instancias conformadas por 20 y 100 clientes. A la izquierda se tienen las soluciones con el VRP y a la izquierda la solución con VRPD. Las líneas continuas azules indican la ruta hecha por el camión, mientras que las líneas discontinuas verdes los viajes de los drones.

8.5 Velocidad de vuelo (UAV)

En el proceso de entrega otro factor que puede incidir es el tiempo, este se calcula mediante la velocidad del dron y la distancia recorrida por este. El valor que se estableció para la velocidad del dron inicialmente fue de 50 mph y es interesante estudiar qué sucedería si el dron operara a velocidades más bajas a causa de la carga que va a llevar este y la influencia que el aire impone sobre él. Al disminuir la velocidad del dron, se amplía el tiempo que le va a tomar a este visitar los nodos, considerando que no puede violar las restricciones establecidas para la capacidad de la batería dada en tiempo de vuelo, es posible que no pueda alcanzar a todos los clientes que atiende en las soluciones actuales que abordan el parámetro fijo de velocidad. Por esta razón se decide evaluar la ecuación de velocidad propuesta en esta investigación en la sección 5.

$$V_{dc} = c_m(V_{fd}) - \bar{v}_v$$

$$V_{dd} = (V_{fd}) - \bar{v}_v$$

Se revisan distintas combinaciones de valores para los parámetros que hacen parte de la ecuación de velocidad y se calcula el tiempo que le tomaría al dron visitar la ruta establecida en la solución de la instancia con la nueva velocidad de vuelo. Como se ve en la Figura 25, la velocidad del dron tiene un impacto significativo a la hora de aceptar un conjunto de salidas; debido a que no podrá llegar a un cliente si este se encuentra a un tiempo de viaje que sobrepasa el tiempo de resistencia de vuelo, el cual es de 21 minutos (el 70% de los 30 minutos de capacidad del batería establecido). Adicionalmente la función objetivo se ve involucrada si no se pueden asignar drones a dichos clientes. Es decir, si disminuye la velocidad del dron, los ahorros que se generan en el VRPD con respecto al caso de solo camiones disminuyen de manera proporcional a la velocidad del dron. Sin embargo, la colaboración de ambos vehículos ofrece un beneficio notable incluso al operar drones a velocidades inferior que la de los camiones.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Cuadrícula de 5x5											
Ruta	Instancia	$m_t + m_b$	m_l	c_m	\bar{v}_v	V_{fd}	V_{dc}	d	t_v	t	T_{Max}
1-3-2	6.5.1	15	0.8688	0.9453	6.211	50	41.0514	7.504	10.9682	9.005	21
5-6-4	6.5.1	15	1.3078	0.9198	3.106	50	42.8846	5.747	8.0406	6.896	21
9-7-8	10.5.2	15	0.8788	0.9447	4.348	50	42.8849	7.744	10.8352	9.293	21
10-2-3	10.5.2	15	1.3377	0.9181	4.969	50	40.9371	6.024	8.8289	7.229	21
1-6-5	10.5.2	15	1.5825	0.9046	3.106	50	42.1229	7.150	10.1847	8.580	21
3-11-10	12.5.4	15	1.38	0.916	3.7	50	42.073	5.0241354	7.165	6.0289624	21
5-12-1	12.5.4	15	1.38	0.916	4.3	50	41.444	9.7218163	14.075	11.66618	21
9-7-8	12.5.4	15	0.01	1.000	4.3	50	45.634	5.3976888	7.097	6.4772266	21
6-2-4	12.5.4	15	0.77	0.951	6.2	50	41.353	3.304111	4.794	3.9649332	21
7-15-4	20.5.1	15	2.31	0.867	1.9	50	41.476	6.6261941	9.586	7.951433	21
2-8-17	20.5.1	15	1.21	0.925	1.9	50	44.391	3.2692179	4.419	3.9230614	21
6-16-20	20.5.1	15	1.75	0.896	1.9	50	42.924	6.3797807	8.918	7.6557368	21
13-3-14	20.5.1	15	1.50	0.909	1.9	50	43.601	8.5076031	11.707	10.209124	21
1-9-10	20.5.1	15	0.63	0.960	1.9	50	46.113	1.070047	1.392	1.2840564	21
18-5-11	20.5.1	15	0.58	0.963	1.9	50	46.274	8.5722374	11.115	10.286685	21
12-19-0	20.5.1	15	2.23	0.870	1.9	50	41.654	4.896361	7.053	5.8756332	21

