

DESARROLLO DE UN MÉTODO HÍBRIDO PARA LA RESOLUCIÓN DEL MDVRP*

A hybrid method for the resolution of the MDVRP

Por: **D. IVÁN GALLEGO MATEOS**

Estudiante predoctorado. Departamento de Administración de Empresas. Universidad de Oviedo. España
ivangallegomateos@outlook.com.

Por: **DR. ALBERTO GÓMEZ GÓMEZ**

Profesor Titular de Universidad. Departamento de Administración de Empresas. Universidad de Oviedo. España
albertogomez@uniovi.es

Por: **D. DANIEL ARGÜELLES MARTINO**

Estudiante predoctorado. Departamento de Administración de Empresas. Universidad de Oviedo. España
leinad87@gmail.com

Por: **DR. JAVIER PUENTE GARCÍA**

Profesor Titular de Universidad. Departamento de Administración de Empresas. Universidad de Oviedo. España
jpuente@uniovi.es

Por: **DR. NAZARIO GARCÍA FERNÁNDEZ**

Profesor Contratado Doctor. Departamento de Administración de Empresas. Universidad de Oviedo. España
ngarciaf@uniovi.es

* Recibido para publicación: 25 de noviembre de 2013.
Enviado para evaluación externa: 26 de noviembre de 2013.
Recibida evaluación externa positiva: 10 de diciembre de 2013.
Aceptado para publicación: 17 de diciembre de 2013.

RESUMEN: Se presenta un algoritmo metaheurístico capaz de resolver de manera eficiente el problema de ruteo de vehículos con múltiples depósitos (MDVRP). Resulta de una síntesis de varios heurísticos aplicados anteriormente en diversos problemas de optimización. En concreto, la estructura general del algoritmo sigue los principios de una búsqueda local iterada y la búsqueda local se realiza mediante una búsqueda por entornos variables descendente. La aplicación sencilla y directa del algoritmo desarrollado, sin apenas algún ajuste de parámetros, facilita una rápida implementación y ejecución. La sección de experimentación presentada al final de este estudio demuestra su eficiencia en relación a los mejores métodos de resolución actuales.

PALABRAS CLAVE: Optimización, multi-depósito, ILS, VNS, metaheurísticas.

ABSTRACT: We present a metaheuristic algorithm which can efficiently solve the vehicle routing problem with multiple depots (MDVRP). This algorithm, called IVNDS, results from a conjunction of well-known heuristics previously applied in several optimization problems. In particular, the main structure of the algorithm follows the principles of an iterated local search and its search method is based in a variable descent neighborhood search. The simple and direct application of the algorithm, with almost any parameter tuning, facilitates a quick implementation and execution. The experimentation presented at the end of this study demonstrates its efficiency in relation to the best known methods for the resolution of the MDVRP in the actuality.

KEY WORDS: optimization, multi-depot, ILS, VNS metaheuristics.

Sumario: I.- INTRODUCCIÓN. II -MULTI-DEPOT VEHICLE ROUTING PROBLEM (MDVRP). III.- ESTADO DEL ARTE. IV.- ALGORITMO IVNDS. 1.- Estructura General. 2.- Inicialización. 3.- Búsqueda Local. A.- *Estructuras de vecindarios*. 4.- Criterio de aceptación. 5.- Perturbación. A.- *Perturbación intra-depósito*. B.- *Perturbación inter-depósito*. 6.- Lista de vecinos. V.- EXPERIMENTACIÓN. 1.- Ajuste de parámetros. 2.- Resultados. VI.- CONCLUSIONES. VII.- BIBLIOGRAFÍA

I. INTRODUCCIÓN

La gestión logística es un elemento clave en la estrategia empresarial, siendo una de sus funciones principales la distribución, y dentro de ella la capacidad para optimizar las rutas de transporte. En este contexto, las empresas deben analizar los factores más relevantes en el diseño de sus rutas vehiculares así como las metodologías más adecuadas para tal optimización¹.

La optimización de una ruta engloba todas las acciones que contribuyen a la mejora de la función de distribución en términos de nivel de servicio, calidad y costes a través de decisiones de carácter estratégico, táctico y operativo.

Las claves para abordar un problema de optimización de rutas son específicas para cada organización en función de sus particularidades, aunque hay ciertos aspectos críticos generalmente comunes en todas ellas.² Así, en primer lugar, deben definirse y priorizarse tanto los factores objeto de la optimización (p.e. nivel de servicio o coste) como las restricciones a satisfacer y la afectación que una ruta modificada pueda tener en otras rutas existentes. En segundo lugar, deben identificarse claramente las características del servicio actual tanto de los productos a distribuir como de las rutas y de la propia organización (tipología de flotas, restricciones -de tipo de vehículos, horarias, volumen/peso, legales, condiciones de mantenimiento, prioridades de entrega, rutas preferentes, existencia de sistemas automáticos de recogida de información, etc-). Por último, se debe establecer el tipo de resultado deseado para el proyecto (modelo de ejecución discreto o continuo, automático o manual, integrado o no con el resto de sistemas de la organización, etc).

La investigación sobre las actividades del transporte, se ha vuelto clave en la adaptación de los operadores de las cadenas de suministro dada la creciente integración de estas cadenas logísticas y su carácter intermodal y multimodal. Así, los sistemas inteligentes de transporte ofrecen un amplio grupo de tecnologías tanto para el control del tráfico de mercancías como para la gestión de vehículos en ruta o la gestión integral de los flujos de transporte. Entre este grupo de tecnologías destacan los sistemas de información geográfica, los sistemas de localización geográfica y las aplicaciones informáticas capaces de calcular modelos matemáticos de optimización de rutas (entre los que se incluiría el objeto del presente trabajo).

El problema de ruteo de vehículos (VRP, Vehicle Routing Problem) se define como una variante específica en el campo de la optimización combinatoria en el que un cierto número de clientes debe ser atendido por una flota de vehículos, de acuerdo a una serie de restricciones que definen las distintas variantes del problema. Cada vehículo debe realizar una ruta que comience y finalice en el mismo depósito. Cada cliente está definido por una cierta demanda y sus coordenadas geográficas, utilizadas para

¹ ARANGO, M. D.; ADARME, W.; ZAPATA, J.A.; “Commodities distribution using alternative types of transport”, Colombia bread SME’s. DYNA, 163, 2010, págs. 222-233

² Documento de trabajo. “Vigilancia tecnológica: Técnicas para la Optimización de Rutas de Transporte y Distribución”. Brain Trust Consulting Services Rev02, 2009 .
http://www.odette.es/SGC/downloads/CAM/Vigilancia_Tecnologica_Tecnicas_Optimizacion_Rutas.pdf.

determinar la distancia entre clientes. Información adicional incluye la capacidad del vehículo y la longitud máxima de una ruta. El objetivo del VRP común sería el de generar un conjunto de rutas para una flota de vehículos que visite un cierto número de clientes geográficamente dispersos y con una demanda conocida, minimizando la distancia total requerida en el proceso.

