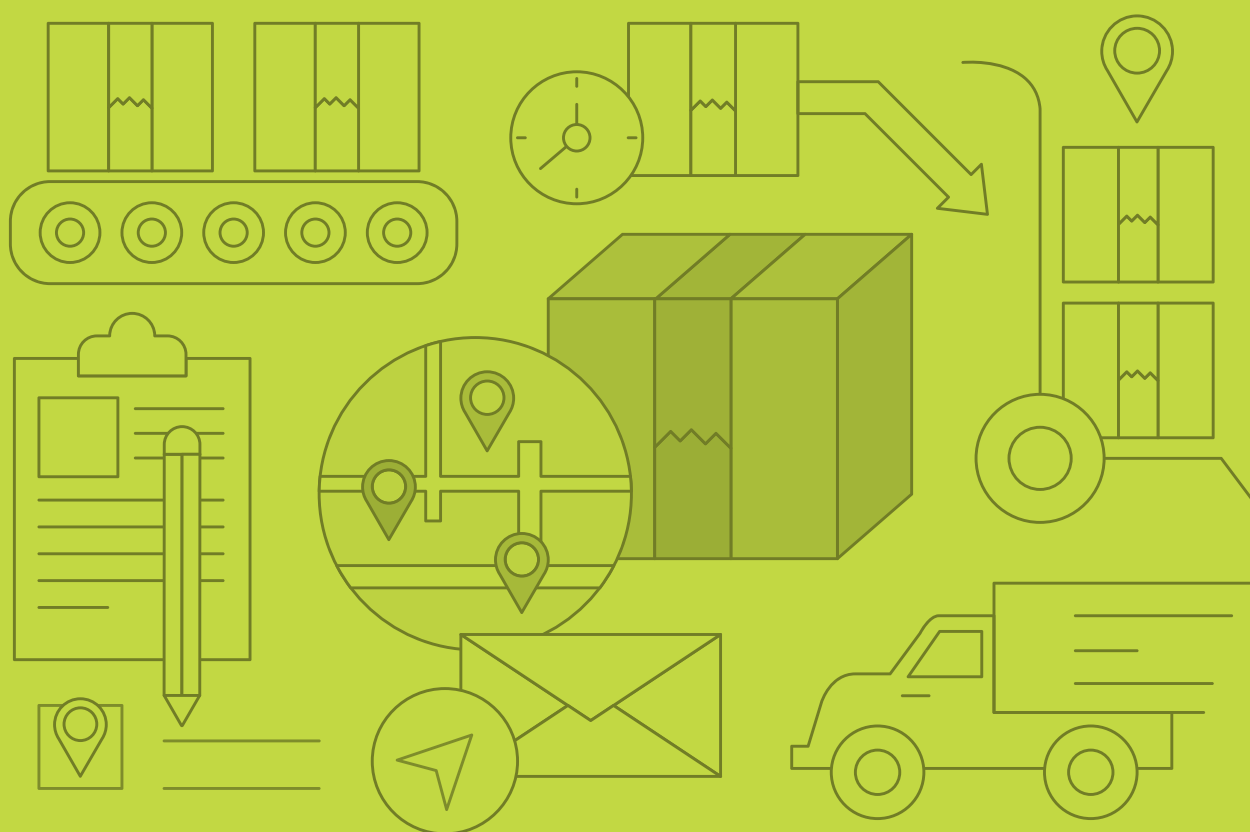


FABIO M. MIGUEL / MARIANO FRUTOS
FERNANDO A. TOHMÉ / MÁXIMO MÉNDEZ BABEY

Ruteo de vehículos y programación de cargas

en la distribución urbana de mercaderías
con introducción de preferencias parciales
en el proceso de toma de decisiones



Serie EXTENSIÓN
Colección CIENCIAS Y TECNOLOGÍA

Ruteo de vehículos y programación de cargas en la distribución urbana de mercaderías con introducción de preferencias parciales en el proceso de toma de decisiones / Fabio M. Miguel ... [et al.]. -1ª ed. - Bahía Blanca: Editorial de la Universidad Nacional del Sur. Ediuns, 2018.

Libro digital, PDF

Archivo Digital: descarga y online

ISBN 978-987-655-187-8

1. Vehículos. 2. Circuitos de Distribución. I. Miguel, Fabio M.

CDD 658.84



Editorial de la Universidad Nacional del Sur

Santiago del Estero 639 - B8000HZK - Bahía Blanca - Argentina

Tel.: 54-0291-4595173 / Fax: 54-0291-4562499

www.ediuns.uns.edu.ar | ediuns@uns.edu.ar



**Libro
Universitario
Argentino**



**Red de Editoriales de
Universidades Nacionales**

Diagramación interior y tapa: Fabián Luzi

No se permite la reproducción parcial o total, el alquiler, la transmisión o la transformación de este libro, en cualquier forma o por cualquier medio, sea electrónico o mecánico, mediante fotocopias, digitalización u otros métodos, sin el permiso previo y escrito del editor. Su infracción está penada por las Leyes 11723 y 25446.

Queda hecho el depósito que establece la Ley 11723.

Bahía Blanca, Argentina, abril de 2018.

© 2018 Ediuns.

ISBN 978-987-655-187-8



9 789876 551878

RESUMEN

Este libro se enmarca en el campo de las Operaciones, un área de las Ciencias de la Administración. El objetivo es el desarrollo y la validación de herramientas tecnológicas para la toma de decisiones en la rama logística de la gestión de la cadena de suministro de la industria alimentaria. Más específicamente, presentamos un novedoso procedimiento inteligente híbrido que ayuda a los responsables de la toma de decisiones a optimizar en un entorno multi-objetivo. Examinamos los modelos habituales de la distribución física de los bienes, clasificándolos según sus principales características. También presentamos los algoritmos evolutivos multi-objetivo que generalmente brindan las soluciones a esos modelos. Nuestro enfoque es introducir una nueva variante multi-objetivo del problema de distribución de bienes en un área urbana. Para modelar la red de centros de distribución, al servicio de los puntos de venta finales de productos, agregamos franjas horarias y otros requisitos. Consideramos la dependencia temporal de los programas óptimos de distribución, a diferencia del caso de los sistemas logísticos de media y larga distancia, para los cuales la distancia es el criterio clave. También agregamos el objetivo de equilibrar cargas entre las diferentes unidades operativas. Una hibridación del algoritmo evolutivo multi-objetivo NSGA-II es nuestra elección de herramienta computacional, junto con el concepto de g-dominación para preferencias parciales, que proporciona la guía informativa en el espacio de búsqueda. Las fases de validación y prueba de este algoritmo utilizan datos del mundo real, comparando sus resultados con los resultados de otros procedimientos evolutivos multi-objetivo utilizados para la solución de problemas complejos de distribución. La información fue proporcionada por un operador logístico, especializado en el transporte y la distribución de cargas fraccionarias. En todos los casos examinados, nuestro algoritmo se desempeñó mejor que los habituales.

ABSTRACT

This book is framed in the field of Operations, an area of Management Science. The goal is the development and validation of technological tools for decision-making in the logistic branch of supply chain management of the food industry. More specifically, we present a novel hybrid intelligent procedure aiding decision-makers optimizing in a multi-objective environment. We examine the usual models of the physical distribution of goods, classifying them according to their main features. We also present the multi-objective evolutionary algorithms that usually yield the solutions to those models. Our approach is to introduce a new multi-objective variant of the distribution problem of goods in an urban area. To model the network of distribution centers, serving the final outlets of goods, we add time frames and other requirements. We consider the time dependence of the optimal programs of distribution, unlike the case of medium and long-distance logistical systems, for which the distance is the key criterion. We also add the objective of balancing loads among the different operating units. A hybridation of the multi-objective evolutionary algorithm NSGA-II is our choice of computational tool, jointly with the concept of g-dominance for partial preferences, which provides the informational guide in the search space. The validation and testing phases of this algorithm uses real-world data, comparing its outcomes to the results of other evolutionary multi-objective procedures used for the solution of complex problems of distribution. The information was provided by a logistic operator, specialized in transporting and distributing fractional loads. In all the cases examined, our algorithm performed better than the usual ones.

ÍNDICE GENERAL

Capítulo I. Introducción	1
1. Problema de Investigación	1
1.1. Introducción.....	1
1.2. Problema objeto de estudio	6
2. Metas, propósitos y objetivos.....	6
2.1. Nivel y tipo de investigación	6
2.2. Objetivos definidos por su alcance.....	7
3. Metodología.....	7
4. Aportes esperados de la investigación.....	9
Capítulo II. Marco Teórico.....	10
1. Enfoque de la gestión de la cadena de suministros	11
1.1. <i>Capacitated Vehicle Routing Problem (CVRP)</i>	15
1.2. <i>Capacitated Vehicle Routing Problem with Time Windows (CVRP-TW)</i> .	15
1.3. <i>Open Capacitated Vehicle Routing Problem (O-CVRP)</i>	15
1.4. <i>Heterogeneous Fleet Capacitated Vehicle Routing Problem (HF-CVRP)</i> ...	16
1.5. <i>Multi Depot Capacitated Vehicle Routing Problem (MD-CVRP)</i>	16
1.6. <i>Capacitated Vehicle Routing Problem with Backhauls (CVRP-B)</i>	16
1.7. <i>Periodic Capacitated Vehicle Routing Problem (P-CVRP)</i>	17
1.8. <i>Time Dependent Capacitated Vehicle Routing Problem (TD -CVRP)</i>	17
1.9. <i>Truck and Trailer Routing Problem (TTRP)</i>	17
2. Tecnología y Dirección de Operaciones.....	21
2.1. Tecnologías de la información	21
2.2. Sistemas de información	23
3. Inteligencia Computacional.....	28
3.1. Introducción.....	29
3.2. Inteligencia Computacional	34
3.3. Computación Evolutiva.....	35
4. Toma de Decisión Multi-objetivo	45
4.1. Introducción.....	45
4.2. Optimización.....	48
4.3. Toma de Decisión	53
5. Métodos de resolución.....	57
5.1. VEGA	59
5.2. MOGA	60

5.3. NSGA	62
5.4. NPGA	64
5.5. SPEA	65
5.6. SPEA 2	65
5.7. NSGA-II	67
Capítulo III. Modelado	69
1. Introducción	69
1.1. Contexto operativo del problema	69
1.2. Objetivos perseguidos por el decisor	71
2. Modelado del MO-MDTDCVRPTW	74
2.1. Variables Binarias	75
2.2. Variables Continuas	75
2.3. Funciones Objetivo:	76
2.4. Restricciones:	77
3. Complejidad Computacional.....	79
Capítulo IV. Resolución e implementación	84
1. Optimizador Propuesto	84
1.1. Introducción.....	84
1.2. Estructura NSGA-II	85
1.3. Mejoras propuestas.....	90
2. Validación y testeo	100
2.1. Caracterización de la red de distribución	101
2.2. Caracterización del problema de planificación modelado.....	102
2.3. Experiencia y evaluación de resultados	105
Capítulo V. Conclusiones y trabajo futuro	119
Anexo	123
Referencias	126

ÍNDICE DE FIGURAS

Figura 1: Variantes de problemas de ruteo de vehículos.....	14
Figura 2: Ejemplo red de distribución urbana simple	18
Figura 3: Sistemas de Información en la SCM.....	26
Figura 4: Herramientas tecnológicas de ayuda a la toma de decisiones	27
Figura 5: Posicionamiento de los AG en las CC	28
Figura 6: Problema simple de ruteo	32
Figura 7: Posicionamiento de la Computación Evolutiva	35

Figura 8: Diagrama de flujo de un AE básico	39
Figura 9: Proceso de toma de decisiones	47
Figura 10: Representación del proceso de mapeo	51
Figura 11: Regiones de dominancia a partir de una solución determinada	52
Figura 12: Representación frontera de Pareto	53
Figura 13: Esquema de Funcionamiento de VEGA.....	59
Figura 14: Asignación de rangos MOGA.....	60
Figura 15: Asignación de rangos NSGA.....	62
Figura 16: Diagrama NSGA.....	63
Figura 17: Diagrama SPEA 2 (Zitzler, Laumanns, & Thiele, 2001)	66
Figura 18: Problema compuesto.....	70
Figura 19: Ordenamiento no dominado de NSGA-II.....	85
Figura 20: Distancia Crowding en NSGA-II	88
Figura 21: Mecanismo de promoción de individuos NSGA-II	90
Figura 22: Esquema del Cromosoma.....	92
Figura 23: Esquema del Operador de Recombinación (ERX-MD)	94
Figura 24: Ejemplo simplificado Re-clustering.....	95
Figura 25: Sub-grafos de cluster 1 y 2 según la asignación del nodo 6.....	96
Figura 26: Concepto de <i>g-dominancia</i>	98
Figura 27: Evolución generacional con <i>g-dominancia</i>	99
Figura 28: Niveles de distribución.....	101
Figura 29: Hipervolumen H del Frente de Pareto A	105
Figura 30: Punto de referencia superior izquierdo, $G1$	108
Figura 31: Punto de referencia central, $G2$	108
Figura 32: Punto de referencia inferior derecho, $G3$	109
Figura 33: Frentes aproximados para tres puntos de referencia	110
Figura 34: Solución A en representación 2D. Pto. Ref.: (440; 1300)	113
Figura 35: Solución A en representación 3D. Pto. Ref.: (440; 1300)	114
Figura 36: Solución B en representación 2D. Pto. Ref.: (450; 1150)	114
Figura 37: Solución B en representación 3D. Pto. Ref.: (450; 1150)	115
Figura 38: Solución C en representación 2D. Pto. Ref.: (500; 1000)	115
Figura 39: Solución C en representación 3D. Pto. Ref.: (500; 1000)	116
Figura 40: Punto de referencia interior	117

CAPITULO I. INTRODUCCIÓN

1. PROBLEMA DE INVESTIGACIÓN

1.1. Introducción

En las últimas décadas hemos vivido un acelerado cambio tecnológico liderado por las *tecnologías de la información* (TI) que ha modificado el mundo moderno. Este cambio tecnológico ha generado un significativo aumento de la velocidad de transferencia, la facilidad de acceso y la exactitud de la información. El permanente aumento de la conectividad inter-organizacional ha permitido generar procesos y operaciones que hace algunos años eran impensables en cuanto a la integración de la cadena de suministros, y a la veracidad, calidad y disponibilidad de la información.

Paralelamente, las prácticas empresariales han evolucionado hacia un nuevo orden de relaciones caracterizado por la cooperación entre empresas que persiguen un objetivo estratégico común, obtener una ventaja competitiva valiosa y difícil de imitar por parte de los competidores. Este esfuerzo cooperativo, surge por la necesidad práctica de adaptarse a los cambios tecnológicos que se daban en el entorno empresarial. En este sentido, la práctica tradicional de canales de distribución fragmentados y con elevados inventarios intermedios, dejó de ser suficiente para sostener una ventaja competitiva si las empresas relacionadas no estaban alineadas con un objetivo estratégico común (Bowersox, Closs, & Cooper, 2007) (Ballou, 2004). La estrategia empresarial de desempeñar una función específica del proceso de distribución de mercaderías, de forma independiente en busca de objetivos estratégicos individuales, se volvió rápidamente obsoleta y poco competitiva en este nuevo milenio.

La tecnología hoy permite una mayor disponibilidad de información de buena calidad, generando cambios profundos en procesos de transformación enfocados en los clientes, que requieren mayor precisión, flexibilidad y menores tiempos de distribución para ser competitivos. Y paralelamente, también ha generado importantes cambios en los mercados, con clientes que exigen un nivel de servicio superior y una mayor variedad de productos que satisfacen las mismas necesidades. Posibili-

tando además un significativo crecimiento del comercio electrónico, que impactó en gran medida sobre la estructura y el desempeño de la distribución urbana de mercaderías. Aumentando los costos y la cantidad de transporte requeridos para realizar la distribución de mercaderías en entornos urbanos, en comparación con las estructuras tradicionales de redes de distribución, con muchos puntos de venta y de inventario, distribuidos en varios escalones a lo largo de la cadena de suministro (Chopra & Meindl, 2008).

