



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA
MAESTRÍA EN OPTIMIZACIÓN Y CÓMPUTO APLICADO

**Diseño de un algoritmo metaheurístico para la
solución del Problema de Ruteo de Vehículos MultiDepósito**
T E S I S

QUE PARA OBTENER EL GRADO DE
MAESTRÍA EN OPTIMIZACIÓN Y CÓMPUTO APLICADO

PRESENTA

JHONY BRITO COLIN

DIRECTOR DE TESIS

DR. FEDERICO ALONSO PECINA

CO-DIRECTOR

DR. MARTIN HERIBERTO CRUZ ROSALES

REVISORES:

DR. FEDERICO ALONSO PECINA

DR. MARCO ANTONIO CRUZ CHÁVEZ

DRA. IRMA YAZMÍN HERNÁNDEZ BÁEZ

DR. MARTIN HERIBERTO CRUZ ROSALES

DR. MARTÍN GERARDO MARTÍNEZ RANGEL



Facultad de Contaduría,
Administración e Informática

CUERNAVACA, MORELOS.

FEBRERO, 2024

El problema de ruteo de vehículos multi-depósitos (MDVRP por sus siglas en inglés) es una variante del problema del ruteo de vehículos (VRP), en el que se puede atender a los clientes desde varios depósitos, se ha estudiado ampliamente debido a sus aplicaciones en muchas situaciones del mundo real. El MDVRP se encuentra relacionado con el problema de logística de distribución, al ser un problema clasificado como NP-Duro, es difícil encontrar su solución óptima usando métodos exactos en un tiempo polinomial.

En esta tesis se propone utilizar un algoritmo metaheurístico para resolver el problema de ruteo de vehículos multi-depósitos. El algoritmo implementado es Recocido Simulado (Simulated Annealing) el cual está inspirado en un proceso metalúrgico, consta de dos fases, y múltiples vecindades. Se utilizaron 23 instancias de la literatura para la experimentación, las cuales han sido utilizadas en todas las investigaciones que resuelven este problema, los resultados mostraron haber obtenido las soluciones óptimas en más del 50% de estas instancias. Asimismo, se generaron 6 instancias aleatorias con más depósitos y clientes de los que se encuentran en la literatura.

ABSTRACT

The multi-depot vehicle routing problem (MDVRP) is a variant of the vehicle routing problem (VRP), in which customers can be served from multiple depots, has been studied extensively due to its applications in many real-world situations. The MDVRP is related to the distribution logistics problem, being a problem classified as NP-Hard, it is difficult to find its optimal solution using exact methods in polynomial time.

In this thesis, it is proposed to use a metaheuristic algorithm to solve the multi-depot vehicle routing problem. The implemented algorithm is Simulated Annealing which is inspired by a metallurgical process, consists of two phases, and multiple neighborhoods. Twenty-three instances from the literature were used for experimentation, which have been used in all the researches that solve this problem, the results showed to have obtained the optimal solutions in more than 50% of these instances. Also, 6 random instances were generated with more depots and customers than those found in the literature.

AGRADECIMIENTOS

ÍNDICE

CAPÍTULO I INTRODUCCIÓN	7
1.1 Antecedentes	7
1.2 Planteamiento del Problema	7
1.3 Objetivos	8
1.3.1 Objetivo General	8
1.3.2 Objetivos Específicos	9
1.3 Hipótesis	9
1.4 Justificación	9
1.5 Alcances y limitaciones	10
1.6.1 Alcances	10
1.6.2 Limitaciones	10
1.6 Organización de la tesis	10
CAPÍTULO II	11
ESTADO DEL ARTE	11
2.1 Introducción	11
2.2 Algoritmos de optimización para MDVRP	13
2.2.1 Métodos Exactos	13
2.2.1.1 Branch and Cut	13
2.2.1.2 Programación Lineal	14
2.2.2 Heurísticas	14
2.2.2.1 Heurística ad hoc	15
2.2.2.2 Merge-Head & Drop-Tail	15
2.2.2.2 Búsqueda Tabú	16
2.2.2.3 Algoritmos Genéticos y evolutivos	17
2.2.2.4 Búsqueda Local Iterada	19
2.2.2.5 Algoritmo de Colonia de Hormigas	19
2.2.2.6 Búsqueda de Vecindad Variable	20
2.2.3 Métodos Híbridos	21
2.3 Recocido Simulado	22
2.4 Problema de Ruteo de Vehículos Multi-Deposito	25
2.4.1 Notación del MDVRP	25
2.4.2 Modelo Matemático para el MDVRP	26

2.4.3 Modelo de grafo no dirigido.....	27
2.4.4 Benchmark	28
CAPÍTULO III.....	31
METODOLOGÍA	31
3.1 Introducción	31
3.2. Generación de la solución inicial para instancias sin restricción distancia máxima.	32
3.3 Generación de la solución inicial para instancias con restricción de distancia máxima de recorrido.	36
3.3 Vecindarios	43
3.3.1 Vecindad 1: Swap Intra-ruta.....	43
3.3.2 Vecindad 2: Swap Inter-ruta.....	44
3.3.3 Vecindad 3: Reinserción Inter-ruta	44
3.3.4 Vecindad 4: Reinserción Intra-ruta	45
3.4 Algoritmo Recocido Simulado.....	46
3.5 Otras Vecindades	48
3.6 Generación de instancias aleatorias.....	48
CAPÍTULO IV.....	51
SINTONIZACIÓN Y RESULTADOS	51
4.1 Sintonización.....	51
4.1.1 Sintonización Temperatura inicial.....	51
4.3 Sintonización Temperatura final	52
4.4 Sintonización Longitud ciclo interno.	53
4.5 Sintonización Alfa.....	53
CAPÍTULO V CONCLUSIONES Y TRABAJOS FUTUROS.....	57
5.1 Conclusiones	57
5.2 Trabajos futuros	57
REFERENCIAS.....	58

CAPÍTULO I

INTRODUCCIÓN

1.1 Antecedentes

El problema de ruteo de vehículos multi-depósito (**Multi-Depot Vehicle Routing Problem**) es un problema que ha sido estudiado desde la década de 1970, esto debido a la creciente necesidad de generar rutas estratégicas de entrega a partir de múltiples centros de distribución, una compleja tarea que las empresas en crecimiento deben atender.

El problema MDVRP es un problema de optimización combinatoria difícil, cuya solución debe agendar la visita hacia todos los clientes haciendo uso de uno de los vehículos disponibles partiendo de algún depósito.

Debido a la complejidad del MDVRP, encontrar una buena solución a este problema sin el apoyo de la tecnología es una manera muy poco práctica, por lo que auxiliarse de las computadoras sea se ha vuelto imprescindible.

Encontrar soluciones “buenas” para este problema sigue siendo un tema de interés en la actualidad, para ello, frecuentemente se utilizan algoritmos heurísticos (o metaheurísticos) para encontrar una solución al problema planteado.

1.2 Planteamiento del Problema

El MDVRP es, al igual que VRP clásico, un problema de optimización combinatoria y categorizado por su complejidad como NP-Hard [Garey, 1979] (M. R. Garey and D. S. Johnson 1979), por lo tanto, es difícil encontrar su solución óptima usando métodos exactos en un tiempo polinomial.

El MDVRP funciona con un horizonte de planificación de un día, y asume que los vehículos tienen su base en más de un depósito.

El objetivo del MDVRP es minimizar el costo de entregar bienes a un conjunto de clientes con demandas conocidas, cumpliendo las restricciones del problema.

Adicionalmente a lo antes mencionado, se deben respetar ciertas características y/o restricciones que una solución debe cumplir para ser considerada factible o viable, las cuales son las siguientes:

- Cada ruta debe comenzar y terminar en el mismo depósito.
- Cada cliente debe ser visitado exactamente una vez.
- La demanda del cliente es un número diferente a cero y no negativo.
- La demanda total de cada ruta no debe exceder la capacidad del vehículo.
- Todos los vehículos tienen una misma capacidad
- Se dispone de un número limitado de vehículos.
- Todos los depósitos tienen el mismo número de vehículos disponibles.
- El recorrido de las rutas no debe exceder la longitud máxima del recorrido del vehículo, cuando exista el límite.

El cumplimiento de los puntos anteriores da como resultado un conjunto de rutas, cada ruta define una secuencia en la que cada cliente debe ser visitado, haciendo uso de un vehículo que pertenece a uno de los depósitos existentes, la distancia recorrida por cada ruta representa un costo. El objetivo es minimizar la suma total de los costos de viaje de todas las rutas. Al existir múltiples soluciones factibles, evaluar su calidad mediante su costo global (suma total de costos) efectuando una comparación entre ellas es esencial.

1.3 Objetivos

1.3.1 Objetivo General

Implementar un algoritmo metaheurístico de Recocido Simulado que pueda minimizar la suma de los costos de viaje de los vehículos que atienden la demanda de cada cliente.

1.3.2 Objetivos Específicos

- Revisar en la literatura los algoritmos que han resuelto con éxito el problema.
- Crear un algoritmo que pueda generar una solución inicial factible.
- Definir las vecindades que serán utilizadas para buscar en el espacio de soluciones.
- Implementar Recocido Simulado utilizando múltiples vecindades.
- Ajustar los parámetros del algoritmo Recocido Simulado (proceso de sintonización).

1.3 Hipótesis

H1: El algoritmo metaheurístico de Recocido Simulado implementado encontrará soluciones factibles que no estén alejadas del resultado mejor conocido en la literatura (a lo más 7% arriba de la cota) en todas de las instancias que se probarán.

1.4 Justificación

Uno de los planes estratégicos en el campo de la gestión de la cadena de suministros y la industria de la logística, es optimizar el problema de la entrega de productos por los vehículos de los proveedores a los clientes.

El problema de ruteo de vehículos multi-depósito (MDVRP) se encuentra en una variedad de contextos, y tiene una gran importancia económica considerable referente a la logística de abastecimiento, ya que está demostrado que se pueden lograr ahorros sustanciales mediante el uso de técnicas de optimización.

La solución eficaz de este problema impacta en 3 aspectos:

1. Al disminuir la distancia recorrida por los vehículos, se reduce el costo por el combustible o energía, también se reduce el desgaste del vehículo o componentes se reduce, tales como llantas, frenos, entre otros;

2. Al disminuir la distancia recorrida, las emisiones del vehículo también se reducen, por lo que la contaminación también se aminora.
3. Al reducir la distancia, se reduce el tiempo para la entrega de bienes y servicios, por lo que los clientes pueden verse más satisfechos.

1.5 Alcances y limitaciones

A lo largo de la experimentación al abordar el MDVRP y la evolución de los resultados obtenidos, se han planteado los siguientes alcances y limitaciones.

1.6.1 Alcances

- El algoritmo de Recocido Simulado trabajará con soluciones factibles.
- Implementar el algoritmo metaheurístico Recocido Simulado.
- Se sintonizarán de parámetros Recocido Simulado.

1.6.2 Limitaciones

- Se trabajará con el conjunto de 23 instancias utilizado en la literatura especializada que aborda este problema.
- El lenguaje utilizado para este proyecto de tesis es C++.

1.6 Organización de la tesis

En el capítulo II se muestra un análisis del estado del arte, evolución de las técnicas y enfoques que han sido utilizados para abordar este problema a lo largo del tiempo. En el capítulo III se expone la metodología utilizada para solucionar el problema, así como el proceso de mejoramiento de los resultados a lo largo de la experimentación. El capítulo IV contiene los resultados finales obtenidos al resolver el conjunto de instancias referentes al MDVRP utilizados en la literatura. En el capítulo V se encuentran las conclusiones del trabajo realizado para resolver el Problema de Ruteo De vehículos Multi-Deposito.

CAPÍTULO II

ESTADO DEL ARTE

2.1 Introducción

Encontrar las mejores rutas para que las flotas lleguen a sus clientes ha sido el enfoque principal del Problema de enrutamiento de vehículos (VRP), que implica calcular las direcciones o rutas de entrega de menor costo, desde un depósito a un conjunto de clientes geográficamente dispersos, Incluso después de realizar numerosos estudios sobre el VRP, dada la complejidad del problema (NP-hard) hasta el momento no se ha encontrado un método exacto que solucione el problema de forma eficiente.

A pesar de muchos estudios realizados con base en el VRP, no presenta aplicaciones muy realistas. Esto ha llamado más la atención sobre un escenario más realista MDVRP: Problema de enrutamiento de vehículos de múltiples depósitos, que es una variante de VRP.

Existen múltiples variantes del VRP que atienden diferentes necesidades e incluyen diversas restricciones o condiciones que deben cumplirse, y a la vez atienden diversas consideraciones impuestas por la dinámica de mercados, esto da como resultado la modelación de diferentes variantes del problema clásico de ruteo (VRP), de entre ellos se destacan las siguientes variantes:

- VRP con recolección y entrega (VRPPD). Estudia el caso donde una empresa debe recolectar y entregar bienes en cantidades específicas para cada cliente visitado.
- VRP con flota heterogénea. Es un problema muy común, donde los vehículos de la empresa tienen diferentes capacidades de carga.
- Open VRP. Se refiere al caso de ruteo abierto cuando el vehículo no regresa al depósito de origen, es decir, que se dirige a otro depósito o sitio diferente para cargar productos, en un contexto de flota rentada.
- VRP con ventanas de tiempo (VRPTW). Plantea que cada cliente tiene que ser atendido de manera obligada dentro de un cierto horario o “ventana de tiempo” específico.
- VRP con depósitos múltiples (MDVRP). Implica que la empresa posee diversos depósitos desde los cuales puede abastecer a los clientes.

- Stochastic VRP (SVRP). Este problema de ruteo propone que las variables involucradas son desconocidas o aleatorias, e incluso que se pueden encontrar en un rango de probabilidad, tales como el número de clientes, sus demandas, tiempo de servicio, tiempo de viaje, entre otras.
- Periodic VRP (PVRP). Analiza un problema donde los pedidos pueden ser llevados sólo en ciertos días.
- VRP multiobjetivo. Busca soluciones óptimas de ruteo considerando dos o más objetivos que se deben satisfacer por la empresa, pero que son contradictorios.

En esta tesis, como se ha señalado anteriormente, se trabajará con la variante MDVRP. En el Problema de generación de rutas para vehículos con múltiples depósitos (MDVRP por sus siglas en inglés), los vehículos comienzan y terminan su ruta en el mismo depósito, y hay varios depósitos para iniciar la ruta. Si un vehículo comienza la ruta desde el depósito A, dicho vehículo completa la ruta asignada una vez que regresa al depósito A.

Con el propósito de minimizar la distancia de la ruta. Entre sus diversos objetivos, MDVRP se centra en mejorar la entrega eficiente. Menos tiempo de entrega puede generar clientes satisfechos y ahorro en el costo de combustible. Una ruta bien planificada reduce la distancia de la ruta y la cantidad de vehículos en operación, lo que en última instancia minimiza el costo operativo de las entregas.

El MDVRP debido a su relevancia como un problema del mundo real, se realizaron los primeros trabajos de estudio en el año 1976, pero aún sin consolidar su nombre, fue en la década de 1990 que [Chao, 1993] propusieron finalmente un benchmark completo compuesto de 23 instancias, de esta manera futuros investigadores podrían resolver dichos problemas y comprar sus resultados obtenidos.

El MDVRP está clasificado como un problema de optimización combinatoria NP-Hard, Garey & Johnson (1979), existen dos tipos de algoritmos que pueden resolver este tipo de problemas, los algoritmos exactos o deterministas, y los algoritmos aproximados. Cada uno de ellos posee diferentes características que pueden considerarse ventajas y/o desventajas que deben ser tomadas en cuenta al ser utilizados.

2.2 Algoritmos de optimización para MDVRP

Para el modelo del MDVRP y sus condiciones de generación de rutas, desde su planteamiento como problema y modelado de restricciones hasta la actualidad se han propuesto diferentes técnicas para su solución, apoyándose tanto de algoritmos exactos como de aproximados.