En el extenso campo que recoge las variantes del VRP, destacamos aquellas que recibieron mayor atención en los últimos años. Generalmente, cada una de estas variantes es resultado de incorporar nuevas restricciones al esquema general, de manera que el problema teórico tenga más parecido con el real. Por ejemplo, el problema de ruteo de vehículos con ventanas de tiempo (VRPTW) establece, para cada uno de los clientes, un intervalo de tiempo en el cual debe ser atendido por alguno de los vehículos. En el VRP periódico (PVRP), los clientes son visitados en días diferentes, dividiendo el problema en varios subgrupos. Otra variante menos conocida es aquella en que los clientes, además de recibir productos, pueden enviarlos, conocida como VRP con envío y recogida (VRPPD). Todas estas variantes comparten la característica de disponer de un único depósito desde el que establecer las rutas. Su contrapartida, denominada VRP multi-depósito (MDVRP), resuelve problemas con varios depósitos de partida, y ha recibido una menor atención a pesar de su importancia en sistemas logísticos actuales. Al contar con depósitos adicionales se presenta un problema previo al establecimiento de las rutas: la asignación de los diferentes clientes a cada uno de los depósitos.

La complejidad NP-completa del MDVRP, que aumenta exponencialmente a medida que lo hace el número de clientes, dificulta el desarrollo de métodos que resuelvan el problema de manera óptima en un tiempo razonable. No obstante, y a pesar de su elevado coste computacional, existen ejemplos prácticos de métodos exactos aplicados al MDVRP que serán tratados posteriormente. El enfoque más habitual a la hora de resolver este problema es el de aplicar métodos heurísticos o metaheurísticos, capaces de generar soluciones cercanas a la óptima sin incurrir en altos tiempos de ejecución y carga computacional.

El presente artículo propone un método metaheurístico simple, denominado IVNDS, orientado a la resolución del MDVRP de manera eficiente y con un reducido coste computacional. Su aplicación es directa y con un limitado número de parámetros, cuyo valor se ajusta en relación a ciertas características del problema. Alternativamente, se introducen conceptos para potenciar las fases de dispersión y diversificación, de vital importancia en los metaheurísticos de mejora, entre los que se encuentra un conjunto de métodos de perturbación y un criterio de aceptación. Cada una de las aportaciones será tratada en detalle en posteriores secciones.

El artículo está organizado como prosigue: En la sección 2 se realiza un repaso a las principales características del MDVRP, seguido de una revisión de los principales métodos que constituyen el estado del arte actual en la sección 3. Los aspectos principales del algoritmo desarrollado, así como las aportaciones que introduce, serán tratadas en la sección 4 de manera individual. Se presenta a su vez, en la sección 5, un apartado de experimentación donde se muestran los resultados obtenidos por el algoritmo desarrollado en conocidos benchmarks. La sección 5.1 comienza con una

breve justificación de las diferentes elecciones previas a la experimentación, como el ajuste de parámetros, seguida de una comparativa con respecto a los resultados de reconocidos métodos en la sección 5.2.. Finalmente, en la sección 6 se presenta una conclusión que resume los principales objetivos del artículo.

II. MULTI-DEPOT VEHICLE ROUTING PROBLEM (MDVRP)

El problema de ruteo de vehículos con múltiples depósitos es una variante del VRP clásico que incorpora varios depósitos con una localización predefinida. Cada depósito cuenta con una flota limitada de vehículos con capacidad fija, utilizada para repartir los productos demandados por los clientes, cuya localización y demanda es también conocida de antemano. Cada vehículo debe comenzar y terminar su ruta en el mismo depósito, y cada cliente debe ser visitado una única vez.

A continuación se presenta el modelo y formulación matemática que formaliza las diferentes restricciones y la función objetivo que definen al MDVRP:

- Objetivo: el objetivo es la distancia total del conjunto de rutas.
- Factibilidad: una solución es factible si todas las rutas satisfacen las restricciones de longitud y distancia y comienzan y terminan en el mismo depósito.
- Formulación: el problema CVRP se extiende al caso donde existen múltiples almacenes, por lo que se denotará el conjunto de vértices como $V = \{v_1, \dots, v_n\} \cup V_0$, donde $V_0 = \{v_{o1}, \dots, v_{od}\}$ son los vértices que representan los almacenes. La ruta i está definida por $R_i = \{d, v_i, \dots, v_m, d\}$ con $d \in V_0$.

La nomenclatura utilizada en la formulación posterior se detalla a continuación:

- Índices de origen (**i**)
- Índices de destino (**j**)
- Grupo de depósitos (**D**)
- Conjunto de clientes (**C**)
- Vehículos (**V_k**)
- Ruta (**R_k**)
- Capacidad del vehículo (**Q_k**)
- Variables de decisión (**X_{ijK}**)

- Distancia entre clientes y depósitos (**Dis ij**)
- Grupo de clientes visitados (**S**)

A continuación se muestra la descripción del MDVRP.

- 1) Posee M depósitos en un grupo D, cada depósito K ($K \in D$) posee un grupo de vehículos exactos.
- 2) Cada vehículo V_k realiza una ruta R_k que contiene un cierto número de clientes, y que comienza y finaliza en un mismo depósito K.
- 3) Todos los clientes (denotados como C) deben ser visitados por un vehículo.
- 4) El total de la demanda de los clientes atendidos en cada ruta R_k no excede la capacidad del vehículo que le atiende Q_k .

Basado en la distancia entre puntos (Clientes y depósitos) dis_{ij} ($i, j \in C \cup D$), calculado por su ubicación, la demanda de los clientes d_i ($i \in C$) y la capacidad del vehículo Q_k ($K \in D$), el MDVRP pretende asignar un cliente a cada depósito y se le asigna una ruta a cada vehículo del depósito para reducir al mínimo la distancia total del recorrido.

