

**Un modelo para la localización de estaciones de intercambio de batería y ruteo de  
vehículos eléctricos**

**John Freddy Acevedo Guarín**

**Erik Danobis Cáceres Entralgo**

**Proyecto de Grado para optar el título de Ingeniero Industrial**

**Director:**

**Karin Julieth Aguilar Imitola**

**M. Sc. En Ingeniería Industrial**

**Codirector(es):**

**Henry Lamos Diaz**

**Ph.D. en Física – Matemáticas**

**Daniel Orlando Martinez Quezada**

**M. Sc. En Ingeniería Industrial**

**Universidad Industrial de Santander**

**Facultad de Ingenierías Físicomecánicas**

**Escuela de Estudios Industriales y Empresariales**

**Bucaramanga**

**2019**

## DEDICATORIA

*A mi madre y mis abuelos por todo su esfuerzo, paciencia y motivación durante esta etapa de formación académica.*

*A Maryerly Jiménez por su amor, compañía y apoyo incondicional.*

*A mis amigos de infancia y compañeros de carrera que sin su compañía y apoyo este trayecto no hubiera sido el mismo.*

**John Freddy Acevedo Guarín**

*A Dios por darme sabiduría, fortaleza y bendiciones en todo momento.*

*A mis padres Luis Cáceres y Teresa Entralgo por el esfuerzo que hicieron en apoyarme durante toda mi carrera.*

*A Tania Arenas por su amor y apoyo incondicional.*

*A mis familiares y amigos que me acompañaron en mi formación universitaria.*

**Erik Danobis Cáceres Entralgo**

### **Agradecimientos**

A los profesores Karin Aguilar y Henry Lamos por su exigencia, compromiso y guía en el desarrollo de este trayecto.

Al grupo de investigación OPALO por su colaboración.

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 ..... 16

1. Planteamiento del problema ..... 18

2. Justificación..... 20

3. Objetivos..... 21

3.1 Objetivo general ..... 21

3.2 Objetivos específicos..... 21

4. Revisión de la literatura..... 22

5. Marco teórico..... 27

5.1 Logística en transporte de mercancías..... 27

5.1.1 Complejidad computacional..... 28

5.1.2 Métodos de solución de un problema de optimización combinatoria..... 29

5.1.2.1 Algoritmos exactos..... 29

5.1.2.2 Algoritmos heurísticos. .... 29

5.1.2.2.1 Heurísticas Constructivas. .... 29

5.1.2.2.2 Métodos de búsqueda local..... 30

5.1.2.3 Algoritmos metaheurísticos. .... 30

5.1.3 Problema de Localización - Ruteo (LRP). .... 30

5.1.4 Problema de Ruteo de Vehículos. .... 30

5.1.5 Problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos (BSS-EV-LRP)..... 31

5.1.5.1 Problema de localización de estaciones de intercambio de batería..... 31

5.1.5.2 Problema de ruteo de vehículos eléctricos (EVRP)..... 32

5.1.6 Clarke & Wright Algorithm..... 32

5.1.7 Métodos de solución BSS-EV-LRP..... 33

5.1.7.1 Heurística ALNS..... 33

5.1.7.2 Heurística AVNS..... 34

6. Problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos ..... 35

6.1 Descripción del problema..... 35

6.2 Formulación del problema..... 35

6.1.1 Función Objetivo..... 37

6.1.2 Restricciones ..... 38

7 AVNS para la solución del BSS-EV-LRP..... 40

7.1 Fase de inicialización ..... 43

7.1.1 Solución inicial ..... 43

7.1.1.1 Algoritmo de Clark and Wright..... 43

7.1.2 Conjunto de estructuras vecinales..... 50

7.1.2.1 2-Opt. .... 51

MODELO PARA DAR SOLUCIÓN AL BSS-EV-LRP	10
7.1.2.2 Or-Opt .....	52
7.1.2.3 Intercambio de vecindario 1-0. ....	53
7.1.2.4 Intercambio de vecindario 2-2. ....	54
7.2 Fase de sacudida .....	55
7.2.1 Agitando vecindarios. ....	55
7.2.2 Métodos de selección. ....	57
7.2.2.1 Selección de ruta. ....	57
7.2.2.2 Selección de secuencia de vértice : .....	58
7.2.3 Mecanismo adaptativo. ....	59
7.3 Fase de búsqueda de vecindario .....	59
7.3.1 Búsqueda local. ....	60
7.4 Fase de evaluación de la solución .....	61
8. Validación del algoritmo propuesto .....	61
8.1. Resultados de la validación del algoritmo AVNS propuesto .....	62
8.2. Resultados de la validación del algoritmo VNS.....	65
8.3. Comparación de los resultados de validación de los algoritmos AVNS y VNS .....	67
8.4. Ejemplo numérico .....	70
9. Conclusiones.....	73
10. Recomendaciones .....	75
Referencias Bibliográficas .....	77

**Lista de Tablas**

Tabla 1. Cumplimiento de objetivos del proyecto .....	18
Tabla 2. Matriz de costos del ejemplo. ....	45
Tabla 3. Cálculo de ahorros del ejemplo. ....	46
Tabla 4. Ahorros $S_{ij}$ ordenados de mayor a menor .....	47
Tabla 5. Uniones factibles del ejemplo.....	49
Tabla 6. Algoritmo AVNS mediante instancias de Tuzun. ....	63
Tabla 7. Validación del algoritmo AVNS mediante instancias de Barreto .....	64
Tabla 8. Validación del algoritmo VNS mediante instancias de Barreto .....	65
Tabla 9. Validación del algoritmo VNS mediante instancias de Tuzun .....	66
Tabla 10. Comparación AVNS vs VNS para instancias de Barreto .....	68
Tabla 11. Comparación AVNS vs VNS para instancias de Barreto .....	68
Tabla 12. Ejemplo numérico con 50 clientes.....	70
Tabla 13. Ejemplo numérico con 75 clientes.....	72

**Lista de Figuras**

Figura 1. Fusión en CWA. ....	33
Figura 2. Diagrama de flujo algoritmo AVNS.....	42
Figura 3. Diagrama de flujo algoritmo de ahorro de Clark and Wright. ....	44
Figura 4. Costos asociados a clientes y posible ubicación del depósito $V_0$ para el ejemplo. ....	45
Figura 5. Ruta para la fusion de nodos 1,2. ....	48
Figura 6. Solución final del ejemplo.....	50
Figura 7. Símbolos para representación de estructuras vecinales. ....	51
Figura 8. Ejemplo de una solución inicial para la estructura 2-opt ....	51
Figura 9. Ejemplo de aplicación de 2-opt.....	51
Figura 10. Ejemplo de una solución inicial para la estructura or-opt.....	52
Figura 11. Ejemplo de aplicación de or-opt.....	52
Figura 12. Ejemplo de una solución inicial para la estructura de intercambio de vecindario 1-0	53
Figura 13. Ejemplo de una solución para la estructura de intercambio de vecindario 1-0.....	53
Figura 14. Ejemplo de una solución inicial para la estructura de intercambio de vecindario 2-2	54
Figura 15. Ejemplo de una solución para la estructura de intercambio de vecindario 2-2.....	54
Figura 16. Ejemplo de un intercambio cíclico con 3 rutas .....	56
Figura 17. Rutas y ubicación de estaciones para 50 clientes .....	71
Figura 18. Rutas y ubicación de estaciones para 75 clientes .....	73

## **Lista de Apéndices**

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

Apéndice A. Código de programación en Python

Apéndice B. Artículo de carácter publicable

Apéndice C. Instancias

## RESUMEN

**TÍTULO:** Un modelo para la localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos\*

**AUTORES:**

**ACEVEDO GUARÍN, John Freddy; CÁCERES ENTRALGO, Erik Danobis\*\***

**PALABRAS CLAVE:**

Localización-ruteo, Búsqueda de vecindario variable adaptativo, Vehículos eléctricos.

**DESCRIPCIÓN:**

La presente investigación aborda el problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos implementando un algoritmo de búsqueda de vecindario variable adaptativo (AVNS por sus siglas en inglés).

Para dar solución a este problema se desarrolla un modelo matemático teniendo en cuenta las restricciones que tienen los vehículos eléctricos. Además, se propone un algoritmo AVNS que consta de 4 fases fundamentales, inicialmente esta la fase de inicialización, la cual genera una solución inicial y posteriormente se trabajará alrededor de esta solución por medio de estructuras de vecindario. En segundo lugar, se encuentra la fase de sacudida, donde se modifica la probabilidad con la cual se aplican las estructuras de vecindario. Posteriormente en tercer lugar se encuentra la fase de búsqueda de vecindario, durante esta fase el algoritmo realiza una búsqueda completa a través de una búsqueda local hasta encontrar el óptimo local. Finalmente se encuentra la fase de evaluación de la solución, en donde la solución obtenida por el procedimiento de búsqueda local se compara con la mejor solución actual.

El algoritmo propuesto es validado con las instancias de la literatura de Barreto y Tuzun para el problema de localización-ruteo capacitado. Los resultados obtenidos presentan variaciones con respecto a las mejores soluciones conocidas en la literatura, mostrando un mejor desempeño para las instancias de Barreto.

---

\* Proyecto de Grado

\*\* Facultad de Ingeniería Físico Mecánicas. Escuela de Estudios Industriales y Empresariales. Programa de Ingeniería Industrial. Director M. Sc Karin Julieth Aguilar Imitola

**ABSTRACT**

**TITLE:** A model for the location of battery exchange stations and electric vehicle routing\*

**AUTHORS:**

ACEVEDO GUARÍN, John Freddy; CÁCERES ENTRALGO, Erik Danobis \*\*

**KEYWORDS:**

Location-routing problem, Adaptive variable neighborhood search, Electric vehicles.

**DESCRIPTION:**

The present investigation addresses the problem of locating battery exchange stations and electric vehicle routing by implementing an adaptive variable neighborhood search algorithm (AVNS).

To solve this problem, a mathematical model is developed considering the constraints that electric vehicles have. In addition, an AVNS algorithm is proposed which consists of 4 fundamental phases, the first of which is the initialization phase, which generates an initial solution and subsequently work around this solution through neighborhood structures. In the second place is the shaking phase, where the probability with which the neighborhood structures are applied is modified. Subsequently in third place is the neighborhood search phase, during this phase the algorithm performs a complete search through a local search until finding the local optimum. Finally, there is the evaluation phase of the solution, where the solution obtained by the local search procedure is compared with the best current solution.

The proposed algorithm is validated with the instances of the Barreto and Tuzun literature for the capacitated location-routing problem. The obtained results present variations with respect to the best known solutions in the literature, showing a better performance for the Barreto instances.

---

\*Graduation Project

\*\* Faculty of Physicomechanical Engineering. School of Industrial and Business Studies. Industrial Engineering. Directed by M. Sc Karin Julieth Aguilar Imitola

## Introducción

Actualmente las organizaciones son más conscientes de su responsabilidad en el impacto que generan sobre el medio ambiente, debido a esto y con el fin de disminuir significativamente los costos de operación, las organizaciones, dentro de sus políticas, han decidido optar por la implementación de tecnologías limpias. En el sector transporte, por ejemplo, multinacionales como PepsiCo, FedEx, AT&T han incorporado vehículos eléctricos a su flota (Lin, J., Wei, Z., y Wolfson., O., 2016), y Colombia recientemente también ha empezado a trabajar en la implementación de este tipo de tecnologías, un ejemplo de esto es Medellín donde se introdujo un bus articulado 100% eléctrico en el sistema integrado de transporte (El Tiempo, 2018) y empresas como TCC, Nutresa y Familia han introducido vehículos eléctricos a sus operaciones logísticas (Zona Logística, 2016).

Los vehículos eléctricos con batería (VEB) son una de las tecnologías más prometedoras para reducir la dependencia de los combustibles fósiles, las emisiones de dióxido de carbono y los costos de transporte (Revista universidad EAFIT 169, s.f). En 2017 el gobierno de Colombia anuncio la importación de 46000 vehículos eléctricos e híbridos sin arancel e IVA del 5% (Ministerio de Comercio, Industria y Turismo. 29 de junio de 2017) como medida para incentivar el uso de energías limpias y para dar cumplimiento al compromiso para el año 2030 de reducir las emisiones de gases de efecto invernadero en un 20%, adquirido en la vigésima primera Conferencia de las Naciones Unidas sobre el Cambio Climático (COP21) realizada en París (Ministerio de Ambiente y Desarrollo Sostenible, 2015).

Sin embargo, a pesar de sus beneficios, los vehículos eléctricos tienen un rango de viaje limitado, y recargar la batería de un vehículo eléctrico puede tardar entre 3 o 4 horas si no se cuenta con las instalaciones adecuadas para esto, o 30 minutos si se dispone de una estación de carga rápida (Shao, S., Guan, W., y Bi, J., 2017), los altos tiempos de espera representan costos para la empresa ya que su operación se vería retrasada. Por otro lado, el cambio de batería en un VEB tomaría sólo de 3 a 10 minutos (Yang, J., Sun, H., 2014) y la adecuación e infraestructura de una estación de intercambio de batería es menor comparada con una estación de recarga.

Teniendo en cuenta que las metodologías tradicionales de planificación de transporte no consideran estas limitaciones de los vehículos eléctricos es necesario trabajar en la construcción de un modelo cuantitativo que permita planificar la localización de estaciones de intercambio de batería y ruteo en los VEB. Para dar solución a este problema se desarrollará un algoritmo de búsqueda de vecindario variable adaptativo (*Adaptive Variable Neighborhood Search*, AVNS).

El AVNS consta de 4 fases fundamentales, la primera de ellas es la fase de inicialización, en la cual se incluirá el algoritmo de ahorro de Clarke y Wright para dar una solución inicial y posteriormente se trabajará alrededor de esta por medio de las estructuras de vecindario que se definirán durante el desarrollo de esta investigación, después se encuentran las fases de sacudida, búsqueda local y evaluación de la solución.

Tabla 1.  
*Cumplimiento de objetivos del proyecto.*

Objetivo	Cumplimiento
Realizar una revisión de literatura sobre el problema de localización-ruteo, y su aplicación con vehículos eléctricos.	Capítulo 4
Desarrollar un modelo matemático para el problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos mediante programación lineal entera mixta.	Capítulo 6
Diseñar un algoritmo AVNS ( <i>Adaptive Variable Neighborhood Search</i> ) modificado para el problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos.	Capítulo 7
Implementar y validar el algoritmo diseñado mediante instancias de literatura.	Capítulo 8
Elaborar un artículo científico de carácter publicable con base a la investigación realizada.	Apéndice B

### 1. Planteamiento del problema

Hoy en día la humanidad se enfrenta a la necesidad de reducir o eliminar las emisiones de gases de efecto invernadero que son los culpables del cambio climático y el deterioro de la capa de ozono. Entre estos gases se encuentra el CO<sub>2</sub>, el cual es producido en mayor cantidad por el hombre a través de las diferentes actividades de los sectores económicos. Entre estos se destaca el sector transporte debido a que los combustibles derivados del petróleo no se queman de forma eficaz emitiendo a la atmosfera grandes cantidades de CO<sub>2</sub>.

Los elevados precios de los combustibles derivados del petróleo afectan directamente los costos de operación en las empresas transportadoras de mercancías. Una alternativa para este tipo de empresas que ofrece una reducción considerable en sus costos de operación son los vehículos eléctricos (EV, por sus siglas en inglés), ya que operan con energía eléctrica la cual es más económica (Yang & Sun, 2014). Adicionalmente los EV reducen significativamente las emisiones de CO<sub>2</sub> hacia la atmosfera, mitigando así el impacto ambiental por parte del sector transporte.

Para la implementación de los EV hay que tener en cuenta factores como el tiempo de ruta o rango de conducción de un EV ya que este es limitado, debido a que su batería que es la fuente de energía se descarga. Otro factor a tener en cuenta es la localización de estaciones de recarga o intercambio de baterías, teniendo en cuenta que recargar la batería de un EV completamente puede demorar entre 3 y 4 horas, esto afectaría en gran manera la logística de transporte de una empresa ya que los tiempos de entrega serán más extensos lo cual a su vez se reflejaría en un incremento de los costos de operación, otra opción viable es cambiar la batería descargada por una totalmente cargada lo cual tomaría 10 minutos aproximadamente en una estación con los equipos e infraestructura pertinentes. Y como último factor, pero no menos importante, tenemos el ruteo de EV para que cumplan satisfactoriamente con la entrega de pedidos a todos los clientes, es por eso que en este trabajo se desarrollara un modelo que localice de forma óptima las estaciones de intercambio de batería y con un ruteo adecuado para los EV permita a las empresas reducir sus costos de transporte.