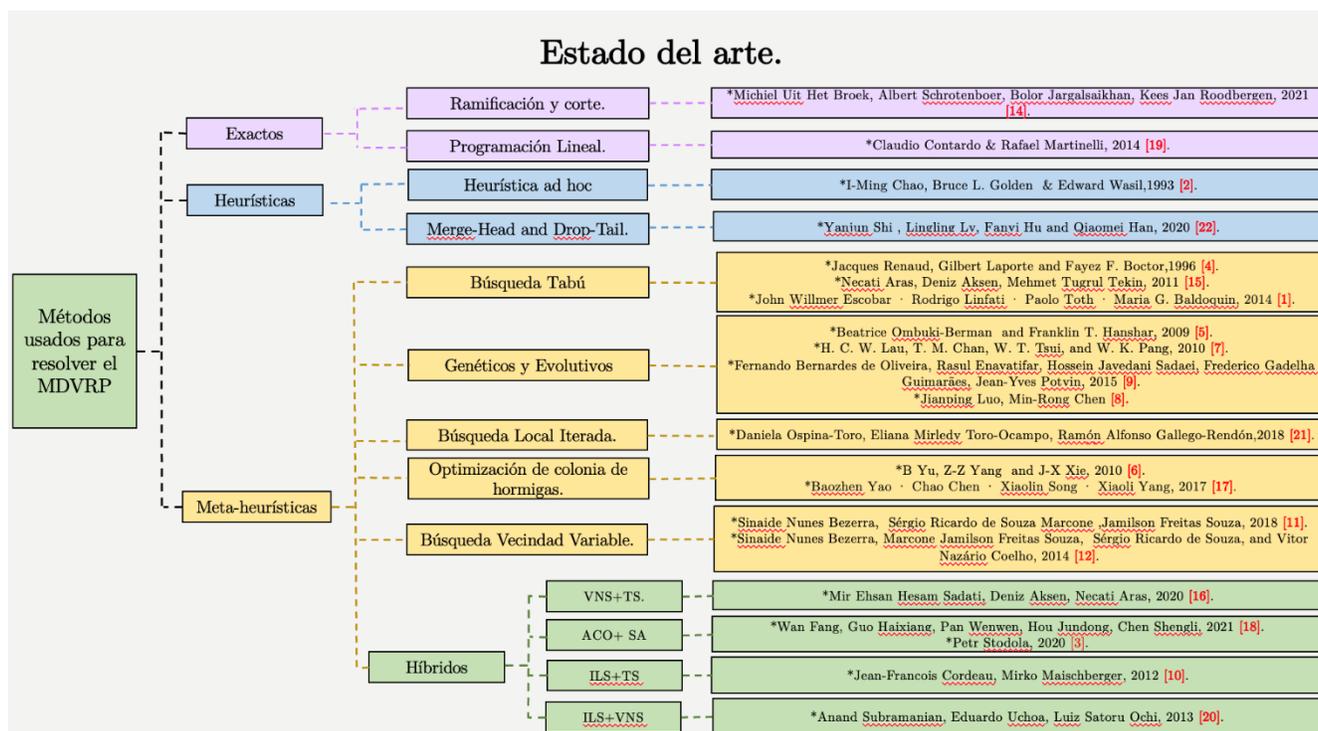


Fig. 2.1 Técnicas utilizadas para la solución del MDVRP

2.2.1 Métodos Exactos

Los métodos exactos son los métodos diseñados para resolver un problema de forma óptima. Sin embargo, no son muy ideales para problemas NP-difíciles a menos que el tamaño del problema sea considerablemente pequeño. Se han llevado a cabo algunas investigaciones introduciendo métodos exactos para MDVRP.

2.2.1.1 Branch and Cut

A comienzo de los 80, se propuso un algoritmo para resolver programas enteros grandes (en donde sus variables eran binarias) y tuvo bastante éxito. Este constaba de aplicar un algoritmo

de planos de corte a la relajación lineal del problema y luego resolver la relajación más ajustada con un algoritmo Branch-and Bound. Esta técnica es ahora conocida como Branch-and-Cut.

Ahora bien, si en cada nodo del árbol ejecutamos un algoritmo de planos de corte sobre la relajación lineal de dicho nodo, se pueden obtener cotas inferiores más ajustadas, lo que puede ocasionar que se incrementen los nodos que son podados por cota, del algoritmo Branch-and-Bound. Esta idea da origen a los algoritmos Branch-and-Cut.

En el trabajo de [uit het Broek, 2021] se presenta un marco genérico de ramificación y corte para resolver problemas de enrutamiento con múltiples depósitos y estructuras de costos asimétricas, que determina un conjunto de recorridos de vehículos (capacitados) que minimizan los costos y satisfacen un conjunto de demandas de los clientes.

2.2.1.2 Programación Lineal

La programación lineal es un método a través del cual se optimiza una función objetivo, bien sea maximizando o minimizando dicha función, en la cual las variables están elevadas a la potencia 1. Este campo de la programación lineal es un área de la programación matemática, dedicada a maximizar o minimizar una función lineal que recibe el nombre de "función objetivo", de manera que las variables de tal función estén sujetas a una serie de restricciones expresadas a través de un sistema de ecuaciones o inecuaciones lineales.

Se utilizó programación lineal entera en el trabajo de [Contardo, 2014], en el presentaron dos formulaciones que utilizaron para modelar y resolver el MDVRP, una formulación de flujo de vehículos y una de partición de conjuntos, las cuales fueron utilizadas en diferentes partes del algoritmo.

2.2.2 Heurísticas

A través del tiempo los problemas de optimización combinatoria han ido creciendo, a la par también lo han hecho las herramientas que sirven de apoyo para poder resolverlos, debido a

su naturaleza de complejidad exponencial, se ha hecho cada vez más difícil resolverlos de manera exacta, el problema del MDVRP es un ejemplo claro, debido al alto costo computacional, utilizar métodos aproximados resulta tener mayor viabilidad en cuanto a tiempo de solución comparado con el tiempo requerido por un método exacto, esto se ve reflejado en que la mayor parte de los trabajos que resuelven el MDVRP recurren a heurísticas y metaheurísticas para su solución.

“Heurística” es un concepto que viene desde la Grecia clásica, originado de la palabra griega *heuriskein* cuyo significado es encontrar o descubrir; según la historia se deriva de eureka, famosa exclamación atribuida a Arquímedes.

Aplicado el término heurística en técnicas para resolver problemas de optimización combinatoria, este toma un significado más exigente, para ilustrar lo anterior se presenta la siguiente definición: “Se califica de heurístico a un procedimiento para el que se tiene un alto grado de confianza en que encuentra soluciones de alta calidad con un coste computacional razonable, aunque no se garantice su optimalidad o su factibilidad, e incluso, en algunos casos, no se llegue a establecer lo cerca que se está de dicha situación. Se usa el calificativo heurístico en contraposición a exacto.” [Melián, 2004].

Los métodos heurísticos se han caracterizado por ser procedimientos con un buen nivel de eficiencia para encontrar “soluciones de calidad” aunque no sea comprobable que seas óptimas. La calidad de la solución obtenida tiene igualdad de importancia que la rapidez en este tipo de métodos.

2.2.2.1 Heurística ad hoc

En 1993 fue presentado una nueva heurística ad hoc en el aporte de [Chao, 1993], su método de solución asigna los clientes a los depósitos de manera fácil y rápida, En el procedimiento de mejora, permite que la distancia total recorrida por la flota aumente con la esperanza de encontrar finalmente rutas muy cortas.

2.2.2.2 Merge-Head & Drop-Tail

Una técnica planteada para resolver el MDVRP es la denominada estrategia Merge-Head and Drop-Tail (MHDT), combinada con optimización de combinación de sectores (SCO),

aplicada por [Shi, 2020], aquí se plantea un comportamiento inverso a la entrega, es decir debe generar rutas de recolección, dada esta única diferencia, el resto de las características y restricciones son idénticas, su propuesta funciona de igual manera para el MDVRP. En este método, primero asignan puntos de recolección de acuerdo con la distancia más cercana a los puntos de recolección, luego cada planta resuelve el problema de enrutamiento de un solo vehículo (VRP) respectivamente, asignando clientes a los vehículos y planificando el orden en que los vehículos visitan a los clientes. En el último paso, proponen el algoritmo de optimización de combinación de sectores (SCO) para generar múltiples soluciones iniciales, y luego estas soluciones iniciales se mejoran utilizando la estrategia de combinación de cabeza y caída (MHDT).

2.2.2.2 Búsqueda Tabú

La Búsqueda Tabú tiene sus antecedentes en métodos diseñados para cruzar cotas de factibilidad u optimalidad local tratadas como barreras en procedimientos clásicos, e imponer y eliminar cotas sistemáticamente para permitir la exploración de regiones no consideradas en otro caso.

Una característica distintiva de este procedimiento es el uso de memoria adaptativa y de estrategias especiales de resolución de problemas. TS es el origen del enfoque basado en memoria y estrategia intensiva en la literatura de las metaheurísticas, en contraposición con los métodos que no tienen memoria o que sólo usan una débil memoria basada en herencia. TS es también responsable de enfatizar el uso de los diseños estructurados para explotar los patrones históricos de la búsqueda, de forma opuesta a los procesos que confían casi exclusivamente en la aleatorización [Glover, 2003].

Unos de los primeros trabajos que se encuentran de Búsqueda Tabú en MDVRP es presentado en [Renaud, 1996], en la que explora el espacio de búsqueda pasando de una solución a su mejor vecino, incluso si esto da como resultado un deterioro del valor de la función objetivo. De esta manera, aumenta la probabilidad de salir de los óptimos locales. Para evitar el ciclo, las soluciones que se examinaron recientemente están prohibidas o declaradas tabú para un cierto número de iteraciones.

En el trabajo de [Aras, 2011] propone una heurística TS con vecindarios enriquecidos (TS-RN) para resolver el problema, donde además se combina con la oscilación estratégica, admitiendo también aquellas soluciones en las que uno o más recorridos violan las restricciones de capacidad del vehículo.

Posteriormente [Escobar, 2014] en su trabajo utiliza vecindarios restringidos llamados vecindarios granulares e implementa tres estrategias de diversificación para permitir la exploración de nuevas partes del espacio de soluciones, construyendo una solución inicial con un procedimiento híbrido para continuar dando entrada a búsqueda Tabú.

2.2.2.3 Algoritmos Genéticos y evolutivos.

Los algoritmos evolutivos (AE) son técnicas de búsqueda inspiradas en la evolución biológica de las especies y aparecieron a finales de la década de 1950. Los algoritmos genéticos (AG) constituyen ciertamente el ejemplo más conocido. Los métodos evolutivos despertaron inicialmente un interés limitado, debido a su costo significativo de ejecución. Pero, en los últimos años, han experimentado un desarrollo considerable, que se puede atribuir al aumento significativo en el poder de cómputo de las computadoras, en particular, luego de la aparición de arquitecturas masivamente paralelas, que explotan el “paralelismo intrínseco”.

En la práctica, se debe seleccionar una representación para los individuos de una población. Clásicamente, un individuo puede ser una lista de enteros para un problema combinatorio, un vector de números reales para un problema numérico en un espacio continuo, o una cadena de números binarios para un problema booleano; incluso se pueden combinar estas representaciones en estructuras complejas si se siente la necesidad. El paso de una generación a la siguiente se desarrolla en cuatro fases: una fase de selección, una fase de reproducción, una fase de evaluación del desempeño y una fase de reemplazo. La fase de selección designa a los individuos que participan en la reproducción. Son elegidos, posiblemente en varias ocasiones, a priori más a menudo los que son más “aptos”. Los individuos seleccionados quedan entonces disponibles para la fase de reproducción. Esta fase consiste en aplicar operadores de variación a las copias de los individuos seleccionados para generar nuevos

individuos; los operadores que se utilizan con más frecuencia son el cruce, que produce uno o dos descendientes a partir de dos padres, y la mutación, que produce un nuevo individuo a partir de un solo individuo. La estructura de los operadores de variación depende en gran medida de la representación seleccionada para los individuos. Los desempeños de los nuevos individuos son luego evaluados durante la fase de evaluación, a partir de los objetivos especificados [Siarry, 2016].

Uno de los trabajos en lo que se emplea esta técnica se encuentra el aporte de [Ombuki-Berman 2009]. El trabajo propuesto emplea una codificación indirecta y una estrategia adaptativa de intercambio de mutaciones entre depósitos para el MDVRP con restricciones de capacidad y longitud de ruta, a cada cliente se le asigna el depósito más cercano en términos de distancia euclidiana. Esta estrategia es simple, rápida y viable ya que no se impone un límite de capacidad en cada depósito. Durante esta asignación inicial, algunos clientes se identifican además como casos límite. Durante el proceso de evolución, una mutación entre depósitos puede reasignar a los clientes límite (es decir, se encuentran a cierta distancia de más de un depósito) a su ubicación final en un depósito determinado.

Un año después fue presentado el trabajo de [Lau, 2010], proponen utilizar una técnica de búsqueda estocástica llamada algoritmos genéticos guiados por lógica difusa (FLGA) para resolver el problema. El papel de la lógica difusa es ajustar dinámicamente la tasa de cruce y la tasa de mutación después de diez generaciones consecutivas.

Un trabajo más reciente fue elaborado por [de Oliveira, 2015], presenta un enfoque de descomposición de problemas para el MDVRP en el que cada subproblema se convierte en un VRP de depósito único y evoluciona de forma independiente en su espacio de dominio. Los clientes se distribuyen entre los depósitos según su distancia de los depósitos y su distancia de su vecino más cercano. A cada depósito se le asocia una población donde los individuos representan soluciones parciales al problema, es decir, conjuntos de rutas sobre clientes asignados al depósito correspondiente.

Otro de los trabajos en los que este tipo de algoritmos es puesto en uso, fue en el elaborado por [Luo, 2014], se presenta un algoritmo mejorado Shuffled Frog Leaping(SFLA), y su modelo multifase para resolver los problemas de enrutamiento de vehículos de múltiples

depósitos (MDVRP). Para mejorar aún más la capacidad de búsqueda local de SFLA y acelerar la convergencia, se introduce en SFLA una búsqueda de vecindad de optimización extrema de ley de potencia (PLEONS, por sus siglas en inglés).

2.2.2.4 Búsqueda Local Iterada

Esta metaheurística se conforma de un procedimiento sencillo y que a la vez es muy efectivo. Se utiliza una solución inicial dada como punto de partida, esta puede ser elaborada utilizando un método aleatorio o un procedimiento muy elaborado. Con la finalidad de llegar a otras regiones en el espacio de soluciones, le es aplicado un proceso de perturbación/diversificación. A la solución obtenida, se le aplica búsqueda local, lo que nos permite obtener un nuevo óptimo. Este óptimo puede ser aceptado o no, ya que existen dos criterios de elección: seleccionar la mejor solución o seleccionar siempre la nueva solución, a pesar de que esta sea peor, ya que de esta manera se diversifica y se evita caer en óptimos locales.

Esta técnica es aplicada por [Toro, 2018] planteando dos etapas, la primera efectúa un proceso de intensificación, la segunda efectúa un proceso de diversificación, con las cuales logra salir de óptimos locales y llegar a otros que eventualmente pueden ser de mejor calidad. De esa forma realiza búsquedas locales, seguido de perturbaciones que le permite continuar la búsqueda local en otras regiones. Así, en cada paso del proceso el ILS concentra la búsqueda en un conjunto reducido de soluciones, en lugar de considerar todo el espacio de búsqueda a la vez.

2.2.2.5 Algoritmo de Colonia de Hormigas

La optimización de colonias de hormigas es una técnica de optimización que se introdujo a principios de la década de 1990, [Dorigo, 1992].

Optimización de colonias de hormigas (ACO), es una de las técnicas más recientes para la optimización aproximada. La fuente inspiradora de los algoritmos ACO son las colonias de hormigas reales. Más específicamente, ACO está inspirado en el comportamiento de búsqueda de alimento de las hormigas. En el centro de este comportamiento está la

comunicación indirecta entre las hormigas por medio de rastros de feromonas químicas, lo que les permite encontrar caminos cortos entre su nido y las fuentes de alimento. Esta característica de las colonias de hormigas reales se aprovecha en los algoritmos ACO para resolver, por ejemplo, problemas de optimización [Bulum 2005].

Esta técnica es utilizada en el trabajo de [Yu, 2011], generando un depósito virtual, donde cada ruta comienza su ruta desde el depósito central virtual a través de un depósito real sin exceder las limitaciones de capacidad del vehículo transfiriendo el problema a V-MDVRP, generando una similitud con el VRP clásico, pero respetando las restricciones del MDVRP.

Años más tarde fue publicado el aporte de [Yao, 2019], utiliza un método para reducir la complejidad del problema con un depósito ficticio, que es similar al VRP con el depósito ficticio como origen. Luego se utiliza ACO con algunas estrategias de mejora para resolver el problema, adopta también una estrategia de escaneo, este escaneo se basa en el ángulo entre el punto del cliente y el almacén. Por lo tanto, determina el ángulo entre el punto del cliente y el depósito del almacén. Dado que hay varios depósitos en MDVRP, las rutas de distribución, que pertenecen al mismo depósito, se optimizan en primer lugar mediante la estrategia de exploración, añadiendo una operación cruzada para mejorar el rendimiento de ACO.

2.2.2.6 Búsqueda de Vecindad Variable

La Búsqueda por Entornos o Vecindarios Variables (Variable Neighborhood Search, VNS, en inglés), es una metaheurística que considera distintas estructuras de vecindarios y las cambia sistemáticamente para escapar de los mínimos locales. El VNS básico, a partir de una solución inicial, ejecuta una Búsqueda Local cuyo procedimiento consiste en reemplazar la solución actual si ha habido una mejora o modificar la estructura del vecindario en caso contrario, [Rivera, 2010].

Podemos ver el uso de esta técnica en el aporte de [Bezerra, 2019], utiliza algunos conocimientos específicos del problema y utiliza un mecanismo de aprendizaje adaptativo para encontrar la estructura de vecindario más adecuada durante el proceso de búsqueda. En

la parte de aprendizaje adaptativo, el algoritmo memoriza la estructura de vecindad que mejora la calidad de la solución.

Anteriormente esta técnica fue utilizada con un diferente comportamiento por [Bezerra, 2018], este trabajo se propone una Búsqueda de Vecindad Variable General (GVNS) para resolver MDVRP el algoritmo propuesto utiliza el Randomized Variable Neighborhood Descent (RVND) como búsqueda local de GVNS. La estrategia utilizada para generar la solución inicial es "agrupar primero y luego enrutar", de esta forma, los clientes son asignados a los depósitos de forma equilibrada.