La función objetivo del modelo es:

$$\text{Minimize } \sum_{i \in C \cup D} \sum_{j \in C \cup D} dis_{ij} \sum_{k \in D} x_{ijk}$$

Sujeto a:

$$\sum_{i \in C \cup D} \sum_{k \in D} x_{ijk} = 1 \quad \forall j \in C \quad (1)$$

$$\sum_{i \in C} \sum_{j \in D} x_{ijk} \leq 1 \quad \forall k \in D \quad (2)$$

$$\sum_{j \in C \cup D} x_{ijk} = \sum_{j \in C \cup D} x_{jik} \quad \forall k \in D, i \in C \cup D \quad (3)$$

$$\sum_{j \in C} d_j \sum_{i \in C \cup D} x_{ijk} \leq Q_k \quad \forall k \in D \quad (4)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1 \quad \forall S \text{ contenido en } C, |S| \geq 2 \quad k \in D \quad (5)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, k \in D, j \in C \quad (6)$$

Donde la restricción 1 propone que todo destino (j) pertenece a un grupo de clientes (C). La restricción 2 propone que todo depósito (k) pertenece a un grupo de depósitos (D). En las restricciones de 1 a 3 se impone que cada cliente sea visitado una única vez, cada ruta sea realizada por un único vehículo, y que el vehículo entra y sale para los clientes asignados respectivamente. En la restricción 4 se aprecia la ecuación de la capacidad de cada vehículo (Q_k). La restricción 5 es la ecuación de restricción generalizada de eliminación de recorrido, la cual restringe que cada vehículo atienda al menos una ruta de cada cliente pasando a un conjunto de visitados (S) por el vehículo

(Vk). Y por último en la restricción 6 se aprecia los requisitos de las variables de decisión, donde las variables de decisión son X_{ijk} que representan una solución de enrutamiento. $X_{ijk} = 1$ si y solo si el cliente j precede al cliente en la ruta R_k , $R_k (i, j \in C \cup D, k \in D)$ de lo contrario el valor es cero.

III. ESTADO DEL ARTE

Durante un largo periodo de tiempo, los métodos heurísticos basados en la búsqueda tabú se mantuvieron como el mejor enfoque para resolver el MDVRP. Renaud et al.³ desarrollaron un método tabú de tres fases que aseguraba la diversificación e intensificación de la búsqueda permitiendo una amplia exploración del espacio de soluciones. Esto se consiguió aceptando ciertos deterioros de la solución y declarando como tabú modificaciones realizadas recientemente. Cordeau et al.⁴ presentaron un método con la búsqueda tabú como núcleo que se mantuvo como estado del arte en la materia durante varios años, junto a la mejora que aplicaron posteriormente en 2001⁵. Estos métodos se caracterizaban por permitir soluciones no factibles en el desarrollo de la búsqueda y por una innovadora fase de diversificación basada en una progresiva penalización de características recurrentes en la solución, guiando la búsqueda hacia zonas más prometedoras. De la misma manera, las soluciones infactibles eran penalizadas en relación a la magnitud de las violaciones, ayudando al algoritmo a evitar regiones infactibles.

En 2002 Giosa et al.⁶ presentaron un conjunto de nuevos métodos de asignación para el MDVRP que organizaba grupos de clientes en determinados depósitos siguiendo diferentes criterios. Estos criterios fueron: asignación según prioridades, asignación cíclica y asignación mediante clusters. Más tarde, Polacek et al.⁷ desarrollaron un efectivo método para el MDVRP con ventanas de tiempo basado en una búsqueda por entornos variables (VNS) que aceptaba soluciones infactibles, presentaba un método de diversificación mediante la modificación cross-exchange y un criterio de aceptación siguiendo el enfoque aplicado por Cordeau et al.⁸. Una de las mejores aportaciones a la resolución del MDVRP fue realizada por Pisinger et al.⁹ en 2007, quienes obtuvieron los mejores resultados conocidos para el MDVRP

³ RENAUD, J.; LAPORTE, G.; FOUAD BOCTOR, F.; "A tabu search heuristic for the multi-depot vehicle routing problem" *Computers & Operations Research*, 23, 1996. Págs 229-235

⁴ CORDEAU, J. F.; GENDREAU, M.; LAPORTE, G.; "A tabu search heuristic for periodic and multi-depot vehicle routing problems.", *Networks*, 30(2), 1997, págs 105-119.

⁵ CORDEAU, J. F.; LAPORTE, G.; MERCIER, A.; "A unified tabu search heuristic for vehicle routing problems with time windows", *Journal of the Operational Research Society*, 52(8), 2001, págs 928- 936.

⁶ GIOSA, I. D.; TANSINI, I. L.; VIERA, I. O.; "New assignment algorithms for the multi-depot vehicle routing problem", *Journal of the Operational Research Society*, 53(9), 2002, págs. :977-984.

⁷ POLACEK, M.; F. HARTL, R.; DOEMER, K.; REIMANN, M.; "A variable neighborhood search for the multi depot vehicle routing problem with time windows", *Journal of Heuristics*, 10(6), 2004, págs 613-627

⁸ CORDEAU, J. F.; LAPORTE, G.; MERCIER, A.; "A unified tabu search heuristic for vehicle routing problems with time windows", (...), opus cit., pág

⁹ PISINGER, D.; ROPKE, S.; "A general heuristic for vehicle routing problems", *Computers & Operations Research*, 34(8), 2007, págs. 2403-2435.

desarrollando un método de búsqueda con vecindarios variables aplicado a grandes instancias (Adaptive Large Neighbourhood Search, ALNS), que implementaba un paradigma de destrucción y mejora de la solución mediante una selección variable y adaptativa de operadores.

En los últimos años, algunos métodos basados en algoritmos poblacionales han sido aplicados satisfactoriamente al MDVRP. Ho et al.¹⁰ propusieron un algoritmo genético híbrido que generaba soluciones iniciales atendiendo a diferentes métodos. Lau et al.¹¹ presentaron un algoritmo genético para tratar el MDVRP basado en los principios de incertidumbre y la lógica difusa para ajustar los parámetros que gobiernan la evolución de las diferentes soluciones y los porcentajes de realización de las operaciones de mezcla y mutación. Los mejores resultados conocidos hasta la fecha fueron obtenidos por el método evolutivo propuesto por Vidal et al.¹² (2012), un algoritmo genético híbrido que incorporaba avanzados mecanismos de guiado para intensificar y diversificar la búsqueda. Su principal contribución fue una nueva regla de selección para el cruce de las diferentes poblaciones que tenía en cuenta no solo la función de coste sino la contribución del nuevo cromosoma en la diversificación de la búsqueda.