Entre los cambios más importantes en los eslabones de las cadenas de suministros más cercanos al consumidor final, se destaca en primer lugar, la reducción de niveles de stock en destino. El control de la demanda ha permitido al comercio tanto minorista como mayorista, ajustar sus pedidos de forma que los costos de inventario disminuyan sustancialmente, pero a costa de un aumento del número de envíos realizados y, por tanto, un incremento de la cantidad de operaciones de carga y descarga. Y en segundo lugar, el incremento de la cantidad de desplazamientos ocasionados por el mayor número de referencias¹. Esto es explicado por el aumento en la variedad de las preferencias de los consumidores, que implicó la aparición de una multitud de productos dentro de una misma línea en diferentes combinaciones de tamaño, ingredientes, colores, texturas, etc. Así por ejemplo, para un tipo de yogurt determinado, existen varias combinaciones posibles de sabores, tamaños, formatos, productos complementarios, etc.

Es importante destacar que la distribución urbana de mercaderías, como último eslabón de la cadena de suministros, impacta directamente sobre las posibilidades de desarrollo económico de cualquier área urbana, especialmente en las que predominen actividades industriales y de servicios. Hay pocas actividades en una ciudad que no requieran por lo menos de alguna mercadería que deba ser movida o trasladada. Al mismo tiempo, la distribución urbana de mercaderías es una fuente importante de empleo (Crainic, Ricciardi, & Storchi, 2004).

Además, es una de las principales causales de la congestión del tránsito en dichas áreas. Es decir, afecta significativamente al resto del transporte urbano, a las velo-

¹ SKU (stock-keeping unit), UPC (Universal Product Code) o EAN (European Article Number).

tidades del tráfico y a las emisiones de contaminación ambiental, generando múltiples efectos negativos sobre las actividades y la calidad de vida de las personas.

Todos estos factores se han potenciado, por los cambios en la estructura y el desempeño de las redes de distribución urbana de mercaderías, donde cualquier usuario (no solo los comercios minoristas) es un destino potencial. Dándose así un considerable aumento del número de desplazamientos realizados por los vehículos, tanto llenos como vacíos, y el consecuente aumento en la congestión del tránsito, generando los efectos negativos antes mencionados, como así también un aumento de los costos asociados a la distribución urbana de mercaderías, que impacta significativamente en los costos finales del producto, determinando finalmente la competitividad empresarial y de la cadena de suministros como un todo.

Como efecto principal que se desprende de la exposición anterior, cabe destacar la creciente complejidad en el proceso logístico de distribución urbana de mercaderías, y sus efectos externos negativos sobre las actividades y la calidad de vida de las personas. Estos efectos, convierten a este proceso de la cadena de suministro, en un área en la que es indispensable mejorar la calidad y la capacidad de las herramientas de apoyo a la toma de decisiones, con el objeto de aumentar su eficiencia, productividad y capacidad de adaptación a las actuales condiciones urbanas en las que debe desarrollarse dicho proceso.

En este contexto, una aplicación imprescindible de las tecnologías de la información como herramientas de apoyo a la toma de decisiones, es el sistema de gestión de transporte, que sirve de apoyo a la planeación, ejecución y administración de las actividades relacionadas al transporte y movimiento de mercadería. Este sistema busca en general, minimizar el costo total del movimiento de materiales para la empresa y también otros eslabones de la cadena de suministro (Bowersox, Closs, & Cooper, 2007), identificando y evaluando estrategias alternativas de transporte y tácticas para determinar los mejores métodos para mover los productos con las restricciones existentes (Taylor, 2007). Entre las funciones que habitualmente desempeñan estos sistemas están: la selección de vehículos, la consolidación del

flete, el ruteo y programación de carga, el rastreo de pedidos, la tarifación, etc. (Ballou, 2004).

Actualmente existen en el mercado, diferentes soluciones informáticas de este tipo, la más básica y habitual es el sistema de seguimiento y control satelital de vehículos. Con esa base, se ofrecen adicionalmente una gama de servicios relacionados a la gestión de flotas, el control del manejo de cargas, etc. A los fines del enfoque de la presente propuesta de investigación, es relevante la función de ruteo de vehículos y programación de cargas. Las otras funciones del sistema, están más relacionadas a la seguridad del vehículo y la carga, además del control para mantenimiento preventivo de vehículos y diagnóstico. Mientras que la función principal del ruteo de vehículos y la programación de cargas es buscar que la operación de la flota sea eficiente.

El enfoque de la presente propuesta, no solo busca la eficiencia desde el punto de vista de los costos operativos, como lo hacen la mayoría de las soluciones mencionadas; sino que permite considerar otros factores como, por ejemplo, el balanceo de carga entre operadores, la congestión en entornos urbanos, las restricciones de circulación en ciertas zonas en horarios determinados con vehículos extra dimensionados, la coordinación de programación de las cargas entre diferentes centros de distribución, etc.

Por otro lado, en la gran mayoría de los estudios encontrados en la literatura académica, que estudian este tipo de problemática, hacen simplificaciones importantes. Una de ellas y quizás la más relevante, es la de suponer que solo existe un criterio de decisión (el costo operativo), utilizando un paradigma mono-objetivo para modelar la realidad logística de la distribución urbana de mercaderías, y en base a estos modelos proponer herramientas para la toma de decisiones. Esta simplificación se vuelve una importante limitación para el modelo y puede quitarle utilidad práctica alejándolo demasiado de la realidad que se pretende modelar.

Dada la complejidad actual en la distribución urbana de mercaderías, es indispensable dotar de un mayor realismo al modelado del problema a resolver, y de esta manera, dar una mayor utilidad a la herramienta de apoyo para la toma de decisiones. En este sentido, es indispensable considerar un enfoque con múltiples obje-

tivos (generalmente en conflicto), en la toma de decisiones diaria de los responsables de logística. Por ejemplo, la reducción de costos de transporte, la maximización del nivel de servicio al cliente, el balanceo de carga por ruta, la subutilización de los vehículos o la minimización del tiempo requerido para realizar toda la distribución. Considerar además el efecto de la coordinación entre diferentes puntos de abastecimiento a establecimientos minoristas o clientes finales, sobre la programación de la secuencia de visitas de cada vehículo y finalmente sobre el armado de la carga del mismo de forma óptima. Esto es de suma importancia en redes de distribución urbanas con estructuras con varios depósitos o centros de distribución, que deben operar de forma coordinada.

A medida que la ciencia ha ido profundizando el análisis formal de estos procesos logísticos, y modelando situaciones que contemplen de manera más completa la complejidad de la realidad de la distribución de mercaderías, se ha observado que los problemas de decisión que se presentan, son en su mayoría de una elevada complejidad computacional. Es decir, son problemas para los que no existen algoritmos que los resuelvan en tiempo polinomial, y que requieren de la utilización de técnicas que permitan aproximar una solución utilizando una cantidad razonable de recursos, como las meta-heurísticas.

En este contexto, la computación evolutiva ha demostrado una gran capacidad de adaptación y eficiencia en la resolución de este tipo de problemas con formulaciones más realistas de dichos procesos. Y aunque estas técnicas meta-heurísticas no garantizan la resolución óptima de un problema, pueden producir rápidamente soluciones de calidad aceptable en casos en los que a los algoritmos exactos conocidos aplicables, les llevaría un tiempo superior al de una vida humana resolverlos.

Por otro lado, se observa en resultados obtenidos en varios trabajos (Wolpert & Macready, 1995) (Wolpert & Macready, 1997), que una meta-heurística puede superar a otra en la resolución de un problema, pero cambiando ciertas características del problema sucede lo contrario. De esta manera no es posible realizar la afirmación de que una heurística sea superior a otra de forma general, y cada pro-

blema requiere del desarrollo, adaptación o mejora del algoritmo para poder resolver con mayor eficiencia el problema con toda su complejidad².

1.2. Problema objeto de estudio

El problema consiste en desarrollar una herramienta algorítmica que permita mejorar la eficiencia operativa del proceso de toma de decisiones, en el ámbito de la distribución urbana de mercaderías, desde un enfoque multi-objetivo.

2. METAS, PROPÓSITOS Y OBJETIVOS

2.1. Nivel y tipo de investigación

El nivel de la investigación es predictivo y del tipo aplicado, dado que los resultados obtenidos pretenden aplicarse y/o utilizarse en forma inmediata para resolver un problema concreto.

Esta investigación se desarrolla en el marco de la Tecnología y la Dirección de Operaciones, como campo del conocimiento de las ciencias de la administración. En este contexto, la administración es una ciencia aplicada a la búsqueda de soluciones de problemas concretos, en situaciones específicas dentro de una organización, donde los agentes toman decisiones y desarrollan su comportamiento moldeando procesos que permiten producir y entregar bienes y servicios a clientes internos o externos de la organización. Se vale de inferencias o predicciones para diseñar herramientas o modelos que realicen las funciones deseadas, que se anticipen o adapten a eventos futuros y que sirvan para dar pautas para resolver problemas.

Estas aplicaciones prácticas o tecnológicas, que están sustentadas a su vez, por teorías, métodos y procedimientos específicos de distintas disciplinas, que potencian las capacidades humanas para la toma de decisiones respecto a la secuencia de

² Que no haya una sola técnica de búsqueda heurística que supere a las demás en todos los casos, es la principal implicación del *No Free Lunch Theorem* (Wolpert & Macready, No free lunch theorems for optimization, 1997).

pautas de actuación que permitan resolver problemas concretos en contextos de alta complejidad.

2.2. Objetivos definidos por su alcance

2.2.1. Objetivo general

Teniendo en cuenta las consideraciones precedentes, como objetivo general del presente trabajo, se planteó el desarrollo de una herramienta de ayuda a la toma de decisiones, que mejore la eficiencia de la planificación operativa de vehículos y la programación de cargas, en la distribución urbana de mercaderías, desde un enfoque multi-objetivo.

2.2.2. Objetivos específicos

- Primero, caracterizar la distribución urbana de mercaderías perecederas.
- Segundo, definir formalmente el problema de ruteo de vehículos en el contexto del servicio logístico objeto de la investigación.
- Tercero, modelar el sistema de distribución bajo la estructura del problema de ruteo definido.
- Cuarto, desarrollar una herramienta algorítmica, un Algoritmo evolutivo multi-objetivo, que permita la resolución del problema con una utilización razonable de recursos.
- Quinto, probar de las soluciones mediante información real del proceso en un caso de aplicación específico, y evaluar de las soluciones generadas.

3. METODOLOGÍA

Un método es un procedimiento regular, explícito y repetible para lograr algo, ya sea material o conceptual. *“Los métodos de investigación constituyen el camino para llegar al conocimiento científico; son un procedimiento o conjunto de procedimientos que sirven de instrumento para alcanzar los fines de la investigación”* (Barchini, 2005).

La metodología que se utilizará para cumplir con los objetivos de la presente investigación, es el Estudio de Caso con un enfoque mixto no estructurado, recolec-

tando, analizando e integrando datos cuantitativos y cualitativos, provenientes de diversas fuentes (Hernández Sampieri, Fernández Collado, & Baptista, 2014).

Respecto a la caracterización del contexto y modelización, se recurre principalmente al enfoque cualitativo, donde la unidad de análisis es el proceso de distribución realizado por la empresa, y su contexto.

Mientras que, en cuanto al desarrollo del optimizador y testeos, la unidad de análisis es el proceso de toma de decisiones que sigue el agente responsable de programar la distribución diaria de mercaderías, en el contexto del proceso de distribución antes mencionado. Aquí se recurre principalmente al enfoque cuantitativo, para comparar el desempeño del modelo y la herramienta algorítmica desarrollados con el de otros algoritmos, utilizando estudios pre-experimentales.

Pasos del método propuesto:

- Revisión del marco conceptual en materia de sistemas de distribución y de información en la distribución física de mercaderías. Recopilando información relevante en el ámbito del servicio logístico objeto de estudio.
- Definición formal del problema objeto de estudio y su modelado matemático. Se busca modelar la realidad de la problemática del ruteo de vehículos y programación de cargas, en una red de distribución urbana de mercaderías, desde un enfoque multi-objetivo, tomándose los controles necesarios para la validación del modelo y verificar que su comportamiento refleje todos los condicionantes y consideraciones del fenómeno real.
- Desarrollo de un procedimiento computacional, que permita la resolución de tal formulación, en un tiempo y con una utilización de recursos razonables.
- Realización de la experimentación mediante ensayos numéricos en herramientas de software, para validar el modelo y el funcionamiento de la herramienta algorítmica propuesta para resolver el problema. De esta manera se intenta demostrar que el algoritmo evolutivo planteado, es efectivo para abordar la problemática objeto de estudio. Se utilizará información real de un operador logístico que presta este tipo de servicios.
- Análisis de los resultados.

4. APORTES ESPERADOS DE LA INVESTIGACIÓN

La primera contribución de este trabajo, es el abordaje realista y desde un enfoque multi-objetivo, del modelado de la problemática que enfrenta un operador logístico en sus procesos de toma de decisiones en ambientes complejos, como es el de la programación de cargas y ruteo de vehículos, con múltiples centros de distribución en contextos urbanos.

Otro aporte original, es la hibridación de un algoritmo evolutivo multi-objetivo para la resolución del problema, presentándose un novedoso procedimiento de recombinación, que permite el clustering y el ruteo de forma simultáneas, que mejora significativamente los resultados obtenidos por otros algoritmos conocidos utilizados en la resolución de este tipo de problemas de optimización combinatoria.

Otra contribución, es la utilización de forma exitosa, en un caso basado en información proveniente de una instancia real, de una estrategia de introducción de preferencias parciales por parte del agente decisor (en nuestro caso el responsable de la planificación de la distribución). Por un lado, la herramienta algorítmica potencia las capacidades computacionales del agente decisor y, por otro lado, mejora el desempeño del algoritmo evolutivo, mediante la introducción de información de su experiencia, que guía el proceso de búsqueda.

De forma general, la línea de investigación introducida, fijará un precedente para el tratamiento formal de los procesos de toma de decisiones en el contexto de la problemática logística de la distribución de mercaderías en general, y de alimentos perecederos en particular. Buscando mejoras en la competitividad de la cadena de suministro y en la sostenibilidad del desarrollo económico. Más concretamente, la investigación presenta un importante impacto socio-económico y productivo, dado que las herramientas tecnológicas desarrolladas podrán ser utilizadas por distintas entidades públicas o privadas, permitiéndoles mejorar su eficiencia y reducir los costos logísticos de distribución.

CAPÍTULO II. MARCO TEÓRICO

Solo es posible alcanzar los objetivos propuestos en el presente trabajo, mediante la abstracción de diferentes desafíos científicos y técnicos que surgen de la conjunción de las diferentes áreas del conocimiento presentes en este libro. El área del conocimiento que actúa como eje central de este libro, es el de Tecnología y Dirección de Operaciones, en el marco de la gestión de la cadena de suministros.

Por lo tanto, en este capítulo primero se presenta el marco conceptual en términos de distribución urbana de mercaderías (DUM) y ruteo de vehículos en el contexto del enfoque de la gestión de la cadena de suministros (Supply Chain Management, SCM).

En este contexto, la tecnología de la información consiste en el hardware, software y el personal que recopila, analiza y ejecuta las operaciones relacionadas a la distribución urbana de mercaderías como último eslabón de la cadena de suministros, captando y analizando la información necesaria que permite tomar buenas decisiones (Chopra & Meindl, 2008). Por lo tanto, en el segundo punto de este capítulo, se presentan distintas tecnologías, aplicaciones y sistemas implicados en la gestión de la cadena de suministros en general y en la distribución urbana de mercaderías en particular, a los efectos de comprender el posicionamiento y las relaciones de la herramienta algorítmica desarrollada, con el resto de los sistemas de apoyo a la toma de decisiones.

Luego, en el punto tres, se presentan los principales paradigmas computacionales que sustentan la base de las técnicas implícitas en la herramienta de ayuda a la toma de decisiones desarrollada en este trabajo.

En el punto cuatro se presentan las fases en que se puede separar un proceso de toma de decisiones multi-objetivo, se realiza una introducción a la teoría de la optimización multi-objetivo, y finalmente se presentan los métodos utilizados para articular las preferencias en el proceso toma de decisiones.

Y finalmente, en el punto cinco, se describen los tipos de métodos utilizados en la literatura para resolver problemas de decisión considerando múltiples objetivos, presentándose de manera más detallada la descripción de los Algoritmos Evolutivos.

vos Multi-objetivo (AEMO), una de las meta-heurísticas más utilizadas para resolver problemas con características similares al problema objeto de estudio.