## 2. Justificación

La implementación de vehículos eléctricos ha tomado gran importancia en los últimos años por parte de las organizaciones debido a que son una fuerte alternativa para reemplazar a los vehículos que dependen de derivados del petróleo para su funcionamiento, esto se ve reflejado en el costo de operación por kilómetro ya que el costo de la energía eléctrica es menor comparado con los combustibles derivados del petróleo, y además, pero no menos importante, los vehículos eléctricos contribuyen en gran medida a la disminución de la contaminación del medio ambiente debido a que no producen residuos de CO<sub>2</sub>, el principal gas contaminante de la capa de ozono.

La duración de las baterías de los vehículos eléctricos es un factor muy importante que se tiene que considerar al usar este tipo de vehículos, ya que el tiempo de recarga de una batería es muy prolongado, mientras que intercambiar una batería descargada por una totalmente cargada es cuestión de minutos, es por esto que se hace necesario realizar un estudio para la localización de estaciones de intercambio de batería y el ruteo de vehículos eléctricos.

Para el grupo de investigación de la universidad industrial de Santander, optimización de sistemas productivos, administrativos y logísticos (OPALO), es de gran interés contribuir con el desarrollo de las organizaciones colombianas al avanzar en el estudio de implementación de energías limpias con el fin de reducir los costos de operación y mitigar el impacto al medio ambiente por parte de la industria del transporte.

### 3. Objetivos

#### 3.1 Objetivo general

Desarrollar un modelo matemático para la solución del problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos.

#### 3.2 Objetivos específicos

- Realizar una revisión de literatura sobre el problema de localización-ruteo, y su aplicación con vehículos eléctricos.
- Desarrollar un modelo matemático para el problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos mediante programación lineal entera mixta.
- Diseñar un algoritmo AVNS (*Adaptive Variable Neighborhood search*) modificado para el problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos.
- Implementar y validar el algoritmo diseñado mediante instancias de literatura.
- Elaborar un artículo científico de carácter publicable con base a la investigación realizada.

#### 4. Revisión de la literatura

El problema de ruteo de vehículos VRP por sus siglas en inglés (*Vehicle Routing Problem*) es una generalización del reconocido problema del vendedor viajero planteado en el artículo “*The Truck Dispatching Problem*” (Dantzig & Ramser, 1959) el cual fue aplicado a entregas de gasolina. El objetivo fundamental del problema es minimizar el costo total de las rutas trazadas entre las estaciones de gasolina y una terminal granelera la cual posee una demanda y debe ser satisfecha por los camiones repartidores. A partir de este problema y con el paso del tiempo han surgido variaciones en el VRP, en donde se tienen en cuenta aspectos como la capacidad del vehículo de transporte, múltiples depósitos, múltiples estaciones, viajes múltiples, ventanas de tiempo, etc. Usualmente en estas variaciones la función objetivo que se plantea es la misma, minimizar los costos totales de las rutas trazadas (Pillac, Gendreau, Guéret, & Medaglia, 2013). En los últimos años se ha popularizado y se ha venido trabajando en una variación en esta área de investigación que tiene relación con el tipo de vehículo. “*Green Vehicle Routing Problem*” (G-VRP, por sus siglas en inglés), conocido como el problema de ruteo del vehículo verde, donde se hace referencia a los vehículos que funcionan con energías alternativas como biodiesel, electricidad, metanol, gas natural, hidrogeno. Las empresas deben considerar numerosos factores al incluir este tipo de vehículos, entre estos factores se encuentran la disponibilidad del combustible y la distribución geográfica de las estaciones de servicio en el área de servicio, el alcance del vehículo, el costo del combustible y del vehículo, la eficiencia del combustible y los costos de mantenimiento de la flota (Erdoğan & Miller-Hooks, 2012).

Las organizaciones están convirtiendo sus flotas de camiones para incluir vehículos de combustible alternativo, como lo son los vehículos eléctricos, ya sea para reducir su impacto ambiental o para cumplir con las reglamentaciones ambientales. A partir del G-VRP deriva el problema de ruteo de vehículos eléctricos (EVRP) en donde se destacan restricciones como el tiempo de carga de la batería del vehículo y disponibilidad de la infraestructura de carga o intercambio de batería a lo largo de la ruta.

El rango de conducción de los EV es limitado debido a la duración de su batería, se han tratado dos alternativas para dar solución al EVRP. Una alternativa consiste en cargar la batería, esta tiene sus variantes las cuales se han tratado en diferentes artículos como lo son el ruteo de vehículos eléctricos con ventanas de tiempo (E-VRPTW) (Schneider, Stenger, & Goeke, 2014), recargas parciales para el problema de ruteo de vehículos eléctricos con ventanas de tiempo (Keskin & Çatay, 2016), ruteo de vehículos eléctricos con función de carga no lineal (Montoya, Guéret, Mendoza, & Villegas, 2017), ruteo de vehículos con flota mixta (Gonçalves et al, 2011). La segunda alternativa para tratar el EVRP consiste en intercambiar la batería descargada por una batería que se encuentre parcial o completamente cargada (Yang & Sun, 2015).

Debido a la limitación del rango de conducción de la batería y la naturaleza del intercambio de batería, la optimización de la red de distribución con la infraestructura de intercambio de batería podría ser una parte importante del establecimiento de la logística ecológica. Además, debido a la falta de estándares de batería unificada para los EV y la insuficiencia de la infraestructura de soporte público, el mejor modelo de propiedad de infraestructura de intercambio de batería es el modelo que indica que las compañías de logística establecen y operan las estaciones de intercambio de batería (BSS, por sus siglas en inglés) para los EV por sí mismas. Por lo tanto, determinar la estrategia de ubicación de BSS ideal y el plan de enrutamiento de vehículos para una red de

distribución es principalmente una cuestión de nivel de servicio y costo operativo para las empresas de logística (Yang & Sun, 2015).

Según Nagy y Salhi (2006) el problema de localización y ruteo (LRP por sus siglas en inglés) se basa desde un punto de vista jerárquico, por lo que el objetivo es resolver el problema de ubicación de la instalación ("problema principal"), pero para lograrlo, simultáneamente se tiene que resolver un problema de enrutamiento del vehículo ("subproblema"). Para el autor, otra característica importante en esta definición es el requisito de la existencia de planificación de viaje, es decir, la existencia de paradas múltiples en las rutas. Esto ocurre si las demandas de los clientes son menores a una carga completa. LRP es computacionalmente difícil de resolver, es por esto que se requiere un método de solución heurístico.

“Problema de enrutamiento de ubicación de estación de intercambio de batería con capacidad vehículos eléctricos” (Yang & Sun, 2015) determinaron la estrategia de ubicación de las estaciones de intercambio de baterías (BSS) y el plan de enrutamiento de una flota de vehículos eléctricos (EV) simultáneamente, bajo la limitación del alcance de duración de la batería. El problema es formulado como un modelo de programación entera bajo los escenarios básicos y extendidos en cuatro fases heurística llamada SIGALNS y una de dos fases de búsqueda tabú modificado que son Clarke y Wright heurísticas (TS-MCWS) que se proponen para resolver el problema. Se genera un plan de enrutamiento inicial con un algoritmo de barrido modificado, lo que lleva al subproblema de ubicación de BSS, que luego se resuelve utilizando una iterada heurística codiciosa. En la tercera fase, las rutas del vehículo resultantes del subproblema de ubicación se determinan mediante la aplicación de una heurística ALNS por sus siglas en inglés (*adaptive large neighborhood search*). Finalmente, la solución en SIGALNS se mejora aún más mediante un procedimiento de división, comparado con el solucionador MIP de CPLEX y TS-MCWS en tres

conjuntos de instancias, SIGALNS busca la solución en el espacio de manera más eficiente, lo que produce buenas soluciones sin un cálculo excesivo en instancias medianas y grandes. Además, en este artículo se lleva a cabo sistemáticamente el desarrollo económico y ambiental de utilizar vehículos eléctricos.

Mladenovic y E. Hansen (1997) mostraron que se puede obtener una metaheurística simple y efectiva procediendo a un cambio sistemático de vecindad dentro de un algoritmo de búsqueda local. Llamaron a este enfoque Búsqueda de Vecindad Variable (VNS por sus siglas en inglés). Contrariamente a la mayoría de los otros métodos de búsqueda locales, VNS no sigue una trayectoria, pero explora los barrios cada vez más distantes de la solución titular actual, y salta de allí a una nueva sí y solo sí se realiza una mejora. De esta manera, a menudo, características favorables de la solución establecida cuando la mayoría de las variables ya están en su valor óptimo, se mantendrán y se utilizarán para obtener soluciones vecinas prometedoras. Además, una rutina de búsqueda local se aplica repetidamente para obtener desde estas soluciones vecinas hasta óptimos locales.

Para dar solución a las diferentes variantes del problema de ruteo de vehículos se ha empleado una metaheurística llamada búsqueda de vecindad variable adaptativa (AVNS por sus siglas en inglés), la cual combina ideas de VNS con otras heurísticas según las necesidades del problema. Stenger, et al. (2012), en su investigación tratan el problema de enrutamiento que surge en la entrega de paquetes pequeños en el último tramo de la ruta, para dar solución a este problema los autores proponen una nueva heurística AVNS que incorpora ideas conocidas de ALNS en el marco VNS. En general, los algoritmos VNS comienzan con una solución inicial y un conjunto preseleccionado de estructuras de vecindad, posteriormente, el procedimiento de solución principal que consiste en un movimiento aleatorio y un componente de búsqueda local, se repite

hasta que se cumple un criterio de detención determinado. En lugar de construir aleatoriamente soluciones vecinas, el AVNS propuesto guía de forma inteligente el paso de agitación a las áreas donde se esperan soluciones de alta calidad. Gracias al mecanismo adaptativo AVNS demostró ser capaz de obtener resultados dentro de un tiempo de cálculo razonable, además muestra que es capaz de mejorar considerablemente la calidad de las soluciones comparado con un VNS.

En el artículo titulado “*Solving the battery swap station location-routing problem with capacitated electric vehicles using an AVNS algorithm for vehicle-routing problems with intermediate stops*” (Hof, Schneider, Goeke, 2016) los autores buscan resolver el problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos (BSS-EV-LRP) ampliando los métodos de solución para problemas de enrutamiento de vehículos con paradas intermedias VRPIS usando el algoritmo AVNS tomando como referencia el trabajo hecho por Yang y Sun (2015), y limitándose a la versión extendida de este, donde se permiten múltiples visitas por vehículo a los BSS. Este trabajo muestra ser capaz de mejorar significativamente las mejores soluciones previamente conocidas, proporcionadas por SIGALNS y TS-MCWS de Yang y Sun (2015), para la gran mayoría de las instancias. Además, el AVNS reduce fuertemente el número de BSS construidos en las soluciones en comparación con SIGALNS.

A través del análisis preliminar de la literatura se puede observar que el LRP y el VRP son temas que desde hace algunas décadas atrás se han venido tratando, pero aplicados a vehículos que funcionan con combustible fósiles, mientras que para dar solución a estos problemas aplicados a vehículos eléctricos las condiciones cambian. El problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos al ser un tema reciente solo se cuenta con los trabajos realizados por Yang y Sun (2015) y Hof, Schneider y Goeke (2016) los cuales

resuelven este problema por diferentes metodologías, y servirán como guía para el desarrollo de este proyecto de grado.

## 5. Marco teórico

### 5.1 Logística en transporte de mercancías.

El transporte de mercancías es una actividad al servicio de la logística de distribución y como tal está sometido a una serie de normas y principios establecidos por la dirección de la empresa.

Uno de los principales aspectos está relacionado con la posesión de la flota, la cual puede ser:

- Flota propia (compra, alquiler o leasing)
- Flota ajena (servicio subcontratado)
- Flota mixta (ambos casos)

Se puede afirmar en términos muy generales, que, dentro de los costes de distribución comercial, el transporte físico de mercancías representa en torno a un 40%, lo que da una idea de su importancia económica y su repercusión en el margen comercial (Tejero, 2015).

**Optimización combinatoria.** La optimización puede definirse como el proceso de tratar de encontrar la mejor solución a un problema determinado, de forma más precisa, los problemas de optimización se pueden expresar como encontrar el valor de unas variables de decisión (sujeto a unas restricciones) para que se alcance el valor máximo o mínimo de una determinada función objetivo (Cunquero, 2003).

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

**5.1.1 Complejidad computacional.** La optimización computacional incluye las disciplinas de investigación operativa para modelar el sistema, matemáticas para formular el modelo, ciencias de la computación para el diseño y análisis de algoritmos e ingeniería del software para implementar el modelo. Matemáticamente, los problemas pueden caracterizarse atendiendo a la dificultad que entraña su resolución por un ordenador (Duarte, Pantrigo y Gallego, 2007).

Teniendo en cuenta la teoría de complejidad computacional, los problemas de optimización pueden ser clasificados de acuerdo al tiempo computacional requerido para encontrar una solución en clases de complejidad, como lo establecen Johnson y Garey (1979):

**Clase P:** está formada por problemas de decisión que pueden ser resueltos mediante algunos algoritmos en un número de pasos acotado por un polinomio fijo en función del tamaño de la entrada.

**Clase NP:** incluye a todos los problemas de decisión cuyas soluciones pueden ser verificadas en un tiempo polinómico dada la información correcta o lo que es lo mismo, pueden ser encontradas en tiempo polinómico haciendo uso de una máquina de Turing no determinista.

**Clase NP-completos:** son los problemas NP más difíciles, debido a que es menos probable que pertenezcan a la clase P, debido a su dificultad por encontrar una forma de resolver el problema rápidamente (en un tiempo polinómico).

**Clase NP-hard:** se caracterizan porque si se resuelve un problema en un tiempo polinómico permite resolver todos los problemas de la clase NP en un tiempo polinómico. (Navarro, 2008).

**5.1.2 Métodos de solución de un problema de optimización combinatoria.** Los métodos empleados para resolver los problemas de optimización combinatoria se clasifican en tres grupos que son: métodos exactos, métodos heurísticos y metaheurísticos.

**5.1.2.1 Algoritmos exactos.** Proporcionan la solución óptima para la instancia planteada en un tiempo acotado; sin embargo, tiene una contraindicación cuando se trata de optimización combinatoria de la clase NP – Hard, ya que el tiempo estimado para la solución aumenta exponencialmente directamente proporcional al tamaño del problema. Entre los más usados están el método Branch and Bound y Programación Lineal (Chicano, 2007).

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

**5.1.2.2.1 Heurísticas Constructivas.** Determinan soluciones por medio de la incorporación iterativa de elementos a una estructura vacía inicial, hasta completar una solución que cumpla con las restricciones. Esta heurística se puede usar cuando el problema se puede solucionar por decisiones sucesivas, por ejemplo, líneas de montaje o rutas como el algoritmo de ahorro de Clarke y Wright, típico del ruteo de vehículos clásico (Díaz, Glover, & H. Ghaziri, 1996).

**5.1.2.2.2 Métodos de búsqueda local.** Procede a una búsqueda en el espacio de solución de un problema determinado el cual examina muchas soluciones diferentes para encontrar una solución óptima en el vecindario que cada vez es más aproximada que la anterior, a esta última solución se le conoce como óptimo local.

**5.1.2.3 Algoritmos metaheurísticos.** Una metaheurística se define formalmente como un proceso iterativo de generación que guía una heurística subordinada combinando inteligentemente diferentes conceptos para explorar y explotar el espacio de búsqueda; las estrategias de aprendizaje se utilizan para estructurar información a fin de encontrar soluciones eficientemente casi óptimas (Osman & Laport, 1996).

**5.1.3 Problema de Localización - Ruteo (LRP).** El LRP se puede definir de la siguiente manera: Un conjunto factible de posibles sitios y ubicaciones de las instalaciones y las demandas esperadas de cada cliente que se otorgan. Cada cliente debe ser asignado a una instalación que abastecerá su demanda. Los envíos de la demanda de los clientes se llevan a cabo mediante vehículos que se envían desde las instalaciones y operan en rutas que incluyen múltiples clientes. La ubicación de las instalaciones de distribución y la distribución de productos desde estas instalaciones a los clientes son dos componentes clave de un sistema de distribución. En diferentes configuraciones, estos dos componentes son interdependientes; por lo tanto, es necesario considerar la ubicación de las instalaciones y las decisiones de distribución de manera simultánea (Hassanzadeh, Mohseninezhad, Tirdad, Dadgostari, & Zolfagharinia, 2009).

**5.1.4 Problema de Ruteo de Vehículos.** El problema de enrutamiento del vehículo (VRP) se define en una red completa no dirigida  $G = (V; E)$  con un conjunto de nodos  $V = \{0; 1; \dots; n\}$  y un conjunto de aristas  $E$ . El nodo  $0$  es un depósito con  $m$  vehículos idénticos de capacidad  $W$ ;  $m$  puede ser arreglado a priori o dejarse como una variable de decisión. Cada otro nodo  $i > 0$  representa

un cliente con una demanda  $q_i$  no negativa y cada borde  $(i, j)$  tiene un costo de viaje no negativo  $c_{ij} = c_{ji}$ . El VRP consiste en determinar un conjunto de  $m$  viajes de vehículos de un costo total mínimo, de modo que cada viaje se inicia y termina en el depósito, cada cliente se visita exactamente una vez y la demanda total manejada por cualquier vehículo no supera  $W$  (Prins, 2004).