Mirabi et al.¹³ presentaron tres heurísticos híbridos para resolver el MDVRP combinando heurísticos de búsqueda y mecanismos de mejora de tipo determinístico, estocástico y de recocido simulado, respectivamente. Yu et al.¹⁴ presentaron un algoritmo paralelo basado en una búsqueda mediante colonia de hormigas (ACO) para la resolución del MDVRP como evolución a un proyecto previo¹⁵ que introducía técnicas de mutación propias de los algoritmos genéticos para facilitar la diversificación y una nueva estrategia para actualizar el nivel de feromona del recorrido. Recientemente, Kuo et al.¹⁶ (2012) desarrollaron un método para el MDVRP que trataba con costes adicionales de utilización de vehículos que el problema tradicional no cubría. El núcleo del algoritmo presenta una búsqueda por entornos variables con una variación del método de aceptación propio del recocido simulado. En el mismo año, Cordeau et

¹⁰ HO, W.; HO, G.; JI, P.; LAU, H.; "A hybrid genetic algorithm for the multi-depot vehicle routing problem." *Engineering Applications of Artificial Intelligence*, 21(4), 2008, :548-557.

¹¹ LAU, H.; CHAN, T.; TSUI, W. T.; PANG, W. K.; "Application of genetic algorithms to solve the multidepot vehicle routing problem". *Automation Science and Engineering, IEEE Transactions on*, 7(2), 2010, págs. 383-392

¹² VIDAL, T.; CRAINIC, T. G.; GENDREAU, M.; LAHRICHI, N.; REI, W.; "A hybrid genetic algorithm for multidepot and periodic vehicle routing problems", *Operations Research*, 60(3), 2012, págs. 611-624.

¹³ MIRABI, M.; FATEMI GHOMI, S. M. T.; JOLAI, F.; "Efficient stochastic hybrid heuristics for the multi-depot vehicle routing problem", *Robotics and Computer-Integrated Manufacturing*, 26(6), 2010, págs. 564 - 569

¹⁴ YU, B.; YANG, Z-Z.; XIE, J-X; "A parallel improved ant colony optimization for multi-depot vehicle routing problem", *Journal of the Operational Research Society*, 62(1), 2010, págs.183-188

¹⁵ YU, B.; YANG, Z-Z ; BAOZHEN, Y.; "An improved ant colony optimization for vehicle routing problem", *European Journal of Operational Research*, 196(1), 2009, págs.171 - 176

¹⁶ KUO, Y.; WANG, C-C.; "A variable neighborhood search for the multi-depot vehicle routing problem with loading cost", *Expert Systems with Applications*, 39(8), 2012, págs. 6949-6954

al.¹⁷ presentaron un método que empleaba una búsqueda tabú iterada basada en anteriores trabajos para resolver diversas variantes del VRP clásico, como el MDVRP, SDVRP y PVRP, además de aquellas que incluyen ventanas de tiempo. Este método se ideó a modo de framework general que tratara diferentes problemas de ruteo de manera simple y directa, obteniendo resultados destacables.

Debido al alto coste computacional de resolver el MDVRP de manera óptima, el desarrollo de métodos exactos fue limitado, existiendo algunos estudios en el campo. Tanto Laporte et al.¹⁸ como Laporte et al.¹⁹ han desarrollado algoritmos basados en métodos de ramificación y poda que funcionaron bien con instancias simétricas y asimétricas relativamente pequeñas, tratando hasta cincuenta clientes. Más recientemente, Baldacci et al.²⁰ (2009) han desarrollado un método exacto para resolver de manera óptima el MDVRP en instancias de hasta 160 clientes, además de resolver varias instancias de otros problemas de ruteo de vehículos como el SDVRP y el HDVRP.

IV. ALGORITMO IVNDS

El método desarrollado es resultado de una conjunción de varios heurísticos bien conocidos en la literatura, con modificaciones y adiciones específicas. El esqueleto principal del algoritmo está basado en una búsqueda local iterada (ILS), cuyo esquema principal se muestra en el Algoritmo 1. Se puede encontrar un estudio completo del ILS en *The Handbook of metaheuristics*²¹. El principal inconveniente de los métodos de búsqueda local es que conducen a la solución a un mínimo que podría estar lejos de la solución óptima, y que no se puede evitar directamente. Como medio para evitar estos mínimos locales, el método ILS propone un proceso iterativo en que se aplica un método de perturbación que modifique la solución local seguido de una búsqueda local. Se pueden aplicar, a su vez, criterios de aceptación que determinen si la solución que genera la búsqueda local es válida para la evolución del algoritmo.

El algoritmo desarrollado, denominado IVNDS, implementa cada uno de los aspectos fundamentales de un ILS, a saber: búsqueda local, criterio de aceptación y perturbación. Cada uno de ellos presenta especificidades concretas que serán tratadas a lo largo de los siguientes apartados.

¹⁷ CORDEAU, J-F.; MAISCHBERGER, M.; "A parallel iterated tabu search heuristic for vehicle routing problems", *Computers & Operations Research*, 39(9), 2012, págs. 2033 – 2050.

¹⁸ LAPORTE, G.; NOBERT, Y.; ARPIN, D.; "Optimal solutions to capacitated multidepot vehicle routing problems", *École des hautes études commerciales*, 1984

¹⁹ LAPORTE, G.; NOBERT, Y.; TAILLEFER, S.; "Solving a family of multidepot vehicle routing and location-routing problems", *Transportation Science*, 22(3), 1988, págs.161–172

²⁰ BALDACCI, R.; MINGOZZI, A.; "A unified exact method for solving different classes of vehicle routing problems". *Mathematical Programming*, 120(2), 2009, págs. 347–380

²¹ LOURENCO, HR., OLIVIER, M.; STÜTZLE, T.; "Iterated Local Search", *The Handbook of Metaheuristics*, (ed. GLOVER), 2003, págs 320-353

Algoritmo 1 Búsqueda Local Iterada

- 1: // S_0 : sol inicial, S^* : sol actual,
 - 2: $S_0 =$ Generar solución inicial
 - 3: $S^* =$ Búsqueda local(S_0)
 - 4: **mientras** Condición de parada **hacer**
 - 5: $S' =$ Perturbación(S^*)
 - 6: $S^{*'}$ = Búsqueda local(S')
 - 7: $S^* =$ Criterio de aceptación(S^* , $S^{*'}$, history)
 - 8: **fin mientras**
-

1. ESTRUCTURA GENERAL

Se procede a caracterizar cada uno de los aspectos fundamentales del algoritmo desarrollado, correspondientes de manera directa con la estructura de alto nivel del ILS presentada en el Algoritmo 1. No obstante, cada una de las fases principales presenta una serie de añadidos y modificaciones que detallaremos en los siguientes apartados. En el Algoritmo 2 se muestra el esqueleto general del algoritmo implementado.