1. ENFOQUE DE LA GESTIÓN DE LA CADENA DE SUMINISTROS

La importancia y complejidad de los sistemas de distribución urbana de mercaderías, justifica la utilización de herramientas de ayuda a la toma de decisiones, que faciliten y/o agilicen la ejecución de dicho proceso. El enfoque de la gestión de la cadena de suministro, da un marco conceptual de referencia desde el cual comenzar el abordaje de los sistemas mencionados.

Actualmente el Consejo de Profesionales de Gestión de la Cadena de Suministro (CSCMP) presenta la siguiente definición de *gestión de la cadena de suministros*:

La gestión de la cadena de suministro, abarca la planificación y gestión de todas las actividades involucradas en compras y abastecimiento, transformación y todas las actividades de gestión logística. De manera muy importante, también incluye la coordinación y colaboración con los socios del canal, que pueden ser proveedores, intermediarios, operadores logísticos y clientes. En esencia, la gestión de la cadena de suministro integra la gestión del abastecimiento y la demanda dentro y a través de las empresas. (Professionals Council of Supply Chain Management, 2015)

En este contexto, cuando se habla de *cadena de suministros* (CS) se hace referencia a la red de relaciones que abarcan todas las actividades vinculadas con el flujo y transformación de bienes, desde la etapa de materia prima hasta el usuario final, así como los flujos de información relacionados. Los materiales y la información fluyen en sentido ascendente y descendente en cada cadena de suministros, conectando cada empresa hacia arriba a proveedores y hacia abajo a mayoristas y minoristas hasta llegar al consumidor final. Cada empresa participa de este flujo como un eslabón de una cadena formada por todas aquellas partes involucradas de manera directa o indirecta en la satisfacción del cliente final de la cadena. En este sentido, la cadena de suministro vincula los procesos de una empresa con los procesos de sus proveedores y sus clientes (Krajewski, Ritzman, & Malhotra, 2007).

La *infraestructura de la red de distribución* puede definirse como la infraestructura a través de la cual los bienes se mueven entre diferentes entidades comerciales

(es decir, proveedores, fabricantes, distribuidores y minoristas), todas pertenecientes a la misma cadena de suministro (Vecchiato, 2012).

Mientras que, con el término *distribución*, se hace referencia a todas las actividades relacionadas con el movimiento físico de bienes, incluido el transporte, pero también el transbordo y el almacenamiento, desde la etapa del proveedor hasta la del cliente final de la cadena de suministro (Bektas, 2017). Con el de *distribución urbana de mercaderías*, se hace referencia a la distribución en el último de los eslabones de una cadena de suministros, y es un factor clave tanto en la competitividad de una organización empresarial individual como de una cadena de suministros como un todo (Antún, 2013).

Las actividades que conforman la distribución urbana de mercaderías, implican un gran número de decisiones interrelacionadas entre sí, que de forma general estarían conformadas por el transporte de mercaderías, y sus operaciones asociadas: entrega, recolección, transferencias, carga, descarga, colocación en puntos de venta, *cross-docking* con o sin almacenamiento transitorio, y retornos de logística inversa en entornos urbanos (Antún, 2013). En la gestión de estas actividades, se busca alcanzar una variedad de objetivos, que van desde un costo bajo hasta una elevada flexibilidad y capacidad de respuesta. Por esta razón es que empresas pertenecientes a la misma industria, pueden seleccionar de forma óptima redes de distribución con estructuras muy diferentes (Chopra & Meindl, 2008).

Hay tres tipos de factores que condicionan las características de la red de distribución urbana de mercaderías que es óptima para una empresa. En primer lugar, factores relacionados a la infraestructura de la red, por ejemplo, la estructura existente en el interior del espacio urbano, como las características de la morfología de la red vial urbana (capacidad, regularidad en el diseño, estado, espacios para estacionamiento, etc.), las restricciones de circulación para vehículos de carga y extra-dimensionados, los problemas por congestión, etc. En segundo lugar, hay aspectos relacionados a la *estrategia de distribución*, como la cantidad de clientes a cubrir, flexibilidad requerida a la red, los tiempos y características del servicio y los horarios para la recepción de mercadería. En tercer lugar, están las *características del vehículo*, ya que este debe adaptarse a las condiciones de in-

fraestructura y la estrategia de distribución. Por ejemplo, el tamaño del vehículo está limitado por la anchura de las calles, y por la facilidad en el acceso a las zonas de carga y descarga (Antún, 2013).

La literatura identifica diferentes problemas de diseño de rutas y programación de vehículos, habitualmente agrupados bajo el nombre genérico de *problemas de ruteo de vehículos*. Irnich, Toth y Vigo (2014) plantean la siguiente definición genérica para esta familia de problemas:

Dado un conjunto de solicitudes de servicios de transporte y una flota de vehículos. El problema de ruteo de vehículos, consiste en determinar un conjunto de rutas, que permitan realizar todas o algunas de las solicitudes de servicios de transporte, con la flota de vehículos dada, a un costo mínimo. De forma que las secuencias de visita en todas las rutas sean factibles de ejecutar (p.1).

Luego, dependiendo de las características de la red de distribución que se esté modelando, se configura una serie de problemas de ruteo de vehículos, que son tratados generalmente como extensiones del problema conocido como el *problema del agente viajero* (o TSP- *traveling salesman problem*) (Miguel, Frutos, & Tohmé, 2015).

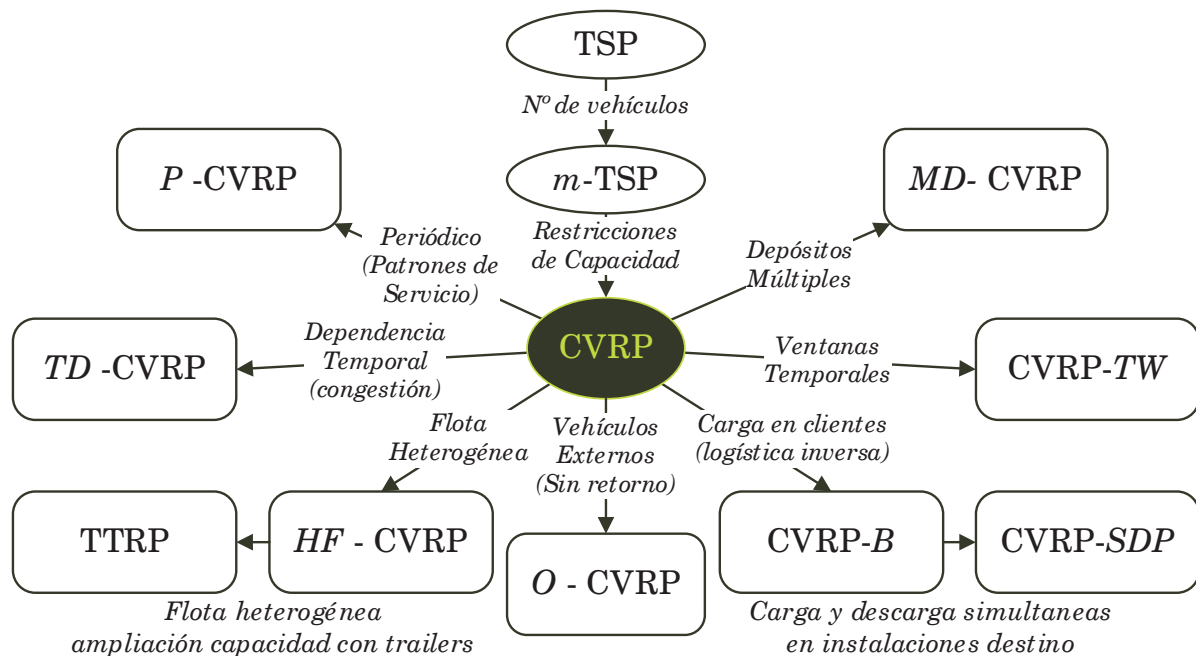
El *Traveling Salesman Problem* (TSP) constituye la situación general de partida para formular otros problemas de distribución más complejos, pero más prácticos, en el sentido de que consideran diferentes aspectos de la realidad logística en la distribución de mercaderías. En el TSP se dispone de un sólo vehículo que debe visitar a todos los clientes en una sola ruta y a costo mínimo. Es decir, la ruta debe ser tal que minimice la distancia recorrida. En el *Multiple Traveling Salesmen Problem* (*m-TSP*), m vendedores deben cubrir un número determinado de clientes, de forma que cada destino sólo se visita por uno de ellos. Cada ruta empieza en un depósito base que es también el final del recorrido. En este caso, se desea minimizar la suma de las distancias recorridas a lo largo de todas las rutas (Ballou, 2004).

En la Figura 1, adaptada de Weise, Podlich y Gorltdt (2009), se presentan esquemáticamente las variantes más conocidas de estos problemas de ruteo de vehículos (VRP- *Vehicle Routing Problem*) derivados como extensiones del TSP, al con-

siderar distintos aspectos de la red de distribución a modelar. Por ejemplo, a partir del TSP, al considerar en el modelo de la red de distribución una flota de m vehículos, se pasa al problema identificado como m -TSP en la figura. Si las características del problema de distribución a modelar implican además, que es relevante considerar las capacidades de los vehículos para determinar las rutas óptimas, se pasaría al problema identificado como CVRP en la figura (descripto posteriormente).

Problemas más realistas, serán combinaciones de los presentados a continuación e incluirán cuestiones adicionales.

Figura 1: Variantes de problemas de ruteo de vehículos



Fuente: Adaptado de Weise, Podlich y Gorltdt (2009)

A continuación, se realiza una descripción de los aspectos más relevantes de cada variante considerada en la Figura 1.

1.1. *Capacitated Vehicle Routing Problem (CVRP):*

El problema de ruteo de vehículos con restricciones de capacidad es uno de los problemas básicos de ruteo de vehículos. Consiste básicamente en que a cada cliente se le debe enviar cierta cantidad de mercadería, conocida con anticipación, desde un único depósito. Para repartirla se dispone de un determinado número de vehículos con cierta capacidad de transporte, los cuales parten y llegan a un único depósito. Cada vehículo visita exactamente a un único cliente, y cada cliente debe ser visitado por un sólo vehículo, y la suma de las demandas de los clientes visitados por un vehículo no debe exceder la capacidad del mismo. El objetivo de este tipo de problemas suele ser determinar una secuencia de abastecimiento y su cronograma de carga correspondiente, que minimice el costo total, donde dicho costo se puede ver como una función del tiempo o la distancia total recorrida, o bien minimizar el número de vehículos a utilizar (Frutos, Tohmé, & Miguel, 2016).

1.2. *Capacitated Vehicle Routing Problem with Time Windows (CVRP-TW):*

Variante del CVRP que se define considerando restricciones temporales conocidas para el servicio o abastecimiento de cada cliente. Existen diversos subtipos y sus combinaciones de este tipo de problemas ya que pueden definirse ventanas de tiempo para diferentes momentos del servicio. Los casos más conocidos pueden resumirse en: restricción de tiempo conocida para la llegada a los clientes; restricción de tiempo conocida para el recorrido del camino hasta el cliente; restricción de tiempo conocida para la llegada a los depósitos de carga; restricción de tiempo conocida para el servicio activo de vehículos o conductores (Miguel, Frutos, & Tohmé, 2014) (Cordeau, Laporte, & Mercier, 2001).

1.3. *Open Capacitated Vehicle Routing Problem (O-CVRP):*

Este problema es una variante del CVRP que se enfoca en el manejo de vehículos externos a la compañía. Por tal motivo, cambia la estructura de costos y la programación dado que, por ejemplo, no se contempla el viaje de regreso (Li, Golden, & Wasil, 2007).

1.4. *Heterogeneous Fleet Capacitated Vehicle Routing Problem (HF - CVRP):*

Es una variación del CVRP que considera una flota de vehículos heterogénea. Es decir con diferentes capacidades y formas para cargas de diferentes tipos, como carga suelta o extra dimensionada entre otras (Li, Golden, & Wasil, A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem, 2007).

1.5. *Multi Depot Capacitated Vehicle Routing Problem (MD -CVRP):*

Problema definido como una variante del CVRP que considera múltiples depósitos o almacenes de carga con distintas ubicaciones geográficas (Gulczynski, Golden, & Wasil, 2011). La variante básica del problema con varios depósitos, aparece por primera vez en la literatura a mediados de la década del 80 en el trabajo de Kulkarni and Bhave (1985), y luego en los trabajos de Laporte, Nobert, y Taillefer (1988) y de Carpaneto, Dell'amico, Fischetti, y Toth (1989).

1.6. *Capacitated Vehicle Routing Problem with Backhauls (CVRP-B):*

Este problema es una generalización del CVRP en el que se permite realizar cargas y descargas en las instalaciones de los clientes. Hay dos modalidades, la conocida como CVRP-B que considera que existen puntos de entrega desde el almacén y otros de recolección hacia el almacén durante el viaje de regreso. Pero no se recoge mercadería hasta que no finalicen las entregas.

Y la conocida como *Capacitated Vehicle Routing Problem with Simultaneous Deliveries and Pickups (CVRP-SDP)*, que considera que existen puntos de entrega y otros de recolección hacia el almacén, pudiendo coincidir ambos simultáneamente. En este caso se permite la entrega y la recolección mientras no se viole la capacidad del vehículo. Es decir, los bienes son entregados desde el depósito al cliente, y bienes adicionales son cargados en las instalaciones del cliente y son llevados de regreso al depósito (Ropke & Pisinger, 2006). Normalmente éste tipo de planteo responde a la necesidad de transporte de regreso con material de empaque para cumplir con la logística inversa.

1.7. *Periodic Capacitated Vehicle Routing Problem (P-CVRP):*

Definido como otra variante del CVRP, este problema consiste en permitir que algunos clientes puedan solicitar el servicio en varios momentos durante un período de tiempo. Así, primero los clientes se asignan a algún patrón de servicio (por ejemplo, clientes requieren tres visitas semanales, se podrían asignar a un patrón de servicio de lunes, miércoles y viernes), y luego se resuelve un VRP para cada día del período de tiempo para todos los clientes programados en ese día. Y el objetivo es minimizar la distancia total recorrida por la flota en todos los días del período de tiempo (Gulczynski, Golden, & Wasil, 2011).

1.8. *Time Dependent Capacitated Vehicle Routing Problem (TD - CVRP):*

Problema que consiste en una variante del CVRP clásico, pero con el objetivo de minimizar el tiempo de las rutas y no el costo de las mismas que normalmente depende de la distancia recorrida por los vehículos (Malandraki & Dial, 1994). En un entorno urbano con congestión, las velocidades no son constantes y el tiempo de viaje entre dos puntos por lo general no es una función sólo de la distancia recorrida. La densidad del tráfico en distintas zonas varía y causa fluctuaciones en la velocidad de desplazamiento y en los tiempos de viaje. Esto puede causar considerables diferencias en los tiempos de viaje en diferentes zonas urbanas. Para tener en cuenta estas variaciones, el tiempo de viaje entre dos puntos puede representarse por medio de una función determinista de la distancia entre dos puntos y también de la hora del día en la que se produce el viaje o de la zona que esté transitando el vehículo, considerando distintas velocidades promedio en distintas zonas urbanas.