**5.1.5 Problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos (BSS-EV-LRP).** El objetivo del BSS-EV-LRP es determinar: la ubicación de las estaciones de intercambio de baterías (BSS), la asignación de clientes a los vehículos eléctricos (EV), la asignación de EV a los BSS, y los recorridos desde el depósito para atender a los clientes con el servicio de BSS considerando el rango de conducción limitado y la capacidad de carga de los EV (Yang & Sun, 2015).

**5.1.5.1 Problema de localización de estaciones de intercambio de batería.** Debido a la capacidad limitada de las baterías, los vehículos eléctricos típicos sólo pueden viajar alrededor de 100 millas con una sola carga y requieren horas para recargarse. La industria ha propuesto una solución en torno al uso de estaciones de intercambio, en las cuales las baterías agotadas se pueden cambiar por recargadas en medio de viajes largos. El posible éxito de esta solución depende de la capacidad del proveedor de servicios de carga para desplegar una red de infraestructura rentable.

Las estaciones de intercambio son necesarias para viajes largos (o viajes de ida y vuelta) que exceden el rango de una batería llena. Para garantizar una cobertura completa de la red de autopistas, las estaciones de intercambio deben ubicarse en ubicaciones convenientes a lo largo de tales viajes largos. Al decidir dónde ubicar las estaciones de intercambio, el proveedor de servicios de carga también debe tener en cuenta que la tasa de demanda de intercambio en una estación determina el número de baterías de repuesto que debe mantener la estación para garantizar que los

usuarios casi siempre puedan obtener baterías completas sin esperar. Debido a que las baterías son caras y tienen una vida útil limitada, los costos de mantenimiento son significativos. Además, tener una mayor tasa de demanda y más baterías de repuesto en una sola estación significa que se recargarán más baterías en paralelo y que la carga en la red eléctrica será más pesada. El proveedor de servicios de carga debe tener en cuenta todos estos factores al diseñar la red de estaciones de intercambio.

Desafortunadamente, debido a la necesidad de implementar una red de infraestructura completa antes de entregar los vehículos, el proveedor de servicios de carga tiene que tomar decisiones estratégicas de diseño de red antes de observar la demanda real. En este punto, los factores que afectan la demanda de intercambio, como la tasa de adopción de EV, la participación de mercado del proveedor de servicios de carga y los patrones de conducción de los usuarios de EV, no se conocen con precisión (Mak, Rong & Shen, 2013).

**5.1.5.2 Problema de ruteo de vehículos eléctricos (EVRP).** El EVRP consiste en encontrar la estrategia óptima de enrutamiento con un costo mínimo de tiempo de viaje y un costo de energía, así como el número de vehículos eléctricos despachados. Debido al límite de rango de conducción, los EV pueden necesitar visitar una estación de carga o intercambio de batería entre las visitas de los clientes durante su operación diaria, lo que se traduce en una considerable cantidad de costos de mano de obra adicionales como consecuencia del largo tiempo de recarga en ruta. Debido a esta característica única, hace que el EVRP sea distintivamente diferente del VRP tradicional (Lin, Zhou & Wolfson, 2016).

**5.1.6 Clarke & Wright Algorithm.** El algoritmo de Clarke y Wright (CWA), como se diseñó originalmente, comienza con la construcción de una solución trivial similar a una flor, en la que cada cliente recibe la visita de una ruta. Luego, cada par de rutas ( $R, S$ ), cuya carga total no

excede la capacidad del vehículo ( $Q$ ), se inspecciona para evaluar el ahorro que resultaría de su fusión. Se ejecuta la fusión que da el mayor ahorro positivo. Este proceso se repite hasta que no se pueda encontrar una fusión de capacidad factible con un ahorro positivo.

Por ejemplo, si  $s$  se denota como el único depósito y los nodos  $i, j, k, l$  representan el primer y último cliente de  $R$  y el primer y último cliente de  $S$ , el ahorro obtenido al fusionar  $S$  después de  $R$  es igual a  $C_{js} + C_{sk} - C_{jk}$ , ver la figura 1. Hay que tener en cuenta que existen cuatro posibles fusiones por par de rutas ( $R, S$ ), ya que cada ruta puede estar invertida o no. Una fusión puede definirse como un arco  $[j, k]$  que unirá un nodo  $j$ , en un extremo de un recorrido, a un nodo  $k$ , en un extremo de otro recorrido.

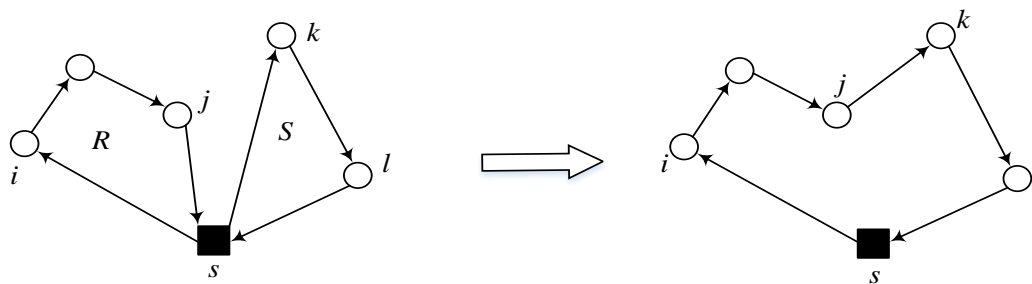


Figura 1. Fusión en CWA. Adaptado de Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking (Prins, C., 2006).

**5.1.7 Métodos de solución BSS-EV-LRP.** En la literatura solo hay 2 artículos que han tratado este problema. Yang y Sun dan solución a este problema por medio de una heurística ALNS modificada, y por otro lado Hof, Schneider y Goeke plantean un método de solución por medio de una heurística AVNS modificada.

**5.1.7.1 Heurística ALNS.** La idea principal de esta heurística es buscar en el gran vecindario eliminando un número dado de clientes de una solución y luego reinsertándolos para construir una nueva solución. En cada iteración, los operadores de extracción e inserción se eligen de forma adaptativa en función de su éxito histórico (Demir E, BektasT, Laporte G., 2012).

**5.1.7.2** *Heurística AVNS*. Para comprender como funcionan las búsquedas de vecindario variable adaptativo (AVNS) se tiene que definir el concepto de búsqueda de vecindario variable (VNS), el cual consiste en hacer un cambio sistemático de vecindario dentro de un algoritmo de búsqueda local posiblemente aleatorizado el cual produce una metaheurística simple y efectiva para la optimización combinatoria y global. Contrario a otras metaheurísticas basadas en métodos de búsqueda local, el VNS no sigue una trayectoria, sino que explora vecindarios cada vez más distantes de la solución actual, y salta de esta solución a una nueva sí y sólo sí se ha realizado una mejora (Hansen & Mladenovic, 2001).

La metaheurística AVNS consiste en combinar ideas de VNS con otras heurísticas según las necesidades del problema (Hof, Schneider & Goeke, 2016).

## **6. Problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos**

### **6.1 Descripción del problema**

Consiste en un problema de localización-ruteo con vehículos eléctricos que al momento de agotarse completamente su batería, esta es intercambiada en un estación por otra batería totalmente cargada, la solución de este problema tiene como objetivo la minimización de los costos totales, los cuales incluyen el costo de construcción de las estaciones de servicio, el costo fijo por vehículo usado para satisfacer la demanda de los clientes en las diferentes rutas, y el costo de envío que dependerá de la distancia recorrida por cada uno de los vehículos.

### **6.2 Formulación del problema**

Para la presente investigación se toma como referencia el modelo matemático propuesto por Yang y Sun (2015). A continuación, se presentan los supuestos sobre los cuales funciona el modelo propuesto

- La flota de EV es homogénea.
- Al salir del depósito todo EV tiene una batería completamente cargada.
- Al salir de una estación de intercambio de batería todo EV tiene su batería completamente cargada.
- Existe una estación de intercambio de batería en el depósito.

- Tanto el depósito como las estaciones de intercambio de batería tendrán suficientes puertos de recarga para que cada EV tenga la potencia suficiente para comenzar o seguir la ruta con una batería completamente cargada.
- Cada instalación tiene una capacidad ilimitada para el intercambio y recarga de las baterías, también puede ser utilizada simultáneamente por cualquier número de vehículos.

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

#### Conjuntos

$c$  Clientes  $c = 1, 2, \dots, C$

$v$  Vehículos  $v = 1, 2, \dots, V$

$e$  Estaciones  $e = 1, 2, \dots, E$

$n$  Nodos  $i, j$

$w$  Depósito  $w = 1$

$w'$  Copia del depósito

$r$  Rutas  $r = 1, 2, \dots, R$

#### Parámetros:

$Q_v$  Capacidad de carga del vehículo  $v$

$A$  Distancia máxima que permite una batería completamente cargada

$M$  Un número muy grande

$F_v$  Costo fijo por vehículo  $v$

$T_{ve}$  Tiempo de cambio de batería de un vehículo  $v$  en una estación  $e$

$L_r$  Tiempo máximo de ruta  $R$

$K_{ijv}$  Costo de envío por kilómetro del nodo  $i$  al nodo  $j$  en el vehículo  $v$

$D_c$  Demanda del cliente  $c$

$S_e$  Costo de construcción de una estación de intercambio de batería

$X_{ij}$  Distancia del nodo  $i$  al nodo  $j$

$q_{cv}$  Carga restante cuando un vehículo  $v$  deja un cliente  $c$

$b_{iv}^1$  Distancia máxima restante cuando un vehículo  $v$  llega al nodo  $i$

$b_{iv}^2$  Distancia máxima restante cuando un vehículo  $v$  sale al nodo  $i$

Variables

$Y_e$  Variable de decisión binaria, tomando un valor de 1 si un BSS es ubicado en el nodo  $e$  y 0 de lo contrario.

$P_{ijv}$  Variable de decisión binaria, tomando un valor de 1 si el vehículo  $v$  va del nodo  $i$  al nodo  $j$  y 0 de lo contrario.

$Z$  Costo.

### 6.1.1 Función Objetivo.

Minimizar  $Z$

$$Z = \sum_{e \in E} Y_e S_e + \sum_{i \in n} \sum_{j \in n} \sum_{v \in V} K_{ijv} X_{ij} P_{ijv} + \sum_{i \in n} \sum_{j \in n} \sum_{v \in V} F_v P_{ijv} \quad (1)$$

La función objetivo (1) minimiza el costo total, incluido el costo de construcción de BSS, el costo de envío y el costo fijo por EV.

### 6.1.2 Restricciones

$$\sum_{i \in n \setminus w', i \neq j} \sum_{v \in V} P_{ijv} = 1 \quad \forall j \in C \quad (2)$$

$$\sum_{i \in n \setminus w', i \neq j} \sum_{v \in V} P_{ijv} \leq M Y_e \quad \forall j \in E \quad (3)$$

$$\sum_{j \in n \setminus w, j \neq i} P_{ijv} - \sum_{j \in n \setminus w', j \neq i} P_{jiv} = 0 \quad \forall i \in n \setminus w, w', v \in V \quad (4)$$

$$\sum_{j \in n \setminus w} P_{wjv} - \sum_{j \in n \setminus w'} P_{jw'v} = 0 \quad \forall v \in V \quad (5)$$

$$\sum_{j \in n \setminus w} P_{wjv} \leq 1 \quad \forall v \in V \quad (6)$$

$$\sum_{v \in V} \sum_{j \in n \setminus w} P_{wjv} \leq |V| \quad (7)$$

$$b_{iv}^1 \leq b_{iv}^2 - X_{ij} P_{ijv} + A(1 - P_{ijv}) \quad \forall i \in n \setminus w', \forall j \in n \setminus w, i \neq j, v \in V \quad (8)$$

$$b_{wv}^2 = A \quad \forall v \in V \quad (9)$$

$$b_{iv}^2 = A Y_i \quad \forall i \in E \quad (10)$$

$$b_{jv}^1 \leq b_{jv}^2 \quad \forall j \in C \quad (11)$$

$$b_{jv}^2 \geq 0 \quad \forall j \in n \quad (12)$$

$$\sum_{c \in n \setminus w', e} q_{cev} = \sum_{c \in n \setminus w, e} q_{ecv} \quad \forall e \in E, v \in V \quad (13)$$

$$\sum_{e \in n \setminus w, c} q_{cev} \leq \sum_{e \in n \setminus w', e} q_{ecv} - D_c \sum_{e \in n \setminus w, c} P_{ecv} + Q_v (1 - \sum_{e \in n \setminus w', c} P_{ecv})$$

$$\forall c \in C, v \in V \quad (14)$$

$$0 \leq q_{cev} \leq Q_v P_{cev} \quad \forall c \in n \setminus w', \forall e \in n \setminus w, c \neq e, v \in V \quad (15)$$

$$T_{ve} \leq 15 \text{ minutos} \quad \forall e \in E \quad (16)$$

$$L_r \leq 8 \text{ horas} \quad \forall r \in R \quad (17)$$

$$Y_e, P_{ijv} \in \{0,1\} \quad \forall e \in E, \forall i \in n \setminus w', \forall j \in n \setminus w, \forall v \in V \quad (18)$$

La restricción (2) asegura de que cada cliente sea visitado por un vehículo, la restricción (3) garantiza que los vehículos eléctricos solo cambien sus baterías en un BSS ubicado. La restricción (4) asegura el balance de flujo para cada vehículo en cada nodo. La restricción (5) denota que cada vehículo debe regresar al depósito. En la restricción (6) se asigna solo un viaje a un vehículo cuando el vehículo sale del depósito. Por medio de la restricción (7) se limita la cantidad de vehículos a  $|V|$ . La restricción (8) rastrea el nivel de potencia de la batería de un EV según la secuencia de nodos. Si el nodo  $j$  es visitado por el vehículo  $v$  después del nodo  $i$  ( $P_{ijv} = 1$ ), la distancia máxima que permite la batería restante al llegar al nodo  $j$  se reduce en función de la distancia entre el nodo  $i$  y el nodo  $j$ . La restricción (9) restablece el nivel de potencia de la batería a  $A$  cuando los EV salen del depósito. La restricción (10) restablece el nivel de potencia de la batería a  $A$  cuando los EV salen de un BSS ubicado. La restricción (11) asegura de que el nivel de energía de la batería del EV debe ser mayor o igual después de visitar un nodo de cliente. La restricción (12) garantiza que cada EV tenga suficiente batería para visitar a los clientes restantes y regresar al depósito. La restricción (13) representa que la suma de la carga restante de un EV que ingresa a un BSS es igual a la suma de la carga restante de un EV que sale de un BSS, lo que garantiza el equilibrio de capacidad del vehículo. Las restricciones (13) y (14) hacen posible que

un EV pase por una estación más de una vez, pero visite a un cliente solo una vez. Además, los dos conjuntos de restricciones son también las restricciones de eliminación de ruta. La restricción (15) limita el rango del nivel de carga restante. Si el vehículo  $v$  visita el nodo  $e$  después de salir de  $c$  ( $P_{ecv} = 1$ ), el nivel máximo de carga restante  $q_{cev}$  es como máximo la capacidad total del vehículo  $Q_v$  de lo contrario,  $q_{cev} = 0$ . La restricción (16) asegura que el tiempo de duración de un vehículo  $v$  en una estación  $e$  no sea mayor a 15 minutos. La restricción (17) asegura que el tiempo máximo que un vehículo demore en una ruta sea de 8 horas. La restricción (18) establece la naturaleza binaria de las variables de decisión.

## 7. AVNS para la solución del BSS-EV-LRP

En este capítulo se describe a detalle el algoritmo AVNS utilizado para dar solución al problema de localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos. El VNS es una metaheurística para resolver problemas de optimización cuya idea básica es el cambio sistemático de entorno o vecindario dentro de una búsqueda local. Un problema de optimización consiste en encontrar, dentro de un conjunto  $Z$  de soluciones factibles, la que optimiza una función  $f(y)$ . Donde  $y$  representa una solución alternativa,  $f$  es la función objetivo y  $Y$  es el espacio de soluciones factibles del problema. Según Hansen, P., Mladenović, N., y Moreno Pérez, J. A., (2009) el algoritmo VNS está basado en tres hechos simples:

- Un mínimo local con respecto a la estructura de un vecindario no es necesariamente un mínimo local para la estructura de otro vecindario.

- Un mínimo global es un mínimo local con respecto a todas las estructuras vecinales posibles.
- Para muchos problemas, los mínimos locales con respecto a uno o varios vecindarios están relativamente cerca uno del otro.