2.2.3 Métodos Híbridos

El rápido éxito de las metaheurísticas se debe a las dificultades que encuentran los métodos tradicionales de optimización en problemas complejos de ingeniería. Tras el éxito inicial de utilizar varias metaheurísticas, llegó el momento de hacer una valoración realista y aceptar la complementariedad de estos nuevos métodos, tanto con otros métodos de este tipo como con otros enfoques: a partir de ahí, vimos la aparición de métodos híbridos.

En los últimos años, el interés por las metaheurísticas híbridas ha aumentado considerablemente entre los investigadores de optimización combinatoria. Los mejores resultados encontrados para muchos problemas de optimización prácticos o académicos se obtienen mediante algoritmos híbridos. Las combinaciones de algoritmos como la búsqueda local de descenso, el recocido simulado, la búsqueda tabú y los algoritmos evolutivos han proporcionado algoritmos de búsqueda muy potentes [Talbi, 2002].

La utilización de dichas técnicas híbridas se ha vuelto un recurso utilizado en múltiples trabajos aplicado al MDVRP. A continuación, abordaremos algunos trabajos que han hibridado metaheurísticas para el MDVRP.

Uno de los aportes en los que se puede ver el uso las hibridaciones es en [Sadati, 2020], en el que se diseñó un algoritmo metaheurístico que combina Búsqueda de Vecindad Variable

y Búsqueda Tabú, y refiere que su estrategia de optimización consta de tres niveles, donde cada uno se conforma de tres componentes con una función objetivo cada uno.

La combinación de Optimización Colonia de Hormigas (ACO) y Recocido Simulado (SA) puede ser encontrado en [Wan, 2021], ya que ACO cae más fácilmente en el óptimo local, introduce SA, para alentar a ACO a saltar fuera del óptimo local, e introduciendo tres operaciones de optimización local: optimización local entre rutas, optimización local en la ruta y optimización de escaneo de ángulo para mejorar la capacidad de optimización del algoritmo.

En el trabajo de [Stodola, 2020], Optimización Colonia de Hormigas (ACO) y Recocido Simulado (SA) fueron las metaheurísticas elegidas su investigación. El algoritmo híbrido ACO+ combina el algoritmo ACO con principios de recocido simulados y un proceso de optimización local determinista. Mientras que la optimización local mejora la mejor solución encontrada en cada generación, el recocido simulado evita que la solución converja con los óptimos locales.

La hibridación de las técnicas Búsqueda Tabú (TS) y Búsqueda Local Iterada (ILS) fue presentada en [Cordeau, 2012], Utilizando el marco de búsqueda local iterado, la heurística combina la búsqueda tabú con un mecanismo de perturbación simple para garantizar una exploración amplia del espacio de búsqueda. Para garantizar una amplia exploración se utilizan mecanismos de diversificación y relajación de restricciones.

Otro trabajo de este tipo es el de [Subramanian, 2013], en él se puede encontrar la hibridación de Búsqueda Local Iterada (ILS) y Búsqueda de Vecindad Variable (VNS), utiliza un procedimiento de inicio múltiple que utiliza heurísticas de inserción en la fase constructiva, un descenso de vecindad variable con ordenamiento de vecindad aleatorio (RVND) en la fase de búsqueda local y movimientos simples como mecanismos de perturbación.

2.3 Recocido Simulado

Kirkpatrick y sus colegas eran especialistas en física estadística, que se interesaban específicamente por las configuraciones de baja energía de los materiales magnéticos desordenados, denominados vidrios de espín. La determinación numérica de estas configuraciones planteaba espantosos problemas de optimización, porque el "paisaje energético" de un cristal de espín contiene varios "valles" de profundidad desigual.

Kirkpatrick e independientemente Cerny, propusieron abordar estos problemas tomando como punto de partida la técnica experimental de recocido utilizada por los metalúrgicos para obtener un estado sólido "bien ordenado", de mínima energía (evitando las estructuras "metaestables" características de los mínimos locales de la energía). Esta técnica consiste en calentar un material a una temperatura elevada y luego bajar esta temperatura lentamente.

El método de recocido simulado traslada el proceso de recocido a la solución de un problema de optimización: la función objetivo del problema, similar a la energía de un material, se minimiza a continuación, con la ayuda de la introducción de una temperatura ficticia, que en este caso es un simple parámetro controlable del algoritmo.

En la práctica, la técnica explota el algoritmo de Metrópolis, que permite describir el comportamiento de un sistema termodinámico en "equilibrio" a una temperatura determinada. A partir de una configuración dada, el sistema se somete a una modificación elemental (por ejemplo, se puede reubicar un componente o intercambiar dos componentes). Si esta transformación hace que disminuya la función objetivo (o la energía) del sistema, se acepta. En cambio, si provoca un aumento de la función objetivo, también puede aceptarse, pero con una probabilidad $e^{-\Delta E/T}$. Este proceso se repite de forma iterativa, manteniendo constante la temperatura, hasta que se alcanza el equilibrio termodinámico, al final de un número "suficiente" de modificaciones. Entonces se baja la temperatura, antes de aplicar una

nueva serie de transformaciones: la regla por la que se baja la temperatura por etapas suele ser empírica, al igual que el criterio de terminación del programa.

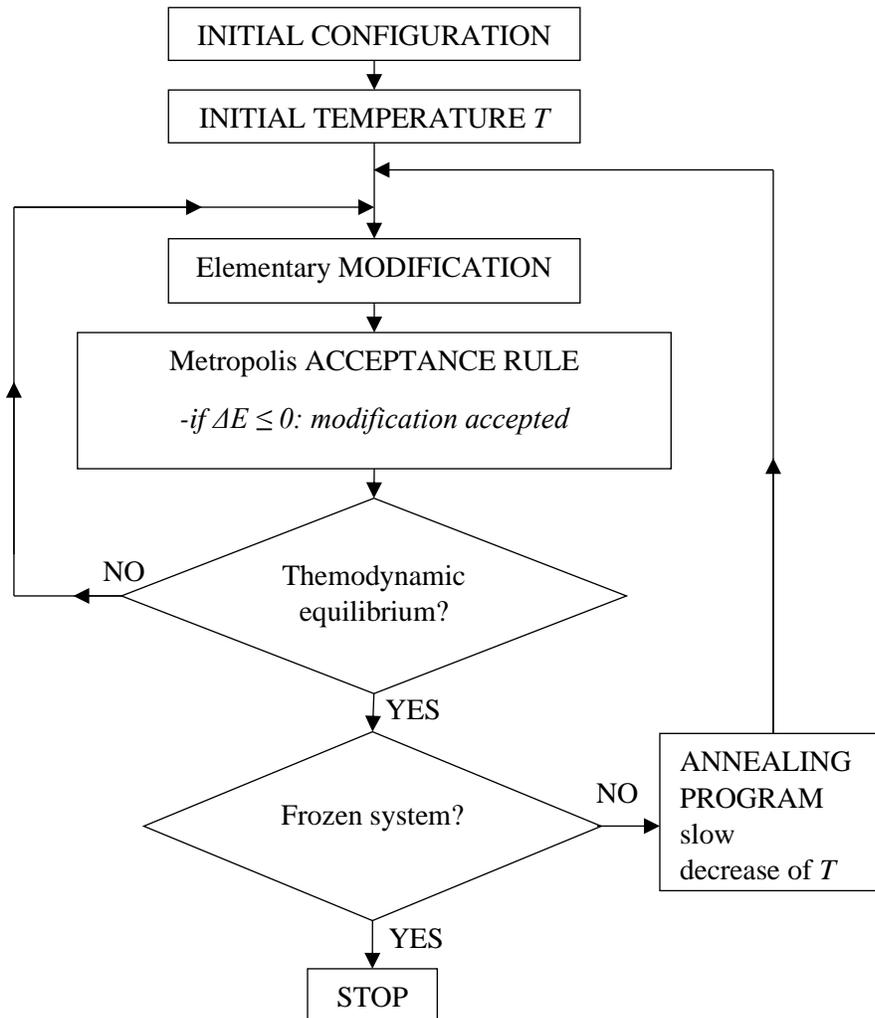


Fig. 2.2 Diagrama de flujo Recocido Simulado.

Los inconvenientes del recocido simulado residen en los "ajustes", como la gestión de la disminución de la temperatura; el usuario debe tener conocimientos técnicos sobre los "buenos" ajustes. Además, el tiempo de cálculo puede llegar a ser muy importante, lo que ha dado lugar a implementaciones paralelas del método. Por otra parte, el método del recocido simulado tiene la ventaja de ser flexible con respecto a la evolución del problema y fácil de

aplicar. Ha dado excelentes resultados en una serie de problemas, generalmente de gran tamaño. [Siarry, 2016].

El comportamiento del recocido simulado puede ser visualizado en el diagrama de flujo en la figura 2.2, de forma general muestra los procesos que lo integran y los criterios que son considerados para determinar la aceptación de un nuevo estado (solución) y los criterios que se deben considerar para determinar el término de la ejecución.

2.4 Problema de Ruteo de Vehículos Multi-Deposito.

Con anterioridad se han mencionado ciertas consideraciones a las que está condicionado el MDVRP y sus principales características, por lo que a continuación se presenta de manera más detallada la manera en la que se compone.

2.4.1 Notación del MDVRP

El problema del MDVRP está formado por un conjunto finito V que es el conjunto de los clientes y los depósitos, dividido en dos subconjuntos, N es el número de clientes, donde $V_C = \{1, 2, \dots, N\}$ compone el conjunto de los clientes, y M el número de depósitos en donde $V_D = \{N+1, N+2, \dots, N+M\}$ que corresponde al conjunto de los depósitos.

Dado que debe se debe generar un conjunto de rutas donde todos los clientes debes ser visitados implica un desplazamiento de los vehículos que los atiende, cada desplazamiento tiene un costo C_{ij} , donde i es el cliente de partida y j el cliente de destino, la suma de los costos que genera un vehículo corresponde al costo de una ruta, y la suma de los costos de todas las rutas es el costo global de la solución.

También existen características físicas que no pueden ser ignoradas, como lo son las capacidades de carga de los vehículos y un límite de distancia que puede recorrer un vehículo, cada cliente i a visitar tiene una demanda d_i , la suma de las demandas de los clientes que componen una ruta no puede ser mayores a la capacidad Q que tienen los vehículos. La máxima distancia T que puede recorrer un vehículo se mide con respecto al costo de la ruta.

Tabla 2.1 Notación del MDVRP.

SIMBOLO	DESCRIPCIÓN
V	Conjunto de los nodos.
A	Conjunto de los arcos.
N	Conjunto de los clientes.
M	Conjunto de los depósitos.
c_{ij}	Costo del arco i al arco j.
d_i	Demanda del nodo i.
K	Número de vehículos disponibles por cada deposito.
Q	Capacidad de los vehículos.
T	Longitud máxima de recorrido por ruta.

Es importante mencionar que el hecho de existir abundantes recursos de entrega (vehículos), no es una necesidad hacer uso de todos ellos, sino encontrar una solución compuesta por un conjunto de rutas que satisfaga las restricciones minimizando el costo global.

En La tabla 2.1 se describen los elementos que componen la descripción anterior.

2.4.2 Modelo Matemático para el MDVRP

El modelo matemático para el MDVRP que representa la función objetivo y las restricciones que se deben tener en consideración se muestra a continuación.

$$\text{Minimizar } \sum_{i=1}^{N+M} \sum_{j=1}^{N+M} \sum_{k=1}^K C_{ij} X_{ijk}, \quad (1)$$

Sujeto a:

$$\sum_{i=1}^{N+M} \sum_{k=1}^K x_{ijk} = 1 (j = 1, \dots, N); \quad (2)$$

$$\sum_{j=1}^{N+M} \sum_{k=1}^K x_{ijk} = 1 (i = 1, \dots, N); \quad (3)$$

$$\sum_{i=1}^{N+M} x_{ihk} - \sum_{j=1}^{N+M} x_{hjk} = 0 (k = 1, \dots, K; h = 1, \dots, N + M); \quad (4)$$

$$\sum_{i=1}^{N+M} \sum_{j=1}^{N+M} d_i x_{ijk} \leq Q (k=1, \dots, K); \quad (5)$$

$$\sum_{i=1}^{N+M} \sum_{j=1}^{N+M} (c_{ij} w_{ij}) \leq T (k=1, \dots, K); \quad (6)$$

$$\sum_{i=N+1}^{N+M} \sum_{j=1}^N x_{ijk} \leq 1 (k=1, \dots, K); \quad (7)$$

$$\sum_{j=N+1}^{N+M} \sum_{i=1}^N x_{ijk} \leq 1 (k=1, \dots, K); \quad (8)$$

$$x_{ijk} \in \{0,1\} \forall i, j, k; \quad (9)$$

La función objetivo (1) minimiza el costo total. Las restricciones (2) y (3) garantizan que cada cliente sea atendido por exactamente un vehículo. La conservación de flujo está garantizada a través de la restricción (4). Las restricciones de capacidad vehicular y duración de la ruta se encuentran en (5) y (6). Las restricciones (7) y (8) verifican la disponibilidad del vehículo. Finalmente, en (9) se define a x como variable binaria, indicando si el desplazamiento x_{ijk} es realizado en alguna ruta, donde i es el nodo de origen, j es el nodo de destino y k es el vehículo que realiza ese desplazamiento. Este modelo fue tomado de [de Oliveira, 2016].

2.4.3 Modelo de grafo no dirigido

El problema del MDVRP también puede ser representado en la forma de un grafo no dirigido de la siguiente manera:

- Sea “ $G = (V, A)$ ” un grafo no dirigido donde V es el conjunto de los nodos, y A es el conjunto de los arcos.
- Los nodos están particionados en dos subconjuntos: los clientes a ser atendidos, $V_C = \{1 \dots N\}$, y los múltiples depósitos $V_D = \{N+1, \dots, N+M\}$, con $V_C \cap V_D = \emptyset$.
- Hay un costo no negativo c_{ij} asociado con cada arco $(i, j) \in A$, la demanda de cada cliente es d_i , (no hay demanda en los nodos de depósito).
- También hay una flota de K vehículos idénticos, cada uno con capacidad Q . El tiempo máximo de duración de la ruta se establece en T .

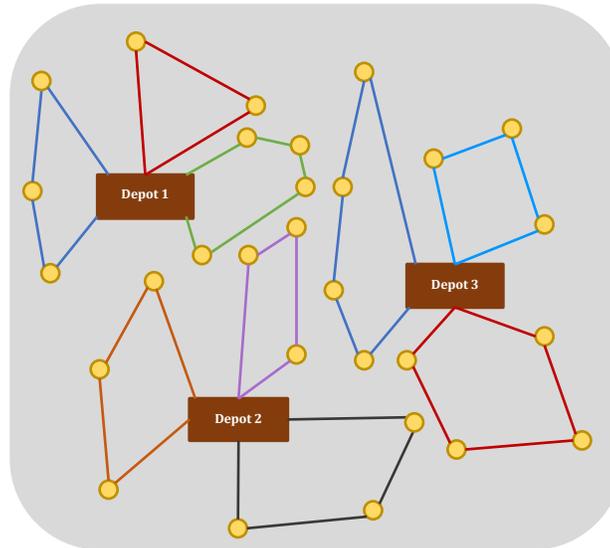


Fig. 2.3 Modelo de grafo para la instancia p01, solución óptima.

2.4.4 Benchmark

Un benchmark o problemas de referencia es un conjunto de problemas de optimización estándar, estos problemas son denominados “instancias”, y estas son valores particulares de entrada, sobre las cuales se puede ejecutar un algoritmo para resolver el problema, tienen la finalidad de ser utilizadas para la evaluación, caracterización y medición de rendimiento del algoritmo de optimización.

El benchmark para el problema del MDVRP consta de 23 instancias con características concretas, se muestran en la tabla 2.2 a

Tabla 2.2. Instancias de referencia para el MDVRP.

Instancia	n	m	K	D	Q
p01	50	4	4		80
p02	50	4	2		160
p03	75	5	3		140
p04	100	2	8		100
p05	100	2	5		200
p06	100	3	6		100
p07	100	4	4		100
p08	249	2	14	310	500
p09	249	3	12	310	500
p10	249	4	8	310	500
p11	249	5	6	310	500
p12	80	2	5		60
p13	80	2	5	200	60
p14	80	2	5	180	60
p15	160	4	5		60
p16	160	4	5	200	60
p17	160	4	5	180	60
p18	240	6	5		60
p19	240	6	5	200	60
p20	240	6	5	180	60
p21	360	9	5		60
p22	360	9	5	200	60
p23	360	9	5	180	60

Las instancias 1-7 han sido creadas por [Christofides, 1969], los casos 8-11 fueron descritas por [Gillett, 1976] y por ultimo los casos 12-23 fueron propuestas por [Chao, 1993].

En la tabla anterior muestra el número de clientes (“n”), número de depósitos (“m”), número de vehículos(“k”), duración o distancia máxima de recorrido (“D”) y por último la capacidad de carga de los vehículos.

También se puede observar en dicha tabla que no todas las instancias tienen un valor de distancia de recorrido máxima, esto es debido a que esta restricción no aplica para todas ellas, permitiendo la creación de rutas muchos más largas a comparación de las que si la tienen, esto repercute en que el espacio de soluciones a explorar es aún mayor respecto a aquellas que si la tienen.