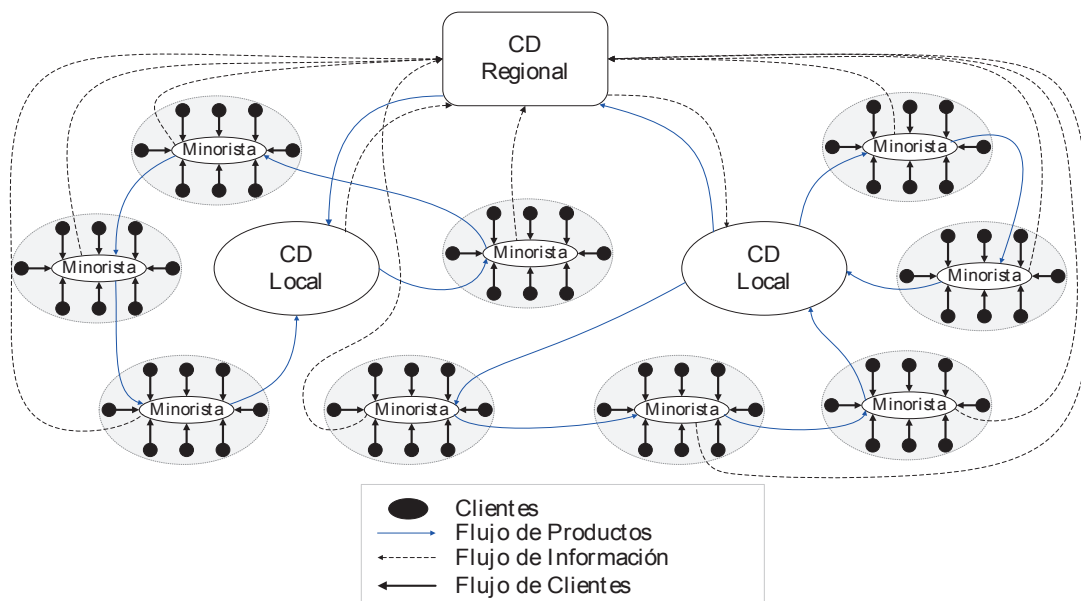
1.9. *Truck and Trailer Routing Problem (TTRP):*

Este problema de ruteo de vehículos, consiste en la atención de clientes con una flota de vehículos de transporte heterogénea, en la que la capacidad de los camiones se incrementa con un tráiler (Chao I. , 2002). Esto implica la consideración de dos restricciones: la primera y más representativa se define por las instalaciones de clientes con espacio de maniobra limitado, lo que restringe la posibilidad de prestación del servicio con el tráiler enganchado al vehículo. La otra restricción, se

contempla en las limitaciones en las calles para el desarrollo de este tipo de transportes con vehículos extra dimensionados. Normalmente existe regulación a contemplar al respecto.

De las variantes presentadas del problema de ruteo de vehículos capacitados, es de especial interés el identificado previamente como MDCVRP, o problema de ruteo de vehículos capacitados con múltiples depósitos, dado que se ajusta a las características básicas de una red de distribución urbana de mercaderías, con varios depósitos y una serie de destinos a abastecer. Por ejemplo, en la Figura 2 se presenta esquemáticamente un ejemplo simplificado de una red de distribución con una estructura de 9 nodos minoristas, dos centros de distribución locales y un centro de distribución regional.

Figura 2: Ejemplo red de distribución urbana simple



Fuente: Elaboración propia en base a Chopra y Meindl (2008)

En los flujos de información representados en la Figura 2, se ejemplifica un proceso de toma de decisiones centralizado en cuanto a distribución, en el que la recepción y la coordinación de la distribución de pedidos, es realizada desde el centro de distribución regional, que abastece a los centros locales, que a su vez abastecen a los minoristas. Esta estructura, responde en general a artículos con un alto nivel de rotación, dado que se mantienen inventarios en diferentes escalones de la red hasta llegar al minorista, y el artículo tiene una elevada disponibilidad. Los costos

de transporte son relativamente bajos comparados con una estructura de distribución con entrega a domicilio a partir de los centros de distribución locales, por ejemplo, dado que en el caso representado el cliente debe trasladarse a las instalaciones del minorista. El minorista transmite el nivel de stock al centro de distribución regional que coordina la distribución, y en base a pronósticos determina los niveles de inventario óptimos.

De esta manera, puede verse que son variados los elementos que caracterizan una red de distribución urbana de mercaderías (clientes, depósitos, vehículos, tipos de mercadería, limitaciones operativas y normativas, etc.), que darán lugar a diferentes formulaciones posibles para el modelo matemático del problema. Por ejemplo, puede darse dentro de un programa de distribución para un día en particular, que un minorista tenga un pedido que requiera de un vehículo completo para ser cubierto. Y al mismo tiempo, darse cargas fraccionadas con pedidos de varios minoristas, y que en zonas muy concurridas no puedan satisfacerse con un solo vehículo, incluso, cada cliente podría ser atendido por varios vehículos si fuera necesario (Tavakkoli-Moghaddam, Safaei, Kah, & Rabbani, 2007) (Belenguer, Benavent, Labadi, Prins, & Reghioui, 2010).

Entonces, puede verse que los problemas de distribución física de mercaderías, constituyen un conjunto variado y complejo, que algunos autores han intentado agrupar atendiendo a sus características más relevantes. Esta simplificación de la realidad permite la adopción de modelos matemáticos que facilitan los procesos de toma de decisiones de los operadores logísticos involucrados (Mula, Peidro, Diaz-Madronero, & Vicens, 2010).

En general, en la literatura estos procesos de toma de decisiones son estudiados siguiendo el paradigma decisional clásico o mono-objetivo. Que como se verá posteriormente, consiste en seleccionar el curso de acción más conveniente a partir de un conjunto de soluciones factibles, basándose en un único criterio³. Sin embargo, en los sistemas reales la mayoría de las decisiones no tienen un único criterio medible

³ En el presente trabajo se utilizará de forma indistinta el término criterio y objetivo. Pero siempre se estará haciendo referencia al concepto de objetivo, en el contexto de optimización.

por el cual pueda declararse que una alternativa sea completamente satisfactoria. Es decir, los agentes implicados, no toman sus decisiones basándose en un único criterio, sino que el proceso decisional está influenciado por una gama de varios criterios y que usualmente están en conflicto. Un ejemplo muy simple para mostrar la racionalidad de este planteo, es analizar la decisión de un individuo respecto de qué automóvil comprar con un determinado presupuesto. En la vida real una persona no toma esta decisión considerando solamente el consumo del vehículo, sino que le interesan también otras prestaciones, como la seguridad, el confort, la potencia, la estética, etc. No obstante, el individuo también puede ver, en general, que al aumentar la cantidad de caballos de fuerza aumenta el costo del vehículo, o que al aumentar la cantidad y calidad de los sistemas de seguridad que el vehículo posee también aumenta el costo del mismo, lo mismo sucede con otras prestaciones relacionadas al confort por ejemplo. Por lo que, si se cuenta con un presupuesto determinado, aumentar alguno de esos criterios puede implicar necesariamente la reducción de algún otro para que el automóvil sea comprado por ese individuo, por lo que se dice que esos criterios están en conflicto.

Cuando un problema de optimización tiene una sola función objetivo a optimizar (un único criterio) se denomina de optimización *mono-objetivo* (o mono-criterio). Y cuando el problema a optimizar posee más de una función objetivo, la tarea de encontrar una o más soluciones óptimas se conoce como optimización *multi-objetivo* (o multi-criterio).

Entre los criterios u objetivos considerados en el contexto de la distribución urbana de mercaderías, además de la *minimización del tiempo total* para realizar la distribución, que generalmente es proporcional al costo, y de la *minimización de la distancia total recorrida*, que en entornos urbanos con congestión no siempre es proporcional al tiempo y los costos, también son relevantes otros criterios como: la *minimización de la cantidad de vehículos*, directamente relacionada a la minimización de la sub-utilización de los vehículos (Ombuki-Berman & Hanshar, 2009), (Tan, Chew, & Lee, 2006); el *balanceo de la carga de trabajo* de los vehículos, ya sea en el tiempo por ruta, en la cantidad de mercadería manipulada en cada ruta, o en la cantidad de clientes servidos por ruta (Lin & Kwok, 2006) (Weise, Podlich, &

Gorltdt, 2009), este objetivo ha suscitado mucho interés dada su importancia para la gestión de los recursos humanos implicados (Lee & Ueng, 1999); la *minimización de riesgos*, presente en la distribución de materiales peligrosos (Zheng, 2010) y residuos hospitalarios; la minimización de las emisiones de CO₂ al ambiente (Urquhart, Hart, & Scott, 2010), la *maximización de la satisfacción al cliente*, derivada generalmente del cumplimiento de ventanas temporales fijadas por los clientes minimizando las llegadas tempranas o tardías.

En resumen, en las cadenas de suministros reales, los problemas de decisión que enfrentan los operadores logísticos, en la distribución urbana de mercaderías, son multi-objetivo por naturaleza. Los agentes tomadores de decisiones tienen que lidiar simultáneamente con varios objetivos a maximizar o minimizar y en general estos objetivos están en conflicto. Como fue mencionado en el capítulo uno, esta es una de las motivaciones del presente trabajo, por un lado por la muy reducida cantidad de trabajos de investigación que tratan este tipo de problemas desde un enfoque multi-objetivo (Montoya-Torres, Franco, Isaza, Jiménez, & Herazo-Padilla, 2015), y por otro, por la relevancia que tiene hoy en día, tanto en Argentina como en el resto del mundo, la mejora en la eficiencia en la distribución urbana de mercaderías, para hacer competitivas tanto a las empresas como a las cadenas de suministro a las que pertenecen.

2. TECNOLOGÍA Y DIRECCIÓN DE OPERACIONES

2.1. Tecnologías de la información

En los últimos años, el desarrollo tecnológico en el ámbito de las tecnologías de la información es tan significativo, que no es posible abstraerse del impacto que esta evolución tiene sobre las ciencias de la administración, y especialmente, sobre la capacidad de toma de decisiones, tanto a nivel individual, como a nivel organizacional.

En el marco de la gestión de la cadena de suministros, el efecto de esta revolución tecnológica, puede verse en el crecimiento exponencial del desarrollo y del uso de sistemas y tecnologías que mejoran la capacidad de gestión de la información, tanto dentro de las organizaciones como a lo largo de toda la cadena de suministro. La

información es la clave del desempeño de la gestión de la cadena de suministro, al proporcionar la base sobre la cual se toman las decisiones y se ejecutan las transacciones. Es la fuente de conocimiento de lo que ocurre en los eslabones inferiores y superiores de la cadena de suministro. Es lo que hace visible la cadena de suministro (Chopra & Meindl, 2008).

Con el término *tecnologías de la información* (TI) se hace referencia, a las ciencias, tecnologías y aplicaciones relacionadas con la adquisición, almacenamiento, tratamiento, comunicación, difusión, y uso de la información. En el contexto de la dirección de operaciones, puede decirse que comprende el conjunto de actividades y soluciones producidas por una fuente computacional y con aplicaciones a distintas áreas organizacionales e inter-organizacionales, abarcando el conjunto de recursos automáticos (no humano) dedicados al almacenamiento, procesamiento y la comunicación de información, así como al modo en que estos recursos están organizados en un sistema capaz de ejecutar dicho conjunto de tareas. Es decir, la TI no se limita a equipos (hardware), software (software) y comunicaciones de datos, sino que cubre todas las actividades desarrolladas en la sociedad mediante el uso de recursos computacionales (Paletta & Dias Vieira Junior, 2008).

La TI en el marco de la gestión de la cadena de suministro, hace posible que la información correcta esté disponible en momento y el lugar adecuado cuando se requiere, lo que permite la sincronización del flujo de información con el flujo físico a través de la empresa y de la cadena de suministro (Zapata Cortés, Arango Serna, & Adarme Jaimes, 2010).

Es importante revisar las tecnologías más relevantes, que han permitido mejorar la integración del flujo de información entre los integrantes de la cadena, facilitando el logro de mayores niveles de eficiencia y eficacia en los procesos de toma de decisiones en contextos logísticos complejos.

Entre las tecnologías de *captura y transferencia de información*, en el marco de la gestión de la cadena de suministros se mencionan Internet, Banda Ancha, EDI (Intercambio Electrónico de Datos), XML (*eXtensible Markup Language*) (Zapata Cortés, Arango Serna, & Adarme Jaimes, 2010).

Las tecnologías más relevantes asociadas al *almacenamiento y la recuperación de la información* son las bases de datos. Existen varios tipos, pero actualmente la más común es la base de datos relacional. Su característica fundamental es que permite el registro, almacenaje y posteriormente la consulta y recuperación de las transacciones. Además, estas transacciones pueden ser observadas en tiempo real (Hugos, 2003), (Zapata Cortés, Arango Serna, & Adarme Jaimes, 2010).

De las tecnologías tradicionales en *identificación y entrega de información*, cabe mencionar, los Código de Barras (Frazelle & Sojo, 2007), que permiten llevar un control y seguimiento precisos y en tiempo real del inventario en las ubicaciones de almacenamiento y entre los integrantes de la cadena de suministro. Registrando y comunicando con rapidez los detalles de movimientos (Bowersox, Closs, & Cooper, 2007).

Otra tecnología de auto identificación son las etiquetas magnéticas o TAGS, que emiten radiofrecuencias (*RFID, Radio Frequency Identification*), que permiten mayor capacidad de almacenamiento de datos que los códigos de barras, son reutilizables y que no es necesario el contacto visual entre el lector y la etiqueta (Mauleón, 2003).

Los sistemas de posicionamiento global (*GPS, Global Positioning System*) permiten determinar en todo el mundo la posición de un objeto con una precisión de hasta 15 metros durante las 24 horas del día (Zapata Cortés, Arango Serna, & Adarme Jaimes, 2010).

2.2. Sistemas de información

Los sistemas de información, son métodos de recopilación, registro, procesamiento, seguimiento y comunicación de la información necesaria para que las personas dentro de una organización tomen decisiones (Fogarty, Blackstone, & Hoffmann, 2007).

Chopra y Meindl (2008) argumentan que estos sistemas abarcan un conjunto de tecnologías, procesos, hardware y software empresarial, que dan soporte a la gestión de tres grupos de procesos que relacionan a la organización con el resto de la cadena de suministro. Estos grupos de procesos (o macro-procesos) pueden resu-

mirse en los procesos enfocados en interacciones corriente abajo, los enfocados en la cadena de suministro interna de la empresa, y los enfocados en interacciones corriente arriba.

Entonces, se pueden clasificar los sistemas de información en el marco de la gestión de la cadena de suministros, tomando como criterio los macro-procesos a los que dan soporte estos sistemas. De esta forma tenemos:

Sistemas enfocados en relaciones empresa-proveedores:

Son sistemas de apoyo a la planificación y gestión de actividades pertenecientes a procesos enfocados en *interacciones corriente arriba*, entre la empresa y sus proveedores. Entre los sistemas más relevantes relacionados a estos procesos están el aprovisionamiento electrónico (*e-procurement*), el inventario administrado por el vendedor (*VMI, Vendor Managed Inventory*), la planeación de reposición continua (*CRP, Continuous Replenishment Planning*), el e-sourcing, el e-procurement. Puede verse una descripción de ellos en (Correa Espinal & Gómez Montoya, 2009).

Sistemas enfocados en procesos internos de la organización:

Son sistemas de apoyo a la planificación y gestión de actividades pertenecientes a procesos enfocados en *interacciones internas a la empresa*, relacionadas a su proceso de transformación de valor. Los sistemas más relevantes relacionados con estos procesos son el ERP (*Enterprise Resource Planning*), el WMS (*Warehouse Management System*), el MRP (*Manufacturing Resource Planning*), los sistemas de picking (*Pick to Light-Voice*) y el LMS (*Labor Management System*). Puede verse una breve reseña de estos sistemas en (Correa Espinal & Gómez Montoya, 2009), a continuación se describen brevemente los necesarios para la construcción del marco teórico en el que se encuentra la herramienta propuesta.

Los sistemas de planificación de recursos empresariales (*ERP, enterprise resource planning*), consisten en un paquete de programas de gestión que permiten administrar el conjunto de procesos de una empresa. Está compuesto por una serie de funcionalidades básicas como cuentas por cobrar, recursos humanos, control de inventarios, administración de pedidos, etc. Pueden incluir otros módulos de pla-

neación, de predicción, de administración de relaciones con los clientes, etc. (Zapata Cortés, Arango Serna, & Adarme Jaimes, 2010).

El sistema de gestión de almacenes (*WMS, Warehouse Management System*), generalmente está integrado al ERP y puede incluir también al sistema de gestión de pedidos (*OMS, Order Management System*), gestiona la información del flujo del producto en los inventarios de la organización. Como el proceso de recepción de materiales, las operaciones de ubicación de los materiales en las instalaciones, y de selección del pedido dentro de las instalaciones, la carga de trabajo para los recolectores y la estimación de disponibilidad del producto. También controla las operaciones internas de abastecimiento de materiales y componentes al proceso de producción (Ballou, 2004).