La generación de la solución inicial mencionada en el Algoritmo 1 se corresponde con las líneas 2 y 3, que en este caso se divide en dos fases: asignación y construcción. Los métodos de búsqueda local y perturbación de la solución responden al mismo patrón que los mencionados en el Algoritmo 1, y sus características adicionales serán tratadas en los siguientes apartados. El criterio de aceptación al que hace referencia el Algoritmo 1 se corresponde con las líneas 8-15 del Algoritmo 2, que en este caso cuenta con dos fases que determinarán si se acepta la solución generada por la búsqueda local. Por último, como condición de parada del algoritmo se establece un número máximo de iteraciones η .

Algoritmo 2 Estructura del algoritmo IVNDS

```
1: //S: Solución actual, S*: Mejor solución
2: S ← Asignar clientes a depósitos con el algoritmo del depósito más cercano.
3: S ← Construir solución inicial
4: Set S* ← S, iterations ← 0
5: mientras iterations < η hacer
6:   Búsqueda Local(S)
7:
8:   //Criterio de aceptación
9:   si f(S) < f(S*) entonces
10:    S* ← S
11:    iterations ← 0
12:   fin si
13:   si iterations ≥ Ω entonces
14:    S ← S*
15:   fin si
16:   Perturbación (S)
17: fin mientras
18: devolver S*
```

2. INICIALIZACIÓN

El proceso de inicialización del MDVRP consta de dos fases. En la primera se debe asignar un conjunto de clientes a cada depósito y en la segunda se genera una serie de rutas factible que contenga a todos los clientes. Este proceso generará una solución inicial válida requerida para el comienzo del proceso de mejora. Aunque no sea necesaria una buena solución inicial para el correcto funcionamiento del algoritmo, es conveniente que está no sea generada de manera aleatoria, pudiendo esto afectar negativamente a los resultados.

Existen diversos métodos para tratar la asignación de clientes a depósitos en la fase de inicialización, siendo un campo objetivo de estudio (ver Giosa et al.²², 2002). En el presente algoritmo se ha seleccionado un criterio en el que cada cliente es asignado a su depósito más cercano. Este reparto es un punto de partida, y será modificado dinámicamente a lo largo de la ejecución.

La generación de las rutas se realiza mediante el heurístico constructivo del vecino más cercano. Según este método, se inicia una ruta con un cliente seleccionado de manera aleatoria, y se van introduciendo nuevos clientes según la cercanía al último introducido en la ruta. Una vez se rompe alguna restricción de distancia o capacidad, el proceso se repite comenzando una nueva ruta, hasta que todos los clientes han sido visitados.

3. BÚSQUEDA LOCAL

²² GIOISA, I. D.; TANSINI, I. L.; VIERA, I. O.; "New assignment algorithms for the multi-depot vehicle routing problem", (...), *opus cit.*, págs. :977–984.

El método de búsqueda local desarrollado en el algoritmo IVNDS está basado en una variante de la conocida metaheurística de búsqueda por entornos variables (Variable Neighbourhood Search, VNS) Hansen et al.²³.

Dada una solución inicial válida, esta metaheurística explora sus diversas estructuras de vecindarios, entendiendo vecindario como el rango de soluciones al que se puede acceder aplicando una transformación específica en la solución inicial. De este modo, cuando se alcanza un mínimo local con respecto a un determinado vecindario, se procede a explorar el siguiente, y así sucesivamente. Si el mínimo local obtenido en un vecindario no supone una mejora, el proceso se repetirá con el siguiente vecindario. Si por el contrario se produce una mejora, el proceso se reiniciará desde el primer vecindario.

La variante implementada en el algoritmo IVNDS se denomina búsqueda por entornos descendente (Variable Neighbourhood Descent, VND) y la diferencia con respecto al VNS clásico es que en cada vecindario se realiza la transformación que genere una solución con menor coste, en vez de realizar una aleatoria a la que aplicar un proceso de mejora. En nuestro caso, la búsqueda termina una vez se exploran todos los vecindarios sin obtener mejora. El mínimo local obtenido una vez finalizado el proceso, resulta ser el mínimo local con respecto a todas las estructuras de vecindarios. Se puede observar el pseudocódigo del proceso en el Algoritmo 3.

Algoritmo 3 Algoritmo VND

```
1: Inicialización
2: Seleccionar el conjunto de estructuras de entornos  $N_k, k = 1, \dots, k_{max}$ 
3: repetir
4:    $k \leftarrow 1$ 
5:   mientras  $k! = k_{max}$  hacer
6:      $x' = \text{BúsquedaLocal}(x, k)$  // Encontrar la mejor solución  $x'$  del  $k$ -ésimo
       entorno de  $x$ 
7:     si  $f(x') < f(x)$  entonces
8:        $x \leftarrow x'$ 
9:        $k \leftarrow 1$ 
10:    si no
11:       $k \leftarrow k + 1$ 
12:    fin si
13:  fin mientras
14: hasta que no se obtenga mejora
```

A. Estructuras de vecindarios

²³HANSEN, P.; MLADENOVIC, N.; PEREZ, M; "Variable neighbourhood search: methods and applications", Annals of Operations Research, Springer US, 175, 2010, págs. 367-407

Como se ha mencionado en el apartado anterior, cada vecindario queda definido por la transformación que aplica a la solución, y contiene el conjunto de las posibles soluciones que se pueden alcanzar aplicando esta transformación a una solución inicial. A continuación se hace referencia a los diferentes vecindarios de búsqueda implementados según la transformación que aplican a la solución. Están basados en las transformaciones λ y λ -opt desarrollados por Lin et al.²⁴:

- 1) **One Point Move**: Varía la posición en la solución de un único cliente.
- 2) **Two Point Move**: Intercambia la posición de dos clientes.
- 3) **Three Point Move**: Intercambia posiciones entre un cliente y otros dos clientes consecutivos.
- 4) **Two OPT**: Elimina dos arcos de la solución (conexión entre dos clientes consecutivos en una misma ruta) y evalúa las posibles reorganizaciones de los clientes.
- 5) **Three OPT**: Elimina tres arcos de la solución y evalúa las posibles reorganizaciones de los clientes.
- 6) **Or-OPT**: Inserta una cadena de clientes consecutivos en una nueva posición en la solución.
- 7) **Cross Exchange**: Eliminando dos arcos en dos rutas diferentes, recombinando los clientes.

4. CRITERIO DE ACEPTACIÓN

El criterio de aceptación del algoritmo determina si la solución generada por la búsqueda local se utilizará en iteraciones sucesivas. A su vez, también regula la importancia de las fases de intensificación y diversificación del proceso de búsqueda. En el caso del algoritmo IVNDS, se mantiene en todo momento constancia de la mejor solución encontrada (líneas 9, 10 del Algoritmo 2), y en cada caso se selecciona la solución que se mantendrá en la siguiente iteración según un parámetro Ω (13, 14).