Cuadrícula 20x20											
Ruta	Instancia	$m_t + m_b$	m_l	c_m	\bar{v}_v	V_{fd}	V_{dc}	d	t_v	t	T_{Max}
8-11-3	12.20.1	15	2.1173	0.8763	1.863	50	41.9519	21.209	30.3335	25.451	21
6-12-2	12.20.1	15	0.8808	0.9445	1.863	50	45.3635	34.329	45.4050	41.195	21
7-9-1	12.20.1	15	1.0471	0.9347	1.863	50	44.8741	29.494	39.4353	35.392	21
5-4-0	12.20.1	15	2.0596	0.8793	1.863	50	42.1002	15.715	22.3972	18.859	21
10-6-14	20.20.4	15	0.43	0.972	1.9	50	46.739	22.299556	28.627	26.759467	21
2-1-12	20.20.4	15	1.78	0.894	1.9	50	42.829	12.524339	17.545	15.029207	21
16-19-9	20.20.4	15	0.93	0.941	1.9	50	45.207	12.667636	16.813	15.201163	21
3-15-7	20.20.4	15	1.83	0.891	1.9	50	42.700	9.0906719	12.774	10.908806	21
8-5-17	20.20.4	15	2.15	0.875	1.9	50	41.865	20.27599	29.059	24.331189	21
20-11-4	20.20.4	15	0.29	0.981	1.9	50	47.195	12.466589	15.849	14.959907	21
18-13-0	20.20.4	15	0.68	0.957	1.9	50	45.966	1.4595915	1.905	1.7515098	21

Figura 25. Resultados tiempos de vuelo del dron al variar su velocidad, probado para 4 instancias de tamaños 6, 10, 12 y 20 clientes, se tienen las rutas que hace el dron dentro de cada instancia. Siendo $m_t + m_b$ la masa del dron (incluyendo batería cargada), m_l la masa del paquete, c_m el factor de carga, \bar{v}_v la velocidad promedio del viento, V_{fd} la velocidad de la ficha del dron, d la distancia empleada en la ruta, t el tiempo sin incluir la influencia del viento y el factor de carga y por último t_v , el tiempo contemplando estos factores.

El artículo científico de carácter publicable con los resultados del presente trabajo de investigación se adjunta en el Apéndice F.

9. Conclusiones

Según la revisión de literatura, son pocos los trabajos de investigación que tratan el problema de ruteo de vehículos con drones (VRPD); por lo tanto, es un tema de investigación que puede ser considerado nuevo e innovador dentro del área del ruteo de vehículos, con múltiples aplicaciones reales.

La influencia que tiene el estudio de los drones en los procesos de distribución de última milla tiene un efecto sumamente positivo. Las nuevas aplicaciones tecnológicas logran desplegar un sin

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

número de intereses en las compañías que a su vez despiertan la atención de los investigadores en plantear tanto cuantiosos modelos matemáticos como estrategias que guíen a la resolución de estos, y, por ende, aportar un amplio beneficio que proviene de la sinergia del VRPD, esto se puede conseguir gracias a la incorporación de los vehículos aéreos no tripulados en las entregas de mercancías, ya que las rutas que se conforman ahorran tiempo, costos y contribuyen a una programación de una forma más eficiente.