Los sistemas de planeación de necesidades de materiales (*MRP, Manufacturing Resource Planning*) generalmente están integrados al ERP, permiten la gestión de requerimientos de materiales del programa maestro de producción, la desglosa y crea programas para los proveedores con tipos de partes, cantidades y fechas de pedido y de entrega. Mostrando información en tiempo real de requerimientos de componentes de cada producto y sus correspondientes niveles de inventario (Chase, Jacobs, & Aquilano, 2010).

Luego están los sistemas de planificación y programación avanzada (*APS, Advanced Planning and Scheduling*). Son aplicaciones muy analíticas que buscan optimizar los procesos de la organización basándose en información transaccional suministrada por el ERP, y que utilizan sofisticadas técnicas matemáticas para realizar la planificación de la producción y las operaciones (Hugos, 2003) (Zapata Cortés, Arango Serna, & Adarme Jaimes, 2010).

Sistemas enfocados en relaciones empresa-clientes:

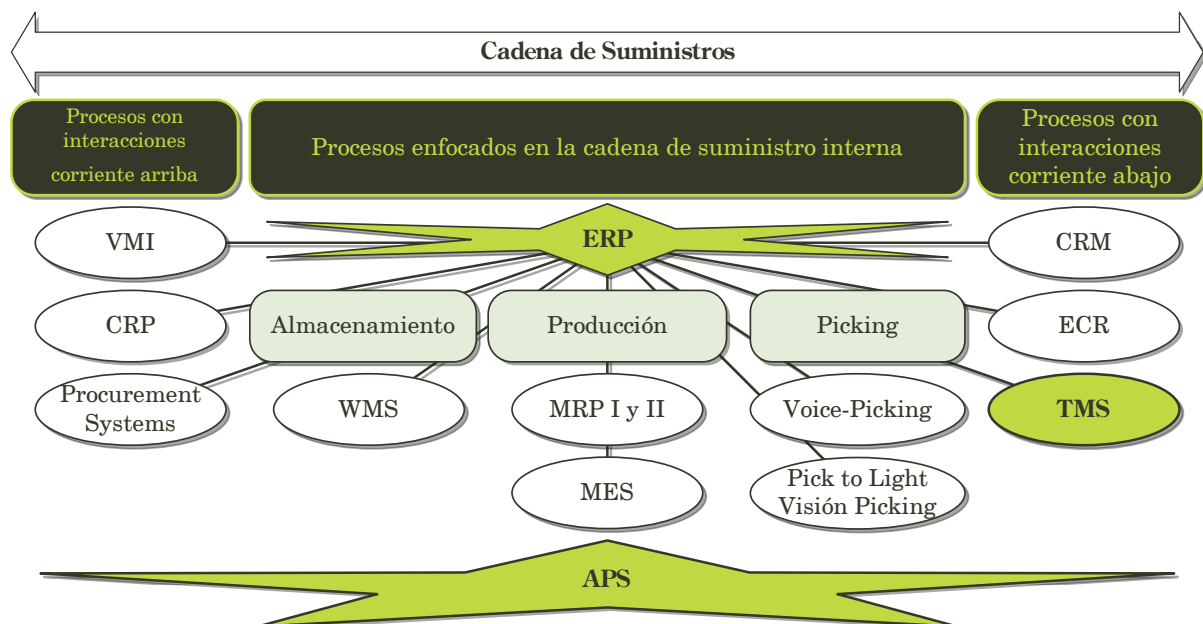
Son sistemas de apoyo a la planificación y gestión de actividades pertenecientes a procesos enfocados en *interacciones corriente abajo*, entre la empresa y sus clientes. Como procesos de distribución y de relación con los clientes finales, además de la coordinación y sincronización con procesos internos. Los más relevantes son el CRM (*Consumer Relationship Management*), el TMS (*Transportation Management System*), y

el ECR (*Efficient Consumer Response*) (Correa Espinal & Gómez Montoya, 2009), (Bowersox, Closs, & Cooper, 2007), (Payne, 2006).

A los fines de este trabajo, es importante en este punto, dar una descripción del sistema de gestión de transporte (*TMS, Transportation Management System*). Este sistema planea, ejecuta y administra las funciones de transporte y movimiento de mercadería. Generalmente está vinculado a través del ERP con el WMS (Bowersox, Closs, & Cooper, 2007).

Este sistema busca identificar y evaluar estrategias alternativas de transporte y tácticas para determinar los mejores métodos para mover los productos con las restricciones existentes (Taylor, 2007). Ballou destaca que su objetivo es dar apoyo a la planificación y control de actividades relacionadas al transporte, como la selección de la modalidad, consolidación o quiebre de las cargas, coordinación de operaciones de Cross-Dock, ruteo y programación de embarques, rastreo de envíos, auditoría de documentación y facturación del flete (Ballou, 2004).

Figura 3: *Sistemas de Información en la SCM*



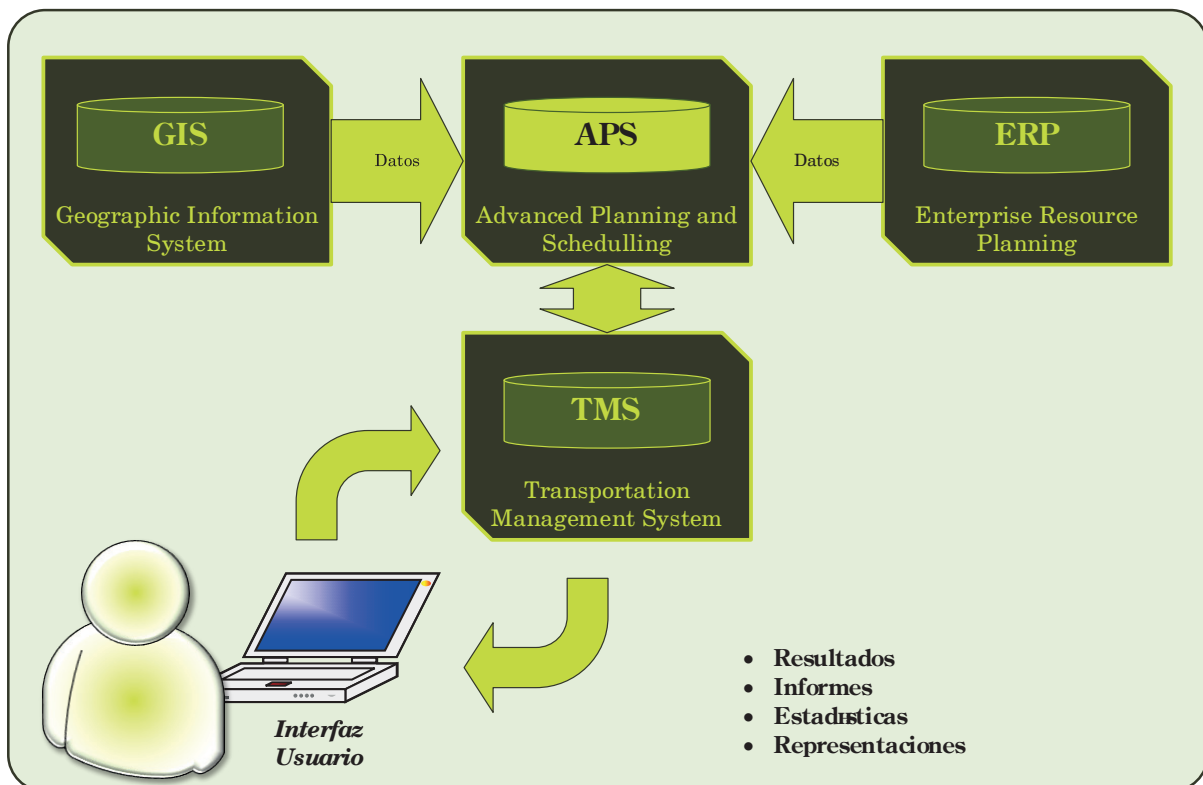
Fuente: *Elaboración propia en base a Chopra y Meindl (2008)*

En la Figura 3 se muestran de forma esquemática los diferentes sistemas y subsistemas de información presentados anteriormente, mostrando su relación con los

macro-procesos que relacionan a la empresa con el resto de la cadena de suministros, siguiendo la argumentación de Chopra y Meindl (2008).

Las técnicas computacionales implementadas en el presente libro, operan como parte de los Sistemas de Planificación y Programación Avanzados (APS) relacionados al Sistemas de Gestión de Transporte (TMS), alimentándose del Sistema de Información Geográfica (GIS) y del Sistema de Planificación de Recursos Empresariales (ERP). Como se comentó anteriormente, el interés sobre estos sistemas, reside en que optimizan la toma de decisiones coordinada en el ámbito de la planificación de la distribución urbana de mercadería en cuanto a actividades de transporte y movimiento de cargas.

Figura 4: Herramientas tecnológicas de ayuda a la toma de decisiones



Fuente: elaboración propia

En la Figura 4, se presenta la relación entre estos sistemas y la interacción con el usuario. El usuario, responsable encargado de determinar el programa de distribución, introduce la información que guiará el proceso de búsqueda de la herramienta computacional. Luego, se ejecuta el proceso de optimización, y se presentan las mejores alternativas encontradas. Finalmente, el usuario selecciona la opción deseada,

entre las alternativas óptimas encontradas. Es decir, selecciona el programa óptimo de distribución y luego se emiten los informes e itinerarios para cada vehículo.

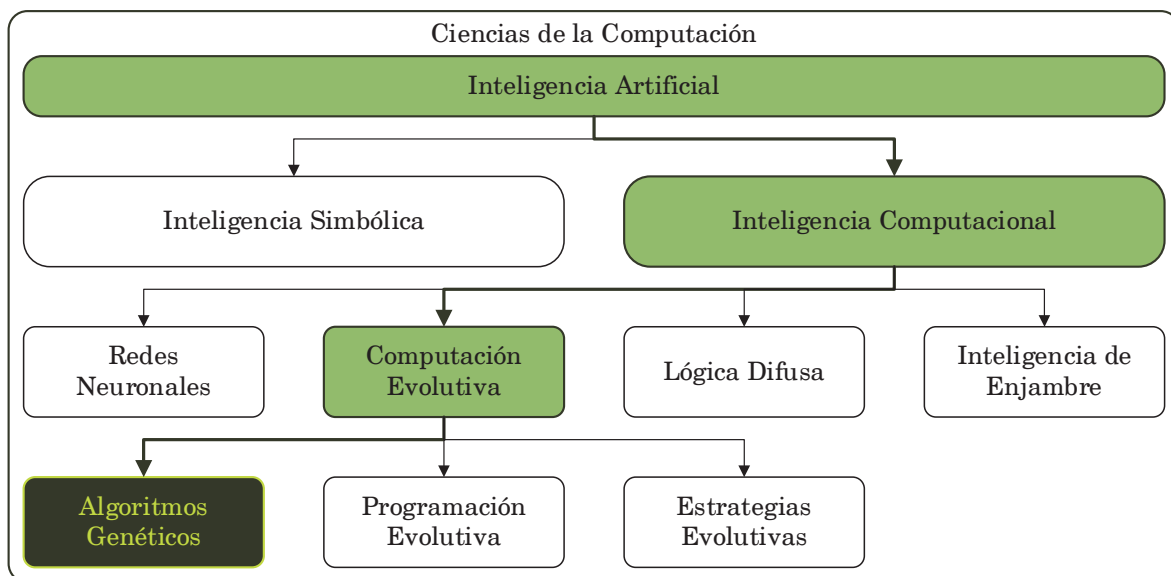
3. INTELIGENCIA COMPUTACIONAL

Como se comentó en los apartados anteriores, el desarrollo que se ve actualmente en la mayoría de los campos del conocimiento, no se hubiera logrado sin la revolución tecnológica que se ha experimentado en la tecnología y los sistemas de información. Obviamente, los sistemas implicados en la gestión de la cadena de suministros en general y en la distribución urbana de mercaderías en particular, no fueron ajenos a los impactos de estos desarrollos.

Es importante en esta instancia, describir los principales paradigmas de la ciencia de la computación, que sustentan la base de las técnicas computacionales implícitas en tales sistemas, por lo que en este apartado se pretende profundizar en estos aspectos, para finalmente presentar de forma más detallada el tipo de técnica de resolución seleccionada para dar solución al problema objeto de estudio, los Algoritmos Genéticos.

En la Figura 5 puede verse el posicionamiento de los Algoritmos Genéticos (AG) en las ciencias de la computación (CC).

Figura 5: Posicionamiento de los AG en las CC



Fuente: elaboración propia

Como se representa en la Figura 5, los AG pertenecen al conjunto de técnicas de resolución de problemas basadas en la emulación de los procesos naturales de evolución englobados bajo el término Computación Evolutiva (CE). La CE pertenece a el conjunto de métodos de búsqueda agrupados bajo nombre de Inteligencia Computacional (IC). La IC es una rama de la Inteligencia Artificial (IA), centrada en el estudio de mecanismos adaptativos para permitir el comportamiento inteligente de sistemas complejos y cambiantes. La IA, forma parte de las ciencias de la computación (CC), es decir, de las ciencias que estudian las bases teóricas de la información y la computación, así como su aplicación en sistemas computacionales. A continuación, se presentan estos conceptos.

3.1. Introducción

Desde la antigüedad el hombre ha diseñado y construido sistemas, máquinas o artefactos, con el objeto de realizar algún tipo de cálculo. Desde Aristóteles, intentando explicar y codificar formas de razonamiento deductivo mediante los silogismos, hasta Turing, tratando de construir máquinas capaces de imitar procesos del cerebro humano. Hoy en día, se dispone de un variado conjunto de modelos, metodologías y técnicas, que dan forma a asistentes inteligentes embebidos en distintos tipos de aplicaciones (desde sistemas de información, como los mencionados previamente, hasta procesadores de textos), que permiten al usuario la resolución de problemas y la realización de tareas de alta complejidad, potenciando sistemas convencionales, mediante la asociación entre inteligencia, información y comunicación (Russell & Norvig, 2004).

El estudio sistemático de procesos algorítmicos que describen y transforman la información: su teoría, análisis, diseño, eficiencia, implementación y aplicación, conforma el cuerpo del conocimiento conocido como las Ciencias de la Computación (Comer, y otros, 1989). Desde un enfoque teórico, la computación puede entenderse como un paradigma o un enfoque para resolver problemas, que se ocupa de construir formalismos matemáticos para determinar algoritmos efectivos para resolver problemas (Contreras, 2012).

La Inteligencia Artificial (IA), es la rama de las Ciencias de la Computación que se encarga del estudio de la inteligencia en elementos artificiales y, desde un punto de vista aplicado, propone la creación de elementos que posean un comportamiento inteligente (Pazos, Pedreira, Rabuñal, & Pereira, 2007).

Esta definición posee dos vertientes, una teórica, que busca una teoría computable del conocimiento humano, es decir, busca comprender y formular la inteligencia humana. Y otra aplicada, que estudia la creación de sistemas informáticos que realicen tareas para las que se precisa inteligencia, buscando la resolución de problemas en los que hay incertidumbre, conocimiento incompleto y necesidad de aprendizaje, sin limitar las técnicas a utilizar a aquellas que utilizan los seres inteligentes. En otras palabras, en esta vertiente, no se busca la comprensión de la inteligencia humana, sino la posibilidad de reescribir de una forma computable los procedimientos usados por los seres humanos para resolver un grupo de problemas científico técnicos, descriptos generalmente de forma poco clara, imprecisa e incompleta (Palma Méndez & Morales, 2008).

Estas dos vertientes, surgen de la evolución histórica de los objetivos de la IA. En agosto de 1955 en el Congreso de Dartmouth, evento es considerado el germen de la IA como campo dentro de las ciencias de la computación, se planteó la conjetura de que cada aspecto del aprendizaje o cualquier otra característica de la inteligencia pueden, en principio, ser descrito con tanta precisión que puede fabricarse una máquina para simularlo (McCarthy, Minsky, Rochester, & Shannon, 1955). En esa conjetura, el campo de la IA no solo intentaría comprender, sino que también se debería construir entidades inteligentes. Posteriormente, los objetivos de la inteligencia artificial han cambiado hacia horizontes más realistas, como el desarrollo de dispositivos que operen como amplificadores de la inteligencia humana. Es decir, la implementación de sistemas convencionales con inteligencia embebida, es un objetivo más práctico, más útil y más factible que la búsqueda de la inteligencia general en un único dispositivo (Palma Méndez & Morales, 2008).