CAPÍTULO III

METODOLOGÍA

3.1 Introducción

En los capítulos anteriores se abordaron temas como las características del MDVRP y técnicas que han sido utilizadas para solucionar el problema en cuestión obteniendo resultados diversos. En este capítulo se describe el método propuesto compuesto de dos fases basado en Recocido Simulado (SA). En la figura 3.1 describe la metodología, donde una heurística en la primera fase se encarga de generar una solución factible, considerando si la instancia a resolver cuenta con la restricción de distancia máxima de recorrido o no, se emplea una de las dos técnicas diseñadas para esta tarea, posteriormente esta solución es tomada por Recocido Simulado que finalmente devuelve la mejor solución encontrada.

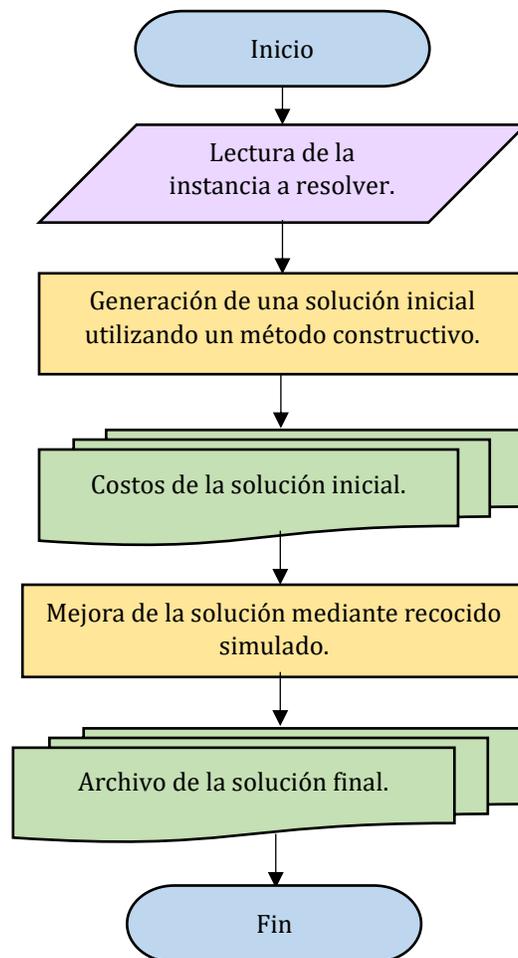


Fig. 3.1 Esquema de la metodología implementada.

3.2. Generación de la solución inicial para instancias sin restricción distancia máxima.

Dado que el algoritmo de recocido simulado trabaja sobre una solución completa, es necesario crearla en una fase previa.

Para ello se hace uso de una heurística constructiva, la cual se encarga de crear una solución que respete las restricciones, generando así una solución factible.

Para el caso en que la instancia no cuenta con la restricción de distancia de recorrido máximo la heurística constructiva difiere al caso en el que la restricción de recorrido si está presente.

El procedimiento que se lleva a cabo se puede visualizar en la figura 3.2 como diagrama de flujo.

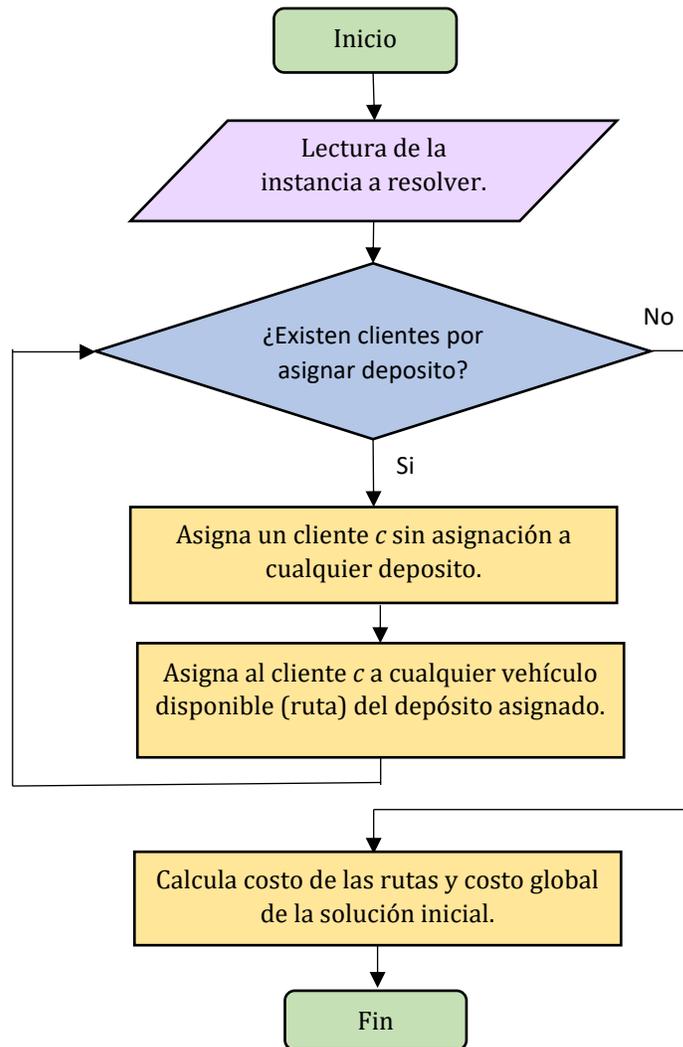


Fig. 3.2 Diagrama de solución inicial para instancias sin restricción de distancia máxima de recorrido.

Inicialmente se lleva a cabo la lectura del archivo contiene los datos de la instancia, se obtiene el número de clientes, número de depósitos, número de vehículos disponibles por cada depósito y finalmente la capacidad de los vehículos sucesivamente. Se continua con los datos de los clientes, como lo es la demanda y sus coordenadas.

La asignación de que depósito y con que vehículo se va a realizar la atención de los clientes es de manera aleatoria, es decir, se elige al azar uno de los depósitos disponibles, posterior a ello se elige de igual manera uno de los vehículos, siempre y cuando la capacidad disponible del vehículo lo permita, en caso contrario se repite la selección aleatoria del depósito y el vehículo hasta que sea posible asignar el cliente. Este proceso se realiza hasta completar la asignación de todos los clientes, finalmente se calcula el costo de todas las rutas, y se calcula el costo global, es decir, la suma del costo de todas las rutas.

Utilizando la instancia p01 ejemplificaremos dicha técnica.

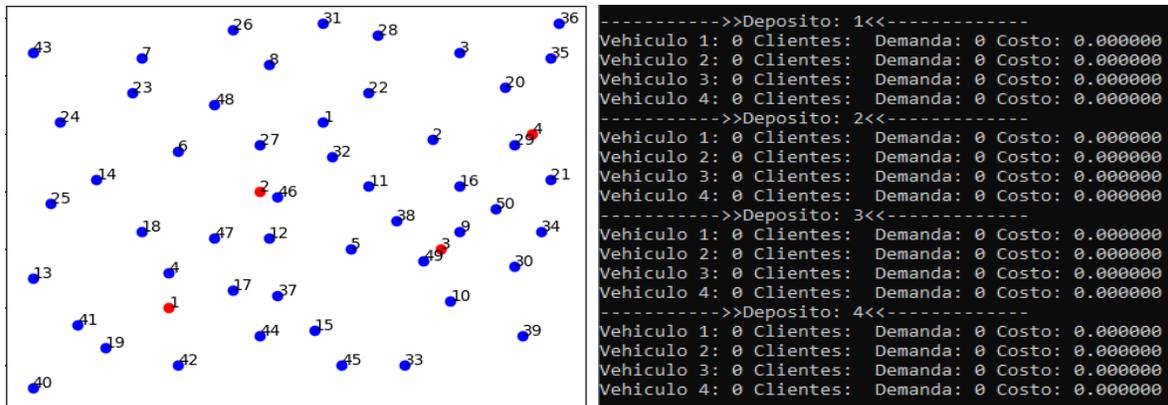


Fig 3.3 Izquierda: distribución de los clientes (puntos azules) y los depósitos (puntos rojos).

Derecha: Asignación de los vehículos que atienden a los clientes imagen derecha. (vehículos vacíos)

En la figura 3.3 se puede observar la distribución geográfica de los clientes y los depósitos que componen la instancia p01 sin haber hecho ninguna asignación podemos observar del lado derecho de la figura que los vehículos al no tener clientes tienen cero capacidades utilizadas, y costo de la atención de ruta de 0.0.

A continuación, realizamos la asignación del cliente 1, que de manera aleatoria se elige al depósito 2 en el vehículo 4, y se actualiza la demanda que acumula el vehículo, en este caso la demanda del cliente es de 7 unidades, al ser el único cliente asignado la demanda del vehículo se mantiene en 7, es primordial mantener actualizado dicho valor al realizar una asignación para conocer hasta qué punto ya no tenemos capacidad disponible asignar más clientes.

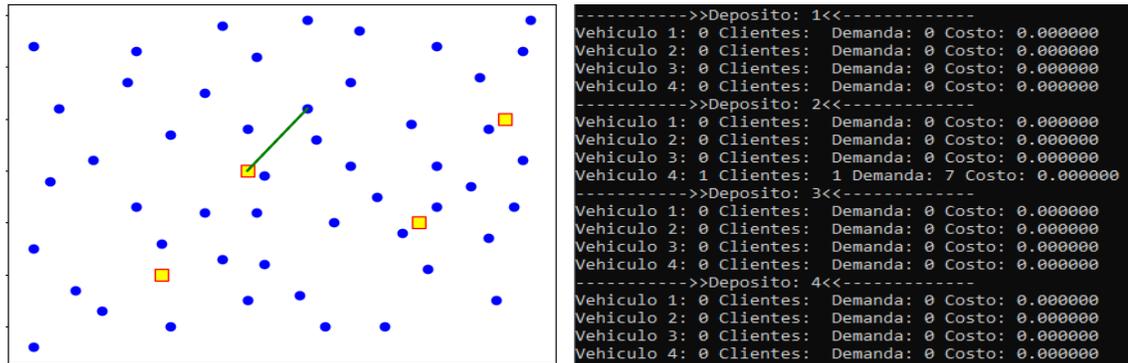


Fig 3.4, Asignación del primer cliente.

Las líneas que se muestran en el lado izquierdo de la figura 3.4 y futuras figuras, representan una ruta y los cuadros amarillos con rojo a los depósitos, donde los puntos azules son los clientes, en este caso al ser solo un cliente, se debe entender que se parte del depósito hacia el cliente y vuelve nuevamente al depósito de partida.

Se continua con la asignación del segundo cliente al depósito 3 en el vehículo 1 como se muestra en la figura 3.5

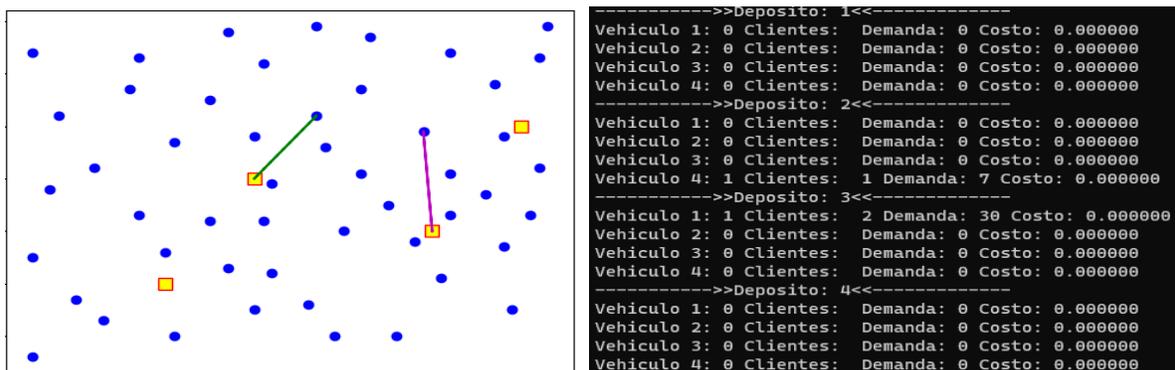


Fig 3.5, Asignación del segundo cliente.

A continuación, se asigna el tercer cliente con una demanda de 16 unidades al depósito 2 en el vehículo 1, mostrando la una actualización de la solución en la figura 3.6.

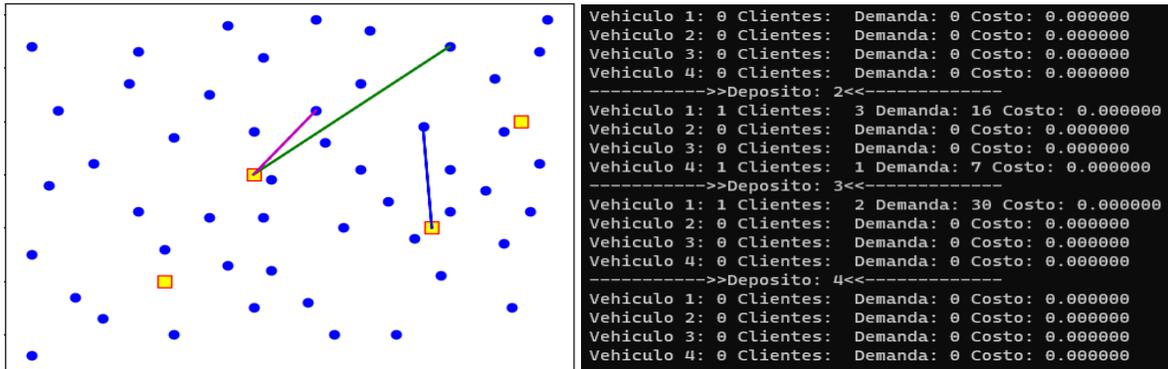


Fig 3.6, Asignación del tercer cliente.

Al llegar a la asignación del cuarto cliente se hace evidente que el criterio de elección del depósito no es el más cercano.

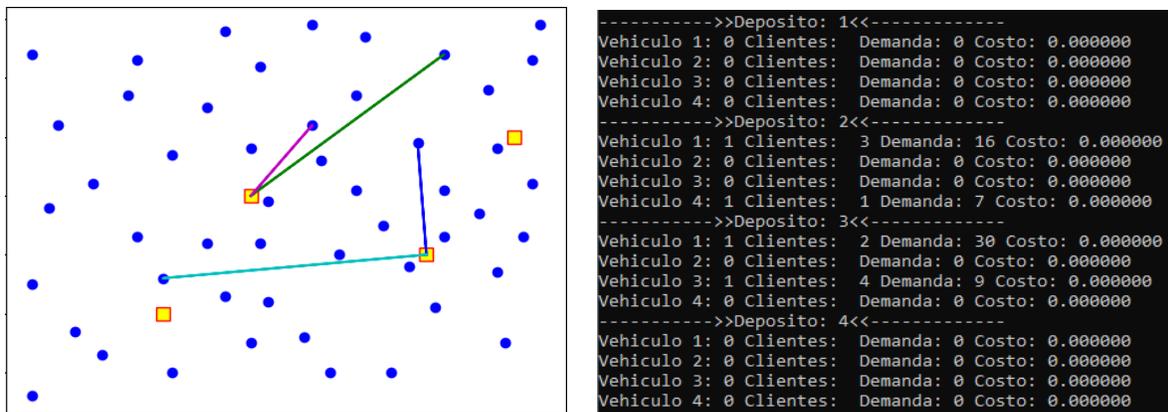


Fig 3.7, Asignación del cuarto cliente.

Al continuar con las asignaciones, en el momento en que hay más de un cliente asignado a un mismo vehículo se puede visualizar como se forman circuitos, que representan las rutas que debe recorrer el vehículo para atender a los clientes asignados, figura 3.8.

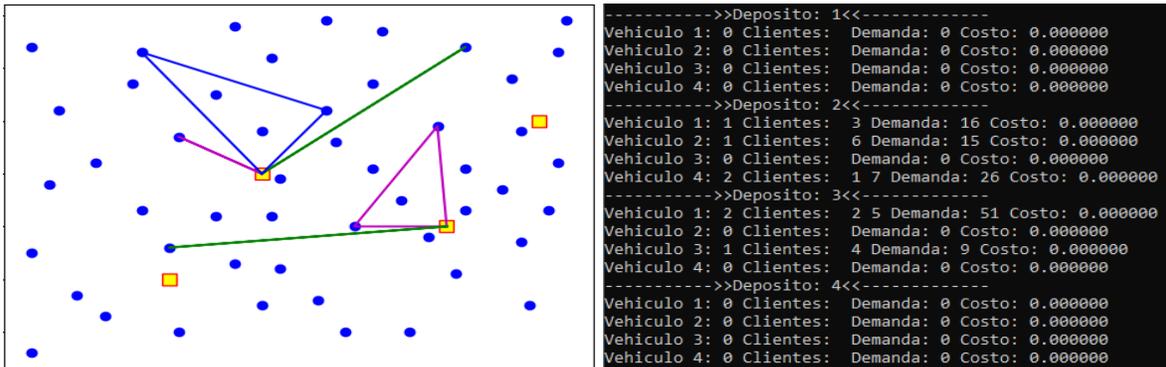


Fig 3.8, Creación de circuitos.

Al haber asignado los clientes en su totalidad, ya existiendo múltiples circuitos, se aprecia que las rutas creadas no emplean ningún tipo de técnica inteligente, permitiendo la existencia de cruces en el recorrido de las rutas, lo que significa que es posible reducir la distancia necesaria para visitar a todos los clientes que debe atender un vehículo, para finalizar la fase de creación de la solución inicial, se calcula el costo de las rutas de los vehículos de todos los depósitos, figura 3.9.