Es posible evidenciar que la corriente de investigación respecto al uso de drones en la distribución de última milla tiene tres vertientes, la primera consiste en que los camiones lanzan y recuperan a los drones en un mismo nodo; en la segunda los camiones lanzan a los drones en un nodo cliente y este es recuperado en otro nodo cliente de la ruta; y la tercera es que los drones pueden ser lanzados y recuperados en cualquier punto de la ruta de los vehículos, lo que ofrece una oportunidad para investigar el uso adecuado del dron dentro del ruteo de vehículos.

Debido a que en las primeras fases la metaheurística ALNS cuenta con un algoritmo de ahorros (CWA) el cual logra encontrar buenas soluciones iniciales, esto influye en gran medida a la aplicación de las heurísticas de destrucción y reparación en instancias pequeñas. De modo que es difícil llegar a mejores soluciones en pocas iteraciones, originando que durante un número significativo de corridas no se encontraran soluciones que fueran aceptadas por tener un costo mayor al generado inicialmente, lo que ocasionaba que la temperatura inicial no disminuyera, lo cual generaba que el algoritmo no avanzara pues no se cumplía el criterio de parada de la temperatura final; por esto, fue necesario establecer un número máximo de iteraciones como criterio adicional de parada; es decir que las instancias pequeñas del problema presentaron serias

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

dificultades para llegar a la temperatura final del recocido simulado, debido a esto se decidió aplicar como criterio de parada un número finito de iteraciones.

Es posible afirmar que la combinación de las heurísticas de destrucción y reparación en la construcción del algoritmo ALNS favorecen en gran medida a que se exploren mayores soluciones en el espacio de búsqueda, esto se debe gracias a que estas cuentan con un componente de aleatoriedad y la adaptabilidad a medida que transcurre el algoritmo.

En la validación del algoritmo se encontró que los valores comparativos de las soluciones iniciales con respecto al mejor valor conocido no sobrepasan el 10.95% en instancias de 200 clientes, además en las instancias de 100 clientes los valores comparativos son como máximo un 8.68% mejores, se puede inferir que este desempeño se debe a las heurísticas seleccionadas y los parámetros que se establecieron en el algoritmo; por lo anterior, se puede observar que el algoritmo ALNS presenta un mejor desempeño cuando se tienen instancias pequeñas; en las de 20 clientes se tiene una mejora de hasta un 26.53% y en las de 12 clientes de hasta un 19.79%. Finalmente, el desempeño del algoritmo no es eficiente en cuestión al tiempo computacional empleado en la solución de instancias de gran tamaño.

Finalmente, en las pruebas realizadas con la velocidad de vuelo del dron en las instancias pequeñas, se encontró que al incluir un factor de carga y la velocidad del viento se influía directamente sobre la decisión de lanzar un dron o no, debido a que al tener en cuenta estas consideraciones, el tiempo de vuelo necesario para completar la entrega era mayor a la resistencia de vuelo del dron, lo que podría genera que un dron deje de volar antes de que el camión lo recuperara. Esto puede ser perjudicial si consideran escenarios en los que se deben incurrir sobrecostos ocasionados por caídas del dron y pérdida de los paquetes.

10. Recomendaciones

Para futuras investigaciones se podrían explorar las otras configuraciones que existen entre camiones y drones, planteando además problemas y escenarios más cercanos a la realidad con la consideración de entregas múltiples, entregas parciales y problemas con una flota mixta de drones, identificando si existen cambios al evaluar la función objetivo.

Abordar en las futuras investigaciones el comportamiento del ALNS en problemas que implican ventanas de tiempo, combinando tanto el servicio orientado al tiempo como el enfoque orientado a la distancia. Utilizando así dentro del algoritmo heurísticas de remoción, que permitan combinar estos dos enfoques, de tal manera que las heurísticas de reparación logren generar rutas que mantengan un equilibrio entre la cercanía de los clientes y sus periodos de atención. Adicionalmente Se recomienda hacer un diseño de experimentos para evaluar el efecto que tiene la variación de algunos parámetros del algoritmo como la tasa de destrucción, el factor de reacción y los puntajes asignados cuando una solución es aceptada.