Palma Méndez y Morales (2008), afirman que actualmente los objetivos básicos de la IA son: modelar conocimiento, formalizar los modelos, programar los operadores formales e implementar físicamente el soporte de esos programas. Es decir, la

IA es en esencia una forma de modelar, formalizar, programar e implementar conocimiento. Inspirándose en la biología, la neurofisiología, la psicología cognitiva y la filosofía.

En este contexto, se define la *resolución de problemas*, como la búsqueda de la mejor secuencia de acciones o decisiones, que será ejecutada con posterioridad por un agente, con el fin de alcanzar un objetivo a partir de una situación inicial dada. Un *algoritmo de búsqueda*, capta como entrada un problema y devuelve una solución en forma de secuencia de acciones. Un *agente*, es cualquier cosa capaz de percibir su medioambiente y actuar en ese medio llevando a cabo una acción o una secuencia de acciones determinada (Russell & Norvig, 2004).

Dependiendo del problema concreto, la ejecución de la secuencia de acciones o decisiones tiene asociado al menos un objetivo, ya sea un costo que se busca minimizar, o un beneficio que se busca maximizar (Palma Méndez & Morales, 2008). En general, “*un agente con distintas opciones inmediatas de valores desconocidos puede decidir qué hacer, examinando las diferentes secuencias posibles de acciones que le conduzcan a estados de valores conocidos, y entonces escoger la mejor secuencia*” (Russell & Norvig, 2004).

Por ejemplo, siguiendo a Russell y Norving (2004), un *problema simple de ruteo* de un vehículo que debe ir desde *D0* hasta *N16* (ver Figura 6), puede plantearse formalmente de la siguiente manera:

- *Estado Inicial*: Es el estado inicial en el que comienza un agente. Por ejemplo, el estado inicial para nuestro agente de la figura siguiente, se describe como *En(D0)*. Es decir, se encuentra en el depósito *D0* (Russell & Norvig, 2004).
- *Acciones*: Hace referencia a una descripción de las posibles acciones disponibles por el agente. Por ejemplo, una forma simple de hacerlo es definir una *función sucesor*, $fncSuc(x)$, que dado un estado particular x , devuelve un conjunto de pares ordenados $\langle acción, sucesor \rangle$. Cada *acción* es una de las acciones posibles en el estado x , por ejemplo, *ir a x'* . Cada *sucesor* es un estado que puede alcanzarse desde el estado x , por ejemplo, *en x'* , ejecutando la acción.

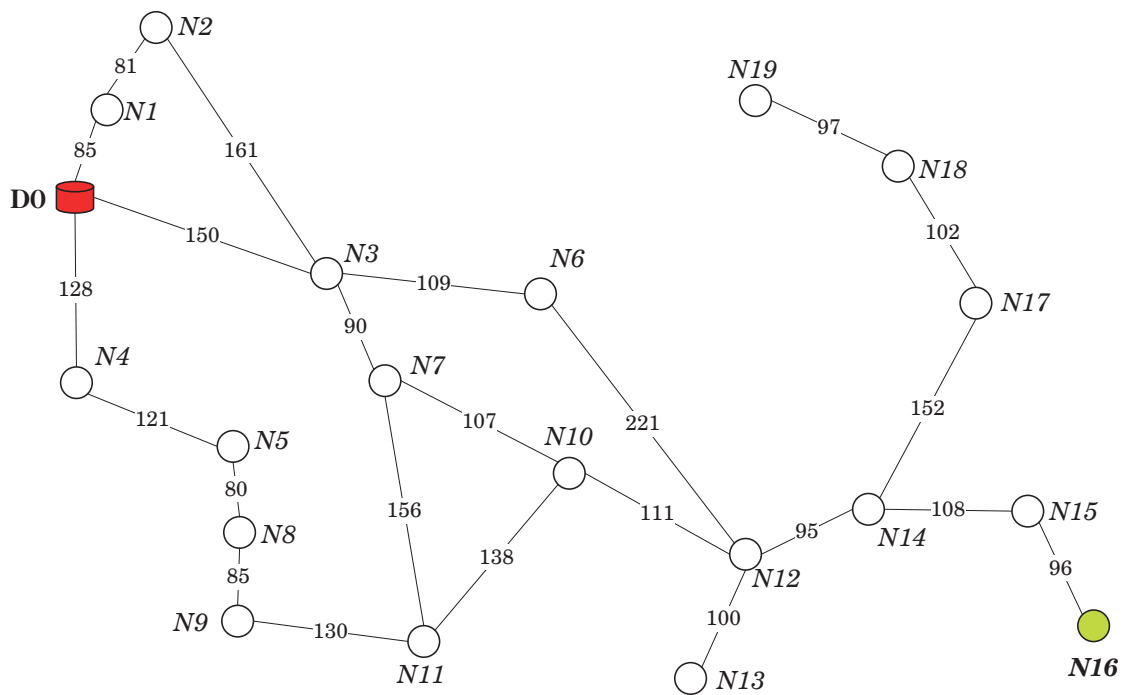
En el ejemplo de ruteo, desde el estado *En(D0)*, la función sucesor devolverá:

$$\{\langle Ir(N1), En(N1) \rangle, \langle Ir(N3), En(N3) \rangle, \langle Ir(N4), En(N4) \rangle\}$$

Es decir, las *acciones* a partir del estado $En(D0)$ son *ir* hacia el nodo 1, 3 o 4, y los estados siguientes serán estar *en* el nodo 1, 3 o 4 (Russell & Norvig, 2004).

- *Espacio de Estados*: Es el conjunto de todos los estados alcanzables desde el estado inicial. A partir del *espacio de estados* se puede construir un grafo⁴ en el cual los *nodos* son los *estados* y los *arcos* entre los nodos son las *acciones*. El gráfico presentado en la Figura 6, puede interpretarse como un grafo del espacio de estados, si vemos cada calle como la acción de que el vehículo recorra ese trayecto. Mientras que un *Camino* en el espacio de estados es una *secuencia de estados conectados por una secuencia de acciones* (Russell & Norvig, 2004).

Figura 6: Problema simple de ruteo



Fuente: adaptado de Russell y Norvig (2004)

⁴ Informalmente, podemos decir que un grafo es una representación de una colección de nodos y de arcos que unen estos vértices. Los vértices se representan como puntos del plano, y las aristas como líneas que unen estos puntos.

- *Test Objetivo*: determina si un estado dado es un estado objetivo. A veces existe un conjunto explicitado de estados objetivo posibles, y el test consiste simplemente en comprobar si el estado pertenece a ese conjunto. El objetivo del agente en nuestro ejemplo, es el conjunto $\{En(N16)\}$, es decir, llegar a estar en el nodo 16. Otras veces, el objetivo puede declararse como una propiedad abstracta más que como un conjunto explicitado de estados. Por ejemplo, el objetivo en el juego de ajedrez, es alcanzar un estado llamado “jaque mate”, que puede describirse como el estado en el que el rey del oponente no tiene escapatoria al ser atacado (Russell & Norvig, 2004).
- *Costo de un camino*: La función *costo de un camino* asigna un costo numérico a cada camino, y se elige de manera que refleje una medida de rendimiento. Por ejemplo, al agente que busca llegar al nodo 16 (*N16*), le interesa la distancia recorrida para lograr este objetivo, así que el *costo del camino* puede describirse como su longitud en kilómetros. Esto puede describirse como la suma de los costos de las *acciones individuales* a lo largo del *camino*. Entonces, el *costo individual* de una acción *a* que va desde un estado *x* al estado *y* se denota por $c(x, a, y)$. Los costos individuales para nuestro ejemplo de ruteo, se muestran en la figura anterior, como las distancias de las calles, por ejemplo, $c(En(N3), ir(N7), En(N7)) = 90$ (Russell & Norvig, 2004).

Estos elementos definen un problema y pueden unirse en una estructura simple de datos que se dará como entrada al algoritmo que lo resolverá. En este contexto con un solo criterio u objetivo, puede asumirse que una *solución* de ese problema, es un *camino* desde el *estado inicial* al *estado objetivo*. Luego, si la *calidad de la solución* se mide por la función *costo del camino*, la solución óptima sería la que tenga el costo del camino más bajo, de entre todas las soluciones o caminos posibles.

Dentro del amplio espectro de las técnicas de búsqueda, una de las ramas con mayor desarrollo en los últimos años, es la de las técnicas inspiradas en la naturaleza, cuya forma de representación de la inteligencia es conocida como *Inteligencia Computacional* (IC).

3.2. Inteligencia Computacional

Uno de los ejes principales en el desarrollo de algoritmos, es el diseño de modelos algorítmicos para resolver problemas cada vez más complejos. Como se verá en el capítulo III, hay una gran cantidad de problemas que requieren algoritmos cuyo esfuerzo computacional es tan grande que, a pesar de la capacidad de cálculo de los ordenadores de hoy en día, se tardaría mucho tiempo en encontrar una solución óptima. Estos problemas se presentan en biología, electrónica, logística, etc., y son problemas de secuenciación, reconocimiento de patrones, visión artificial, percepción, etc. A pesar de la complejidad de estos tipos de problemas, los organismos biológicos los resuelven todos los días, por esta razón han servido como fuente de inspiración para nuevos enfoques para la resolución de dichos problemas (Pérez, 2010).

Tal es así que, en los últimos años se han logrado grandes éxitos en el desarrollo de algoritmos para resolver este tipo de problemas a través del modelado de la inteligencia biológica y natural, dando lugar a la Inteligencia Computacional. Esta rama de la IA inspirada en la naturaleza, puede ser vista entonces como una combinación de varias disciplinas, por ejemplo, informática, fisiología, filosofía, sociología y biología (Engelbrecht, 2007).

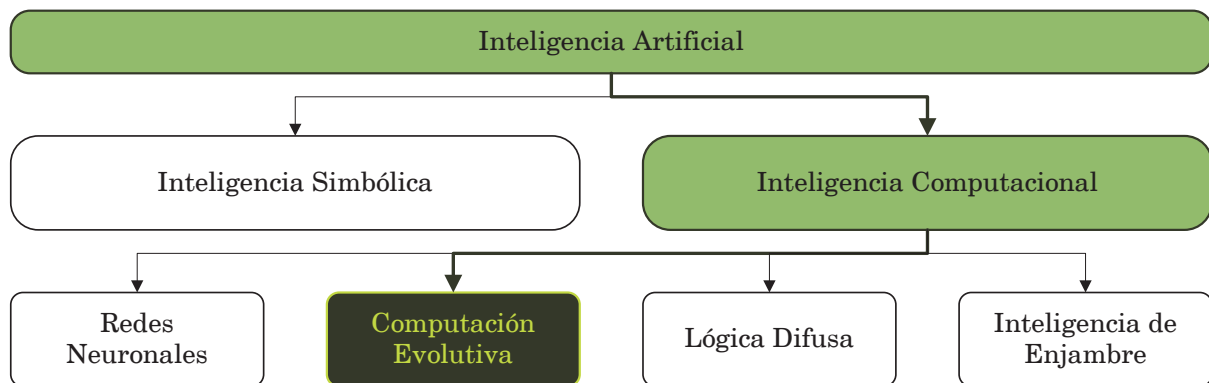
La *Inteligencia Computacional* (IC) es la rama de la inteligencia artificial que se ocupa del estudio de mecanismos adaptativos para permitir o facilitar el comportamiento inteligente en entornos complejos y cambiantes. Estos mecanismos incluyen una variedad de paradigmas inspirados en la naturaleza que muestran una habilidad para aprender o adaptarse a nuevas situaciones (Engelbrecht, 2007).

En otras palabras, la IC se ocupa de la teoría, diseño, desarrollo y aplicaciones de paradigmas computacionales que utilizan una nueva forma de representación de la inteligencia inspirada en la naturaleza, y aglutina metodologías como las *Redes Neuronales*, inspiradas en el conocimiento que tenemos actualmente del funcionamiento del cerebro (Engelbrecht, 2007). La *Lógica Difusa*, para la representación de conocimiento ambiguo y la imitación de los procesos del pensamiento humano con información vaga e imprecisa (Engelbrecht, 2007). La *Computación Evolutiva*, inspirada en los mecanismos de la evolución biológica (selección natural y evolución). Las técnicas basadas en la interacción social de los animales, deno-

minada también inteligencia de enjambre (SI, *swarm intelligence*) (Denby & Le Hégarat-Masclé, 2003) (Lim & Jain, 2009), como es el caso del comportamiento colectivo de las *Colonias de Hormigas* (ACO, *Ant Colony Optimization*) (Dorigo, Maniezzo, & Colorni, 1996). O las técnicas inspiradas en las sincronías de las bandadas de pájaros o peces, que definen el algoritmo de optimización por enjambre de partículas (PSO, *Particle Swarm Optimisation*) (Kennedy & Eberhart, 1995).

Entre este conjunto de metodologías de búsqueda agrupadas bajo el nombre de Inteligencia Computacional, se profundizará en el espectro de técnicas basadas en la emulación de procesos naturales de evolución, aglutinadas bajo el término de Computación Evolutiva.

Figura 7: Posicionamiento de la Computación Evolutiva



Fuente: elaboración propia

3.3. Computación Evolutiva

La *Computación Evolutiva* (CE) comprende una serie de modelos computacionales inspirados en procesos evolutivos, que son usados para abordar problemas complejos de búsqueda y aprendizaje (Coello & Zacatenco, 2015). Las implementaciones de esos modelos, se conocen como *Algoritmos Evolutivos* (AEs).

Como algoritmos de búsqueda, los AEs, se diseñan para explorar sistemáticamente el espacio de estados, inspirándose en el proceso de evolución natural. Se comienza con k estados (llamados población) generados aleatoriamente y luego se generan los estados sucesores (descendientes), combinando (reproducción sexual) dos estados padres. Luego se seleccionan aleatoriamente k sucesores, con la probabilidad

de elegir a un sucesor como una función creciente del valor de la función de costo. Posteriormente se repite el proceso, hasta alcanzar un estado objetivo (Russell & Norvig, 2004).

Desde el punto de vista de la *Algoritmia*, los AEs son algoritmos probabilistas, en el sentido de que toman decisiones de forma aleatoria y por lo tanto el mismo algoritmo puede comportarse de forma diferente al aplicarse sobre los mismos datos. Mientras que desde el punto de la *inteligencia artificial*, los AEs son algoritmos heurísticos que permiten encontrar buenas soluciones en tiempos razonables mediante la evaluación del progreso logrado en la búsqueda de un resultado final (Palma Méndez & Morales, 2008). En resumen, son algoritmos cuyo propósito es guiar un proceso de búsqueda estocástica, haciendo evolucionar un conjunto de estructuras y seleccionar de modo iterativo las más adecuadas (Aguilar, 2007).

3.3.1. Fundamentos Biológicos

El fundamento biológico de los algoritmos evolutivos está en los principios de la Teoría Neo-Darwiniana de la *evolución natural*. Según estos principios, la vida en este planeta se puede explicar por un conjunto de procesos que actúan sobre, y dentro, de las poblaciones y especies, la *reproducción*, la *mutación*, la *competencia* y la *selección* (Hoffman, 1989). En un sistema que se *reproduce* continuamente a sí mismo, manteniendo un cierto equilibrio, está garantizada la *mutación* (Fogel D. B., 1995). Además, el hecho de contar con un *espacio restringido* para el desarrollo de la vida en este sistema, garantiza la existencia de la *competencia*. La *selección*, sería la consecuencia natural del *exceso de organismos que han llenado el espacio de recursos disponibles*.

El resultado de la interacción entre estos procesos estocásticos, de forma iterativa y a través de las generaciones, es la *evolución* (Coello & Zacatenco, 2015). El proceso de evolución puede ser visto como una búsqueda de los mejores individuos entre un conjunto enorme de posibles candidatos, mediante una serie de transformaciones de la composición genética de las poblaciones, debidas principalmente a las interacciones entre los individuos, y de estos con el ambiente.

3.3.2. Funcionamiento genérico de un AE

Previo al análisis de la emulación de estos procesos evolutivos, mediante procedimientos computacionales, es conveniente explicitar una serie de *conceptos biológicos* que se repetirán reiteradamente con posterioridad.