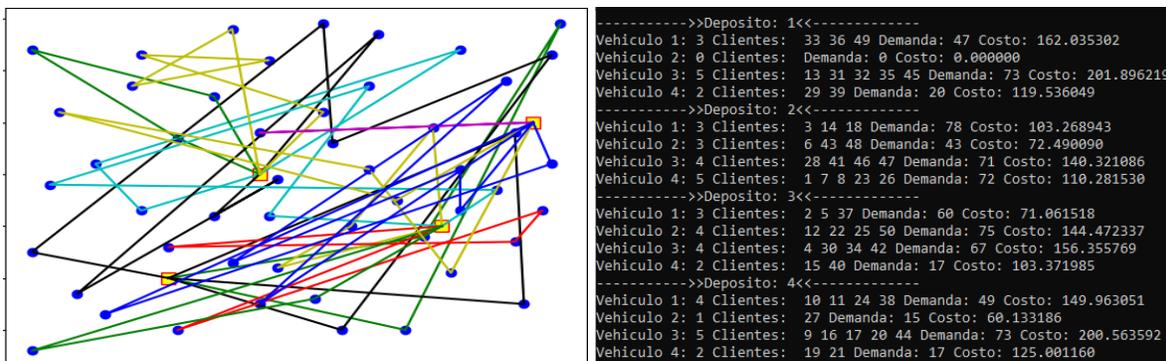


Fig 3.9, generación de la solución inicial terminada, todos los costos de las rutas calculados.

3.3 Generación de la solución inicial para instancias con restricción de distancia máxima de recorrido.

Para el caso de las instancias con restricción de recorrido máximo se hace uso de una técnica denominada “Max-Min”, ya que al hacer una asignación aleatoria resulta muy improbable que dé como resultado una solución inicial factible.

INICIALIZACIÓN MAX_MIN para el MDVRP

Para la generación de una solución inicial de las instancias que cuentan con la restricción de “distancia de recorrido máxima” se realiza una asignación tomando en cuenta el depósito más cercano al cliente, pero en estos casos se toma en consideración un criterio de asignación por priorización. Esta prioridad se lleva a cabo de acuerdo con el cliente que tenga la mayor distancia respecto al depósito más lejano a él, una vez determinado este cliente, se activa el uso de un vehículo del depósito más cercano y se asigna a él, se realiza el cálculo del costo la ruta, posteriormente se hace una búsqueda global de los clientes que pueden ser asignados a este vehículo, y uno a uno es asignado a esta ruta de acuerdo con el que tenga menos repercusión en el aumento del costo de recorrido del vehículo, cada que un cliente es asignado a un vehículo es actualizado el valor de la demanda de atención y el costo de desplazamiento del vehículo debe ser actualizado, este procedimiento se repite hasta que la capacidad libre ya no permita agregar más clientes y no debe aplicarse en caso de que una asignación cause un excedente en la distancia de recorrido máxima, aunque aún haya capacidad disponible en el vehículo con en el que se están realizando las asignaciones. El resto de los vehículos son activados repitiendo este proceso hasta que no haya más clientes que asignar. Al igual que con las instancias que no cuentan con la restricción de “distancia de recorrido máxima” es posible que no necesariamente todos los vehículos disponibles sean utilizados.

Este procedimiento puede observarse en el diagrama de flujo mostrado en la figura 3.11:

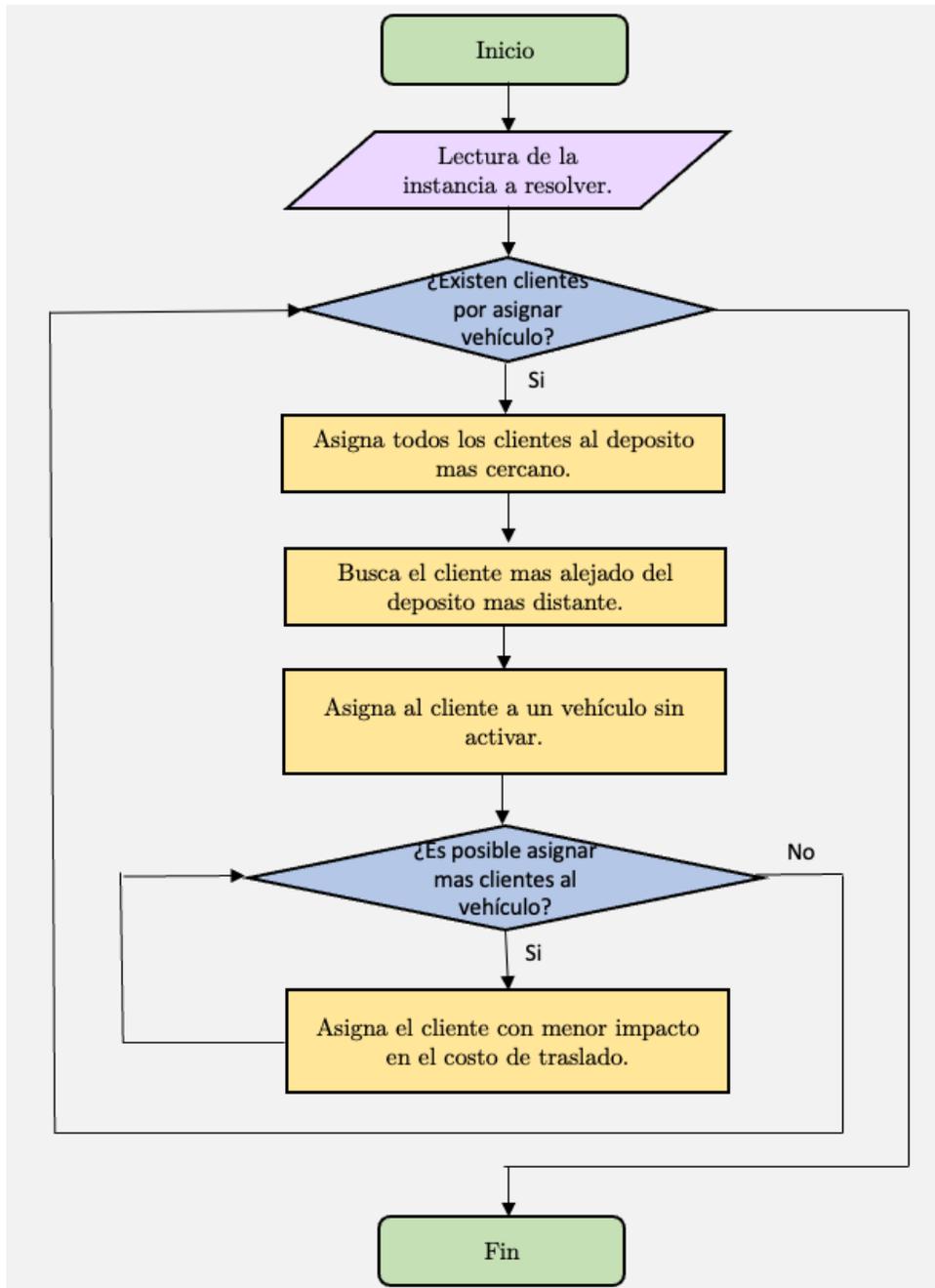


Fig. 3.11 Diagrama de solución inicial para instancias con restricción de distancia máxima de recorrido.

A continuación, se muestra un ejemplo de este procedimiento utilizando la instancia p08, cuyos vehículos cuentan con una capacidad 500 unidades, y una longitud máxima de recorrido (costo) de 310 unidades.

Inicialmente se cuenta con una solución vacía, donde se observan los dos depósitos y sus 249 clientes. Figura 3.12

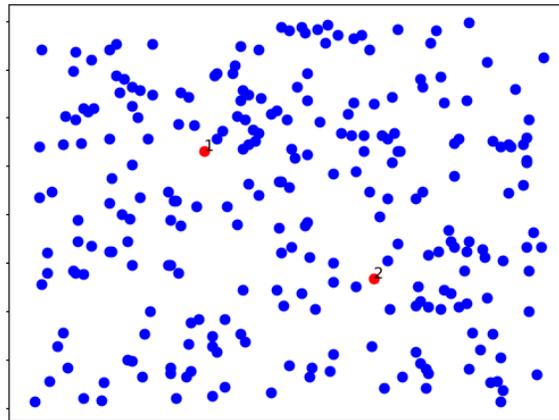


Fig. 3.12 Dispersión geográfica de la instancia p08.

Una vez hecha la lectura de las coordenadas de todos los clientes y los dos depósitos, se procede a realizar la búsqueda del cliente más lejano respecto a todos los depósitos existentes, una vez hecho esto se asigna al depósito con mayor cercanía a este, en este caso se trata del cliente 1. Por lo que se activa el vehículo 1 del depósito 1. Como se muestra en la figura 3.13.

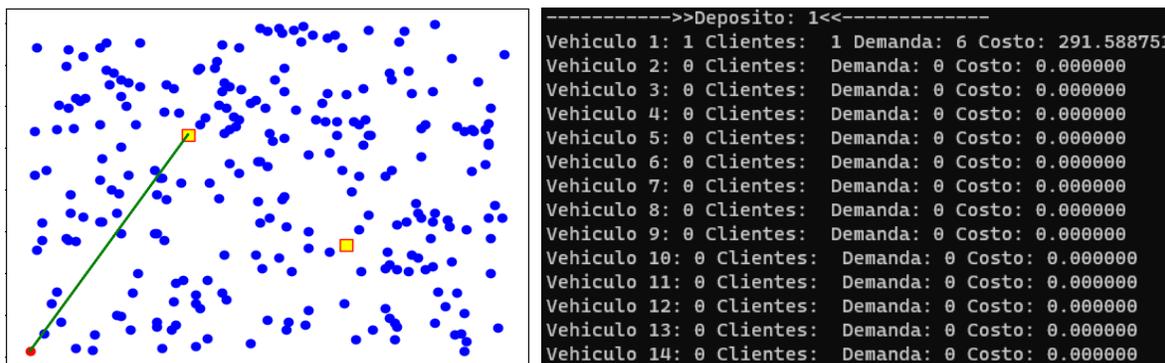


Fig. 3.13 Activación de un vehículo con el cliente 1(nodo rojo).

Una vez activado un vehículo se busca de manera global los clientes cuya demanda pueda ser atendida por el vehículo en cuestión, y se realiza un cálculo de cual cliente el mínimo incremento en el costo de la ruta del vehículo, y se realiza su asignación, en este caso el cliente 229 incrementa el costo de 291.588751 a 291.594742, debido a este mínimo incremento es la razón para que el cliente 229 sea añadido a la ruta. Como se muestra en la figura 3.14

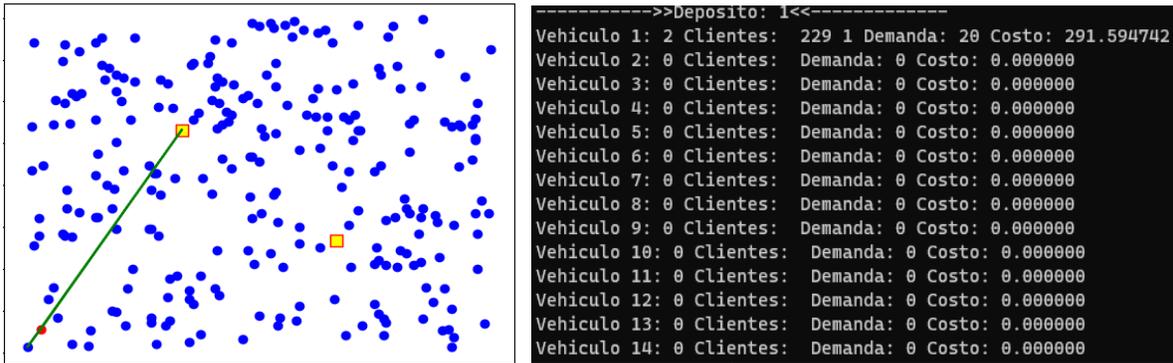


Fig. 3.13 Asignación del cliente 229 (nodo rojo).

Bajo el mismo criterio, el cliente 198 es asignado, generando un incremento en el costo de la ruta a 291.616563. Figura 3.14.

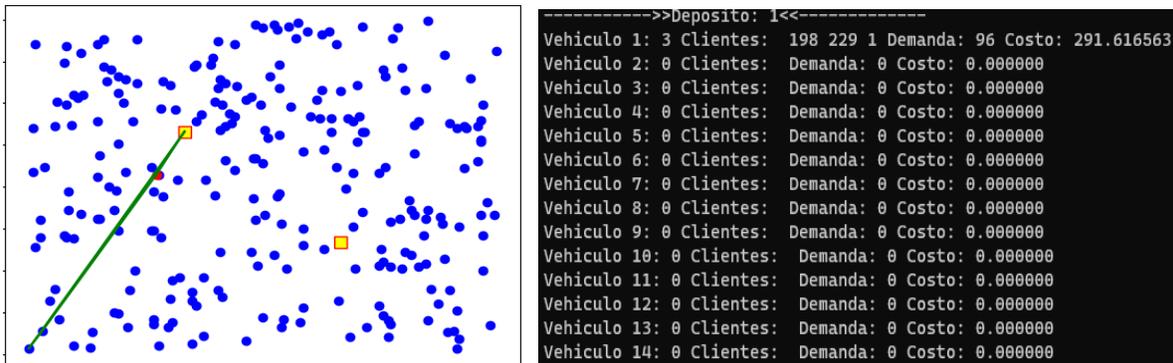


Fig. 3.14 Asignación del cliente 198 (nodo rojo).

Al asignar el cliente 79, se hace visible que la ruta empieza a tomar forma de un polígono, dejando de verse como una sola recta.

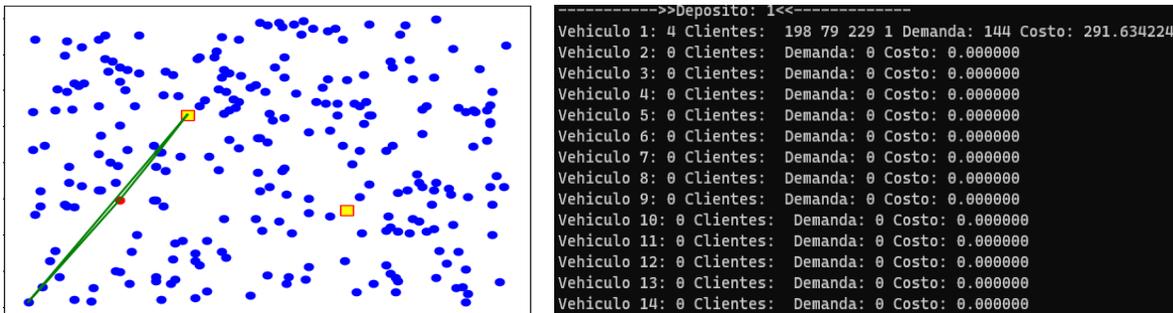


Fig. 3.15 Asignación del cliente 79 (nodo rojo).

Se siguen asignando clientes otros clientes. Y al asignar el doceavo cliente de la ruta (el cliente 166), es el último cliente que es posible asignar a la ruta debido a que la capacidad del vehículo de 500 unidades está completamente utilizada, donde los clientes a atender son 12 en total, con un costo final de la ruta de 301.611794.

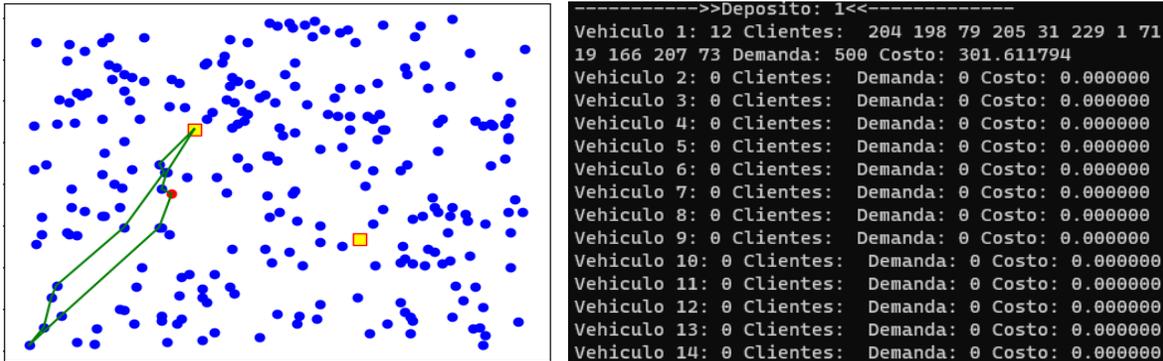


Fig. 3.16 Asignación del cliente 166 (nodo rojo).

Se procede a activar un nuevo vehículo bajo los mismos criterios antes mencionados, por lo que se activa el vehículo 1 del depósito 2 con el cliente 159.