Una primera fase favorece la diversificación de la búsqueda, manteniendo la solución generada por el proceso de búsqueda local, sin tener en cuenta la calidad de la misma. La segunda fase, relativa a la intensificación, recupera la mejor solución obtenida hasta el momento, centrando los esfuerzos del proceso de búsqueda a la región más prometedora del espacio de soluciones.

Durante las primeras Ω iteraciones, la búsqueda explora progresivamente espacios de la solución más amplios, tratando la misma solución. Tras Ω iteraciones sin ninguna mejora del coste global, se restaura la mejor solución S^* asegurando la intensificación de la búsqueda en regiones prometedoras del espacio de soluciones, aquellas que circundan la mejor solución alcanzada hasta el momento.

²⁴ LIN, S; "Computer solutions of the traveling salesman problem", Bell System Technical Journal, 44(10), 1965, 2245-2269.

El número de iteraciones realizadas por el algoritmo se reinicia cada vez que la búsqueda local mejora el fitness de la mejor solución encontrada, por lo que las fases de intensificación y diversificación se alternan varias veces a lo largo de todo el proceso, antes de que se cumpla la condición de parada una vez el número de iteraciones iguala un determinado parámetro η .

5. PERTURBACIÓN

Como medio para evitar el mínimo local que ha absorbido a una solución y favorecer la diversificación de la búsqueda, se aplica un mecanismo de perturbación. En este caso, para facilitar la evasión de mínimos, se implementaron 4 métodos de perturbación, seleccionándose uno de ellos de manera aleatoria en cada iteración del algoritmo. Tres de estos métodos realizan transformaciones locales en un único depósito (intra-depósito), y un cuarto aplica transformaciones en dos o más depósitos (inter-depósito). Durante la experimentación inicial se observó que el algoritmo actuaba de manera cíclica a lo largo de la búsqueda, ya que el proceso de búsqueda local deshacía las transformaciones realizadas en la fase de perturbación. Para impedirlo, se decidió reforzar las perturbaciones de manera que se prohibiera revertir sus modificaciones durante un cierto número de iteraciones.

Un factor importante en este proceso es el de la violencia de la perturbación, representada en este caso mediante un parámetro ϕ . Un valor reducido de ϕ podría no ser suficiente para escapar al mínimo local, fallando por lo tanto en la diversificación de la búsqueda. Si el valor es demasiado alto, la perturbación deteriorará la solución de manera que las buenas características de la solución se perderán, degenerando la búsqueda local iterada en un método de reinicio aleatorio.

En el siguiente apartado se presenta una breve descripción sobre los métodos de perturbación desarrollados, denominados swap, insert, exchange y insert MD.

B. Perturbación intra-depósito

Swap: Selecciona dos clientes asignados a un mismo depósito e intercambia sus posiciones. El primer cliente es seleccionado de manera aleatoria, mientras que el segundo es seleccionado de entre la lista de vecinos del primero. Además, el segundo cliente debe pertenecer a una ruta diferente a la del primero. Este proceso se repite ϕ veces.

Insert: Selecciona un cliente aleatorio y $\phi-1$ clientes vecinos del primero. Estos clientes son extraídos de la solución y reinsertados, por orden de extracción, en la posición que represente menor coste en la solución.

Exchange: Intercambia una cadena de clientes perteneciente a una ruta con otra cadena de clientes perteneciente a una segunda ruta. El tamaño de cada cadena es generado aleatoriamente entre 2 y ϕ .

C. *Perturbación inter-depósito*

Una de las aportaciones del presente estudio es el del desarrollo de un nuevo método de perturbación cuyas modificaciones afectan a más de un depósito. Está basado en una estructura de inserción y eyección que cambia los clientes asignados a cada depósito. En primer lugar se selecciona un depósito de origen de manera aleatoria, y un depósito de destino por proximidad al primero. A continuación se construye una lista que contiene, para cada cliente en el depósito origen, el número de vecinos que tiene en el depósito destino. Esto determinará la probabilidad de inserción de cada uno de los clientes, que será más alta en el caso de tener un gran número de vecinos en el depósito de destino. La reinsertión de los clientes en otra posición dentro de la solución se realiza de manera que se minimiza el coste. Esta estrategia está diseñada de manera que la búsqueda evolucione hacia regiones del espacio de soluciones más prometedoras.

6. LISTA DE VECINOS

Ya que evaluar cada una de las posibles combinaciones de clientes está fuera de alcance en términos computacionales, cada cliente en la solución presenta dos listas de vecinos que restringe esta búsqueda. La primera de ellas, llamada la lista global, almacena los α clientes más cercanos en la solución, siendo α un parámetro que define el tamaño de la lista. La segunda lista de vecinos o lista local, un subconjunto de la global, contiene los clientes más cercanos que además están asignados al mismo depósito. La lista local es utilizada en todos los métodos que implican modificaciones locales de un único depósito, es decir, optimización intra-depósito y los métodos de perturbación. La lista de vecinos global es necesaria en modificaciones inter-depósito. Esta división de listas reduce el tiempo de computación ya que no es necesario evaluar relaciones con vecinos globales en modificaciones locales.

El tamaño de la lista global, α , restringe la profundidad de la exploración. Un gran α implicará un alto coste computacional ya que se realizará gran parte del conjunto de posibles evaluaciones. Podría parecer que esto conduciría la búsqueda hacia mejores resultados que un menor valor de α pero esto no es necesariamente cierto. Ya que los vecinos son seleccionados según proximidad a un cliente, una lista de vecinos de gran tamaño contendría clientes lejanos cuya distancia podría implicar violaciones de longitud máxima de ruta. No se permiten soluciones infactibles, por lo que las evaluaciones de posibles modificaciones que resultaran en violaciones de restricciones no se realizarían y el tiempo consumido en la evaluación se perdería. Por el contrario, un valor de α demasiado bajo podría dejar de realizar modificaciones que mejorarán la solución global. Por lo tanto, es necesario encontrar un balance entre un valor alto de α y uno bajo.

V. EXPERIMENTACIÓN

El algoritmo IVNDS fue probado en las instancias del MDVRP presentadas por Cordeau et al.²⁵, que contienen 23 instancias de varios tamaños, desde 50 a 360 clientes. Sus resultados han sido comparados con los obtenidos por los mejores métodos en la actualidad: la búsqueda tabú de Cordeau et al.²⁶, el ALNS de Pisinger et al.²⁷ y el algoritmo evolutivo presentado por Vidal et al.²⁸.