Investigar sobre posibles algoritmos de búsqueda local simple y operadores de mutación que comparen todas las combinaciones válidas posibles de los mecanismos de intercambio en las rutas, generando así un aporte de mayor rendimiento a las rutas que se le ofrecen al algoritmo ALNS.

El presente trabajo de investigación presentó limitaciones con respecto a la capacidad computacional disponible para realizar la validación de las instancias, se encontró que, al tratarse de un algoritmo robusto, debido a que se compone de varias heurísticas, los tiempos de ejecución fueron un factor limitante en problemas de gran tamaño, ya que el número de posibilidades de inserción en un conjunto de nodos removidos es directamente proporcional a la cantidad de clientes que se encuentran en este. Por esta razón si lo que se requiere es dar una respuesta inmediata en el

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

campo que se esté efectuando la investigación, lo más recomendable sería usar tecnología de punta para dar una pronta respuesta del problema que se esté considerando.

Referencias Bibliográficas

- Agatz, N., Bouman, P., & Schmidt, M. (2018). Optimization Approaches for the Traveling Salesman Problem with Drone. *Transportation Science*.
<https://doi.org/10.1287/trsc.2017.0791>
- Allain, R. (2013). *Physics of the Amazon Octocopter Drone*. Wired.
<https://www.wired.com/2013/12/physics-of-the-amazon-prime-air-drone/>
- Alotaibi, K. A., Rosenberger, J. M., Mattingly, S. P., Punugu, R. K., & Visoldilokpun, S. (2018). Unmanned aerial vehicle routing in the presence of threats. *Computers and Industrial Engineering*. <https://doi.org/10.1016/j.cie.2017.10.030>
- Amazon.com, I. or its affiliates. (2016). *Amazon.com: Prime Air*.
<https://www.amazon.com/b?node=8037720011>
- Baker, B. M., & Ayechev, M. A. (2006). A genetic algorithm for the vehicle routing problem. *Beijing Huagong Daxue Xuebao (Ziran Kexueban)/Journal of Beijing University of Chemical Technology (Natural Science Edition)*, 33(4), 84–87. [https://doi.org/10.1016/S0305-0548\(02\)00051-5](https://doi.org/10.1016/S0305-0548(02)00051-5)
- Balfour Beatty. (2017). *Flying into the future of bridge inspections - News - Balfour Beatty plc*.
<https://www.balfourbeatty.com/news/flying-into-the-future-of-bridge-inspections/>
- Baños Navarro, R. (2006). *Meta-heurísticas híbridas para optimización mono-objetivo y multi-objetivo: paralelización y aplicaciones*.
- Benito Quintanilla, A. (2015). *Problemas de Rutas de Vehículos: modelos, aplicaciones logísticas y métodos de resolución*. 128. <http://uvadoc.uva.es/handle/10324/13287>
- Blum, C., & Roli, A. (2003). Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison Metaheuristics in Combinatorial Optimization. En *ACM Computing*

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

Surveys (Vol. 35, Número 3).

https://www.iiia.csic.es/~christian.blum/downloads/blum_rolis_2003.pdf

Bonn. (2014). *DHL / Press Release / English.*

https://www.dhl.com/en/press/releases/releases_2014/group/dhl_parcelcopter_launches_initial_operations_for_research_purposes.html

Boysen, N., Briskorn, D., Fedtke, S., & Schwerdfeger, S. (2018). Drone delivery from trucks:

Drone scheduling for given truck routes. *Networks*. <https://doi.org/10.1002/net.21847>

Brown Eric. (2019). E-commerce spurs innovation in last-mile logistics. *MIT News*.

<http://news.mit.edu/2018/mit-e-commerce-spurs-innovations-last-mile-logistics-0904>

Černý, V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient

simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1), 41–51.

<https://doi.org/10.1007/BF00940812>

Clarke, G., & Wright, j. w. (1964). *Scheduling of Vehicles from a Central Depot to a Number of*

Delivery Points. 12, 568–581. <https://www.jstor.org/stable/167703>

Coelho, B. N., Ochi, L. S., Haghazari K., R., Zuidema, D., Lima, M. S. ., & da Costa, A. R. (2017).