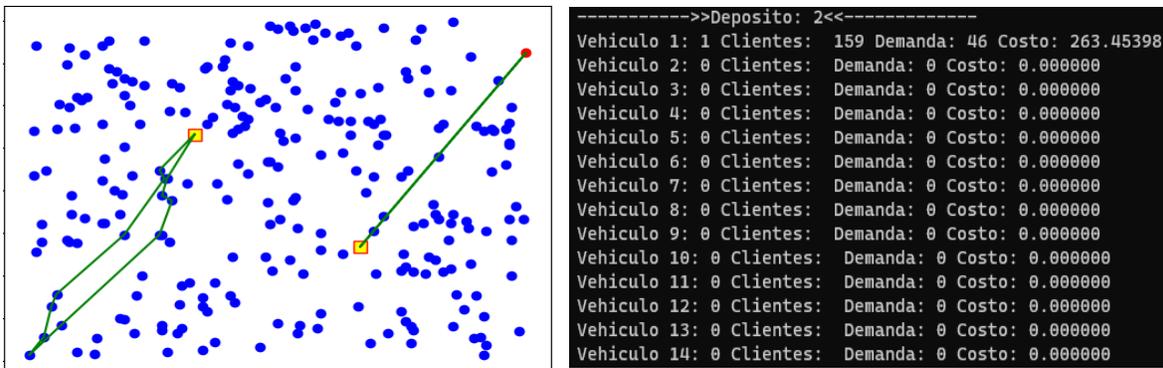


Fig. 3.16 activación de un nuevo vehículo con el cliente 159 (nodo rojo).

Se sigue la misma lógica de asignar clientes uno por uno, de tal forma que el incremento en la longitud de la ruta sea lo mínimo posible y que la capacidad del vehículo lo permita. Este vehículo atiende finalmente a 13 clientes, atendiendo una demanda de 490 y costo de ruta de 283.042769, el último cliente es ser asignado es el 165 y cuyo circuito de la ruta se visualiza en la siguiente figura.

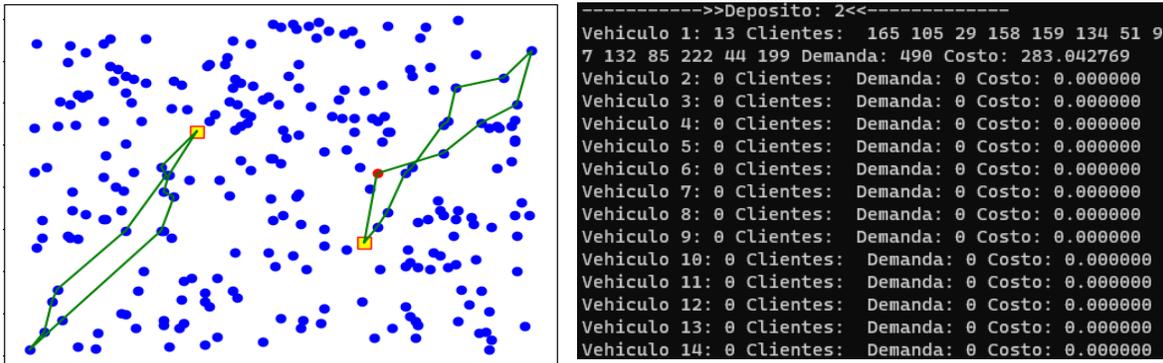


Fig. 3.17 Circuito final del de la segunda ruta generada, ultimo cliente asignado es el 165 (nodo rojo).

Finalmente, tras activar todos los vehículos necesarios para atender a todos los clientes, la solución inicial aplicando la técnica Max-Min nos devuelve el siguiente conjunto de rutas.

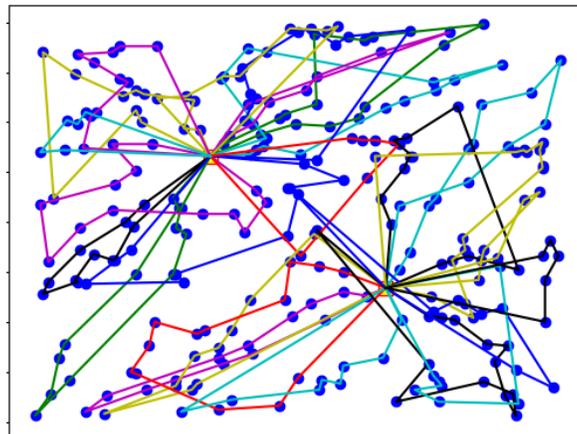


Fig. 3.18 Solución inicial termina tras la utilización de la técnica Max-Min.

Es evidente que la utilización de esta técnica arroja una solución de mejor calidad respecto al criterio de asignación aleatorio, y es requerido para las instancias con una cota en la distancia máxima por recorrido, donde la asignación aleatoria no es útil.

3.3 Vecindarios

Para la segunda fase de algoritmo, que corresponde al Recocido Simulado (SA) se implementaron 4 vecindades. Con estas 4 vecindades se concluyó que son suficientes para generar una buena diversidad de soluciones al momento de aplicar una perturbación a una solución.

La aplicación de estas vecindades está sujeta al cumplimiento de dos restricciones concretas, como los son la capacidad de carga los vehículos en la/las rutas afectadas y los nuevos costos de las rutas resultantes posterior a los movimientos.

Estas 4 vecindades se describen a continuación:

3.3.1 Vecindad 1: Swap Intra-ruta

Esta vecindad consiste en seleccionar dos clientes de una misma ruta e intercambiar sus posiciones en el orden en el que la ruta establece que deben ser visitados, de manea grafica puede visualizarse de la siguiente manera:

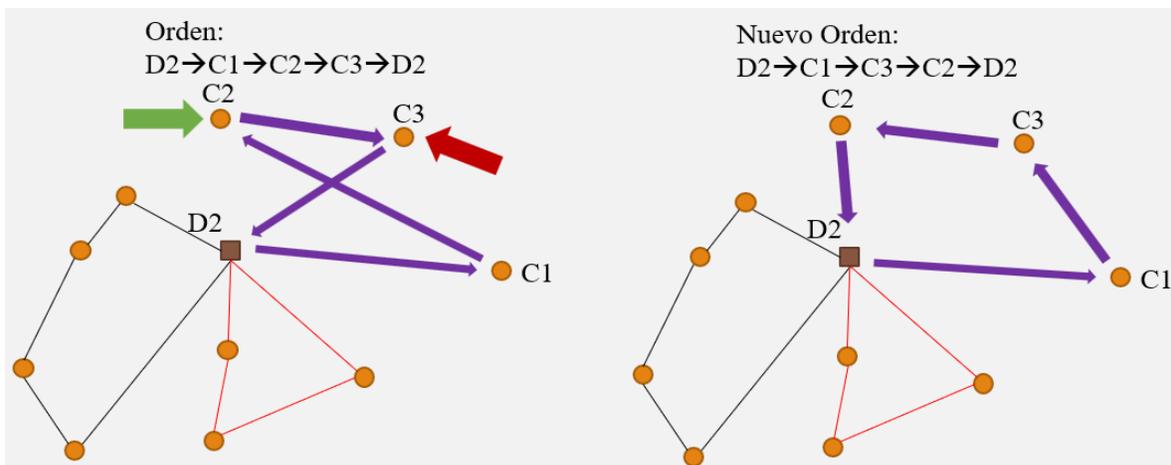


Fig. 3.19 Vecindad 1: Swap Intra-ruta.

Como se puede apreciar, del lado izquierdo de la figura 3.19, Existe un orden establecido en el que los clientes deben ser visitados, es visible que existe un cruce en esa ruta, por lo que aplicar la vecindad 1 genera un nuevo orden de visita, intercambiando los clientes 2 (C2) y el cliente 3 (C3), dando como resultado la eliminación de este cruce.

3.3.2 Vecindad 2: Swap Inter-ruta

Esta vecindad al igual que la vecindad 1 intercambia dos clientes, pero con la diferencia que este intercambio debe efectuarse entre dos clientes que pertenezcan a dos rutas diferentes, pudiendo ser del mismo depósito o uno diferente.

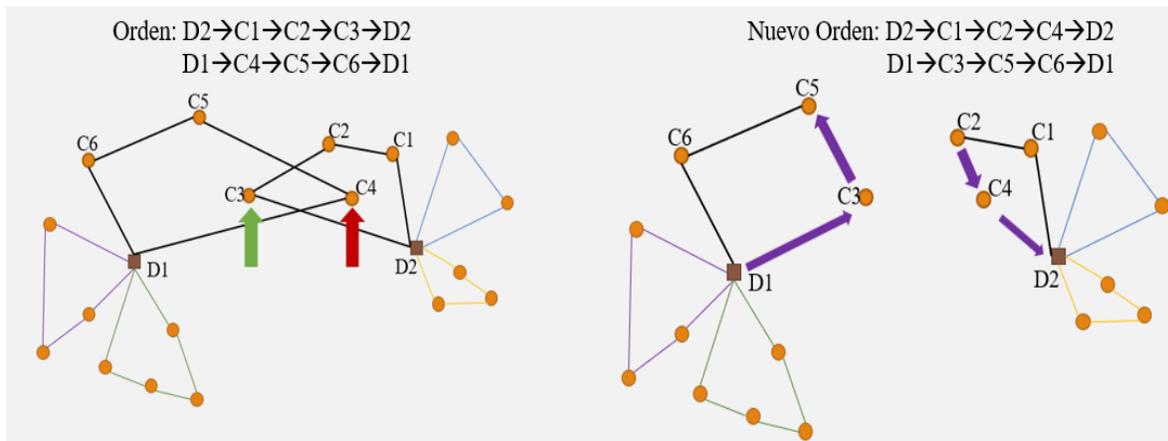


Fig. 3.20 Vecindad 2: Swap Inter-ruta.

La vecindad 2 puede ser útil en los casos en que haya cruces entre rutas aplicando un solo movimiento, en la figura 3.20 se muestra como es esto posible, en las que los clientes 3 (C3) y 4 (C4) se encuentran asignados para su atención a depósitos que a simple vista puede no ser el más conveniente por cuestión de distancia, por lo que la aplicación de esta vecindad puede realizar una mejora significativa en cuanto a distancia de recorrido.

3.3.3 Vecindad 3: Reinserción Inter-ruta

Esta vecindad se encarga de tomar un cliente de una ruta y reinsertarlo en cualquier otra, pudiendo ser en cualquier posición, ya sea en otra ruta de este depósito o de cualquier otro.

Como se muestra en la figura 3.21, esta vecindad puede ser de utiliza en el caso en el que un cliente esté cerca de una ruta distinta a la que pertenece, por lo que realizar una reasignación puede resultar conveniente siempre y cuando las restricciones de capacidad y longitud máxima seas respetadas en la ruta en que se reasignaría.

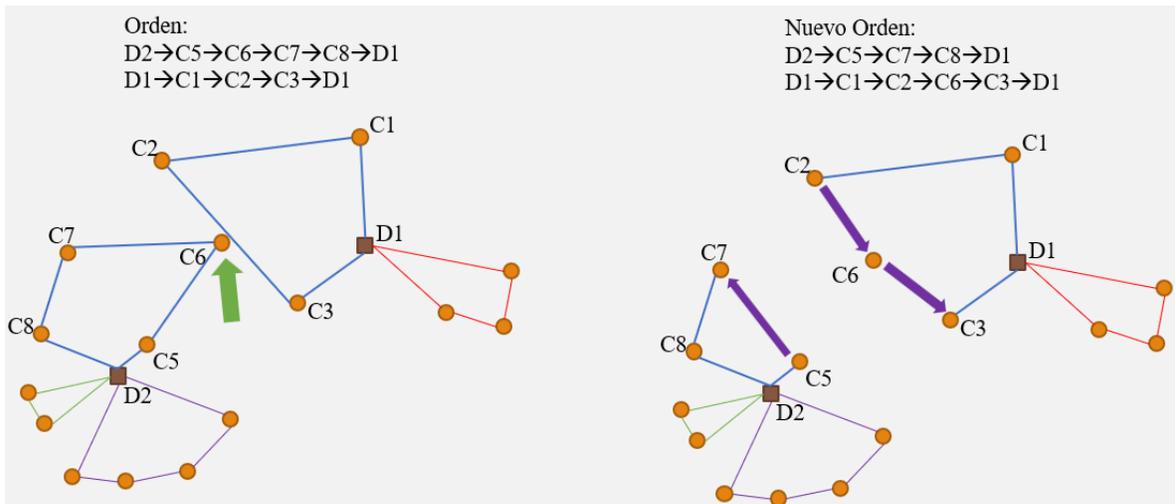


Fig. 3.21 Vecindad 3: Reinserción Inter-ruta.

3.3.4 Vecindad 4: Reinserción Intra-ruta

Esta vecindad realiza la reasignación de un cliente en una posición diferente de la misma ruta a la que pertenece, dado que no supone rehacer un cálculo de la capacidad de carga del vehículo ya que al hacer este movimiento no sufre un cambio, si debe realizarse un cálculo de la distancia resultante tras la reasignación.

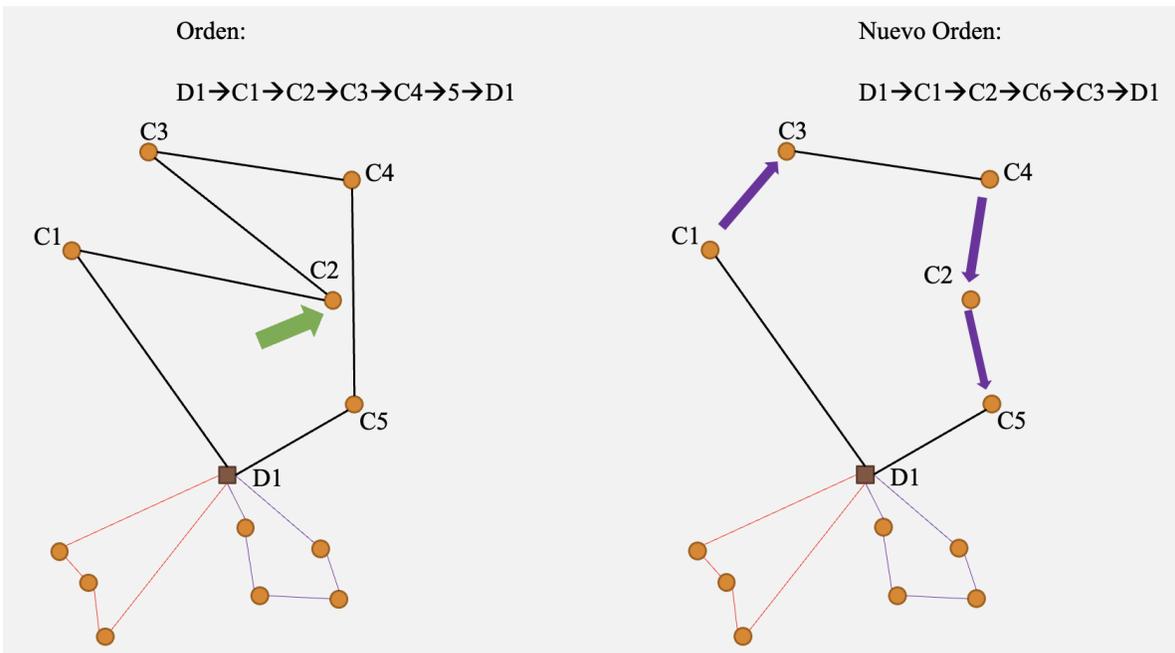


Fig. 3.22 Vecindad 4: Reinserción Intra-ruta.

En la figura 3.22 se muestra un caso de ejemplo en el que esta vecindad puede dar resultados favorables, en el que el cliente 2 (C2) es reasignado en otro orden de atención en la misma ruta dando como resultado una ruta con un recorrido optimizado.

3.4 Algoritmo Recocido Simulado

```

1 INPUT (T0, a, L, Tf)
2 T ← T0
3 Sact ← Genera_Solucion_INICIAL();
4 SBest ← Sact
5 WHILE T ≥ Tf DO
6   BEGIN
7     FOR cont ← 1 TO L(T) DO
8       BEGIN
9         tipo = rand() % NUM_VECINDADES+1 //Elige un tipo de vecindad
10        switch(tipo)
11          case 1:
12            Scand ← Vecinda01(Sact);
13            Break;
14          case 2:
15            Scand ← Vecinda02(Sact);
16            Break;
17          case 3:
18            Scand ← Vecinda03(Sact);
19            Break;
20          case 4:
21            Scand ← Vecinda04(Sact);
22            Break;
23        δ ← coste(Scand)-coste(Sact)
24        IF (U (0,1) < e(-δ/T)) OR (δ < 0) THEN Sact ← Scand
25        IF Sact < SBest THEN SBest ← Sact
26      END
27    T ← a(T)
28  END
29 RETURN SBest

```

Fig. 3.23 Seudocódigo Recocido Simulado.

A continuación, se explica el pseudocódigo que describe el procedimiento del algoritmo de recocido simulado mostrado en la figura 3.23.