La heurística propuesta se presenta en el diagrama de flujo de AVNS, ver figura 2. Esta empieza con una solución factible,  $y$ , dada por el algoritmo de ahorro de Clarke y Wright (CWA, por sus siglas en inglés). Existen dos parámetros más que configuran el algoritmo:  $maxiter$  es el número máximo de iteraciones, y  $kmax$  es el número máximo de vecindades a explorar. En cada iteración, la solución se perturba de manera aleatoria con el procedimiento de Sacudida con tamaño  $k$ , obteniendo una nueva solución  $y'$ , y se aplica un proceso de búsqueda, obteniendo una solución mejorada  $y''$ . Si  $y''$  es mejor que  $y$ , se actualiza a  $y''$  como la mejor solución encontrada hasta el momento, y  $k$ , la estructura de vecindario, vuelve a 1; en cualquier otro caso,  $k$  se incrementa. Este proceso se repite hasta que se alcance  $kmax$ , siempre que no se haya llegado al máximo número de iteraciones permitidas. Por lo tanto,  $maxiter$  y  $kmax$  son parámetros de la búsqueda (Menéndez, B., Pardo, E., y Duarte A., 2015).

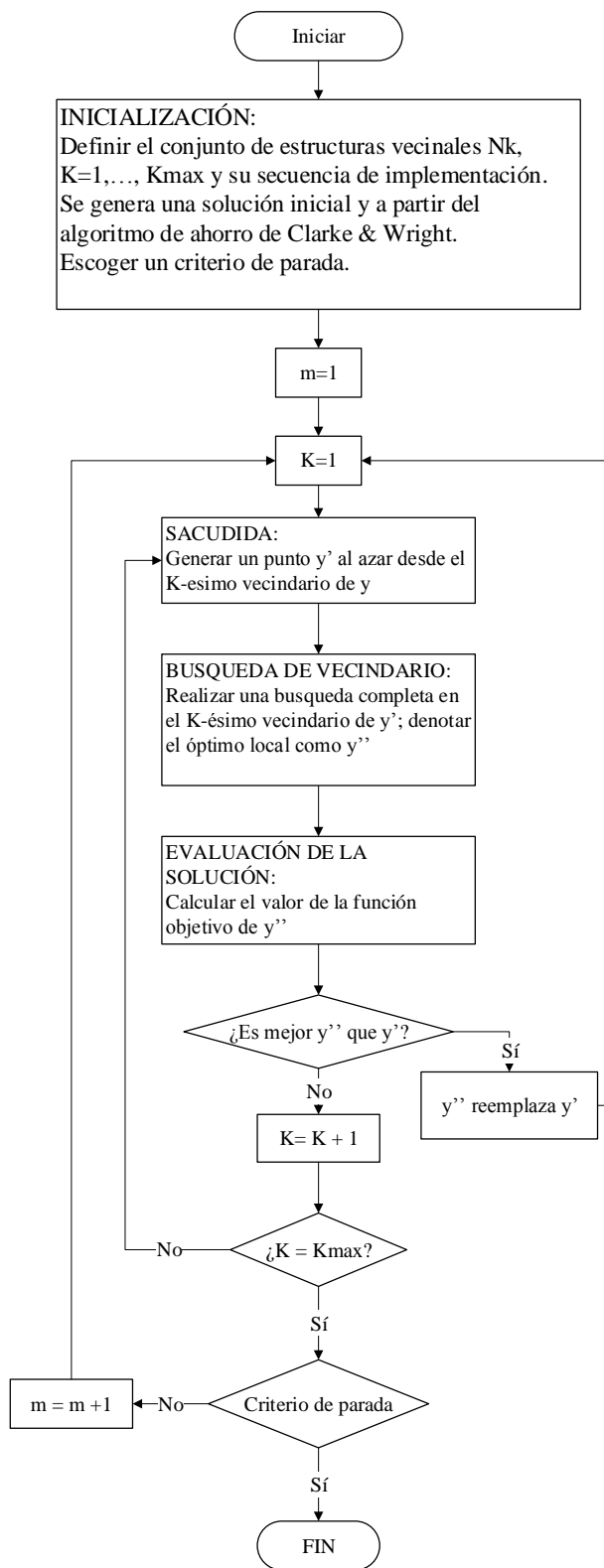


Figura 2. Diagrama de flujo algoritmo AVNS. Adaptado de Variable neighbourhood search for redundancy allocation problems (Liang, Y., Lo, M. y Chen, Y., 2007).

## 7.1 Fase de inicialización

En esta fase se define el conjunto de estructuras vecinales las cuales se implementarán de manera aleatoria, también se genera una solución inicial, y además se escoge el criterio de parada.

**7.1.1 Solución inicial.** La solución inicial es generada a partir del algoritmo de ahorro de Clark and Wright.

**7.1.1.1 Algoritmo de Clark and Wright.** El procedimiento de esta heurística es simple, dado que, por medio de una exploración limitada del espacio de búsqueda, se encuentra una solución que es más o menos aceptada con un tiempo de cálculo moderado (Sardo Va, 2007).

El algoritmo de ahorros no proporciona una solución óptima al problema; pero a menudo este método rinde una solución relativamente buena, es decir; una solución que se desvía poco de la solución óptima (Aquino y Jiménez, 2010).

La aplicación de este algoritmo busca minimizar el costo de tal modo que se satisfaga la demanda de cada cliente (Barajas, 2010).

Si en una solución dos rutas diferentes  $(0, \dots, i, 0)$  y  $(0, j, \dots, 0)$  pueden ser combinadas formando una nueva ruta  $(0, \dots, i, j, \dots, 0)$ , el ahorro (en distancia o costo) obtenido por dicha unión es:

$$S_{ij} = C_{i0} + C_{0j} - C_{ij} , \text{ (Clarke y Wright, 1964).}$$

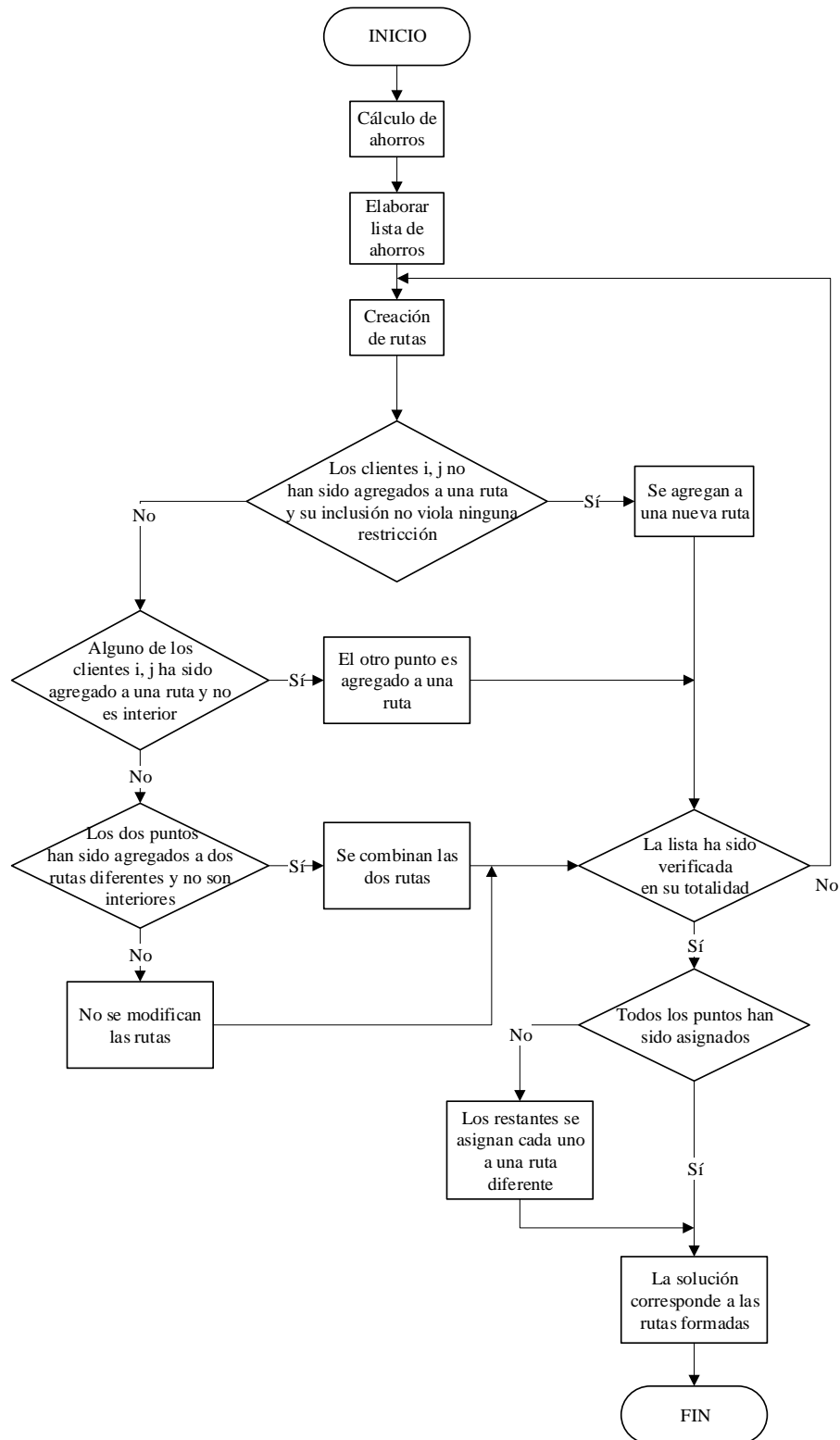


Figura 3. Diagrama de flujo algoritmo de ahorro de Clark and Wright. Adaptado de An improved Clarke and Wright savings algorithm for the capacitated vehicle routing problem (Pichpibul, T. y Kawtummachai, R., 2012).

A continuación, desarrollaremos un ejemplo con 5 clientes que van desde 1 hasta 5 y 1 depósito  $V_0$  con el propósito de ilustrar el funcionamiento del algoritmo de ahorros de Clarke y Wright.

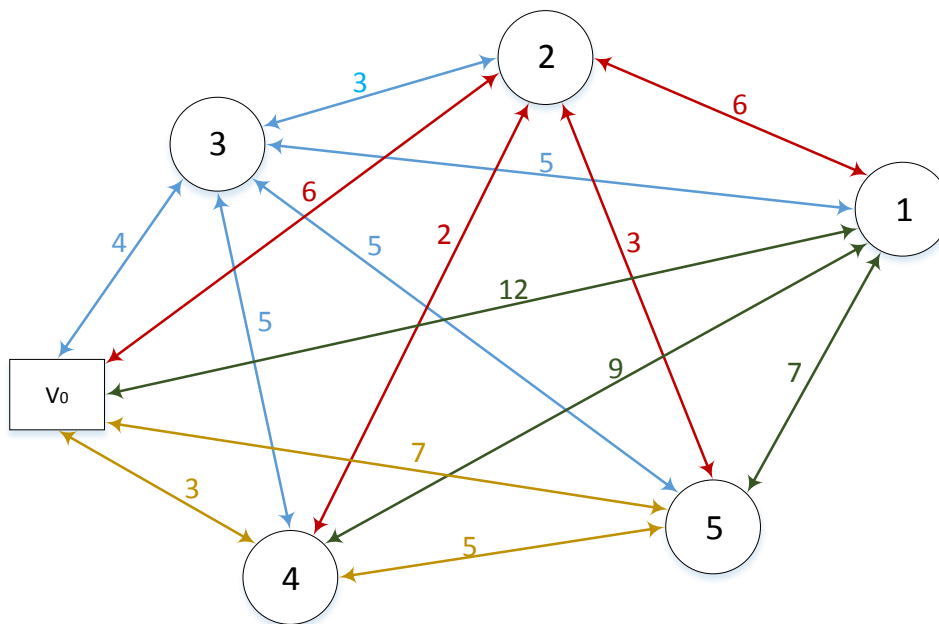


Figura 4. Costos asociados a clientes y posible ubicación del depósito  $V_0$  para el ejemplo.

Los datos tomados para el ejemplo, (ver tabla 2), provienen de la figura 4 en la cual se encuentran los costos asociados a los clientes y la ubicación del depósito. Las filas están etiquetadas desde  $i = 0, 1, 2, 3, 4, 5$  y  $j = 0, 1, 2, 3, 4, 5$  donde 0 representa el depósito. El vector demanda para cada cliente respectivamente es:  $D = [24, 25, 26, 28, 29]$ . La capacidad de cada vehículo es de 55 unidades.

Tabla 2.  
Matriz de costos del ejemplo.

	0	1	2	3	4	5
0	0	12	6	4	3	7
1	12	0	6	5	9	7
2	6	6	0	3	2	3
3	4	5	3	0	5	5
4	3	9	2	5	0	5
5	7	7	3	5	5	0

El primer paso en el algoritmo es calcular todos los ahorros  $S_{ij}$  para todas las parejas de clientes  $(i, j)$  involucradas en el ejercicio, con  $i \neq j$ . En este caso los ahorros por calcular son:  $S_{12}, S_{13}, S_{14}, S_{15}, S_{23}, S_{24}, S_{25}, S_{34}, S_{35}, S_{45}$  por ser un problema simétrico.

Para el cálculo de estos ahorros la fórmula que se utilizará es la de Clark y Wright (1964), la cual está dada por:  $S_{ij} = C_{i0} + C_{0j} - C_{ij}$ , donde  $C_{i0}$  es el costo de ir del nodo  $i$  al depósito  $0$ ,  $C_{0j}$  es el costo de ir del depósito  $0$  al nodo  $j$ , y  $C_{ij}$  es el costo de ir desde el nodo  $i$  al nodo  $j$ .

El cálculo de la ruta está dado por la fórmula:  $2C_{0i} + 2C_{0j}$

El cálculo de la nueva ruta está dado por la fórmula:  $C_{0i} + C_{ij} + C_{j0}$

Ejemplo de cálculo de ruta, cálculo de nueva ruta y ahorro obtenido de los nodos 0,1,2

Ruta: (0,1,0) y (0,2,0)

Cálculo de ruta:  $2(12) + 2(6) = 36$

Nueva ruta: (0,1,2,0)

Cálculo nueva ruta:  $12 + 6 + 6 = 24$

Ahorro obtenido.  $S_{12} = 12 + 6 - 6 = 12$

Tabla 3.  
*Cálculo de ahorros del ejemplo.*

Ruta	Cálculo ruta	Nueva ruta	Cálculo nueva ruta	Ahorro obtenido
(0,1,0) y (0,2,0)	36	(0,1,2,0)	24	12
(0,1,0) y (0,3,0)	32	(0,1,3,0)	21	11
(0,1,0) y (0,4,0)	30	(0,1,4,0)	24	6
(0,1,0) y (0,5,0)	38	(0,1,5,0)	26	12
(0,2,0) y (0,3,0)	20	(0,2,3,0)	13	7
(0,2,0) y (0,4,0)	18	(0,2,4,0)	11	7
(0,2,0) y (0,5,0)	26	(0,2,5,0)	16	10
(0,3,0) y (0,4,0)	14	(0,3,4,0)	12	2
(0,3,0) y (0,5,0)	22	(0,3,5,0)	16	6
(0,4,0) y (0,5,0)	20	(0,4,5,0)	15	5

En el segundo paso del algoritmo, los ahorros obtenidos se ordenan de mayor a menor en una lista, que denominaremos lista de los ahorros  $S_{ij}$ . La lista para el ejemplo que se está desarrollando aparece a continuación.

Tabla 4.  
Ahorros  $S_{ij}$  ordenados de mayor a menor.

Ruta	Ahorro obtenido
$S_{12}$	12
$S_{15}$	12
$S_{13}$	11
$S_{25}$	10
$S_{23}$	7
$S_{24}$	7
$S_{14}$	6
$S_{35}$	6
$S_{45}$	5
$S_{34}$	2

Como tercer paso en el algoritmo, de la lista de ahorros cada par de puntos se considera paso a paso. Para el par de puntos  $i-j$  que está en consideración, las dos rutas que visitan a  $i$  y a  $j$  se fusionan, siempre y cuando cumplan las condiciones enunciadas anteriormente, en la explicación del funcionamiento del algoritmo que se encuentra en la figura 3.

Para este ejemplo se inicia con el ahorro  $S_{12}$ , el cual presenta el mayor ahorro, y se revisa si está ruta es factible verificando que la demanda de los clientes sea menor a la capacidad del vehículo. En caso de que la capacidad del vehículo se exceda, la unión no se realiza y se selecciona el siguiente ahorro según la tabla 4. Cuando el primer vehículo complete su capacidad, se inicia otra ruta con otro vehículo.

En la figura 5 aparece en color verde la ruta que resulta de fusionar las rutas  $(0,1,0)$  y  $(0,2,0)$  esta cumple con la capacidad ya que la suma de las demandas de los clientes 1 y 2 es igual a 49 y

la capacidad del vehículo es 55 por lo que la hace una solución factible. Si la ruta resultante es factible se agrega a la solución, de lo contrario se rechaza la unión y no puede ser agregada a la solución.