Primero, está el concepto de *individuo*, que es un solo miembro de una población. Una *población*, es un grupo de individuos, entre los que se da una interacción permanente, por ejemplo, para reproducirse. Cada individuo tiene *rasgos visibles* (u *observables*), denominados en conjunto *fenotipo*. También tiene una *composición genética*, denominada *genotipo*. La relación entre estos dos conceptos consiste en que el genotipo da origen al fenotipo después de un proceso de desarrollo (Coello & Zacatenco, 2015).

Por otro lado, el conjunto de informaciones genéticas que llevan todos los miembros de una población, se denomina *acervo génico*. Una *especie*, es un conjunto de poblaciones de individuos, aislado reproductivamente de otras especies, que pueden reproducirse entre sí (Mayr, 1982). Una población es una entidad genética abierta, es decir que puede intercambiar genes con otras poblaciones de la misma especie. Una especie es una entidad genética cerrada, es decir que no puede intercambiar genes con otras especies.

El *proceso de evolución natural*, consiste en el cambio de las características hereditarias de grupos de organismos a través de las generaciones, y ocurre como consecuencia de la interacción básica, y sostenida en el tiempo, de cuatro procesos elementales (Cabrero & Camacho, 2002):

- *Reproducción*, proceso mediante el que los individuos de una misma población *generan descendencia*. Puede ser *asexual* (un solo padre) o *sexual* (dos padres).
- *Mutación*, proceso que introduce modificaciones al azar en la información genética transmitida de padres a hijos.
- *Competencia*, consecuencia natural de que las poblaciones crezcan en un *espacio finito de recursos*.

- *Selección*, es el resultado generado por las diferencias en la eficacia biológica, y se manifiesta en la tasa de supervivencia o reproducción entre individuos portadores de diferentes genotipos. Es decir, la diferente viabilidad de los individuos, se traduce en un número diferente de descendientes (por lo tanto de genes) a la siguiente generación (Cabrero & Camacho, 2002). Consecuentemente, puede definirse la *Selección Natural* como “el proceso por el que las formas de organismos en una población, que están mejor adaptadas al ambiente, incrementan su frecuencia relativamente a las formas menos adaptadas en un número de generaciones” (Ridley, 1993).

Es importante aclarar algunos términos relevantes a los fines de que se comprenda la terminología utilizada *en el campo de la computación evolutiva*.

Cuando se habla de *Individuo*, se hace referencia a una solución del espacio de búsqueda, representada por un cromosoma.

Un *Cromosoma* es una estructura de datos que contiene una cadena de parámetros o genes, y representa un individuo de la población, una solución. Normalmente, es un arreglo o cadena de enteros donde cada posición es conocida como gen. Cada *gen* usualmente codifica el valor de un sólo parámetro o atributo de la solución del problema. Y el valor que puede tomar cada gen se conoce como *alelo* (por ejemplo, en representación binaria los alelos son 0 y 1), y la posición de un gen en el cromosoma, es conocida como *locus*.

Los objetos que forman posibles soluciones dentro del contexto del problema original, se los denomina *Fenotipos*, mientras que su codificación, es decir, los individuos dentro del algoritmo evolutivo, se les llama *Genotipos*.

El *Genotipo* es el individuo codificado en los cromosomas y genes. Y el *Fenotipo* son los valores decodificados del cromosoma, de manera que puedan sustituirse en la función de aptitud. Así se constituyen los dos niveles de representación: genotípico y fenotípico.

Una *Población*, es un conjunto finito de individuos.

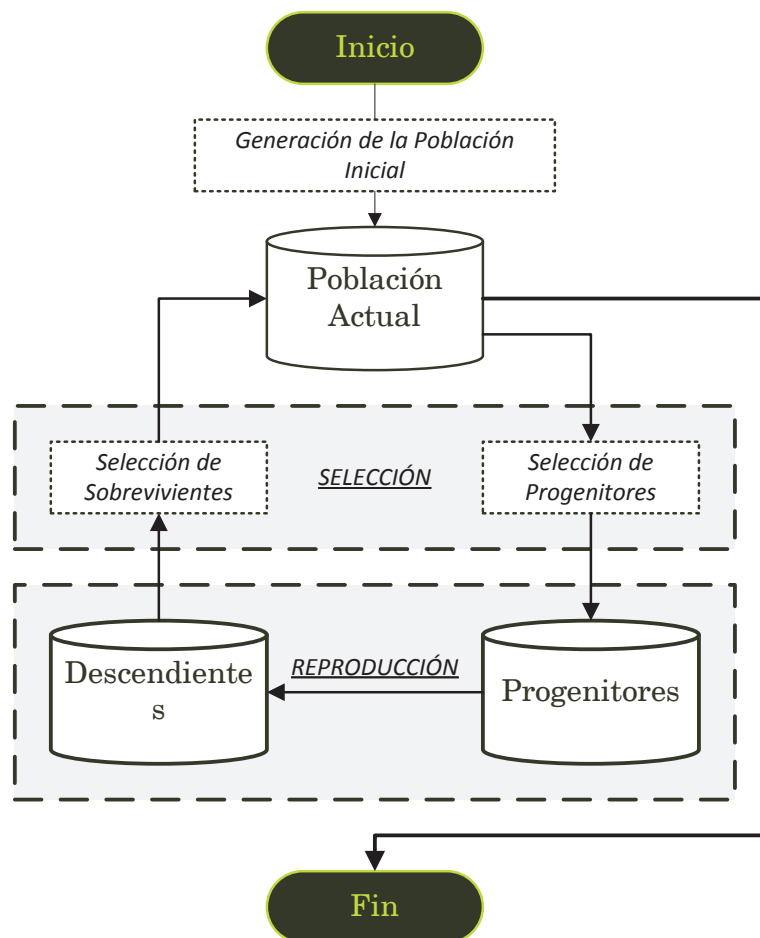
La *Aptitud*, es el valor que se le asigna a cada individuo de una población, y que indica que tan bueno es con respecto a los demás para la solución de un problema.

Se calcula mediante la función de aptitud, la cual, en ocasiones puede ser igual al valor de la función objetivo del problema.

Un *Superindividuo*, es un cromosoma que tiene un valor de adaptación muy alto en comparación con el resto de los cromosomas de la población.

En la Figura 8, puede verse el esquema genérico de un algoritmo evolutivo muy básico, que busca emular el proceso de evolución natural.

Figura 8: Diagrama de flujo de un AE básico



Fuente: Elaboración propia en base a Coello y Zacatenco (2015)

En el esquema se muestra que, el algoritmo evolutivo inicia generando una *población inicial* de posibles soluciones del problema, correctamente representadas mediante individuos creados aleatoriamente. Luego se somete a esta población inicial a un *proceso de selección* de los mejores individuos que, serán los progenitores de la próxima generación, basado en las medidas de calidad de cada individuo, conocidas como *valor de aptitud* (o *fitness*). Posteriormente, se toma esta población de

progenitores y se los somete a un *proceso de transformación o reproducción* (cruce, mutación, etc.) para generar, a partir de estos padres actuales, nuevos individuos candidatos a solución (*población de descendientes*), que a su vez competirán por su sobrevivencia con los anteriores, para tener lugar en la población de la próxima generación.

Una vez conformada la población de la nueva generación, se repite todos los procesos (selección de progenitores, reproducción y selección de sobrevivientes) hasta que se encuentra un *candidato con calidad suficiente* o se alcanza un *límite de cómputo establecido previamente* (satisfacción de una condición de paro).

3.3.3. Modelos Evolutivos

Históricamente sus bases teóricas, se remontan a fines de los años 60s y principios de los 70s, pero en la práctica, estas ideas resultaban poco eficientes. Recién en la década del 80, con la aparición de computadoras con altas prestaciones y bajo costo, es que cambia la aplicabilidad para estas técnicas computacionales. Empiezan a ser usadas para resolver problemas de ingeniería y ciencias sociales que hasta ese momento eran abordados de forma precaria con métodos convencionales. A fines de los 80s inicios de los 90s, aparecen los libros de Goldberg "*Genetic Algorithms in Search, Optimization and Machine Learning*" (1989), y de Davis "*Handbook of Genetic Algorithms*" (1991), se plantean y resuelven problemas "de la vida real" con los que se empiezan a producir intereses económicos (Palma Méndez & Morales, 2008).

Por las diferencias en sus fundamentos biológicos, tradicionalmente se identifican por lo menos tres modelos de algoritmos evolutivos (Coello & Zacatenco, 2015):

Programación Evolutiva (PE)

Modelo propuesto en 1966 por Lawrence J. Fogel (1966), para simular la evolución como un proceso de aprendizaje, con el objetivo de crear inteligencia artificial (Fogel D. B., 1995). Este enfoque da relevancia a los nexos hereditarios y de comportamiento entre padres y sus descendientes, sin buscar emular operadores genéticos específicos. Ve al comportamiento adaptativo como una forma de inteligencia,

conciéndola como la capacidad de un sistema para adaptar su comportamiento a entornos diferentes y así lograr cumplir ciertas metas específicas.

El modelo opera a nivel de especies, es decir que cada individuo se ve como correspondiente a una especie distinta (Eiben & Smith, 2015). Por lo tanto, no utiliza operador genético de cruce (diferentes especies no pueden cruzarse entre sí). La *selección* de individuos se realiza de manera probabilística y el *operador genético* principal es la mutación; además opera a nivel fenotipo.

Algoritmo básico de la programación evolutiva (Coello & Zacatenco, 2015):

1. Generar aleatoriamente una población inicial.
2. Aplicar mutación.
3. Calcular aptitud de todos los individuos de la población.
4. Seleccionar mediante torneo (normalmente probabilístico) los individuos que sobrevivirán.

Estrategias Evolutivas (EE)

Las estrategias evolutivas (*Evolutionstrategien*) fueron desarrolladas inicialmente en Alemania en 1964 por Ingo Rechenberg, Hans-Paul Schwefel y Paul Bienert (Schwefel, 1995) (Beyer & Schwefel, 2002). Surgieron como métodos estocásticos diseñados específicamente para resolver problemas hidrodinámicos de alto grado de complejidad. Con el tiempo fueron incorporando distintos aspectos de la computación evolutiva, hasta convertirse en un modelo más de este tipo de procedimientos. Actualmente son algoritmos evolutivos que utilizan una representación de vectores reales, selección determinista y operadores específicos de mutación.

Estos modelos operan a nivel fenotípico de los individuos, por lo que existe un operador de cruce, sexual (dos padres) o asexual (un sólo padre). Sin embargo, es un operador secundario con respecto al de mutación.

La mutación es el operador primario y se utiliza con valores Gaussianos. Estos valores son los que permiten que las EE sean auto-adaptativas, ya que evoluciona no sólo a las variables del problema, sino también los parámetros mismos de la técnica (es decir, las desviaciones estándar).

Su mecanismo de selección es determinístico y extintivo (los peores individuos tienen probabilidad cero de sobrevivir).

En su versión original (1+1)-EE no contemplaban el concepto de población. Existía un sólo padre y a partir de él se genera un nuevo individuo mediante la siguiente expresión: $x_{t+1} = x_t + N(0, \sigma)$ donde t es la generación y $N(0, \sigma)$ es un vector de números Gaussianos independientes con media cero y desviación estándar σ . Si el hijo es mejor en aptitud se mantiene, de lo contrario se elimina.

En versiones posteriores las EE comienzan a utilizar una población, en la que hay μ padres y se genera un sólo hijo el cual puede reemplazar al peor padre.

En 1975 Schwefel introdujo el uso de múltiples hijos en las EE, considerando dos técnicas. Una preservando los mejores individuos obtenidos de la unión de padres e hijos. Otra permitiendo que sobrevivan sólo los mejores hijos de la siguiente generación.

Algoritmos Genéticos (AG)

Los algoritmos genéticos, fueron creados a inicios de la década del 60 por John H. Holland (Holland, 1962 a) (Holland, 1962 b) (Holland, 1992). Posiblemente sean la técnica evolutiva más popular. Definidos por Goldberg (1989) como:

“algoritmos de búsqueda basados en el mecanismo de selección natural y genética natural. Combinan la supervivencia del más apto entre una estructura dada de cadenas con un intercambio aleatorio de información para conformar un algoritmo de búsqueda con algo del talento de la búsqueda humana”. (Goldberg, 1989)

El AG trabaja a nivel genotípico y su operador primario es el de cruce sexual, ya que modela la evolución a nivel de individuos. Su operador secundario es la mutación. La manera de seleccionar los individuos es probabilística basada en sus aptitudes. Normalmente no son auto-adaptativos.

El algoritmo opera sobre una población P de soluciones codificadas (individuos). La representación tradicional usada para codificar este conjunto de soluciones es el esquema binario. Para cada individuo $i \in P$ se define una función de aptitud o *fitness* $f(i)$ de modo que cuanto mayor es el *fitness* de un individuo mejor es la solución que éste representa. Se aplican operadores evolutivos en cada iteración que combinan y modifican a los individuos de la población creando una nueva.

Mediante el *operador de selección* se eligen individuos de la población que tendrán la posibilidad de reproducirse. En general, este operador genera una población intermedia cuya cantidad de individuos depende de las características del operador de cruzamiento utilizado.

Habitualmente los operadores de selección son probabilísticos y suelen dar una mayor ponderación a individuos con mayor *fitness* en la población. Las técnicas de selección usadas en algoritmos genéticos básicos pueden clasificarse en tres grandes grupos (Coello & Zacetenco, 2015):

Selección proporcional :

Técnicas propuestas originalmente por Holland (1975) en las que se eligen individuos de acuerdo a su contribución de aptitud con respecto al total de la población. En la selección proporcional la probabilidad de seleccionar al individuo i es proporcional a $f(i)/\bar{f}$, siendo $\bar{f} = (1/|P|) \times \sum_{i \in P} f(i)$ el *fitness* promedio de la población.

Existen diferentes variantes dentro de este grupo, por ejemplo la ruleta, el sobran-te estocástico, el muestreo determinístico, etc.

Selección por torneo :

Técnica propuesta por Wetzel (1983). En la selección por torneo, cada individuo se toma eligiendo el mejor de un conjunto de k individuos tomados al azar de la población actual. Esto se repite iterativamente hasta completar la muestra. El parámetro k suele ser 2 o 3.

Selección Uniforme :

Técnica propuesta por Whitley (1989). En algunos casos es común el uso de mecanismos en los que cada persona tiene la misma oportunidad de ser seleccionada. Esto iría en contra del principio de la selección natural en el que se basan los AGs, pero no es así, al combinarlo con un mecanismo de selección de sobrevivientes fuertemente basado en la aptitud (Eiben & Smith, 2015).

Una vez que se generó la población intermedia se aplican repetidas veces operadores de reproducción para combinar individuos de dicha población y originar una nueva.

El procedimiento básico de un AG es el siguiente (Buckles & Petry, 1992):

Algoritmo Genético Simple

- 1: Generar aleatoriamente una población inicial $P(0)$ y hacer $t=0$
 - 2: **Repetir**
 - 3: Evaluar el *fitness* (aptitud) de cada individuo en $P(t)$
 - 4: Seleccionar padres de $P(t)$ de acuerdo al fitness
 - 5: Aplicar operadores de reproducción para producir la siguiente generación, población $P(t+1)$
 - 6: Hacer $t=t+1$
 - 7: **Hasta** alcanzar criterio de parada.
-

Es importante destacar, antes de cerrar este apartado en el que describimos las características de los AGs, algunos aspectos relevantes de la representación.

Holland (1975), comparo diferentes representaciones y concluyó que las codificaciones con reducido número de alelos y cadenas largas, favorecían un elevado paralelismo implícito, mejorando la eficiencia de los AGs. Planteándose de esta manera a la codificación binaria, como la representación más adecuada. Pero también demostró que el paralelismo implícito de los AGs no impide usar alfabetos de mayor cardinalidad.

Esto es importante, dado que en muchas aplicaciones la codificación binaria resulta muy rígida, generando que los AGs tengan serias dificultades para considerar restricciones no triviales del espacio de búsqueda, particularmente en el caso de problemas de optimización combinatoria basados en el orden. Por ejemplo, en el caso de problemas de optimización combinatoria como el CVRP donde la representación más natural son permutaciones de números enteros (conocida en la literatura inglesa como *Path Representation*).