1. Se definen los parámetros que utiliza el algoritmo, como lo son: temperatura inicial (T_0), alfa o factor de decremento (α), número de iteraciones (soluciones a explorar) previas a realizar un decremento a la temperatura y finalmente la temperatura final (T_f), que actúa como criterio de parada, que detiene completamente la ejecución del algoritmo.
2. Se asigna el valor de la temperatura inicial previamente definida a la variable que actúa como temperatura actual (T).
3. La solución creada en la fase 1 que utiliza una heurística constructiva es tomada como solución inicial y se asigna como solución actual.
4. La solución actual se guarda como la “mejor solución”.
5. Se inicia uno de los dos ciclos de repeticiones que conforman el algoritmo, en este caso se trata del ciclo externo, el cual se encarga de detener el algoritmo cuando la temperatura actual sea menor a la temperatura final, cuyo valor ha sido definido previamente a la iniciación del algoritmo.
7. Inicia el segundo ciclo (ciclo interno) en el cual se lleva a cabo un conteo de repeticiones el cual se encarga de explorar L soluciones previo a hacer un decremento en la temperatura.
9. De manera aleatoria se genera un número del uno al cuatro, por consecuencia el número resultante representa el número de vecindad que se utilizará de las cuatro vecindades mencionadas en el subtema 3.3, generando una nueva solución vecina denominada solución candidata.
10. Se inicia un switch, en el cual, dependiendo del número aleatorio generado, se selecciona el número de vecindad a utilizar, se aplica dicha vecindad a la solución actual y se genera una nueva solución candidata.
23. Se calcula la diferencia de costo entre la solución candidata y la solución actual.
24. Se genera un número aleatorio entre 0 y 1, si el valor generado es menor al exponencial de la diferencia de costos la solución candidata es tomada como solución actual, en caso de

no ser aceptada la solución candidata, se valida si el costo de la solución candidata es menor a la solución actual, es decir si la diferencia de costos es menor a cero, la solución candidata es aceptada.

25. En caso de el costo de la solución actual sea menor a la “mejor solución”, la solución actual es guardada como mejor solución.

26. Termina el ciclo interno.

27. Se realiza un decremento de la temperatura, multiplicando el valor de la temperatura actual por el valor de alfa (α).

28. Fin del ciclo externo tras cumplirse el criterio de paro.

29. Se escribe el archivo de salida con la información que compone la mejor solución encontrada.

3.5 Otras Vecindades

Durante el periodo de experimentación se realizaron pruebas a la implementación de dos vecindades extras a las que fueron utilizadas en el algoritmo final, la primera de ellas denominada “2-opt” la cual se encargaba de realizar intercambio de dos nodos adyacentes en una misma ruta con la finalidad de eliminar cruces en un circuito, y la segunda denominada “doble vecindad”, la cual toma solución actual e inicialmente genera una solución vecina, una vez hecho esto se genera por segunda ocasión una nueva vecindad, el resultado tras aplicar por segunda ocasión la vecindad es denominada solución candidata, y esta será evaluada para determinar su aceptación.

Dados los resultados obtenidos de las pruebas tras la implementación estas dos vecindades, se concluyó que no se generaba una mejora en la calidad de las soluciones finales con respecto a los resultados obtenidos donde solo se hacía uso de las 4 vecindades que fueron utilizadas de manera final.

3.6 Generación de instancias aleatorias.

Con el fin de evaluar el rendimiento del algoritmo metaheurístico propuesto se generaron instancias aleatorias de mayor tamaño que la literatura en lo que respecta al número de

clientes. Los clientes poseen una demanda más diversa con respecto a los problemas de la literatura, pero sin restricción de distancia de recorrido máxima. En la literatura, las instancias p22 y p23 son las de mayor tamaño, manejando 360 clientes. Las instancias generadas en este trabajo cuentan con hasta 800 clientes. Dado el número de depósitos y clientes, se genera de manera aleatoria las coordenadas de los clientes, sus demandas, y las coordenadas de los depósitos.

Para generar dichas instancias aleatorias se generó un programa que crea archivos con el mismo formato que manejan los problemas de referencia, cuyos datos de entrada son: número de clientes, número de depósitos, coordenadas máximas y mínimas para los valores positivos y negativos en los ejes X y Y, así como la capacidad con la que deben contar los vehículos.

El número de instancias generadas fue 6, las cuales se denominaron p30, p31, p32, p33, p34 y p35. Cuyas características de los problemas propuestos se visualizan en la tabla 3.1. Como se mencionó anteriormente, las características de las instancias se presentan con los siguientes valores: n=número de clientes, m=número de depósitos, k=número de vehículos por depósito y Q representa la capacidad de cada vehículo.

Tabla 3.1 Características de las instancias aleatorias.

Instance	n	m	k	Q
p30	500	16	7	500
p31	500	16	7	500
p32	600	18	7	500
p33	700	19	7	500
p34	600	17	5	500
p35	800	19	6	500

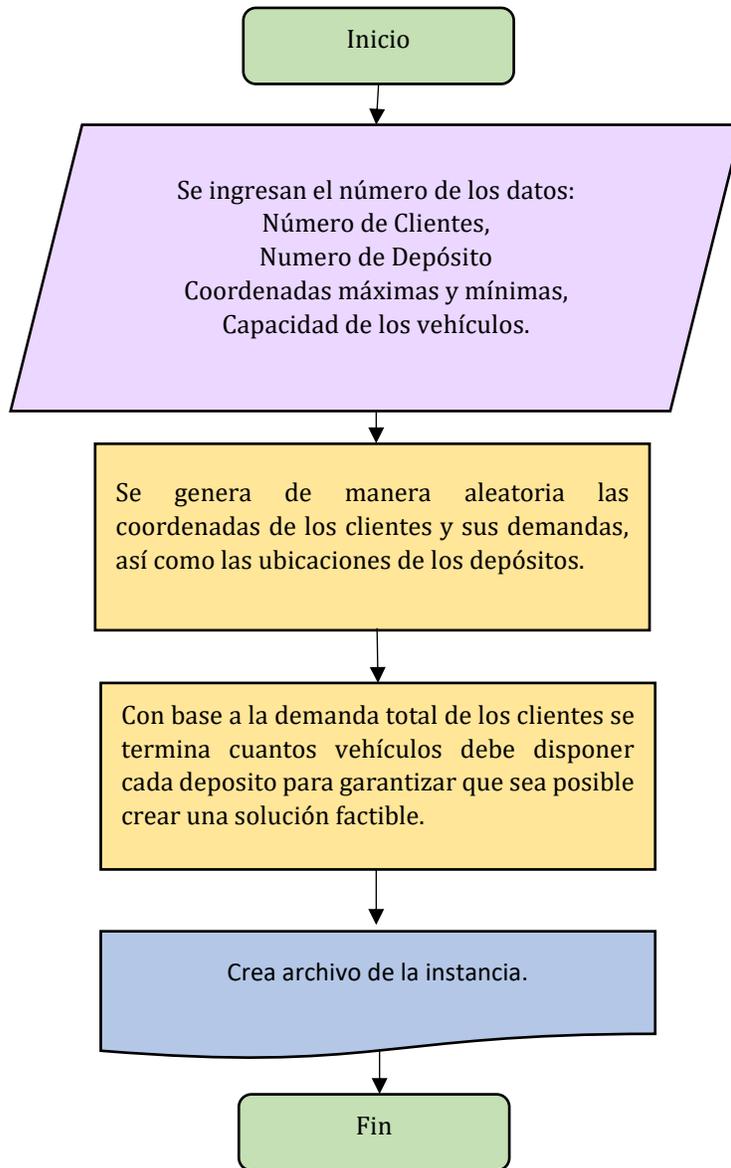


Fig. 3.24 Generación de una instancia aleatoria.

CAPÍTULO IV

SINTONIZACIÓN Y RESULTADOS

En este capítulo se aborda la sintonización de los parámetros de control de Recocido Simulado y los resultados finales obtenidos tras la experimentación en este proyecto de tesis.

4.1 Sintonización

Para la sintonización de los parámetros de Recocido Simulado se llevó a cabo múltiples experimentaciones con diferentes valores, con los cuales se realizaron ajustes en los valores de temperatura inicial, temperatura final, longitud del ciclo interno y finalmente, alfa o factor de decremento con el fin de mejorar el rendimiento del algoritmo.

4.1.1 Sintonización Temperatura inicial.

Para determinar el ajuste de la temperatura inicial (T_0) se debe de tomar en cuenta que se busca que la temperatura sea lo más baja posible, pero al mismo tiempo se requiere que la probabilidad de aceptación sea bastante alta, de manera que al inicio del algoritmo casi todas las soluciones generadas sean aceptadas sin importar la calidad, para que el algoritmo tenga una amplia capacidad de exploración al principio. Este parámetro se toma del costo de la solución inicial que es generado en la primera fase de nuestro algoritmo, partiendo de ahí se hicieron pruebas con los 4 problemas de referencia más difíciles de solucionar: p09, p15, p18 y p21, añadiendo 2 instancias ficticias: p32 y p33, creadas con el propósito de evaluar la capacidad del algoritmo con el manejo de un mayor número de nodos con respecto a los que se manejan en las instancias de referencia.

Como se puede apreciar en la tabla 4.1, se realizaron divisiones al costo de la solución inicial, se determinó que realizar la división entre dos daba un buen porcentaje de aceptación, por lo que el valor de la temperatura inicial se fijó en $C/2$, donde C es costo de la solución inicial.

Tabla 4.1 Sintonización de la temperatura inicial.

PORCENTAJE DE ACEPTACION					
SINTONIZACIÓN TEMPERATURA INICIAL					
Instancia	C	C/2	C/4	C/8	C/16
p09	99.73 %	99.60 %	99.10 %	98.56 %	96.96 %
p15	99.30 %	98.86 %	97.63 %	95.86 %	90.23 %
p18	99.60 %	99.00 %	98.00 %	95.96 %	92.93 %
p21	99.70 %	99.30 %	98.83 %	97.16 %	93.50 %
p32	99.50 %	99.46 %	99.30 %	98.36 %	95.90 %
p33	99.76 %	99.80 %	99.20 %	98.43 %	96.80 %

4.3 Sintonización Temperatura final

Tabla 4.2 Sintonización de la temperatura final.

PORCENTAJE DE ACEPTACION					
TEMPERATURA FINAL					
Instancia	1	0.1	0.01	0.001	0.0001
p09	1.36 %	0.1 %	0 %	0 %	0 %
p15	3.46 %	1.3 %	1.53 %	2.3 %	1.16 %
p18	4.93 %	4.66 %	4.06 %	3.63 %	2.06 %
p21	2.26 %	2.16 %	2.23 %	1.93 %	2.243 %
p32	1.97 %	1.86 %	1.53 %	0.7 %	1.06 %
p33	0.96 %	0 %	0.03 %	0.06 %	0.13 %

La temperatura final (T_f) se determinó tomando en cuenta el criterio en que las soluciones aceptadas que empeoren la solución sea baja, ya que en el punto final del algoritmo generalmente es poco probable que se sigan encontrando soluciones mejores a la actual, y las soluciones que empeoran la calidad de la solución actual deben ser rechazadas, en su mayoría. Por lo que se determinó que fijar la temperatura final (T_f) con un valor de 0.001, esto se puede ver representado en la tabla 4.2.

4.4 Sintonización Longitud ciclo interno.

Para determinar la longitud de ciclo interno (L) se realizó un ajuste en que el número de iteraciones, en el que con cada experimentación el valor de L se duplicaba, empleando un análisis del costo de la solución final y el tiempo que toma con respecto a la calidad de anterior, se pudo concluir que el valor sería de 8000, por lo que el tiempo necesario para las ejecuciones aún es permisible, la comparativa de estos es mostrada en la tabla 4.3.

Tabla 4.3 Sintonización de la longitud del ciclo interno.

SINTONIZACIÓN NÚMERO DE ITERACIONES												
L	500		1000		2000		4000		8000		16000	
INST.	C	T	C	T	C	T	C	T	C	T	C	T
p09	4292.28	14.54	4206.64	28.48	4126.80	55.23	4053.73	111.52	4043.56	198.29	3990.84	394.38
p15	3264.16	1.46	3122.39	2.84	3002.39	5.77	2953.05	11.34	2849.75	19.60	2789.85	39.46
p18	5475.01	1.62	5020.70	3.17	4695.33	6.24	4534.82	12.28	4362.71	21.28	4294.67	42.57
p21	9553.68	1.90	8507.41	3.66	7650.12	7.04	7204.32	13.68	6847.26	23.79	6563.59	46.99
p32	27844.60	2.73	23241.39	5.26	19856.56	10.03	17867.69	19.15	16359.95	32.97	15381.98	64.66
p33	38547.55	2.95	32194.55	5.79	27349.85	10.97	24046.75	20.66	21685.25	35.74	20207.06	69.21

4.5 Sintonización Alfa

Finalmente, se sintonizó el último parámetro pendiente por ajustar que es Alfa (a), este parámetro determina la velocidad de enfriamiento, es decir la disminución de la temperatura. Mientras el valor de alfa sea más pequeño la velocidad de enfriamiento se reduce considerablemente, generando así un aumento en el tiempo de ejecución. Nuevamente se realizó un análisis considerando la calidad de la solución final y el tiempo requerido, variando el valor de alfa en solo de centésimas y milésimas, generando los datos obtenidos que se muestran en la tabla 4.4, se pudo determinar que el valor de alfa se fijaría en 0.999.

Tabla 4.4 Sintonización de alfa.

FACTOR DE DECREMENTO																		
F.	0.80		0.85		0.90		0.95		0.99		0.995		0.999		0.9995		0.9999	
Inst.	C	T	C	T	C	T	C	T	C	T	C	T	C	T	C	T	C	T
p09	4249.47	14.44	4227.03	17.67	4181.80	25.02	4126.06	44.14	4058.05	210.10	4037.38	405.18	4010.04	2105.56	4002.33	4251.59	3990.80	22128.51
p15	3169.13	1.50	3102.44	1.84	3048.83	2.62	2988.99	4.72	2866.94	20.30	2796.90	40.49	2728.03	200.45	2720.51	433.80	2704.78	2040.54
p18	5292.01	1.64	5053.36	2.28	4911.99	3.17	4699.00	5.56	4346.27	22.05	4265.79	43.46	4136.36	213.85	4074.88	467.23	4024.50	2291.76
p21	9236.51	2.12	8858.05	2.65	8440.26	3.76	7605.41	6.70	6850.51	24.39	6646.14	47.57	6312.03	233.28	6207.02	486.22	6110.26	2530.20
p32	26807.30	3.27	24764.14	3.94	22791.70	5.12	19992.47	8.91	16745.39	32.08	15925.82	59.86	15370.16	289.38	15242.34	599.95	14976.19	3079.35
p33	38681.76	3.66	35557.40	4.47	32296.72	5.59	27792.15	8.99	21911.58	36.24	21010.88	64.39	19787.14	306.20	19626.58	620.99	19296.10	3146.95

4.2 Resultados

La implementación de la metodología propuesta que está compuesta de dos fases, la primera emplea una heurística constructiva y la segunda que implementa la metaheurística Recocido Simulado, con la sintonización de sus parámetros y utilizando 4 vecindades, se logró encontrar la solución mejor conocida (*BKS*) en la actualidad, en más del 60% de los problemas de referencia.

Estos resultados se muestran en la tabla 4.5, también muestra una comparativa de los resultados obtenidos previos a la sintonización y posteriores a ella, además del margen de error con respecto al *BKS*, resaltando en color verde los valores iguales a la solución mejor conocida, y en color rojo la soluciones que superan el 7% de margen de error.

Es importante mencionar que como consecuencia de la sintonización los resultados obtenidos mejoraron, aumento el número de soluciones *BKS* encontradas, y en ningún caso la mejor solución encontrada supera el 7% de margen de error, cumpliendo así la hipótesis definida en la sección 1.3.

Tabla 4.5 Resultados obtenidos en el benchmark del MDVRP

Instancia	R. Simulado Sin Sintonizar	% Error	R. Simulado Sintonizado	% Error	Tiempo (seg)	BKS
p01	576.86	0	576.86	0.00	106.66	576.86
p02	473.53	0.00	473.53	0.00	161.20	473.53
p03	645.14	0.62	641.18	0.00	134.01	641.18
p04	1015.67	1.46	1001.04	0.00	142.14	1001.04
p05	755.7	0.76	750.03	0.00	211.97	750.03
p06	885.48	1.02	876.5	0.00	135.52	876.5
p07	894.88	1.46	881.97	0.00	136.88	881.97
p08	4550.2	4.08	4429.15	1.32	3753.23	4371.66
p09	4007.82	9.23	3896.31	6.19	2562.83	3669.31
p10	3813.43	5.06	3669.09	1.09	2738.97	3629.6
p11	3683.67	3.91	3561.47	0.46	2983.10	3545.18
p12	1353.14	2.59	1318.95	0.00	166.42	1318.95
p13	1318.95	0.00	1318.95	0.00	3905.74	1318.95
p14	1360.12	0.00	1360.12	0.00	8320.47	1360.12
p15	2766.83	10.44	2552.89	1.90	184.25	2505.29
p16	2572.23	0.00	2572.23	0.00	5297.14	2572.23
p17	2709.09	0.00	2709.09	0.00	11128.84	2708.99
p18	4276.1	15.49	3881.33	4.83	198.98	3702.55
p19	3827.06	0.00	3827.06	0.00	7323.65	3827.06
p20	4097.05	0.96	4058.07	0.00	13865.97	4058.07
p21	6870.95	25.50	5805.65	6.04	216.40	5474.74
p22	5765.29	1.11	5714.45	0.22	11011.11	5702.06
p23	6145.58	1.10	6078.75	0.00	18738.24	6078.75

EL algoritmo metaheurístico de dos fases diseñado fue utilizado para solucionar un total de 23 instancias que conforman el benchmark del MDVRP, en 14 de ellas (60.86%) se logró encontrar la mejor solución publicada en la literatura hasta el momento, el margen de error más alto que se obtuvo en las soluciones con respecto al BKS fue de 6.19% con la instancia p09, para la instancia p17 no fue posible encontrar la solución óptima a pesar de que el margen de error es aproximadamente del 0.003691, para siete de las nueve instancias no se logró encontrar la mejor solución conocida, pero la desviación de las soluciones con respecto

a la mejor conocida están por debajo del 5% de margen de error, por lo que se considera que los resultados son buenos.