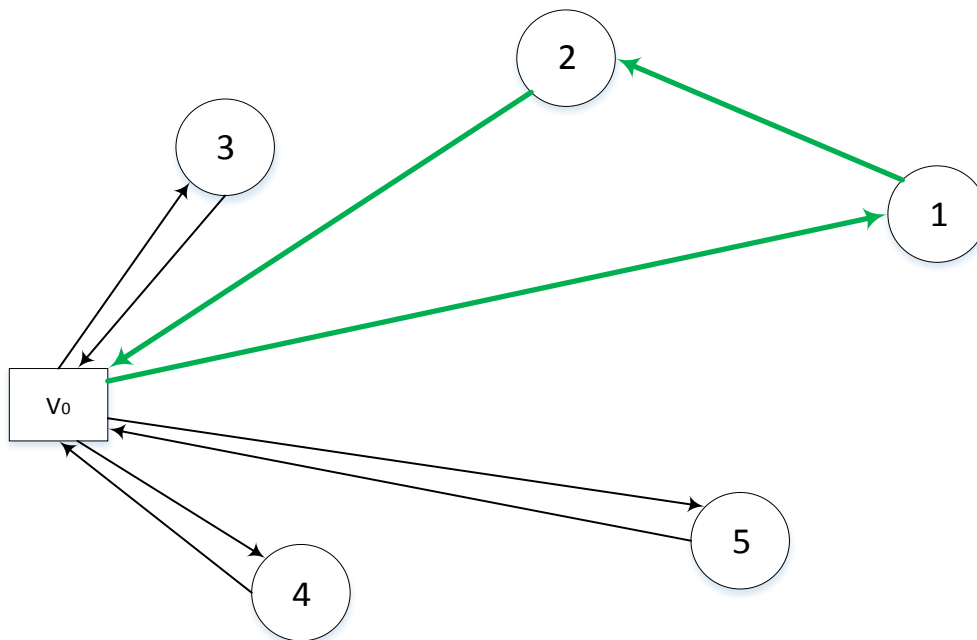


Figura 5. Ruta para la fusión de nodos 1,2.

En el cuarto paso se escoge el siguiente mejor ahorro según la tabla 4, también se examina si los clientes contenidos en la nueva ruta no hacen parte de conexiones directas previas, además si es factible o no su unión, también se revisa si la suma de sus demandas cumplen con la capacidad del vehículo. Si es factible su unión, incluir la nueva ruta en la solución, en caso contrario continuar con el siguiente ahorro de la lista y volver al paso 3. Este paso se repite hasta que se hayan considerado todos los ahorros comprendidos en la tabla 4. En caso de que un cliente no sea incluido en una ruta será necesario crear una sola para este cliente y esta ruta no presentará ningún ahorro. A continuación, se presenta una tabla con los tramos no factibles del ejemplo.

Tabla 5.  
Uniones factibles del ejemplo.

Ruta	Factibilidad	Explicación
S <sub>12</sub>	Factible	La ruta (0,1,2,0) hace parte de la solución ya que los clientes 1 y 2 no han hecho conexiones previas con otros clientes. Además, la suma de las demandas de los clientes cumple con la capacidad del vehículo.
S <sub>15</sub>	No factible	La ruta (0,1,5,0) no se puede unir debido a que el cliente 1 tiene una conexión previa en la ruta (0,1,2,0) por lo que no se puede incluir al cliente 1 en otra ruta.
S <sub>13</sub>	No factible	La ruta (0,1,3,0) no se puede unir debido a que el cliente 1 tiene una conexión previa en la ruta (0,1,2,0) por lo que no se puede incluir al cliente 1 en otra ruta.
S <sub>25</sub>	No factible	La ruta (0,2,5,0) no se puede unir debido a que el cliente 2 tiene una conexión previa en la ruta (0,1,2,0) por lo que no se puede incluir al cliente 2 en otra ruta.
S <sub>23</sub>	No factible	La ruta (0,2,3,0) no se puede unir debido a que el cliente 2 tiene una conexión previa en la ruta (0,1,2,0) por lo que no se puede incluir al cliente 2 en otra ruta.
S <sub>24</sub>	No factible	La ruta (0,2,5,0) no se puede unir debido a que el cliente 2 tiene una conexión previa en la ruta (0,1,2,0) por lo que no se puede incluir al cliente 2 en otra ruta.
S <sub>14</sub>	No factible	La ruta (0,1,4,0) no se puede unir debido a que el cliente 1 tiene una conexión previa en la ruta (0,1,2,0) por lo que no se puede incluir al cliente 1 en otra ruta.
S <sub>35</sub>	Factible	La ruta (0,3,5,0) hace parte de la solución ya que los clientes 3 y 5 no han hecho conexiones previas con otros clientes. Además la suma de las demandas de los clientes cumple con la capacidad del vehículo.
S <sub>45</sub>	No factible	La ruta (0,4,5,0) no se puede unir debido a que el cliente 5 tiene una conexión previa en la ruta (0,3,5,0) por lo que no se puede incluir al cliente 5 en otra ruta.
S <sub>34</sub>	No factible	La ruta (0,3,4,0) no se puede unir debido a que el cliente 3 tiene una conexión previa en la ruta (0,3,5,0) por lo que no se puede incluir al cliente 3 en otra ruta.

De la anterior tabla se forman dos rutas que son la  $S_{12}$  y  $S_{35}$ , y es necesario añadir una tercera ruta,  $S_4$ , para el cliente 4 ya que no fue posible unirlo a una de las rutas ya establecidas. En la ruta

$S_{35}$  se aprovecha la capacidad del vehículo al 100% debido a que la demanda de los clientes y la capacidad del vehículo es la misma, mientras que para las rutas  $S_{35}$  y  $S_4$  se subutiliza la capacidad de los vehículos debido a que la demanda de los clientes es menor a la capacidad.

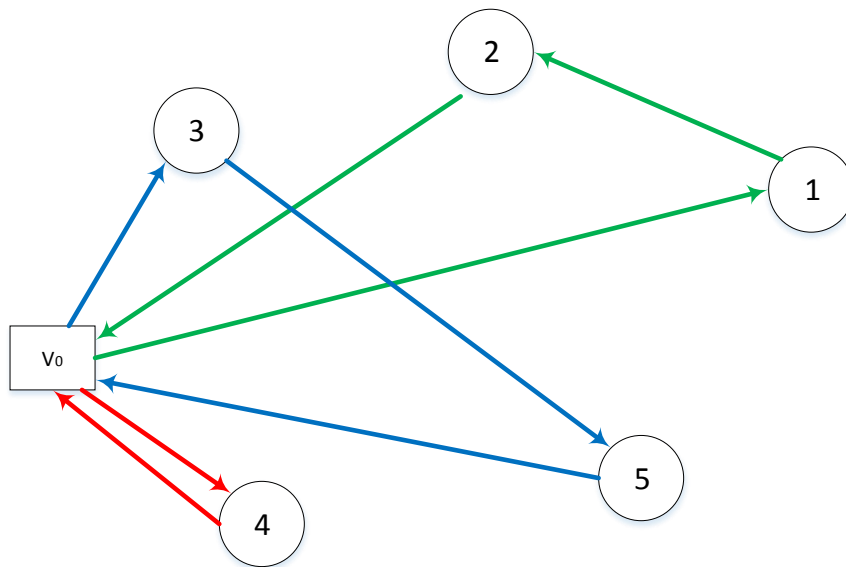


Figura 6. Solución final del ejemplo.

Para obtener la solución final mostrada en la figura 6 se unen todas las rutas que fueron factibles y aquellas que se añadieron para visitar los clientes que no se incluyeron en las rutas factibles. Las flechas de color verde representan la ruta que realiza el primer vehículo, las flechas azules representan la ruta del segundo vehículo y las flechas de color rojo representan la ruta que realiza el tercer vehículo.

**7.1.2 Conjunto de estructuras vecinales.** A continuación, se definirán las estructuras vecinales, para las cuales se toman como referencias las estructuras trabajadas por Pop, Fuksz y Marc (2014), en su artículo “*A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem*”. A continuación, se relaciona los símbolos con las cuales se representarán los ejemplos de las diferentes estructuras vecinales.

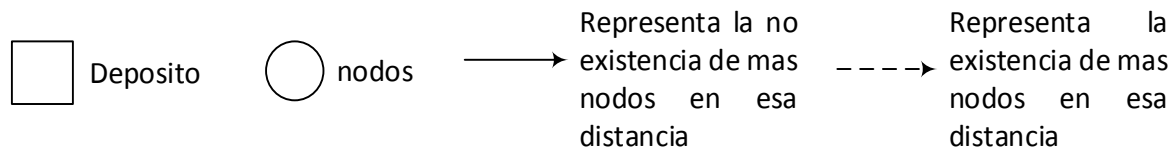


Figura 7. Símbolos para representación de estructuras vecinales.

**7.1.2.1 2-Opt.** Dos arcos globales que pertenecen a una sola ruta se reemplazan por otros dos arcos globales (Lin 1965) para mejorar el costo total de la ruta. Luego, utilizando el procedimiento local-global, se encuentra la mejor solución factible correspondiente y verificamos si el costo de la ruta mejoró. El tamaño del vecindario 2-Opt es cuadrático (con el número de agrupaciones) y solo hay un tipo de movimiento adecuado. Las siguientes figuras ilustran este proceso.

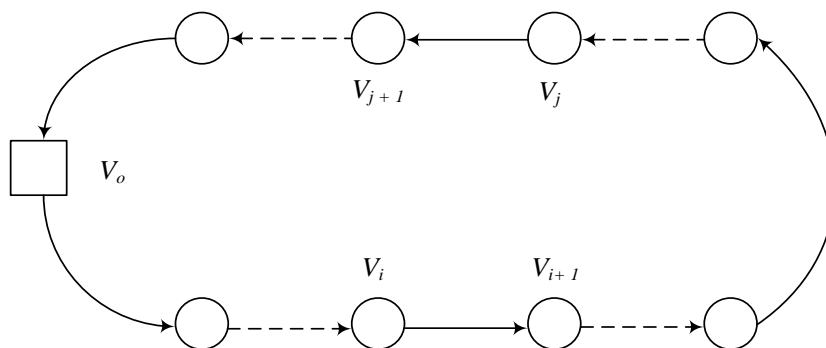


Figura 8. Ejemplo de una solución inicial para la estructura 2-opt. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., Fuzsz, L. y Marc, A., 2014).

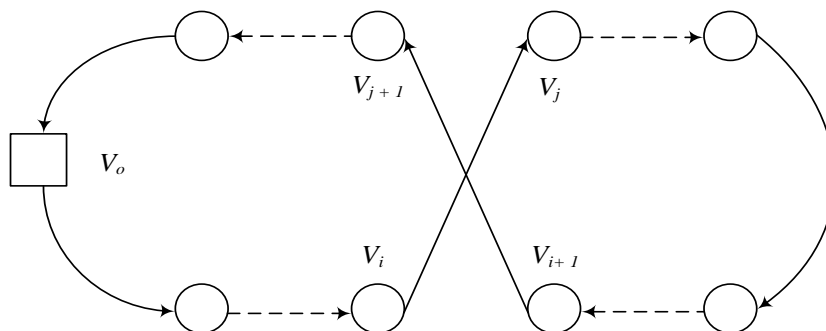


Figura 9. Ejemplo de aplicación de 2-opt. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., et al., 2014).

En la figura 8 se observa una solución inicial en la cual se describe el recorrido de un vehículo a través de los diferentes nodos pertenecientes a una ruta, la cual cuenta con un depósito  $V_0$  en donde el vehículo inicia y termina su recorrido. En la figura 9 se muestra la aplicación de la estructura vecinal 2-opt a la solución inicial, en donde 2 arcos son reemplazados por otros dos arcos dando como resultado el cambio de la trayectoria del vehículo en la ruta.

**7.1.2.2 Or-Opt.** La heurística Or-opt intenta mejorar el recorrido actual moviendo primero una cadena de 1, 2 o 3 vértices consecutivos en una ubicación diferente (y posiblemente invirtiéndola) hasta que no se pueda obtener ninguna mejora adicional (Or, 1976). El tamaño del vecindario Or-opt es cuadrático con la condición de que la longitud de la secuencia sea delimitada. La siguiente figura ilustra este proceso y contiene un ejemplo de un movimiento adecuado.

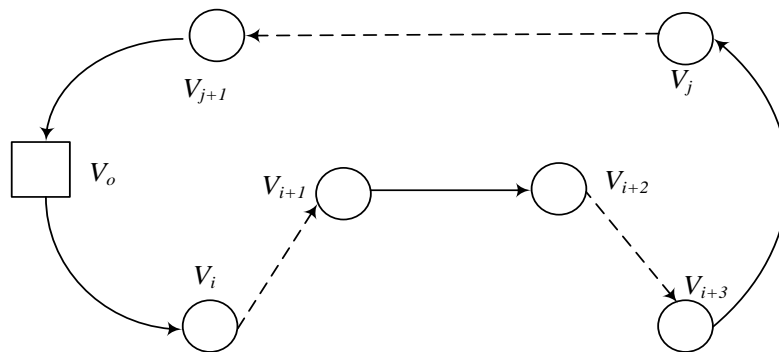


Figura 10. Ejemplo de una solución inicial para la estructura or-opt. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., et al., 2014).

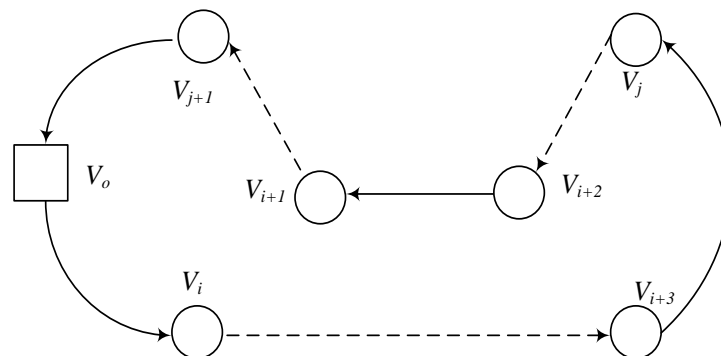


Figura 11. Ejemplo de aplicación de or-opt. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., et al., 2014).

En la figura 10 se observa una solución inicial en la cual se describe el recorrido de un vehículo a través de los diferentes nodos pertenecientes a una ruta, la cual cuenta con un depósito  $V_0$  en donde el vehículo inicia y termina su recorrido. En la figura 11 se muestra la aplicación de la estructura vecinal or-opt a la solución inicial, en donde se mueven 3 arcos consecutivos dando como resultado el cambio de la trayectoria del vehículo en la ruta.

**7.1.2.3 Intercambio de vecindario 1-0.** Dado un par de rutas globales que corresponden a una solución inicial del problema, el intercambio de vecindario 1-0 mueve un nodo de una ruta global a la otra (Pop, Fuksz & Marc, 2014), reemplazando tres arcos globales. Luego, utilizando el procedimiento local-global, se determina la mejor solución factible. La siguiente figura ilustra este proceso.

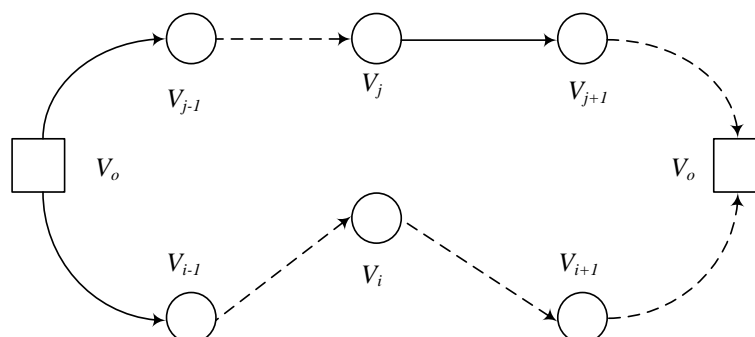


Figura 12. Ejemplo de una solución inicial para la estructura de intercambio de vecindario 1-0. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., et al., 2014).

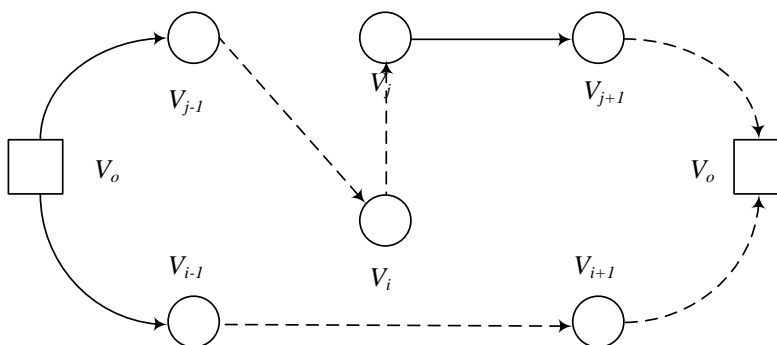


Figura 13. Ejemplo de una solución para la estructura de intercambio de vecindario 1-0. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., et al., 2014).