A multi-objective green UAV routing problem. *Computers and Operations Research*.

<https://doi.org/10.1016/j.cor.2017.04.011>

Colomi, A., Dorigo, M., & Maniezzo, V. (1991). Distributed Optimization by Ant Colonies.

Conference: Proceedings of ECAL91 - European Conference on Artificial Life.

https://www.researchgate.net/publication/216300484_Distributed_Optimization_by_Ant_Colonies

Cordeau, J.-F. O., Gendreau, M., & Laporte, G. (1997). *A Tabu Search Heuristic for Periodic and*

Multi-Depot Vehicle Routing Problems (Vol. 30). John Wiley & Sons, Inc. Networks.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Cruz Jimenez, C., & Gonzalez Villamizar, L. J. (2013). *Desarrollo de un algoritmo hibrido para la resolucio del problema de ruteo de vehiculos con entrega y recogida simultaneas (VRPSPD)* [Universidad Industrial de Santander].
<http://tangara.uis.edu.co/biblioweb/tesis/2013/149648.pdf>
- Daneshzand, F. (2011). The Vehicle-Routing Problem. En *Logistics Operations and Management* (pp. 127–153). Elsevier Inc. <https://doi.org/10.1016/B978-0-12-385202-1.00008-6>
- Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Source: Management Science*, 6(1), 80–91.
- ECN. (2018). *¿Qué Es La Última Milla? Analizamos Sus Problemas Y Sus Costes*.
<https://www.ecommerce-nation.es/que-es-ultima-milla-problemas-costes-ecommerce/>
- El tiempo. (2017). *Cali, pionera en el uso de drones para uso médico en la ladera - Cali - Colombia - ELTIEMPO.COM*.
- Emeç, U., Çatay, B., & Bozkaya, B. (2016). An Adaptive Large Neighborhood Search for an E-grocery Delivery Routing Problem. *Computers and Operations Research*, 69, 109–125.
<https://doi.org/10.1016/j.cor.2015.11.008>
- Ercan, C., & Gencer, C. (2018). *A Decision Support System for Dynamic Heterogeneous Unmanned Aerial System Fleets*. <http://dergipark.gov.tr/gujs>
- Erdoğan, S., & Miller-Hooks, E. (2012). A Green Vehicle Routing Problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1), 100–114.
<https://doi.org/10.1016/j.tre.2011.08.001>
- Faulin, J. (2003). Applying MIXALG procedure in a routing problem to optimize food product delivery. *Omega*, 31(5), 387–395. [https://doi.org/10.1016/S0305-0483\(03\)00079-3](https://doi.org/10.1016/S0305-0483(03)00079-3)
- Ferrandez, S. M., Harbison, T., Weber, T., Sturges, R., & Rich, R. (2016). Optimization of a truck-