Como se puede observar en la tabla de resultados (tabla 4.5) existe una notable diferencia en los tiempos entre las 23 instancias, cabe mencionar que las instancias que superan los mil segundos son aquellas que cuentan con la restricción de distancias máxima de recorrido, por lo que respetar esta restricción aumenta el tiempo necesario considerablemente.

Tomando como ejemplo las instancias p12 y p13, ambas instancias son casi idénticas, con la única diferencia es que existe una restricción de recorrido máximo de 200 unidades, esta característica repercute en que el tiempo de ejecución necesario para las instancias p13 sea en promedio más de veinte veces mayor comparado con la p12.

La experimentación con las instancias aleatorias arrojó los datos mostrados en la tabla 4.6, la cual contiene la mejor solución encontrada con el algoritmo propuesto y el tiempo promedio de ejecución.

Tabla 4.6 Resultados de las instancias aleatorias creadas.

Instancia	Mejor Solución	Solución Promedio	Tiempo promedio(seg)
p30	11920.25	12223.61	248.91
p31	11090.13	11290.19	243.28
p32	14048.64	14408.41	304.14
p33	18155.99	18587.20	320.20
p34	23614.38	24097.63	253.84
p35	31802.52	32429.28	280.10

El equipo utilizado para realizar experimentación a lo largo del desarrollo este trabajo de tesis cuenta con las siguientes características:

- Procesador: Intel Core i5 10400 2.90 GHz.
- RAM: 16 GB DDR4 2666 MHz.
- Sistema Operativo: Ubuntu Desktop 22.04.02.

CAPÍTULO V

CONCLUSIONES Y TRABAJOS FUTUROS

5.1 Conclusiones

Se presentó un algoritmo metaheurístico de dos fases para solucionar el Problema de Ruteo de Vehículos Multi-deposito (MDVRP), en el cual la primera fase consta de una heurística constructiva que genera una solución inicial, la cual es mejorada en la segunda fase por el algoritmo Recocido Simulado, implementando 4 vecindades se logró resolver el MDVRP.

Para evaluar el desempeño del algoritmo propuesto se resolvieron los 23 problemas de referencia que componen el benchmark del MDVRP, realizando comparaciones con las mejores soluciones conocidas en la literatura hasta la actualidad.

El algoritmo fue capaz de resolver las 23 instancias encontrando la solución mejor conocida en más de 60% de los casos, para la mayoría de las instancias que no se lograron encontrar la solución óptima, el margen de error no superaba el 5% de margen de error con respecto a al BKS, y solo en dos instancias su solución está entre el 5% y el 7% de margen de error.

Dado que ninguna solución supera el 7% de margen de error se concluye que los resultados obtenidos por algoritmo propuesto son competitivos con la literatura.

Además, se generaron 6 instancias aleatorias con el fin de evaluar el rendimiento del algoritmo propuesto, dichas instancias cuentan con un mayor tamaño que los problemas de la literatura en lo que respecta al número de clientes.

5.2 Trabajos futuros

A continuación, se enlistan algunas propuestas para posibles trabajos futuros a realizarse de este trabajo de tesis.

- Implementar otros algoritmos metaheurísticos diferentes tales como algoritmos genéticos o búsqueda tabú. para solucionar el MDVRP y comparar su eficacia.
- Solucionar otras variantes del VRP con otras restricciones del problema, como puede ser con ventanas de tiempo o con recolección y entrega.
- Realizar una hibridación entre dos o más metaheurísticas.
- Realizar una paralelización a dos o más metaheurísticas y evaluar el margen de mejora en tiempos entre los algoritmos utilizados.
- Generar instancias para el MDVRP que consideren la restricción de máxima distancia por recorrido.

REFERENCIAS

- Aras, N., Aksen, D., & Tekin, M. T. (2011). Selective multi-depot vehicle routing problem with pricing. *Transportation Research Part C: Emerging Technologies*, 19(5), 866-884.
- Bezerra, S. N., de Souza, S. R., & Souza, M. J. F. (2018). A GVNS algorithm for solving the multi-depot vehicle routing problem. *Electronic Notes in Discrete Mathematics*, 66, 167-174.
- Blum, C. (2005). Ant colony optimization: Introduction and recent trends. *Physics of Life reviews*, 2(4), 353-373.
- Černý, V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of optimization theory and applications*, 45, 41-51.
- Chao, I. M., Golden, B. L., & Wasil, E. (1993). A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions. *American Journal of Mathematical and Management Sciences*, 13(3-4), 371-406.
- Christofides, N., Eilon, S.: An algorithm for the vehicle-dispatching problem. *Oper. Res. Q.* 20(3), 309–318 (1969).
- Contardo, C., & Martinelli, R. (2014). A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. *Discrete Optimization*, 12, 129-146.
- Cordeau, J. F., & Maischberger, M. (2012). A parallel iterated tabu search heuristic for vehicle routing problems. *Computers & Operations Research*, 39(9), 2033-2050.
- de Oliveira, F. B., Enayatifar, R., Sadaei, H. J., Guimarães, F. G., & Potvin, J. Y. (2016). A cooperative coevolutionary algorithm for the multi-depot vehicle routing problem. *Expert Systems with Applications*, 43, 117-130.

- Dorigo, M.: Optimization, learning and natural algorithms [in Italian]. Ph.D. thesis, Politecnico di Milano, Italy (1992)
- Escobar, J. W., Linfati, R., Toth, P., & Baldoquin, M. G. (2014). A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem. *Journal of heuristics*, 20, 483-509.
- Gillett, B., Johnson, J.: Multi-terminal vehicle-dispatch algorithm. *Omega* 4(6), 711–718 (1976).
- Glover, F., & Melián, B. (2003). Búsqueda tabú. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 7(19), 0.
- GONZÁLEZ, J. G., Moscato, P., & Tseng, F. T. (1996). Optimización Heurística y Redes Neuronales en Dirección de Operaciones e Ingeniería. A. Díaz, coord. Paraninfo.
- Kirkpatrick, S., Gelatt Jr, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science*, 220(4598), 671-680.
- Lau, H. C., Chan, T. M., Tsui, W. T., & Pang, W. K. (2009). Application of genetic algorithms to solve the multidepot vehicle routing problem. *IEEE transactions on automation science and engineering*, 7(2), 383-392.
- Luo, J., & Chen, M. R. (2014). Improved shuffled frog leaping algorithm and its multi-phase model for multi-depot vehicle routing problem. *Expert Systems with Applications*, 41(5), 2535-2545.
- Melián, B., Pérez, J. A. M., & Vega, J. M. M. (2003). Metaheurísticas: Una visión global. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, 7(19), 0.
- Nunes Bezerra, S., Souza, M. J. F., de Souza, S. R., & Nazário Coelho, V. (2019). A VNS-Based Algorithm with Adaptive Local Search for Solving the Multi-Depot Vehicle Routing Problem. In *Variable Neighborhood Search: 6th International Conference, ICVNS 2018, Sithonia, Greece, October 4–7, 2018, Revised Selected Papers 6* (pp. 167-181). Springer International Publishing.

- Ombuki-Berman, B., & Hanshar, F. T. (2009). Using genetic algorithms for multi-depot vehicle routing. *Bio-inspired algorithms for the vehicle routing problem*, 77-99.
- Renaud, J., Laporte, G., & Boctor, F. F. (1996). A tabu search heuristic for the multi-depot vehicle routing problem. *Computers & Operations Research*, 23(3), 229-235.
- Rivera, J. C., & Celín, A. J. (2010). Hybrid variable neighborhood and simulated annealing heuristic algorithm to solve rcpsp. *Revista Facultad de Ingeniería Universidad de Antioquia*, (56), 255-267.
- Sadati, M. E. H., Aksen, D., & Aras, N. (2020). A trilevel r-interdiction selective multi-depot vehicle routing problem with depot protection. *Computers & operations research*, 123, 104996.
- Shi, Y., Lv, L., Hu, F., & Han, Q. (2020). A heuristic solution method for multi-depot vehicle routing-based waste collection problems. *Applied Sciences*, 10(7), 2403.
- Siarry Patrick, 2016, *Evolutionary Algorithms, Metaheuristics* 1st ed. 2016, (pp.25-26), Springer.
- Stodola, P. (2020). Hybrid ant colony optimization algorithm applied to the multi-depot vehicle routing problem. *Natural Computing*, 19(2), 463-475.
- Subramanian, A., Uchoa, E., & Ochi, L. S. (2013). A hybrid algorithm for a class of vehicle routing problems. *Computers & Operations Research*, 40(10), 2519-2531.
- Talbi, E. G. (2002). A taxonomy of hybrid metaheuristics. *Journal of heuristics*, 8, 541-564.
- Toro, D. O., Ocampo, E. M. T., & Rendón, R. A. G. (2018). Solución del MDVRP usando el algoritmo de búsqueda local iterada. *REVISTA COLOMBIANA DE TECNOLOGIAS DE AVANZADA (RCTA)*, 1(31), 120-127.
- uit het Broek, M. A., Schrottenboer, A. H., Jargalsaikhan, B., Roodbergen, K. J., & Coelho, L. C. (2021). Asymmetric multidepot vehicle routing problems: Valid inequalities and a branch-and-cut algorithm. *Operations Research*, 69(2), 380-409.
- Wan, F., Gou, H., Pan, W., Hou, J., & Chen, S. (2021). A Mathematical Method for Solving Multi-Depot Vehicle Routing Problem. Available at SSRN 3943419.

Yao, B., Chen, C., Song, X., & Yang, X. (2019). Fresh seafood delivery routing problem using an improved ant colony optimization. *Annals of Operations Research*, 273, 163-186.

Yu, B., Yang, Z. Z., & Xie, J. X. (2011). A parallel improved ant colony optimization for multi-depot vehicle routing problem. *Journal of the Operational Research Society*, 62(1), 183-188.

Cuernavaca, Morelos a 13 de septiembre del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA.
PRESENTE

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Jhony Brito Colin, con matrícula 10053357, con el título **Diseño de un algoritmo metaheurístico para la solución del Problema de Ruteo de Vehículos MultiDepósito**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Federico Alonso Pecina
Profesor- investigador
Facultad de Contaduría, Administración e Informática



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

FEDERICO ALONSO PECINA | Fecha:2023-09-13 22:36:33 | Firmante

Drj5bzEKuvPKHDI+VL0mJYgCC3R72tRliBj07iOXfeRyXsD2iafcUJnJtFyfRFy+f9r5DUoi1E6EAjhfV8+Np0UDhW5M1nLHimO7wLizLrfMLmccu8zw78ntsnaIoizflTibKK+eGICIYgxA
qquckeu8OlnA93wNbYDMD1HBj5IO1d+3WNoK/4CUbEqhQXWi4HP81MH+0TnbjYWGnlGclceGvvb3ump8SdWTSE/VYFMqlYRHtHuOIJhfVwCu+UtdUv37zWwH3xyE9KWkDFFV
Auxlf5RfCM55iLGHtOfVgf/60n+5vm9twh4E8wHJZMj07/Lk1XUWFyTqVxrijRERAQ==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



t8g3FZ0pB

<https://efirma.uaem.mx/noRepudio/iKuXsrd87p219qmgETPq8dYn3nGfEbBk>



Cuernavaca, Morelos a 8 de septiembre del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA.
PRESENTE

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Jhony Brito Colín, con matrícula 10053357, con el título **Diseño de un algoritmo metaheurístico para la solución del Problema de Ruteo de Vehículos Multidepósito**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Martín Heriberto Cruz Rosales
Profesor investigador
Facultad de Contaduría, Administración e Informática



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

MARTIN HERIBERTO CRUZ ROSALES | Fecha:2023-09-10 22:15:05 | Firmante

k1xDUHKgOQhc9EFQNfmGsaBChHkUbkS2TZf+RrkjJn6fqC6t0vHfqK/z+kFCmDjf/x1MBWMMRiAwv2DBppYkT37iqVmy5dfTfPLqtddec9t1Y30aY3hxFRVHh3aT4fsF3n CZFT7aSC9xcuJcj1WWIh6RePCQSkGvNHNKnekawj5gnCTiEJLO5m/fQF9W9rrv6l0xPYneFx1ETJQ91TuDph6mrSt0vGBwM55q8MvshzFKbeS4LjYvCOQrNhXEY9tt/l2r0G8Xj8sxxvXRHm8yNpjVqbeqGdggOzthgghFu5pK4Uadxrot37Mo6rN7z2+ggb2me/c9MZn5o5nfAfhwcOsQw==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:



[nyCsuJUYw](#)

<https://efirma.uaem.mx/noRepudio/UwSDhLneEss1XygGT8cw7kPKuM1DL7xv>





UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS



Cuernavaca, Morelos a 8 de septiembre del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA.
PRESENTE

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Jhony Brito Colin, con matrícula 10053357, con el título **Diseño de un algoritmo metaheurístico para la solución del Problema de Ruteo de Vehículos MultiDepósito**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Marco Antonio Cruz Chávez
Profesor- investigador
Centro de Investigación en Ingenierías y Ciencias Aplicadas.



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

MARCO ANTONIO CRUZ CHAVEZ | Fecha:2023-09-08 20:37:32 | Firmante

sZ9J7rO0uMDiAXJQxjoC9B0/IApP0ocMzAE7svjEg5rAS7xXHr5VFB9lf+0NTPqAMiDADF45fGbEo59QyErjB3cj+AtHuh/f+N5AEWwEkzmzGcyfzKtc2yDfvK1j1yeEu/mEQYes2Oc3ayUMRWDXPJiZhfOdNOWSQw5AOyAnoskFaYJOL3f24ObzSsKmwOPxdEfsTviZTOqsKrkOeYZP867BnrON8psAEsW0hiJIRMqc2rLsPMolRowalxMTyUh1lu5zpeNhw+33b1cuJl ttAvX/0EDwRFz1wH03Xat6g+eCSBCA4AdLn/L+LtrXfhK9e/ulAhoirkBaOK4joZC8Q==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:



[nt4hPlvW6](#)

<https://efirma.uaem.mx/noRepudio/g3jVvevZE7jjDI1cLLuRGPscJZpJD52x>



Cuernavaca, Morelos a 18 de febrero del 2024.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA.
PRESENTE

En mi carácter de revisor de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Jhony Brito Colin, con matrícula 10053357, con el título **Diseño de un algoritmo metaheurístico para solución del Problema de Ruteo de Vehículos MultiDepósito**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Martín Gerardo Martínez Rangel
Profesor- investigador
Facultad de Contaduría, Administración e Informática



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

MARTIN GERARDO MARTINEZ RANGEL | Fecha:2024-02-19 15:36:36 | Firmante

GpZUZj7bzVJqcgD4TPbYebI0JbnRBoNfZ24IYKWcDpeiEqLRuEM/qjpcX9z6SLppittmw2n6iohTskqZO/aPankn70kD6kNLfRd/OKy4uRS3B93puY6ud2o7LLLJGBHKyTNgDMI4ga
ukdH/NQ81QZ6gMRKfgrM8pmmelC0AyEmRLThbb+NzGDmcCp5KO2rLEuyfC8XXLBI+UHKd7i6dqmZLVLHM1ojcTPraV6UI3J1wVDFf92oLkFKz3t3tVsGslR+wNEMDh4HqN9tu
+aYyvm2SelceailPBMb0YoieXNMYcCLsOjY0akFJFcP660szpkaiQEZIlwDdbruyqRNw==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



[ntegkbsjd](#)

<https://efirma.uaem.mx/noRepudio/853yCRPVtXgbT7UDnuuZTaCgLTmYMx>



UAEM
RECTORÍA
2023-2029



Cuernavaca, Morelos a 8 de septiembre del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA.
PRESENTE

En mi carácter de revisora de tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante Jhony Brito Colin, con matrícula 10053357, con el título **Diseño de un algoritmo metaheurístico para la solución del Problema de Ruteo de Vehículos MultiDepósito**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que la Universidad Autónoma del Estado de Morelos tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dra. Irma Yazmín Hernández Báez
Profesora de Tiempo Completo
Universidad Politécnica del Estado de Morelos



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

IRMA YAZMIN HERNANDEZ BAEZ | Fecha:2023-09-12 11:57:12 | Firmante

oiH31CJliwB3eUhhYhxwblLigsa5Clfs2rK7xjr4qydtlPmzzQrwzx00f0TK8Noh1a07jAeng7SF2ITiZaObHWkU5obpnIrfvyCgtgsfxKcnLjty0DadqIFdaPwQea50IuyyMp2MRR0e7ErFsn
bhj29fq0QM/NcFDeUSUX+VrvVgcuM9rF48vOrcCgNVC1mnWLU0NcYB7PsMh/bE55Xnv6kZr4YdNm9/p18irqMyjIVkmDlftL2FQ86kWs+YdweJ09M6/WCWxWr6wohCvuHgXUgh
DCZANZszEFBkeWs6eQ11aDq1vIEKqobNgriThStnMK3PZ7FK7kWXBPUY9/e4JA==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



w8AuxiXZY

<https://efirma.uaem.mx/noRepudio/YBBRpYxdGe1GgZRFEvVb6L8UhiH8nEhG>