En la figura 12 se muestra una solución inicial en la cual se describe el recorrido de dos vehículos a través de los diferentes nodos pertenecientes a dos rutas, que cuentan con un depósito  $V_0$  en donde los vehículos inician y terminan su recorrido. En la figura 13 se muestra la aplicación de la estructura vecinal intercambio de vecindario 1-0 a la solución inicial, en donde un nodo de una ruta es cambiado a la otra ruta, reemplazando tres arcos.

**7.1.2.4 Intercambio de vecindario 2-2.** Dado un par de rutas globales que corresponden a una solución actual del problema, el vecindario de intercambio 2-2 intercambia las posiciones de 2 nodos consecutivos pertenecientes a una ruta global con dos nodos consecutivos de la otra ruta global (Savelsbergh 1992), eliminando cuatro arcos globales y creando cuatro nuevos. La siguiente figura ilustra este proceso.

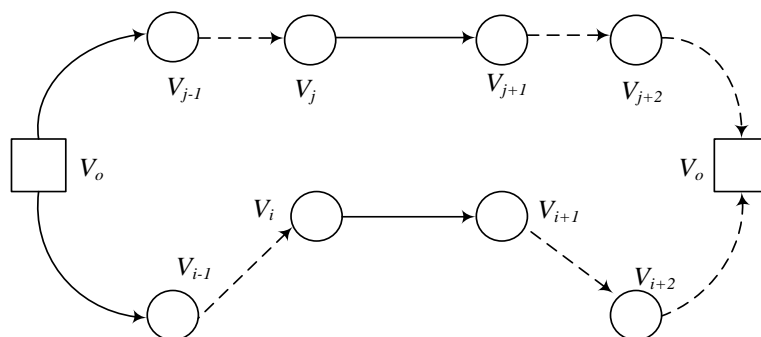


Figura 14. Ejemplo de una solución inicial para la estructura de intercambio de vecindario 2-2. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., et al., 2014).

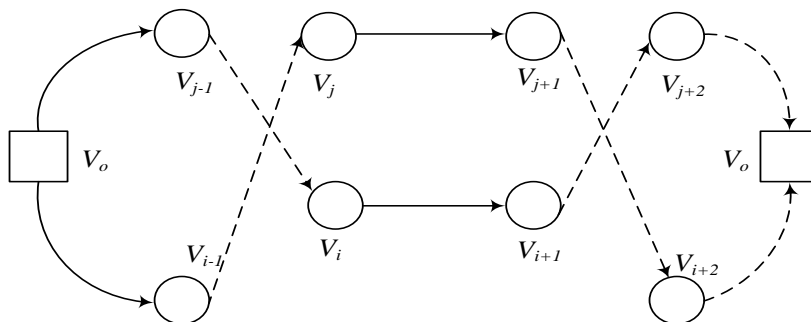


Figura 15. Ejemplo de una solución para la estructura de intercambio de vecindario 2-2. Adaptado de A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem (Pop, P., et al., 2014).

En la figura 14 se muestra una solución inicial en la cual se describe el recorrido de dos vehículos a través de los diferentes nodos pertenecientes a dos rutas, que cuentan con un depósito  $V_0$  en donde los vehículos inician y terminan su recorrido. En la figura 15 se muestra la aplicación de la estructura vecinal intercambio de vecindario 2-2 a la solución inicial, en donde 4 arcos son reemplazados por otros 4 arcos dando como resultado el cambio de la trayectoria de los vehículos en las rutas.

## 7.2 Fase de sacudida

En el paso de agitación del AVNS, se generan nuevas soluciones de acuerdo con estructuras de vecindario predefinidas (Sección 7.2.1). Los métodos específicos del problema se utilizan para la selección de las rutas y vértices que intervendrán en la sacudida (Sección 7.2.2). El algoritmo guía el paso de agitación adaptando las probabilidades de selección de estos métodos de acuerdo con su desempeño previo durante la búsqueda (Sección 7.2.3).

**7.2.1 Agitando vecindarios.** Similar a Stenger et al. (2013), se emplea un operador de intercambio cíclico para generar soluciones vecinas, originalmente introducido por Thompson y Orlin (1989). El intercambio cíclico mueve vértices entre rutas de una manera cíclica. Se caracteriza por dos parámetros el número de rutas involucradas  $\Omega$  y el número máximo de vértices a intercambiar  $\Gamma^{max}$ .

Para cada ruta  $k$  la secuencia de vértices  $\Psi_{jk, \Gamma_k}^k$  con el vértice de inicio  $j_k$  y la longitud  $\Gamma_k$  se transfiere a la ruta  $k + 1$  en la posición anterior de la secuencia  $\Psi_{jk+1, \Gamma_{k+1}}^{k+1}$ . En la Figura 16, el operador de intercambio cíclico se representa con  $\Omega = 3$  rutas, intercambiando  $\Gamma_1 = 1$ ,  $\Gamma_2 = 2$  y  $\Gamma_3 = 2$  vértices. Tenga en cuenta que, si el número total de rutas existentes es inferior al número



anteriores sobre problemas de enrutamiento.

Por otro lado, se implementan métodos específicos para problemas que tienen en cuenta las nuevas características de las paradas intermedias, por ejemplo, los desvíos asociados a las visitas por parte de los EV a las estaciones de intercambio de batería. Cada uno de los métodos se elige con una cierta probabilidad, que se actualiza dinámicamente durante la búsqueda en función del éxito del método en las iteraciones anteriores. Los métodos de selección y el mecanismo de adaptación se detallan a continuación.

**7.2.2.1 Selección de ruta.** La primera de las rutas  $\Omega_k$  del vecindario actual  $N_k$  se elige según uno de los siguientes cinco métodos de selección de ruta:

- **Aleatorio.** La probabilidad de ser seleccionado es igual para cada ruta.
- **Longitud de la ruta.** La probabilidad de que una ruta sea seleccionada es proporcional a la distancia de viaje asociada. La intención es eliminar los vértices de rutas largas y reinsertarlos en rutas más cortas para reducir los costos totales.
- **Longitud de la ruta por unidad de demanda.** La probabilidad de selección de una ruta es proporcional a la relación de la distancia total y la demanda acumulada de una ruta. Este criterio conducirá a una mejora de las rutas ineficientes.
- **Densidad de la instalación.** La probabilidad es proporcional a la relación entre el número de paradas intermedias y el número de clientes dentro de una ruta. El objetivo es favorecer las rutas que posiblemente contengan visitas redundantes a las instalaciones.
- **Desvío de la instalación.** La probabilidad es proporcional al desvío total resultante de las paradas intermedias. El objetivo es reducir los desvíos asociados y, por lo tanto, los costos generales.

Después de elegir la primera ruta por medio de uno de los procedimientos anteriores, las otras rutas que participarán en la sacudida se determinan iterativamente de la siguiente manera: la siguiente ruta se elige aleatoriamente del conjunto de todas las rutas restantes que están espacialmente más cerca que un umbral predefinido  $d^{max}$  a la ruta previamente seleccionada (Stenger et al. 2013a).

**7.2.2.2 Selección de secuencia de vértice.** Una vez que se determinan las rutas a involucrar, se deben identificar las secuencias de vértices que se eliminarán de cada ruta. Los siguientes tres métodos se utilizan para esta decisión de selección:

- **Aleatorio.** Cada secuencia de vértices se elige con la misma probabilidad.
- **Distancia a la siguiente ruta.** La probabilidad de seleccionar una secuencia de vértice es inversamente proporcional a la distancia de la secuencia a la ruta en la que se insertará. Esto se mide por la suma de las distancias de los vértices al centro de gravedad de la ruta objetivo.
- **Distancia a los vértices vecinos.** La probabilidad de seleccionar una secuencia es proporcional a la distancia de la secuencia a los vértices circundantes. Está dada por la suma de la distancia entre el primer vértice y su predecesora y la distancia entre el último vértice y su sucesor. Eliminar una secuencia que está lejos de los otros vértices de la ruta puede reducir los costos totales.

**7.2.3 Mecanismo adaptativo.** En cada paso de agitación, la elección de los métodos de selección de ruta y vértice se basa en las probabilidades. A cada método se le asigna la misma probabilidad al comienzo de la búsqueda. La probabilidad de cada método se actualiza dinámicamente en el curso de la búsqueda en función de su éxito en la mejora de la solución actual.

Para seleccionar los métodos, utilizamos el procedimiento de selección de la rueda de la ruleta según lo propuesto por Pisinger y Ropke (2007) para ALNS. Dados los métodos de selección  $h$ , a cada método  $s$  se le asigna un peso  $w_s$ . La probabilidad de seleccionar el método  $s$  se define entonces por  $w_s / \sum_{i=1}^h w_i$ . Después de  $\gamma$  iteraciones del AVNS, el peso de cada método se actualiza en función de su éxito durante estas iteraciones. El rendimiento de un método se mide mediante un sistema de puntuación. Se agrega un puntaje de nueve al puntaje total de un método cada vez que se logra una mejor solución general, un puntaje de tres si se mejora la solución actual y un puntaje de uno si la solución es peor que la actual, pero se acepta de acuerdo con el criterio de aceptación. Si  $\phi_s$  denota la puntuación actual del método  $s$  y  $\chi_s$  el número de aplicaciones del método desde la última actualización de peso, entonces el nuevo peso se calcula como  $w_s = w_s(1 - \rho) + \rho \frac{\phi_s}{\chi_s}$ . El parámetro del sistema  $\rho \in [0, 1]$  permite controlar en qué medida el valor pasado del peso influye en el nuevo. Los valores  $\phi_s$  y  $\chi_s$  se restablecen a cero después de cada actualización.

### 7.3 Fase de búsqueda de vecindario

Durante esta fase el algoritmo realiza una búsqueda completa en el  $k$ -ésimo vecindario de  $(y')$  a través de una búsqueda local hasta encontrar el óptimo local de ese vecindario el cual denotaremos como  $(y'')$ .

**7.3.1 Búsqueda local.** Para Rahmani, Cherif-Khettaf, y Oulamara, (2015) la búsqueda local es un proceso iterativo que empieza en una solución y la mejora realizando modificaciones locales. Básicamente empieza con una solución inicial y busca en su vecindad por una mejor solución. Si la encuentra, reemplaza su solución actual por la nueva y continua con el proceso,

hasta que no se pueda mejorar la solución actual.

La solución generada en el paso de agitación se mejora posteriormente mediante varios procedimientos de búsqueda local. Todos los operadores se implementan de manera tal que se acepte el primer movimiento de mejora.

Primero, las posibles violaciones de capacidad de la batería de los EV's dentro de una ruta se manejan agregando visitas a estaciones intermedias de intercambio de batería. Si la distancia entre dos visitas consecutivas a las estaciones de intercambio de batería supera la capacidad de carga de la batería del EV, el nivel de carga de la batería cae por debajo de cero en cierto punto. Por lo tanto, al menos una estación de intercambio de batería debe visitarse antes de este punto. Denotemos  $\phi$  la posición de la última instalación de reabastecimiento visitada y  $\sigma$  la del último vértice al que se puede acceder desde allí. La mejor posición de inserción se determina, por lo tanto, dentro de la trayectoria  $\{ \phi + 1, \dots, \sigma + 1 \}$ . Para cada posición posible, se calcula el costo de insertar la estación de intercambio de batería más cercana. Se realiza la inserción con el menor incremento de costo, pero en este paso, se opta por las inserciones que conducen a soluciones factibles en lugar de las que conducen a soluciones inviables. La inserción de las estaciones de intercambio de batería se realiza de manera análoga.

En un segundo paso, nuestro objetivo es mejorar el enrutamiento por medio de las estructuras de vecindarios mencionadas en la sección 7.2.1 las cuales se aplican en orden aleatorio. Finalmente, se lleva a cabo una aplicación de un operador de eliminación de estaciones de intercambio de batería, cuyo objetivo es eliminar las visitas redundantes a las estaciones de intercambio de batería. En un paso final, aplicamos dos operadores entre rutas. El operador de reubicación entre rutas mueve a un cliente de su ruta actual a otra, y el operador de intercambio intercambia dos clientes entre dos rutas.

#### **7.4 Fase de evaluación de la solución**

La solución  $y''$  obtenida por el procedimiento de búsqueda local se compara con la mejor solución actual  $y$ . Si  $y''$  es aceptada, reemplaza a  $y$  como solución inicial y  $k$  se restablece a 1.  $y''$  solo se acepta si está mejorando la solución inicial  $y$ . Después  $\epsilon$  iteraciones si no ha mejorado la solución actual se restablece a la mejor solución encontrada hasta el momento.

### **8. Validación del algoritmo propuesto**

Para validar el desempeño del algoritmo de búsqueda de vecindario variable adaptativo (AVNS) propuesto, el cual fue desarrollado en el software Python 3.7.1 y puesto en marcha en un computador ASUS modelo X555Q con procesador AMD A12 de 7ma generación, 3.4 GHz y 8GB de RAM.

Durante el desarrollo del BSS-EV-LRP no se encontraron instancias establecidas para la validación del mismo, por lo tanto, se utilizaron los resultados presentados por Prodhon (2014) para las instancias Tuzun (1999) y Barreto (2004) para el problema de localización-ruteo capacitado (CLRP, por sus siglas en inglés) ya que estos problemas son los más trabajados en la literatura y poseen características similares al problema tratado durante la presente investigación.

Prodhon en este estudio muestra la mejor solución conocida (Best known solution, BKS) hasta el momento para cada una de las instancias del CLRP, las cuales se tomarán como referencia para la validación del algoritmo AVNS desarrollado.

La notación utilizada en las tablas de resultados de la validación del algoritmo es la siguiente:

- Instancia: Es el nombre de cada instancia según el autor.
- BKS: Es el valor del costo de la mejor solución conocida en la literatura
- Valor obtenido: Es el resultado del costo de la función objetivo que se obtiene a través del algoritmo AVNS o VNS.
- Tiempo Computacional: Es el tiempo de procesamiento computacional para conocer el valor obtenido de cada instancia.
- N° de Estaciones: Es el número de estaciones óptimas que arroja el algoritmo para cada instancia.
- N° de Vehículos: Es el número de vehículos óptimos que arroja el algoritmo para cada instancia.
- Gap %: Es el porcentaje de diferencia entre el valor obtenido y el BKS.

### **8.1. Resultados de la validación del algoritmo AVNS propuesto**

En las tablas 6 y 7 se muestran los resultados obtenidos del algoritmo AVNS, la mejor solución conocida, el tiempo computacional, y el diferencial de Gap en porcentaje entre las soluciones con respecto a las instancias de Tuzun (1999) y Barreto (2004) respectivamente.

Estos resultados se obtuvieron después de que cada instancia, con la cual se valida el algoritmo, fuera ejecutada una única vez con un criterio de parada que corresponde a 400 iteraciones.

Tabla 6.  
*Algoritmo AVNS mediante instancias de Tuzun.*

<b>Instancia</b>	<b>BKS</b>	<b>Valor obtenido AVNS</b>	<b>Tiempo computacional [sg]</b>	<b>N° de Estaciones</b>	<b>N° de Vehículos</b>	<b>Gap %</b>
111112	1468,4	1952,78	234,70	4	7	32,99
111122	1449,2	1717,35	93,64	4	7	18,50
111212	1396,46	1848,43	52,32	4	7	32,36
111222	1432,29	1896,30	204,28	6	7	32,40
112112	1167,53	2005,42	56,27	4	7	71,77
112122	1102,7	1243,94	129,94	3	6	12,81
112212	793,97	542,28	160,27	0	5	-31,70
112222	728,3	753,75	78,71	2	5	3,49
113112	1238,49	1262,62	56,14	3	7	1,95
113122	1246,34	1682,53	253,62	4	7	34,00
113222	1021,31	1177,47	60,64	2	6	15,29
131122	1841,86	2260,31	308,41	6	9	22,72
131212	1981,37	2099,32	94,10	6	10	5,95
131222	1809,25	2684,65	95,40	9	10	48,38
132112	1448,27	1650,76	133,72	4	8	13,98
132122	1444,25	2270,69	51,38	7	9	57,22
132212	1206,73	1281,26	63,90	3	8	6,18
132222	931,94	2096,85	194,73	3	9	124,99
133122	1401,82	1420,21	113,77	3	8	1,31
133212	1199,51	699,34	80,37	0	7	-41,69
133222	1152,86	1499,69	134,56	4	8	30,08
121122	2185,55	3517,77	121,94	9	14	60,96
121212	2234,78	2847,88	248,65	5	12	27,43
121222	2259,52	2306,35	171,33	6	12	2,07
122112	2101,9	2041,96	136,28	3	12	-2,85
122122	1709,56	1969,27	104,70	4	11	15,19
122212	1467,54	1486,82	235,66	3	10	1,31
122222	1084,78	1944,34	315,10	5	11	79,24
123212	1771,06	1133,81	309,85	1	9	-35,98
123222	1393,62	1769,64	150,30	5	11	26,98

En la tabla 6, para las 30 instancias tratadas de Tuzun (1999) el algoritmo AVNS logra resultados óptimos en las instancias 112212, 133212, 122112, 123212 con Gaps de 31.70%, 41.69%, 2.85% y 35.98% respectivamente, superando así a las mejores soluciones conocidas en la

literatura y dando cumplimiento a la función objetivo de disminuir los costos. Para las demás instancias, la implementación del algoritmo AVNS muestra resultados de baja calidad ya que las soluciones difieren en comparación con las mejores soluciones conocidas en la literatura.