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- drone in tandem delivery network using k-means and genetic algorithm. *Journal of Industrial Engineering and Management*. <https://doi.org/10.3926/jiem.1929>
- Figliozzi, M. A. (2017). Lifecycle modeling and assessment of unmanned aerial vehicles (Drones) CO2e emissions. *Transportation Research Part D: Transport and Environment*. <https://doi.org/10.1016/j.trd.2017.09.011>
- Gallego Rendón, R. A., Escobar Zuluaga, A., & Romero Lazaro, R. A. (2006). *Técnicas de optimización combinatorial*. Universidad Tecnológica de Pereira.
- Gaskell, T. J. (1967). Bases for Vehicle Fleet Scheduling. *Journal of the Operational Research Society*, 18(3), 281–295. <https://doi.org/10.1057/jors.1967.44>
- Gevaers, R. (2013). *Evaluation of innovations in B2C last mile*.
- Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5), 533–549. [https://doi.org/10.1016/0305-0548\(86\)90048-1](https://doi.org/10.1016/0305-0548(86)90048-1)
- Góez Sánchez, G. D. (2016). *Metodología para el planeamiento automático de rutas en vehículos aéreos no tripulados usando algoritmos bio-inspirados sobre sistemas embebidos*. INSTITUTO TECNOLÓGICO METROPOLITANO-ITM.
- Golden, B., Assad, A., Levy, L., & Gheysens, F. (1984). The fleet size and mix vehicle routing problem. *Computers and Operations Research*, 11(1), 49–66. [https://doi.org/10.1016/0305-0548\(84\)90007-8](https://doi.org/10.1016/0305-0548(84)90007-8)
- Goodman, R. W. (2005). Whatever You Call It, Just Don't Think of Last-mile Logistics, Last. *Global Logistics and Supply Chain Strategies*, December, 1–6.
- Grupo Valora. (2016). *La última milla: punto crítico de la logística en ecommerce - Grupo Valora*. <https://www.grupovalora.es/blog/la-ultima-milla-lo-que-no-sabes-sobre-la-logistica-en->

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

ecommerce/

- Ham, A. M. (2018a). Integrated scheduling of m-truck, m-drone, and m-depot constrained by time-window, drop-pickup, and m-visit using constraint programming. *Transportation Research Part C: Emerging Technologies*. <https://doi.org/10.1016/j.trc.2018.03.025>
- Ham, A. M. (2018b). Test instances and CP log file. *Transportation Research Part C*. https://drive.google.com/file/d/1ooWstfwP4wQ7qHAs_CY6otXZ904io0Ag/view
- Hillier, F. S., & Lieberman, G. J. (2013). *Introducción a la Investigación de operaciones*. <https://doi.org/10.1017/CBO9781107415324.004>
- Homberger, J., & Gehring, H. (2005). A two-phase hybrid metaheuristic for the vehicle routing problem with time windows. *European Journal of Operational Research*, 162(1), 220–238. <https://doi.org/10.1016/j.ejor.2004.01.027>
- Keskin, M., & Catay, B. (2016). Partial recharge strategies for the electric vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies*, 65, 111–127. <https://doi.org/10.1016/j.trc.2016.01.013>
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671–680. <https://doi.org/10.1126/science.220.4598.671>
- Laporte, G., Ropke, S., & Vidal, T. (2014). *Classical Heuristics for the Vehicle Routing*. 87–116.
- Lara, E. R., María, I., & Barrera, A. (2016). RENTOL: Un algoritmo de agrupamiento basado en K-means RENTOL: A Clustering Algorithm Based on K-means. En *Research in Computing Science* (Vol. 128).
- Li, W., Wu, Y., Kumar, P. N. R., & Li, K. (2019). Engineering Optimization Multi-trip vehicle routing problem with order release time Multi-trip vehicle routing problem with order release time. *Optimización de ingeniería*. <https://doi.org/10.1080/0305215X.2019.1642880>

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Lin, S., & Kernighan, B. W. (1973). Effective heuristic algorithm for the traveling-salesman problem. *Operations Research*, *21*(2), 498–516. <https://doi.org/10.1287/opre.21.2.498>
- Lutz, R. (2014). *Adaptive Large Neighborhood Search*. <https://d-nb.info/1072464683/34>
- Manalo, M. (2015). *EasyJet UK utiliza drones para la inspección de aviones - Business Insider*. <https://www.businessinsider.com/easyjet-uk-uses-drones-airplane-inspection-2015-6>
- Marinelli, M., Caggiani, L., Ottomanelli, M., & Dell’Orco, M. (2018). En route truck–drone parcel delivery for optimal vehicle routing strategies. *IET Intelligent Transport Systems*. <https://doi.org/10.1049/iet-its.2017.0227>
- Megan Willett. (2013). A London Sushi Chain Delivers Dinner By Drone. *Business Insider*. <https://www.businessinsider.com/yo-sushi-debuts-flying-trays-2013-6>
- Mesa, V. G., Barajas, D. P., Rajabifard, A., & Puccini, M. S. (2016). *The colombian emergency response platform (PCRE): design and testing of a sdi-based system to facilitate disaster management in Colombia* [Universidad de los Andes]. <http://biblioteca.uniandes.edu.co/acepto201699.php?id=10284.pdf>
- Murray, C. C., & Chu, A. G. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies*. <https://doi.org/10.1016/j.trc.2015.03.005>
- Observatorio eCommerce & Transformación Digital Centro de Estudios EY. (2017). *Informe de evolución y perspectivas eCommerce 2017*. <https://observatorioecommerce.com/informe-evolucion-perspectivas-ecommerce-2017/>
- Osman, I. H. (1993). Metastrategy simulated annealing and tabu search for combinatorial optimization problems. *Annals of Operations Research*, *41*, 421–451. <http://hdl.handle.net/10044/1/7596>