Para evitar la aleatoriedad en el proceso, cada experimento fue realizado 10 veces. Las soluciones y tiempos presentados a continuación se corresponden con las mejores soluciones encontradas y el tiempo total de duración del experimento al completo. La experimentación se llevó a cabo en un AMD FXTM6100 Six-Core Processor a 3.30GHz con Windows Server 2008 y fue implementado en C++.

1. AJUSTE DE PARÁMETROS

Como la mayoría de metaheurísticos, IVNDS depende de un conjunto de parámetros correlacionados y opciones de configuración para su correcto funcionamiento. Todos estos parámetros son dependientes de cada instancia, relacionados tanto con el tamaño como la distribución de clientes en cada problema. Tratando de evitar un ajuste específico de parámetros para cada instancia, se realizaron una serie de estudios para explorar las posibles relaciones entre estos parámetros y las características fundamentales de las instancias. Estas experimentaciones nos permitieron fijar estos parámetros sin un empeoramiento destacable en los resultados, simplificando la utilización del algoritmo, uno de los propósitos de este estudio.

Los principales parámetros del algoritmo, mencionados en secciones previas, son: el número de iteraciones (η), que regula la condición de parada del bucle, Ω , utilizado para establecer un balance entre las fases de intensificación y diversificación y el valor asociado a los métodos de perturbación, ϕ , que establece la intensidad de la perturbación.

En relación a la condición de parada y al parámetro η , percibimos que según las instancias contaban con más clientes se requerían más iteraciones para resolverlas de manera eficiente. Las instancias pequeñas podían ser resueltas de manera óptima en pocas iteraciones y todo el proceso posterior a la obtención de la mejor solución conduciría a una pérdida de tiempo y coste computacional. De la manera contraria, instancias con muchos clientes necesitan más iteraciones para mejorar la solución global. La experimentación realizada nos llevó a fijar el parámetro η como dos veces el tamaño del problema, ya que los resultados obtenidos son similares a aquellos con un mayor valor de η y el tiempo requerido se redujo drásticamente.

²⁵ CORDEAU, J. F.; GENDREAU, M.; LAPORTE, G.; “A tabu search heuristic for periodic and multi-depot vehicle routing problems.”, (...), *opus cit*, pág. 4

²⁶ CORDEAU, J. F.; LAPORTE, G.; MERCIER, A.; “A unified tabu search heuristic for vehicle routing problems with time windows”, (...), *opus cit*, pág. 4.

²⁷ PISINGER, D.; ROPKE, S.; “A general heuristic for vehicle routing problems”, (...), *opus cit*, pág. 5

²⁸ VIDAL, T.; CRAINIC, T. G.; GENDREAU, M.; LAHRICHI, N.; REI, W.; “A hybrid genetic algorithm for multidepot and periodic vehicle routing problems”, (...), *opus cit*, pág. 5.

Fijamos Ω como el tamaño del problema dividido por 10. Aplicado en la experimentación este valor demostró dar buenos resultados y un buen balance entre las fases de intensificación y diversificación a lo largo de la búsqueda. Su valor necesita ser lo suficientemente grande como para garantizar una aceptable dispersión en la exploración de soluciones, pero no demasiado alto con respecto a la fase de intensificación, centrada en una exploración exhaustiva de una región prometedora. Ya que el número de iteraciones se reinicia cada vez que se mejora la mejor solución global alcanzada, la fase de diversificación se aplica varias veces a lo largo del proceso.

La intensidad de los métodos de perturbación está definida por un parámetro ϕ . Una vez más, se debe encontrar un balance entre un valor alto que desvíe la búsqueda realizando modificaciones violentas en la solución, y un valor bajo que no permita evitar el mínimo local alcanzado en la fase de mejora. Aunque los métodos de perturbación están implementados de manera que realicen modificaciones posiblemente beneficiosas para la solución, el factor aleatorio que integran podría degenerar la búsqueda en un método de reinicio aleatorio. Decidimos fijar ϕ como la raíz cuadrada del tamaño del problema.

2. RESULTADOS

El primer grupo de columnas (1-4) en la Tabla 1 muestra el identificador de la instancia, el número de clientes, el máximo número de vehículos y el número de depósitos, respectivamente. A continuación se muestra la mejor solución conocida para cada una de las instancias y la mejor solución obtenida por nuestro algoritmo, IVNDS. Las soluciones mostradas en la tabla son las mejores obtenidas de entre las 10 repeticiones realizadas y el tiempo representa el total empleado en la experimentación de cada instancia. La última fila muestra la media del error cometido por el algoritmo con respecto a la mejor solución conocida.

Inst	N	V	D	BKS	VID	PS	CGL	IVNDS	Error	t(10 rep)
p01	50	4	4	576.87	576.87	576.87	576.87	576.87	0.00	0.12
p02	50	2	4	473.53	473.53	473.53	473.87	477.95	0.93	0.1
p03	75	3	2	641.19	641.19	641.19	645.15	641.19	0.00	0.33
p04	100	8	2	1001.04	1001.04	1001.04	1006.66	1010.24	0.92	0.54
p05	100	5	2	750.03	750.03	751.26	753.34	765.27	2.03	0.39
p06	100	6	3	876.5	876.5	876.7	877.84	880.69	0.48	0.39
p07	100	4	4	881.97	881.97	881.97	891.95	892.70	1.22	0.49
p08	249	14	2	4372.78	4372.78	4390.8	4482.44	4414.43	0.95	9.18
p09	249	8	3	3858.66	3858.66	3873.64	3920.85	3909.11	1.31	6.19
p10	249	12	4	3631.11	3631.11	3650.04	3714.65	3750.43	3.29	8.02
p11	249	6	5	3546.06	3546.06	3546.06	3580.84	3627.62	2.30	10.68
p12	80	5	2	1318.95	1318.95	1318.95	1318.95	1318.95	0.00	0.2
p13	80	5	2	1318.95	1318.95	1318.95	1318.95	1329.70	0.82	0.13
p14	80	5	2	1360.12	1360.12	1360.12	1360.12	1360.12	0.00	0.25
p15	160	5	4	2505.42	2505.42	2505.42	2534.13	2526.06	0.82	0.98
p16	160	5	4	2572.23	2572.23	2572.23	2572.23	2572.23	0.00	0.9
p17	160	5	4	2709.09	2709.09	2709.09	2720.23	2742.80	1.24	0.61
p18	240	5	6	3702.85	3702.85	3702.85	3710.49	3783.02	2.17	6.67
p19	240	5	6	3827.06	3827.06	3827.06	3827.06	3842.46	0.40	1.96
p20	240	5	6	4058.07	4058.07	4058.07	4058.07	4102.80	1.10	2.81
p21	360	5	9	5474.84	5474.84	5474.84	5535.99	5604.69	2.37	42.02
p22	360	5	9	5702.16	5702.16	5702.16	5716.01	5733.22	0.55	11.94
p23	360	5	9	6078.75	6078.75	6078.75	6139.73	6225.41	2.41	7.63
Mean					0	0.07	0.63		1.10	4.9