Tabla 7.  
*Validación del algoritmo AVNS mediante instancias de Barreto.*

<b>Instancia</b>	<b>BKS</b>	<b>Valor obtenido AVNS</b>	<b>Tiempo computacional [sg]</b>	<b>N° de Estaciones</b>	<b>N° de Vehículos</b>	<b>Gap %</b>
Christofides69-50x5	565,6	495,22	152,04	1	3	-12,44
Christofides69-75x10	861,6	648,00	90,42	1	4	-24,79
Christofides69-100x10	842,9	840,19	75,94	0	5	-0,32
Gaskell67-21x5	424,9	282,92	43,09	1	2	-33,41
Gaskell67-22x5	585,1	500,38	44,16	0	2	-14,48
Gaskell67-29x5	512,1	449,38	45,48	0	2	-12,25
Gaskell67-32x5	571,7	463,87	164,90	0	2	-18,86
Gaskell67-32x5	504,3	192,33	72,68	0	2	-61,86
Gaskell67-36x5	460,4	372,43	107,64	0	2	-19,11
Min92-27x5	3062	7019,7	19,54	0	8	129,2

Como se evidencia en la tabla 7, para las 10 instancias tratadas de Barreto (2004) el algoritmo AVNS logra resultados óptimos en las 9 primeras con Gaps que oscilan entre 0.32% y 61.86%, superando así a las mejores soluciones conocidas en la literatura y dando cumplimiento a la función objetivo de disminuir los costos. Para la última instancia, Min92-27x5, la implementación del

algoritmo AVNS muestra un resultado de baja calidad ya que la solución presenta gran discrepancia en comparación con la mejor solución conocida.

### 8.2. Resultados de la validación del algoritmo VNS

En esta sección se valida el algoritmo VNS, para el cual se implementa el mismo código del algoritmo AVNS y se omite la adaptación en la fase de sacudida. Esto con el fin de observar si los resultados obtenidos del algoritmo AVNS presentan mejoras en comparación con los resultados obtenidos del algoritmo VNS. Esta validación se realizará a través de las mismas instancias y condiciones con las que se validó el algoritmo AVNS en la sección 8.1.

En las tablas 8 y 9 se muestran los resultados obtenidos del algoritmo VNS, la mejor solución conocida, el tiempo computacional, y el diferencial de Gap en porcentaje entre las soluciones con respecto a las instancias de Barreto (2004) y Tuzun (1999) respectivamente. Estos resultados se obtuvieron después de que cada instancia, con la cual se valida el algoritmo, fuera ejecutada una única vez con un criterio de parada que corresponde a 400 iteraciones.

Tabla 8.  
*Validación del algoritmo VNS mediante instancias de Barreto.*

<b>Instancia</b>	<b>BKS</b>	<b>Valor obtenido VNS</b>	<b>Tiempo computacional [sg]</b>	<b>N° de estaciones</b>	<b>N° de vehículos</b>	<b>Gap %</b>
Christofides69-50x5	565,6	499,97	130,32	1	3	-11,60
Christofides69-75x10	861,6	671,549	95,55	1	4	-22,06
Christofides69-100x10	842,9	835,29	66,53	0	5	-0,90
Gaskell67-21x5	424,9	283,6	38,69	1	2	-33,25
Gaskell67-22x5	585,1	509,18	41,65	0	2	-12,98
Gaskell67-29x5	512,1	432,32	67,8	0	2	-15,58

[Continuación Tabla 8]

<b>Instancia</b>	<b>BKS</b>	<b>Valor obtenido VNS</b>	<b>Tiempo computacional [sg]</b>	<b>N° de estaciones</b>	<b>N° de vehículos</b>	<b>Gap %</b>
Gaskell67-32x5	571,7	485,79	117,186	0	2	-15,03
Gaskell67-32x5	504,3	487,72	116,82	0	2	-3,29
Gaskell67-36x5	460,4	375,68	70,86	0	2	-18,40
Min92-27x5	3062	6816,85	31,07	0	8	122,63

Como se evidencia en la tabla 8, para las 10 instancias tratadas de Barreto (2004) el algoritmo VNS logra resultados factibles en las 9 primeras con Gaps que oscilan entre 0.90% y 33.25%, superando así a las mejores soluciones conocidas en la literatura y dando cumplimiento a la función objetivo de disminuir los costos. Para la última instancia, Min92-27x5, la implementación del algoritmo VNS muestra un resultado de baja calidad ya que la solución presenta gran discrepancia en comparación con la mejor solución conocida.

Tabla 9.

*Validación del algoritmo VNS mediante instancias de Tuzun.*

<b>Instancia</b>	<b>BKS</b>	<b>Valor obtenido VNS</b>	<b>Tiempo computacional [sg]</b>	<b>N° de estaciones</b>	<b>N° de vehículos</b>	<b>Gap %</b>
111112	1468,4	1937,3696	85,5119	4	7	31,94
111122	1449,2	1749,063	103,44	4	7	20,69
111212	1396,46	1860,198	188,39	4	7	33,21
111222	1432,29	1835,205	181,4699	6	7	28,13
112112	1167,53	1759,8868	49,42	4	7	50,74
112122	1102,7	1187,41	237,538	3	6	7,68
112212	793,97	539,82	328,66	0	5	-32,01
112222	728,3	754,17	357,35	2	5	3,55
113112	1238,49	1259,74	96,863	3	7	1,72
113122	1246,34	1614,879	89,82	4	7	29,57
113222	1021,31	1239,01	95,2	2	6	21,32
131122	1841,86	2414,7	313,28	6	9	31,10
131212	1981,37	2027,61	163,28	6	10	2,33

[Continuación Tabla 9]

Instancia	BKS	Valor obtenido VNS	Tiempo computacional [sg]	Nº de estaciones	Nº de vehículos	Gap %
131222	1809,25	2449,98	259,55	9	10	35,41
132112	1448,27	1694,23	229,14	4	8	16,98
132122	1444,25	2080,91	107,7	7	9	44,08
132212	1206,73	1267,29	87,62	3	8	5,02
132222	931,94	2005,64	95,21	3	9	115,21
133122	1401,82	1308,849	83	3	8	-6,63
133212	1199,51	700,76	134,42	0	7	-41,58
133222	1152,86	1481,42	268,45	4	8	28,50
121122	2185,55	3769,85	150,25	9	14	72,49
121212	2234,78	2887,64	108,93	5	12	29,21
121222	2259,52	2367,209	69,78	6	12	4,77
122112	2101,9	2001,25	85,33	3	12	-4,79
122122	1709,56	1929,99	609,15	4	11	12,89
122212	1467,54	1469,14	92,59	3	10	0,11
122222	1084,78	1917,7	380,15	5	11	76,78
123212	1771,06	1139,69	132,7	1	9	-35,65
123222	1393,62	1831,87	105,98	5	11	31,45

En la tabla 9, para las 30 instancias tratadas de Tuzun (1999) el algoritmo VNS logra resultados factibles en las instancias 112212, 133122, 133212, 122112, 123212 con Gaps de 32.01% , 6.63%, 41.58%, 4.79% y 35.65% respectivamente, superando así a las mejores soluciones conocidas en la literatura y dando cumplimiento a la función objetivo de disminuir los costos. Para las demás instancias, la implementación del algoritmo VNS muestra resultados de baja calidad ya que las soluciones difieren en comparación con las mejores soluciones conocidas en la literatura.

### 8.3. Comparación de los resultados de validación de los algoritmos AVNS y VNS

Con el fin de observar si la adaptación propuesta realizada al algoritmo VNS muestra mejores resultados, se realiza una comparación entre las soluciones de los mismos mediante las instancias de Barreto y Tuzun.

Tabla 10.  
Comparación AVNS vs VNS para instancias de Barreto.

<b>Instancia</b>	<b>N° de Estaciones</b>	<b>N° de Vehículos</b>	<b>Solución AVNS</b>	<b>Solución VNS</b>	<b>Gap %</b>
Christofides69-50x5	1	3	495,22	499,97	0,96
Christofides69-75x10	1	4	648,00	671,549	3,63
Christofides69-100x10	0	5	840,19	835,29	-0,58
Gaskell67-21x5	1	2	282,92	283,6	0,24
Gaskell67-22x5	0	2	500,38	509,18	1,76
Gaskell67-29x5	0	2	449,38	432,32	-3,80
Gaskell67-32x5	0	2	463,87	485,79	4,73
Gaskell67-32x5	0	2	192,33	487,72	153,6
Gaskell67-36x5	0	2	372,43	375,68	0,87
Min92-27x5	0	8	7019,74	6816,85	-2,89

En la tabla 10 se muestra un mejor desempeño del algoritmo AVNS comparado con el algoritmo VNS para las instancias de Barreto, ya que en 7 de las 10 instancias comparadas se observan mejores resultados y en las 3 instancias restantes la diferencia en el Gap no supera el 5%.

Tabla 11.  
Comparación AVNS vs VNS para instancias de Barreto.

<b>Instancia</b>	<b>N° de Estaciones</b>	<b>N° de Vehículos</b>	<b>Solución AVNS</b>	<b>Solución VNS</b>	<b>Gap %</b>
111112	4	7	1952,78	1937,3696	-0,79
111122	4	7	1717,35	1749,063	1,85
111212	4	7	1848,43	1860,198	0,64
111222	6	7	1896,30	1835,205	-3,22
112112	4	7	2005,42	1759,8868	-12,24

[Continuación Tabla 11]

<b>Instancia</b>	<b>N° de Estaciones</b>	<b>N° de Vehículos</b>	<b>Solución AVNS</b>	<b>Solución VNS</b>	<b>Gap %</b>
112122	3	6	1243,94	1187,41	-4,54
112212	0	5	542,28	539,82	-0,45
112222	2	5	753,75	754,17	0,06
113112	3	7	1262,62	1259,74	-0,23
113122	4	7	1682,53	1614,879	-4,02
113222	2	6	1177,47	1239,01	5,23
131122	6	9	2260,31	2414,7	6,83
131212	6	10	2099,32	2027,61	-3,42
131222	9	10	2684,65	2449,98	-8,74
132112	4	8	1650,76	1694,23	2,63
132122	7	9	2270,69	2080,91	-8,36
132212	3	8	1281,26	1267,29	-1,09
132222	3	9	2096,85	2005,64	-4,35
133122	3	8	1420,21	1308,849	-7,84
133212	0	7	699,34	700,76	0,20
133222	4	8	1499,69	1481,42	-1,22
121122	9	14	3517,77	3769,85	7,17
121212	5	12	2847,88	2887,64	1,40
121222	6	12	2306,35	2367,209	2,64
122112	3	12	2041,96	2001,25	-1,99
122122	4	11	1969,27	1929,99	-1,99
122212	3	10	1486,82	1469,14	-1,19
122222	5	11	1944,34	1917,7	-1,37
123212	1	9	1133,81	1139,69	0,52
123222	5	11	1769,64	1831,87	3,52

En la tabla 11 se muestra un desempeño intermedio del algoritmo AVNS ya que en 12 de las 30 instancias de Tuzun comparadas arrojó mejores resultados que el algoritmo VNS, además en las 18 instancias restantes la diferencia en el Gap oscila entre 0.2% y 12.2%.

#### 8.4. Ejemplo numérico

A continuación, se realizará un ejemplo aplicado del algoritmo propuesto para observar el comportamiento del mismo, este ejemplo tomará como referencia las instancias Christofides69-50x5 y Christofides69-75x5 de Barreto ya que como se demostró en las secciones 8.1 y 8.2 son las que más se ajustan al algoritmo AVNS propuesto.

Estos resultados se obtuvieron después de que cada instancia fuera ejecutada 15 veces cada una. Además se modifica los siguientes parámetros de entrada:

- Capacidad de cada vehículo = 200
- Iteraciones = 400
- Demanda aleatoria = [1-50]
- Dimensiones = 200x400 pixeles.
- Clientes = 50, 75

Tabla 12.  
*Ejemplo numérico con 50 clientes.*

Corridas	Tiempo Computacional [sg]	Nº de Estaciones	Nº de Vehículos	Valor Obtenido AVNS
1	114	2	5	2897,9
2	42,98	3	5	2929,3
3	79,9	2	5	2941,7
4	62,13	2	5	2878,6
5	64,11	3	5	3016,7
6	51,39	3	5	3053,6

[Continuación Tabla 12]

Corridas	Tiempo Computacional	N° de Estaciones	N° de Vehículos	Valor Obtenido AVNS
7	59,62	3	5	3029,8
8	92,79	3	5	2941,5
9	56,05	2	5	2893,6
10	62,86	3	5	2966,3
11	70,46	3	5	2888,2
12	59,82	3	5	2964,9
13	150,51	2	5	2772,5
14	90,7	3	5	2854,9
15	67,86	3	5	2964,9

En la tabla 12 se muestran los resultados para el ejemplo planteado. En la corrida número 13 se muestra la mejor solución de la función objetivo del modelo, la cual dice que, mediante el algoritmo AVNS, para 50 clientes la cantidad de vehículos necesarios para cubrir las rutas y satisfacer la demanda de los clientes son 5, y que el número de estaciones de intercambio de batería a las que deben acudir los EV antes de que se agote la batería y así cumplir su ruta son 2.

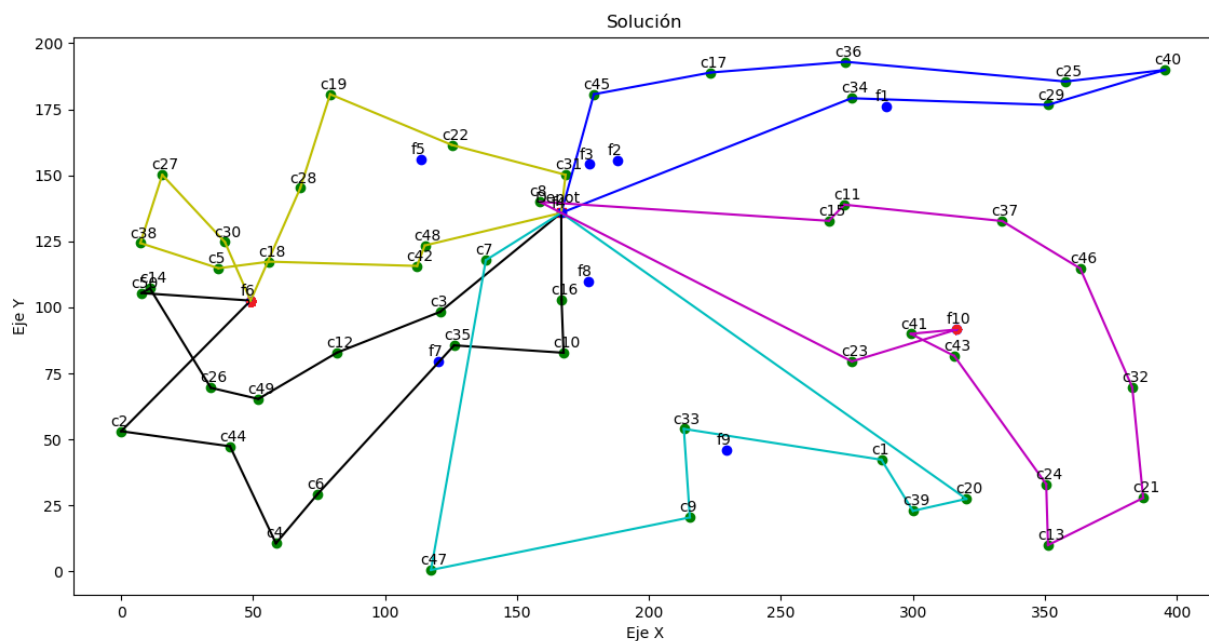


Figura 17. Rutas y ubicación de estaciones para 50 clientes. Adaptado del software Python.

En la figura 17 se muestran las 5 rutas que deben realizar los EV, la ubicación de las 2 estaciones (f6 y f10) y la ubicación de los 50 clientes para el ejemplo numérico según la programación del modelo mediante el algoritmo AVNS propuesto.