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Overly, S. (2017). *A drone carrying humans prepares to take flight in Dubai - The Washington Post*. <https://www.washingtonpost.com/news/innovations/wp/2017/02/15/a-drone-carrying-human-passengers-prepares-to-take-flight-in-dubai/?noredirect=on>
- Paessens, H. (1988). The savings algorithm for the vehicle routing problem. *European Journal of Operational Research*, 34(3), 336–344. [https://doi.org/10.1016/0377-2217\(88\)90154-3](https://doi.org/10.1016/0377-2217(88)90154-3)
- Pereira, F. B., Tavares, J., Machado, P., & Costa, E. (2002). GVR: A new genetic representation for the vehicle routing problem. *Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science)*, 2464, 95–102.
- Pérez-Jiménez, M. J., & Sancho-Caparrini, F. (2003). Máquinas moleculares basadas en ADN. *Colección de divulgación científica*, 2(1866), 255. <http://www.cs.us.es/~marper/investigacion/libroadn1.html>
- Phan, A. T., Nguyen, T. D., & Pham, Q. D. (2018). Traveling salesman problem with multiple drones. *ACM International Conference Proceeding Series*, 46–53. <https://doi.org/10.1145/3287921.3287932>
- Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8), 2403–2435. <https://doi.org/10.1016/j.cor.2005.09.012>
- Pisinger, D., & Ropke, S. (2010). *Large Neighborhood Search* (pp. 399–419). Springer, Boston, MA. https://doi.org/10.1007/978-1-4419-1665-5_13
- Pix4D. (2017). “Drone mapping saves lives: Canadian RescUAV team helps 4000 people return home after fatal landslide in Colombia”. <https://www.pix4d.com/blog/drone-mapping-saves-lives-rescuav-landslide-in-colombia>
- Poikonen, S., Wang, X., & Golden, B. (2017). The vehicle routing problem with drones: Extended models and connections. *Networks*. <https://doi.org/10.1002/net.21746>

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Ponza, A. (2016). *Optimization of Drone-Assisted Parcel Delivery*. 80.
- Rand, G. (2009). The life and times of the Savings Method for Vehicle Routing Problems. *ORiON*, 25(2). <https://doi.org/10.5784/25-2-78>
- Reid, D. (2016). *Domino's delivers world's first ever pizza by drone*. <https://www.cnbc.com/2016/11/16/dominos-has-delivered-the-worlds-first-ever-pizza-by-drone-to-a-new-zealand-couple.html>
- Ropke, S. (2006). *General rights Heuristic and exact algorithms for vehicle routing problems*.
- Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4), 455–472. <https://doi.org/10.1287/trsc.1050.0135>
- Rose, C. (2013). *Jeff Bezos de Amazon mira hacia el futuro - CBS News*. <https://www.cbsnews.com/news/amazons-jeff-bezos-looks-to-the-future/>
- Ruiz García, R. (2004). *Técnicas metaheurísticas para la programación flexible de la producción* (M. C. Maroto Álvarez & Universidad Politécnica de Valencia Departamento de Estadística e Investigación Operativa Aplicadas y Calidad (eds.)) [Document]. Editorial UPV.
- Sacramento, D., Pisinger, D., & Ropke, S. (2018). *Vehicle Routing Problem with Drones Instances*. <https://zenodo.org/record/2572764#.XVXs-Oj0nIV>
- Sacramento, D., Pisinger, D., & Ropke, S. (2019). An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. *Transportation Research Part C: Emerging Technologies*. <https://doi.org/10.1016/j.trc.2019.02.018>
- Sánchez, E. M., & Penín, M. L. (2003). Inteligencia artificial: revista iberoamericana de inteligencia artificial. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 11(33), 7–12. <http://redalyc.uaemex.mx>