Table 1: Resultados instancias MDVRP

VI. CONCLUSIONES

Se ha presentado el método IVNDS, un algoritmo capaz de resolver el VRP multi-depósito, una variante no tan explorada de este extenso campo de optimización, con un particular interés en su sencilla y directa aplicación e implementación general. Se han discutido los aspectos relevantes de su estructura y de su naturaleza híbrida, como conjunción de otros conocidos heurísticos como la búsqueda local iterada (ILS) y la búsqueda descendente por entornos variables (VNDS). Una de las principales desventajas de los métodos metaheurísticos es la gran cantidad de parámetros que se deben configurar antes de la ejecución del algoritmo. En el desarrollo del IVNDS, se ha tratado de minimizar el número de parámetros utilizado, además de desarrollar un análisis posterior en busca de relaciones entre los mismos y características particulares de los problemas. Esto condujo a la inicialización de estos parámetros en el propio algoritmo, con una reducida pérdida en términos de los resultados finales y un reducido decremento del tiempo total empleado en la ejecución. Finalmente, se ha realizado una experimentación para justificar el correcto funcionamiento del algoritmo comparando nuestros resultados con las mejores soluciones conocidas hasta la fecha, obteniendo destacables resultados teniendo en cuenta la sencillez del algoritmo y su aplicación directa con un mínimo ajuste de parámetros.

VII. BIBLIOGRAFÍA

ARANGO, M. D.; ADARME, W.; ZAPATA, J.A.; “Commodities distribution using alternative types of transport”, Colombia bread SME’s. DYNA, 163, 2010, págs. 222-233

BALDACCI, R.; MINGOZZI, A.; ”A unified exact method for solving different classes of vehicle routing problems”. Mathematical Programming, 120(2), 2009, págs. 347–380

CORDEAU, J. F.; GENDREAU, M.; LAPORTE, G.; “A tabu search heuristic for periodic and multi-depot vehicle routing problems.”, Networks, 30(2), 1997, págs 105–119.

CORDEAU, J. F.; LAPORTE, G.; MERCIER, A; “A unified tabu search heuristic for vehicle routing problems with time windows”, Journal of the Operational Research Society, 52(8), 2001, págs 928– 936.

CORDEAU, J-F.; MAISCHBERGER, M.; ”A parallel iterated tabu search heuristic for vehicle routing problems”, Computers & Operations Research, 39(9), 2012, págs. 2033 – 2050.

DOCUMENTO DE TRABAJO; “Vigilancia tecnológica: Técnicas para la Optimización de Rutas de Transporte y Distribución”. Brain Trust Consulting Services Rev02, 2009 .
http://www.odette.es/SGC/downloads/CAM/Vigilancia_Tecnologica_Tecnicas_Optimizacion_Rutas.pdf.

GIOSA, I. D.; TANSINI, I. L.; VIERA, I. O.; “New assignment algorithms for the multi-depot vehicle routing problem”, Journal of the Operational Research Society, 53(9), 2002, págs. :977–984.

HANSEN, P.; MLADENOVIC, N.; PEREZ. M; “Variable neighbourhood search: methods and applications”, Annals of Operations Research, Springer US, 175, 2010, págs. 367-407

HO, W.; HO, G; JI, P.; LAU, H; “A hybrid genetic algorithm for the multi-depot vehicle routing problem.” Engineering Applications of Artificial Intelligence, 21(4), 2008, :548-557.

KUO, Y.; WANG, C-C.; “A variable neighborhood search for the multi-depot vehicle routing problem with loading cost”, Expert Systems with Applications, 39(8), 2012, págs. 6949-6954

LAPORTE, G.; NOBERT, Y.; ARPIN, D.; ”Optimal solutions to capacitated multidepot vehicle routing problems”, École des hautes études commerciales, 1984

LAPORTE, G.; NOBERT, Y.; TAILLEFER, S.; "Solving a family of multidepot vehicle routing and location-routing problems", *Transportation Science*, 22(3), 1988, págs.161–172

LAU, H.; CHAN, T.; TSUI, W. T.; PANG, W. K.; "Application of genetic algorithms to solve the multidepot vehicle routing problem". *Automation Science and Engineering, IEEE Transactions on*, 7(2), 2010, págs. 383–392

LIN, S.; "Computer solutions of the traveling salesman problem", *Bell System Technical Journal*, 44(10), 1965, 2245-2269.

LOURENCO, HR., OLIVIER, M.; STÜTZLE, T.; "Iterated Local Search", *The Handbook of Metaheuristics*, (ed. GLOVER), 2003, págs 320-353

MIRABI, M.; FATEMI GHOMI, S. M. T.; JOLAI, F.; "Efficient stochastic hybrid heuristics for the multi-depot vehicle routing problem", *Robotics and Computer-Integrated Manufacturing*, 26(6), 2010, págs. 564 – 569

PISINGER, D.; ROPKE, S.; "A general heuristic for vehicle routing problems", *Computers & Operations Research*, 34(8), 2007, págs. 2403-2435.

POLACEK, M.; F. HARTL, R.; DOEMER, K.; REIMANN, M.; "A variable neighborhood search for the multi depot vehicle routing problem with time windows", *Journal of Heuristics*, 10(6), 2004, págs 613-627

RENAUD, J.; LAPORTE, G.; FOUAD BOCTOR, F.; "A tabu search heuristic for the multi-depot vehicle routing problem" *Computers & Operations Research*, 23, 1996. Págs 229-235

VIDAL, T.; CRAINIC, T. G.; GENDREAU, M.; LAHRICHI, N.; REI, W.; "A hybrid genetic algorithm for multidepot and periodic vehicle routing problems", *Operations Research*, 60(3), 2012, págs. 611–624.

YU, B.; YANG, Z-Z.; XIE, J-X; "A parallel improved ant colony optimization for multi-depot vehicle routing problem", *Journal of the Operational Research Society*, 62(1), 2010, págs.183–188

YU, B.; YANG, Z-Z; BAOZHEN, Y.; "An improved ant colony optimization for vehicle routing problem", *European Journal of Operational Research*, 196(1), 2009, págs.171 – 176.