Tabla 13.  
*Ejemplo numérico con 75 clientes.*

<b>Corridas</b>	<b>Tiempo Computacional [sg]</b>	<b>N° de Estaciones</b>	<b>N° de Vehículos</b>	<b>Valor Obtenido AVNS</b>
1	38,62	6	8	4422,6
2	72,88	6	8	4429,9
3	56,23	4	8	4319,5
4	81,95	4	8	4200,5
5	60,45	6	8	4421,4
6	41,66	6	8	4418,9
7	100,56	5	8	4274,2
8	53,99	6	8	4285,3
9	32,66	6	8	4442,4
10	37,36	6	8	4458,4
11	62,84	5	8	4233,9
12	68,14	5	8	4301,5
13	76,99	5	8	4337,1
14	36,28	6	8	4430,4
15	46,76	4	8	4225,5

En la tabla 13 se muestran los resultados para el ejemplo planteado. En la corrida número 4 se muestra la mejor solución de la función objetivo del modelo, la cual nos dice que, mediante el algoritmo AVNS, para 75 clientes la cantidad de vehículos necesarios para cubrir las rutas y satisfacer la demanda de los clientes son 8, y que el número de estaciones de intercambio de batería a las que deben acudir los EV antes de que se agote la batería y así cumplir su ruta son 4.

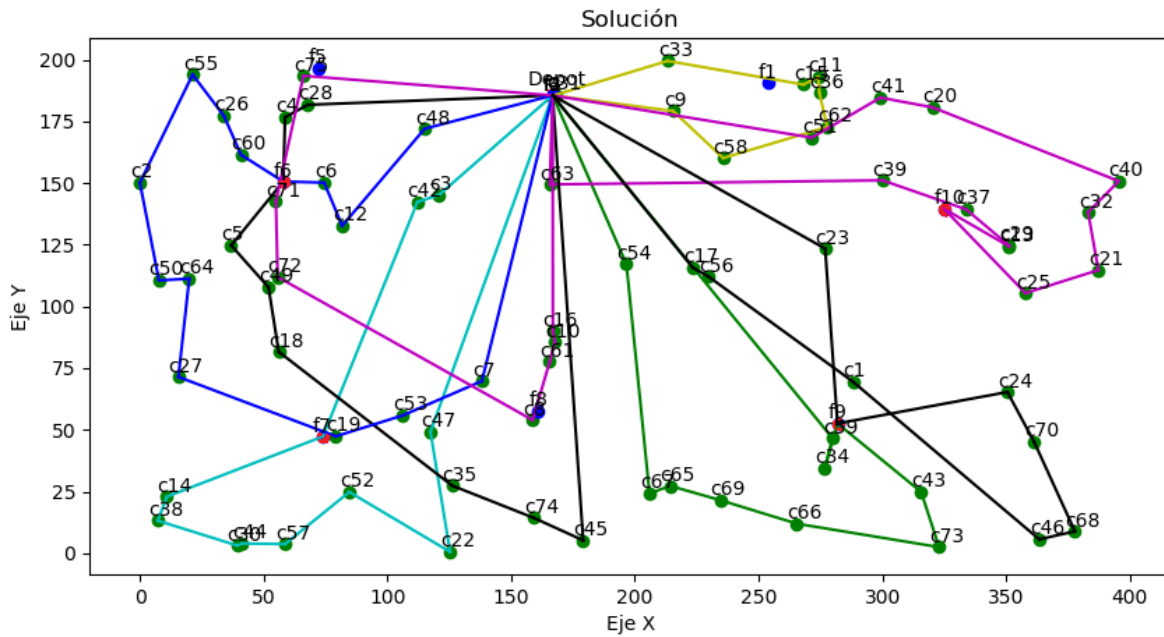


Figura 18. Rutas y ubicación de estaciones para 75 clientes. Adaptado del software Python

En la figura 18 se muestran las 8 rutas que deben realizar los EV, la ubicación de las 4 estaciones (f6, f7, f9, f10) y la ubicación de los 75 clientes para el ejemplo numérico según la programación del modelo mediante el algoritmo AVNS propuesto.

### 9. Conclusiones

Según la revisión de literatura, son pocos los trabajos de investigación que tratan la localización y ruteo de vehículos eléctricos, por lo tanto, esta investigación puede ser considerada como referencia para futuras investigaciones.

Los objetivos propuestos en la presente investigación titulada “Un modelo para la localización de estaciones de intercambio de batería y ruteo de vehículos eléctricos” se cumplen en su totalidad. El algoritmo diseñado arroja resultados factibles que dan solución al problema y cumple con los parámetros y restricciones del modelo propuesto.

El algoritmo AVNS propuesto para las instancias tratadas de Barreto (2004) logra resultados factibles en 9 de 10, con Gaps que oscilan entre 0.32% y 61.86%, superando así a las mejores soluciones conocidas en la literatura. Para la instancia, Min92-27x5, el algoritmo AVNS muestra un resultado alejado a la mejor solución conocida en la literatura.

El algoritmo AVNS propuesto para las instancias tratadas de Tuzun (1999) logra resultados óptimos en las instancias 112212, 133212, 122112, 123212 con Gaps de 31.70%, 41.69%, 2.85%, y 35.98% respectivamente, superando así a las mejores soluciones conocidas en la literatura. Sin embargo, para las demás instancias, el algoritmo AVNS muestra resultados alejados a la mejor solución conocida en la literatura.

La adaptación realizada en la fase de sacudida muestra un mejor desempeño del algoritmo AVNS comparado con el algoritmo VNS debido a que para las instancias de Barreto, en 7 de las 10 instancias comparadas arrojó mejores resultados y en las 3 instancias restantes la diferencia en el Gap no supera el 5%. También en 12 de las 30 instancias de Tuzun comparadas, el algoritmo AVNS arrojó mejores resultados que el algoritmo VNS, y en las 18 instancias restantes la diferencia en el Gap oscila entre 0.2% y 12.2%.

La representación de la solución es diseñada con el objetivo de separar la localización y el ruteo en la misma solución, como se evidencio en el ejemplo numérico esto permitió tener una mejor comprensión del resultado generado por el algoritmo.

La herramienta computacional propuesta en la presente investigación a través del algoritmo AVNS se convierte en una nueva alternativa de solución para problemas de localización de estaciones y de ruteo de vehículos eléctricos que permite obtener resultados óptimos para las organizaciones que consideren su implementación.

El desarrollo de este proyecto de grado en la modalidad de investigación represento un gran reto debido a que involucro conocimiento de diferentes áreas necesarias para una competente formación profesional en la ingeniería industrial.

## **10. Recomendaciones**

Ampliar el campo de aplicación de la presente investigación considerando variaciones que se puedan presentar en el problema de localización y ruteo de vehículos eléctricos.

Existe la posibilidad de utilizar otras heurísticas o metaheurísticas para resolver el problema de localización y ruteo de vehículos eléctricos, tales como: algoritmos genéticos, búsqueda tabú, colonia de hormigas; y de esta manera aprovechar los beneficios que se tienen al utilizar estos diferentes métodos de solución.

Crear instancias para el BSS-EV-LRP que permitan evaluar el desempeño de futuros algoritmos desarrollados para este problema.

Incentivar en estudiantes la realización de proyectos de grado que aborden investigaciones relacionadas con este proyecto de grado usando otros métodos de solución.

### Referencias Bibliográficas

- Aguilar, K. (2017). Un modelo multiobjetivo de localización-ruteo para la planeación logística en la fase de preparación a sismos en Bucaramanga. (Tesis de maestría). Universidad Industrial de Santander, Bucaramanga. Recuperado de <http://tangara.uis.edu.co/biblioweb/tesis/2017/165948.pdf>
- Barreto, M., y Niño, P. (2016). Un algoritmo memético para el problema de localización-ruteo con ventanas de tiempo para la atención de desastres sísmicos en la ciudad de Bucaramanga. Universidad Industrial de Santander, Bucaramanga. Recuperado de <http://tangara.uis.edu.co/biblioweb/tesis/2016/165444.pdf>
- Chicano, J. (2007). Metaheurísticas e ingeniería de software. *Lenguaje y ciencias de la computación*. Málaga, España: Universidad de Málaga.
- Cunquero, R. M. (2003). Algoritmos heurísticos en optimización combinatoria. *Universidad de Valencia, Facultad de Ciencias Matemáticas*. Recuperado de <http://yalma.fime.uanl.mx/~roger/work/teaching/mecbs5122/1-Introduction/Intro-byRafa%20Marti.pdf>
- Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Management Science*, 6(1), 80–91. <https://doi.org/10.1287/mnsc.6.1.80>
- Demir E, BektasT, Laporte G. (2012). Anadaptive large neighborhood search heuristic for the pollution-routing problem. *Eur J Oper Res* 223:346–59.

- Díaz, A., Glover, F., & Ghaziri, H. (1996). *Optimización Heurística y Redes Neuronales en dirección de Operaciones e Ingeniería*. Paraninfo S.A.
- Duarte, A., Pantrigo, J., & Gallego, M. (2007). *Metaheurísticas*. Madrid: DYKINSON.
- Erdoğ an S, Miller-Hooks, E. (2012) A green vehicle routing problem. *Transportation Res. Part E: Logistics Transportation Rev.* 48(1):100–114.
- Gagliardi, E. O., Hernández, G., Leguizamón, G., & Taranilla, M. T. (2007). Algoritmo genético para la Descomposición de Minkowski de poligonos convexos. Recuperado de <https://www.infor.uva.es/egc07/articulos/13.pdf>
- García, M. S. (2000). Optimización combinatoria. *Números*, (43), 115-120. *Recuperado de* <http://www.bdigital.unal.edu.co/26526/1/24129-84462-1-PB.pdf>
- Gonçalves F, Cardoso S.R., Relvas S, Barbosa-Póvoa, A. (2011). Optimization of a distribution network using electric vehicles: A VRP problem. IO2011—15 Congresso da associação portuguesa de Investigação Operacional (INESC, Lisboa, Portugal).
- Hansen P., Mladenovic´ N, 2001. Variable neighborhood search: Principles and applications. *European Journal of Operational Research*; 130:449-467.
- Hassanzadeh, A., Mohseninezhad, L., Tirdad, A., Dadgostari, F., & Zolfagharinia, H. (2009). Location-Routing Problem. *Contributions to Management Science*, 395–417.
- Hof, J., Schneider, M., & Goeke, D. (2016). Solving the battery swap station location-routing problem with capacitated electric vehicles using an AVNS algorithm for vehicle-routing problems with intermediate stops. *Transportation Research Part B*, 102-112.

- Johnson, D. S., & Garey, M. (1979). *Computers and Intractability: A Guide to the Theory of NPCompleteness*. New York: FREEMAN AND COMPANY.
- Keskin M, Çatay B (2016) Partial recharge strategies for the electric vehicle routing problem with time windows. *Transportation Res. Part C: Emerging Tech.* 65:111–127.
- Liang, Y.-C., Lo, M.-H., & Chen, Y.-C. (2007). Variable neighborhood search for redundancy allocation problems. *IMA Journal of Management Mathematics*, 18(2), 135–155.  
doi:10.1093/imaman/dpm010
- Lin, J., Zhou, W., Wolfson, O., (2016). Electric Vehicle Routing Problem. *Transportation Research Procedia* 12, 508 – 521
- Lin, S. (1965). Computer solutions to the traveling-salesman problem. *Bell Syst Techn J* 44(10):2245–2269
- Mak, Rong, and Shen. (2013). Infrastructure Planning for Electric Vehicles with Battery Swapping. *Management Science* 59(7), pp. 1557–1575
- MinComercio. (2017). Decreto Número 1116 de 2017. Retrieved from [http://es.presidencia.gov.co/normativa/normativa/DECRETO\\_1116\\_DEL\\_29\\_DE\\_JUNIO\\_DE\\_2017.pdf](http://es.presidencia.gov.co/normativa/normativa/DECRETO_1116_DEL_29_DE_JUNIO_DE_2017.pdf)
- Ministerio de Ambiente y Desarrollo Sostenible. (Septiembre de 2015). Miniambiente. Obtenido de El ABC de los compromisos de Colombia para la COP21:  
[http://www.minambiente.gov.co/images/cambioclimatico/pdf/colombia\\_hacia\\_la\\_COP21/ABC\\_de\\_los\\_Compromisos\\_de\\_Colombia\\_para\\_la\\_COP21\\_VF\\_definitiva.pdf](http://www.minambiente.gov.co/images/cambioclimatico/pdf/colombia_hacia_la_COP21/ABC_de_los_Compromisos_de_Colombia_para_la_COP21_VF_definitiva.pdf)
- Mladenovic´ N, Hansen P. 1997. Variable neighborhood search. *Comput Oper Res*; 24:1097–100.

- Montoya, A., Guéret, C., Mendoza, J. E., & Villegas, J. G. (2017). The electric vehicle routing problem with nonlinear charging function. *Transportation Research Part B: Methodological*, 103, 87–110. <https://doi.org/10.1016/j.trb.2017.02.004>
- Mundi, I. (2018). Index Mundi. Obtenido de <https://www.indexmundi.com/g/r.aspx?v=91&l=es>
- Mundial, B. (2018). Grupo Banco Mundial. Obtenido de [https://datos.bancomundial.org/indicador/EN.ATM.CO2E.KT?end=2014&name\\_desc=false&start=2014&view=chart](https://datos.bancomundial.org/indicador/EN.ATM.CO2E.KT?end=2014&name_desc=false&start=2014&view=chart)
- Nagy G, Salhi S. 2007. Location routing: issues, models and methods. *Eur J Oper Res*; 177(2):649–72.
- Navarro, R. B. (2008). *Meta-Heurísticas híbridas para optimización mono-objetivo y multiobjetivo. Paralelización y aplicaciones*. Universis Almenía.
- Nier, A., y Niño, J. (2015). Desarrollo de un modelo híbrido para resolver el problema de localización-ruteo (LRP). Universidad Industrial de Santander, Bucaramanga. Recuperado de <http://tangara.uis.edu.co/biblioweb/tesis/2015/157662.pdf>
- Niño, G., y Vargas, D. (2018). Algoritmo memético para el problema de localización y ruteo de vehículos con ventanas de tiempo en el proceso de recolección de residuos sólidos en el centro de la ciudad de Bucaramanga de la empresa de aseo de Bucaramanga-EMAB. Universidad Industrial de Santander, Bucaramanga. Recuperado de <http://tangara.uis.edu.co/biblioweb/tesis/2018/172947.pdf>
- Or, I. (1976). Traveling salesman-type problems and their relation to the logistics of regional blood banking. Ph.D. thesis, Department of Industrial Engineering and Management Sciences, Northwestern University Evanston, USA

- Osman, I. H., & Laport, G. (1996). Metaheuristics: A bibliography. *Annals of Operations Research*, 63, 513–623.
- Pichpibul T., Kawtummachai R., (2012). An improved Clarke and Wright savings algorithm for the capacitated vehicle routing problem. *ScienceAsia* 38(2012): 307-318. doi: 10.2306/scienceasia1513-1874.2012.38.307
- Pillac, V., Gendreau, M., Guéret, C., & Medaglia, A. L. (2013, February 16). A review of dynamic vehicle routing problems. *European Journal of Operational Research*. NorthHolland. <https://doi.org/10.1016/j.ejor.2012.08.015>
- Pop, P. C., Fuksz, L., & Marc, A. H. (2014). A Variable Neighborhood Search Approach for Solving the Generalized Vehicle Routing Problem. *Hybrid Artificial Intelligence Systems*, 13–24. doi:10.1007/978-3-319-07617-1\_2.
- Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, 31(12), 1985–2002.
- Prins, C., Prodhon, C. (2014). Métaheuristiques pour les problèmes de tournées de véhicules, dans. *Métaheuristiques*, pp. 387-412.
- Prins, C., Prodhon, C., & Calvo, R. W. (2006). Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking. *4OR*, 4(3), 221–238. doi:10.1007/s10288-006-0001-9
- Savelsbergh, M (1992). The vehicle routing problem with time windows: minimizing route duration. *ORSA J Comput* 4(2):146–154

- 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>
- Schneider, M., Stenger, A., Hof, J., 2015. An adaptive VNS algorithm for vehicle routing problems with intermediate stops. *OR Spectr.* 37 (2), 353–387.
- Stenger, A., Vigo, D., Enz, S., & Schwind, M. (2012). An Adaptive Variable Neighborhood Search Algorithm for a Vehicle Routing Problem Arising in Small Package Shipping. *Transportation Science*, 47(1), 64–80.[doi:10.1287/trsc.1110.0396](https://doi.org/10.1287/trsc.1110.0396)
- Tejero, J.J. (2015). El transporte de mercancías 2ª edición: Enfoque logístico de la distribución. ESIC Editorial.
- Thompson, P.M., Orlin, J.B. (1989). Theory of cyclic transfers. Working Paper, MIT, Operations Research Center, Cambridge, USA
- Yang, J., Sun, H. (2015). Battery swap station location-routing problem with capacitated electric vehicles. *Computers & Operations Research*, 1-16.
- Zona logística. (2016). Revista zona logística edición 90. Recuperado de <https://www.zonalogistica.com/wp-content/uploads/2016/05/Revista-zonalogistica-edicion-90.pdf>