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Schermer, D., Moeini, M., & Wendt, O. (2019). A matheuristic for the vehicle routing problem with drones and its variants. *Transportation Research Part C: Emerging Technologies*, 106, 166–204. <https://doi.org/10.1016/j.trc.2019.06.016>
- Schiffer, M., & Walther, G. (2017). An adaptive large neighborhood search for the location-routing problem with intra-route facilities. *Transportation Science*, 52(2), 331–352. <https://doi.org/10.1287/trsc.2017.0746>
- Schneider, M., Stenger, A., & Goeke, D. (2014). The Electric Vehicle-Routing Problem with Time Windows and Recharging Stations. *Transportation Science*, 48(4), 500–520. <https://doi.org/10.1287/trsc.2013.0490%0D>
- Shaw, P. (1997). *A new local search algorithm providing high quality solutions to vehicle routing problems*. <https://pdfs.semanticscholar.org/8ac9/d5287ede9463d33cc0a1daf4191a5e34608d.pdf>
- Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1520, 417–431. https://doi.org/10.1007/3-540-49481-2_30
- Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2), 254–265. <https://doi.org/10.1287/opre.35.2.254>
- Stewart J. (2014). *Google tests drone deliveries in Project Wing trials - BBC News*. <https://www.bbc.com/news/technology-28964260>
- Taillard, E. (1993). Parallel Iterative Search Methods for Vehicle Routing Problems. *Networks*, 23, 661–673.

UN AG ALNS PARA EL VRPD EN LA DISTRIBUCIÓN DE UM

- Toth, P., & Vigo, D. (2000). An Overview of Vehicle Routing Problems. *SIAM*, 1–26.
<https://epubs.siam.org/doi/abs/10.1137/1.9780898718515.ch1>
- Toth, P., & Vigo, D. (2014). *Vehicle Routing: Problems, Methods, and Applications*, Second Edition. En *SIAM*. <https://research.vu.nl/en/publications/vehicle-routing-problems-methods-and-applications-second-edition>
- Trop, J. (2016). *La entrega de drones está a punto de remodelar el transporte por carretera*.
<https://www.trucks.com/2016/06/21/drone-delivery-reshape-trucking/>
- UPS. (2017). *Peso y tamaño: UPS - Estados Unidos*. <https://www.ups.com/us/en/help-center/packaging-and-supplies/weight-size.page>
- W. Hämäläinen. (2006). Class NP, NP-complete, and NP-hard problems. *Sort*, 1–7.
- Wang, X., Poikonen, S., & Golden, B. (2017). *The vehicle routing problem with drones: several worst-case results*. *II(4)*, 679–697. <https://doi.org/10.1007/s11590-016-1035-3>
- Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. *Transportation Research Part B: Methodological*. <https://doi.org/10.1016/j.trb.2019.03.005>
- Wen, T., Zhang, Z., & Wong, K. K. L. (2016). Multi-objective algorithm for blood supply via unmanned aerial vehicles to the wounded in an emergency situation. *PLoS ONE*.
<https://doi.org/10.1371/journal.pone.0155176>
- Yellow, P. C. (1970). A Computational Modification to the Savings Method of Vehicle Scheduling. *Journal of the Operational Research Society*, 21(2), 281–283.
<https://doi.org/10.1057/jors.1970.52>