Michalewicz (1992) propuso una estrategia de incorporación directa de conocimiento específico del problema en el AG, mediante la representación. De esa manera pudo dar mayor flexibilidad a la representación, para considerar restricciones en el espacio de búsqueda. A esta variante de los algoritmos genéticos, se la conoce como *programas de evolución*. Comparten la estructura básica de un AG, diferenciándose solamente en dos aspectos. Primero en que buscan utilizar una represen-

tación lo más natural posible para el problema que se busca resolver (vectores de números reales, de números naturales, matrices, árboles, etc.). Y segundo, en que los operadores genéticos deben ser específicos para esa representación, adaptables al dominio y lo más cerrados⁵ posible.

La metodología de resolución seleccionada para resolver el problema objeto de estudio, pertenece a este tipo de algoritmos genéticos, también conocidos como programas de evolución, especializados en optimización combinatoria.

4. TOMA DE DECISIÓN MULTI-OBJETIVO

4.1. Introducción

La viabilidad de una organización, está ligada al nivel de desempeño de sus sistemas de inteligencia, y una parte importante de estos, son los que están vinculados a su proceso de toma de decisiones y solución de problemas, incluyendo la experimentación en los estados más maduros de los procesos decisionales (Puche Regaliza, Costas Gual, & Arranz Val, 2016).

Siguiendo a Simon, el proceso de toma de decisiones por un agente, implica una racionalidad, un tipo de conducta (o modo de proceder), que es apropiado para la obtención de determinados fines, dentro de los límites dados por las condiciones y restricciones impuestas por una situación o contexto específico (Simon, 1972). La toma de decisiones se da en un contexto, que moldea el proceso de selección de una opción entre varias, dado que requiere del empleo de facultades inherentes al agente decisor, y a la situación en particular. Luego, la conducta racional en la realidad está limitada tanto por el *entorno interno* de las mentes de las personas (capacidades cognitivas, capacidades computacionales, contenidos en memoria y procesos), como por el *entorno externo* del mundo en el que actúan, y al que deben adaptarse, y por lo tanto, que actúa en ellos (Simon, Racionalidad limitada en Ciencias Sociales: Hoy y mañana, 2003).

⁵ Cuando hablamos de operadores cerrados, hacemos referencia a que las entradas y las salidas son individuos factibles. Es decir, que cumplen con todas las restricciones impuestas en el modelo. Por lo que no se deben aplicar estrategias de penalización o de reparación por fuera de los operadores.

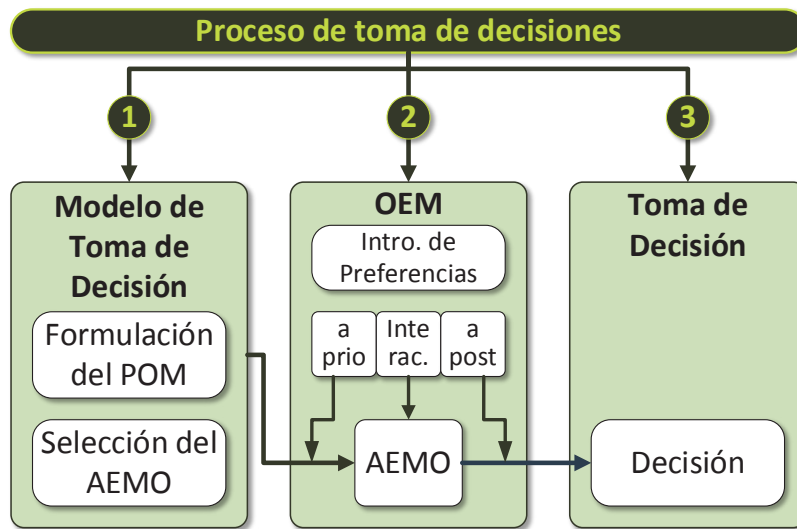
Elementos como el conocimiento del mundo que tienen o dejan de tener quienes toman decisiones, o su habilidad o falta de habilidad para recordar ese conocimiento en el momento en que sea relevante, limitan la racionalidad del proceso de toma de decisiones, y llevan a que personas o equipos diferentes, tomen decisiones distintas contando con exactamente la misma información (Bonome, 2010).

Entendiendo que el comportamiento es racional si la conducta del agente se adapta bien a sus fines, entonces la racionalidad está condicionada por el conjunto de habilidades o aptitudes que se usa para encontrar el curso de acción que conduzca al agente a la realización de sus fines. Entonces en este sentido, el problema pasa por, mejorar las aptitudes o capacidades que restringen este proceso de toma de decisiones, buscando aumentar su racionalidad.

La idea de racionalidad limitada de Herbert Simon, es lo que motivó el interés en el uso de un enfoque para resolver problemas de decisión, en el que se permite al agente decisor incorporar sus preferencias. Es decir, se asume que el agente decisor posee conocimiento específico del problema y que puede establecer preferencias parciales. Luego esta información es potenciada por las capacidades de la inteligencia computacional para guiar el proceso de búsqueda hacia regiones de interés. Es decir, no se busca modelar las preferencias del agente decisor y a partir de ellas determinar la acción que le permita maximizar la utilidad subjetiva, sino que se utiliza un enfoque de mejora en el proceso de decisión, usando la inteligencia computacional como herramienta de apoyo a la toma de decisiones de seres racionalmente limitados, que permite aumentar su racionalidad mediante el incremento en la capacidad de procesamiento simbólico y de almacenamiento información, interactuando con el decisor al permitirle guiar el proceso de búsqueda, hacia donde de otra forma no llegaría debido a su racionalidad limitada.

Salazar, Carrasquero y Galván (2005), plantean que un proceso de toma de decisiones con un enfoque multi-objetivo, se puede separar en tres fases (ver Figura 9):

Figura 9: Proceso de toma de decisiones



Fuente: elaboración propia en base a Salazar, Carrasquero y Galván (2005)

Como se representa en la Figura 9, el proceso consta de una primera fase relacionada a la *construcción del modelo del problema de decisión*. En esta fase se realiza la construcción del modelo analítico del problema de optimización multi-objetivo (POM), tema que será presentado de forma introductoria en la siguiente sección, y se llevará a cabo formalmente en el capítulo III. En esta fase también se selecciona un método o técnica de resolución adecuado, estos métodos se presentan en la sección 5 de este capítulo y en el capítulo IV se presenta la técnica seleccionada, que como se verá posteriormente, es un AEMO (Algoritmo Evolutivo Multi-objetivo).

En una segunda fase, se lleva a cabo el *proceso de optimización* o de búsqueda de buenas soluciones. En esta fase se debe considerar los diferentes enfoques en cuanto al momento en el que el decisor expresa sus preferencias. Como veremos en la subsección 4.3 de este capítulo, se identifican tres estrategias principales en este sentido: La introducción de preferencias *a priori*, en la que las preferencias son introducidas antes de que inicie la exploración. Por lo que implica que el decisor tiene gran conocimiento del dominio del problema. La introducción de preferencias de forma interactiva, que incluye métodos en los que la toma de decisión y la optimización se desarrolla en forma interactiva. Y finalmente la introducción de preferencias *a posteriori*, en las que el decisor elige a partir de todo el conjunto completo de soluciones eficientes. En esta fase se realiza la ejecución del proceso de optimización, esto se presenta en el capítulo IV.

Finalmente, en la última fase es la conclusión del proceso, y consiste en la selección por parte del decisor de una sola alternativa como solución del problema.

4.2. Optimización

La solución final de un problema de decisión es resultado de las tres fases descritas en el apartado anterior. Ahora se busca realizar una breve introducción a los conceptos relativos a la optimización en el proceso de toma de decisiones. Para poder introducirnos posteriormente en el abordaje de las técnicas de resolución relevantes para el presente trabajo.

La *teoría de optimización* es la rama de la matemática aplicada, que se dedica al estudio de los óptimos y las técnicas para determinarlos. Las *técnicas de optimización*, son las técnicas, métodos, procedimientos y algoritmos que se utilizan para hallar un óptimo. En general puede decirse que una técnica o proceso de optimización, es un proceso mediante el que se busca obtener lo “mejor” (si es posible) de una determinada medida (Antoniou & Lu, 2007). Es decir, buscamos lo mayor (o máximo), o lo menor (o mínimo), según sea lo “mejor”. Luego, mediante el término *óptimo*, se hace referencia a un máximo o mínimo, dependiendo de las circunstancias; y mediante el término *optimizar*, al proceso para alcanzar dicho óptimo.

Para poder realizar este proceso de optimización, el problema debe admitir más de una solución, entonces la optimización puede lograrse mediante la búsqueda de la mejor solución al problema en términos de algún criterio de desempeño. Puede verse, que búsqueda y optimización son dos caras de una misma moneda, la diferencia está en que al usar el término búsqueda, se trata de enfatizar el proceso en sí, mientras que, al usar el término optimización, se destaca más el resultado de dicho proceso.

4.2.1. Optimización mono-objetivo

El objetivo principal de un proceso de optimización mono objetivo, es encontrar el óptimo (o los óptimos) globales de un problema de optimización mono-objetivo (Coello & Zacatenco, 2015).

En general, un *problema de optimización mono-objetivo* se define como, el problema de buscar un $x \in \Omega$ que minimice $f(x) \in \mathbb{R}$, sujeto a $g_i(x) \leq 0$, para $i = 1, \dots, m$, y a $h_j(x) = 0$ para $j = 1, \dots, p$.⁶

Donde, $\Omega \in \mathbb{R}^n$, y a cada elemento x de Ω se le dice *solución factible*, dado que debe satisfacer las restricciones (las inecuaciones $g_i(x) \leq 0$ y las ecuaciones $h_j(x) = 0$). Luego a Ω , se le llama *región factible*. A $f: \Omega \rightarrow \mathbb{R}$, se le llama *función objetivo*.

Esta búsqueda de x^* de manera que $f(x^*)$ sea lo menor posible, puede concluir con uno de los resultados siguientes (Salazar González, 2001):

Problema no factible: cuando no existe ninguna solución factible, $\Omega = \emptyset$.

Problema no acotado: cuando existen soluciones que hacen descender infinitamente a la función objetivo. Es decir, para cualquier $v \in \mathbb{R}$, siempre existe un $x \in \Omega$ tal que $f(x) < v$.

Problema con óptimo: cuando existe un $x^* \in \Omega$ tal que $f(x^*) \leq f(x)$ para todo $x \in \Omega$. En este caso x^* recibe el nombre de solución óptima (global) siendo no necesariamente la única en Ω con esta propiedad. Por otro lado, un punto x' se dice que es solución óptima local cuando existe un $\varepsilon > 0$ tal que $f(x') \leq f(x)$ para todo $x \in \Omega$ con $\|x' - x\| < \varepsilon$, para alguna norma predefinida en el espacio vectorial \mathbb{R}^n .

4.2.2. Optimización multi-objetivo

La diferencia principal entre los *Problemas de Optimización Multi-objetivo* (POM) y los mono-objetivo, es que ahora un decisor desea optimizar simultáneamente varios objetivos, que conforman un espacio multi-dimensional de objetivos que usualmente están en conflicto entre sí.

En general, un POM consiste en buscar un vector de variables de decisión, que satisfaga las restricciones y optimice una función vectorial cuyos elementos son las funciones objetivo. Estas funciones son una descripción matemática de los criterios de performance del decisor. Por lo tanto, en este contexto, optimizar es encon-

⁶ Si se busca un máximo, puede reformularse así: $\max \{f(x): x \in \Omega\} = -\min \{-f(x): x \in \Omega\}$

trar aquella solución que daría valores, aceptables para el decisor, a todas las funciones objetivo (Osyczka, 1985).

Más formalmente, un POM puede definirse como, el problema de buscar un $x \in \Omega$ que minimice $F(x) \in \Delta$, sujeto a $g_i(x) \leq 0$ para $i = 1, \dots, m$, a $h_j(x) = 0$ para $j = 1, \dots, q$, y a $B_k^{inf} \leq x_k \leq B_k^{sup}$ para $k = 1, \dots, n$. $F: \Omega \rightarrow \Delta$, donde $\Omega \in \mathbb{R}^n$ y $\Delta \in \mathbb{R}^p$ con $p \geq 2$.

Siendo B^{inf} y B^{sup} los vectores de valores de los límites inferiores (B_k^{inf}) y superiores (B_k^{sup}) respectivamente, de cada una de las x_k variables del vector de variables de decisión x . Los valores de x que verifican las n restricciones de $B^{inf} \leq x \leq B^{sup}$, definen el *Espacio de Búsqueda* (ver Figura 10).

Los valores de x que verifican las $n + m + q$ restricciones (conformadas por las q ecuaciones $h_j(x) = 0$ y las $n + m$ inecuaciones $g_i(x) \leq 0$ y $B_k^{inf} \leq x_k \leq B_k^{sup}$) definen el *Espacio Factible*, Ω (ver Figura 10).

Luego el vector $x = (x_1, \dots, x_n) \in \Omega$ es un *vector solución* de las n variables de decisión. $F(x)$ se conoce como *función de evaluación* del MOP, y es una función vectorial que agrupa las p funciones objetivo a minimizar⁷. Esta función mapeará las n variables de decisión $x = (x_1, \dots, x_n) \in \Omega$ a un vector $z = (z_1, \dots, z_p) \in \Delta$.

Siguiendo a Méndez Babey (2008), en un proceso de optimización multi-objetivo, el decisor razona en términos de la evaluación de una solución x con respecto a cada objetivo $f_1(x)$, situándose así en el espacio de objetivos. Y para representar el conjunto de soluciones factibles en el espacio de los objetivos, es necesario determinar las imágenes de cada una de las soluciones factibles del espacio de decisión. Obteniéndose así el conjunto:

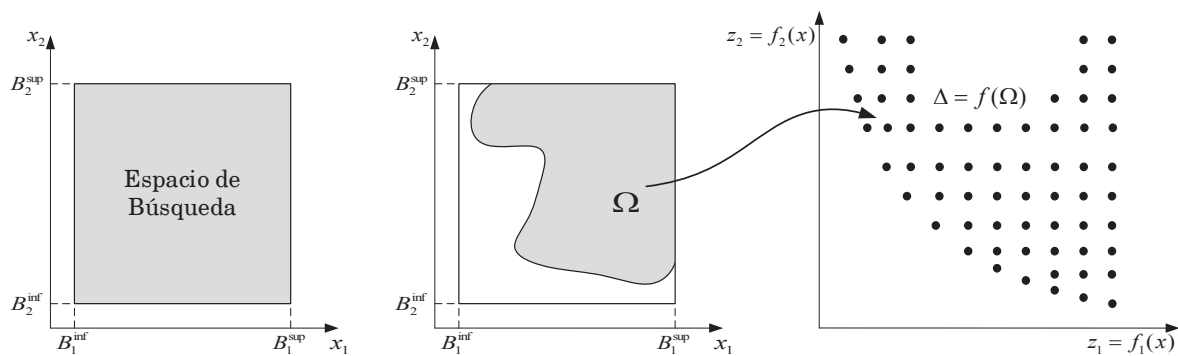
$$\Delta = \{z = (z_1, \dots, z_p) \in \mathbb{R}^p: z_1 = f_1(x), z_2 = f_2(x), \dots, z_p = f_p(x), \forall x \in \Omega\}$$

⁷ Si se busca maximizar, puede reformularse así: $\max\{F(x): x \in \Omega\} = -\min\{-F(x): x \in \Omega\}$

Donde Δ representa el conjunto de imágenes de las soluciones factibles, en el espacio de los objetivos, y el vector $z = (z_1, \dots, z_p)$ representa la imagen de una solución factible en el espacio de los objetivos⁸ (ver Figura 10).

En la Figura 10 se presenta la representación de los mencionados espacios, para un MOP con $\Omega \in \mathbb{R}^2$ y $\Delta \in \mathbb{R}^2$.

Figura 10: Representación del proceso de mapeo



Fuente: elaboración propia en base a Méndez Babey (2008)

Como puede verse en la Figura 10, el hecho de que existan múltiples objetivos implica que no existirá una única solución óptima al problema, sino un conjunto de soluciones que implicarán diferentes compromisos entre los valores de las funciones a optimizar. Es decir, al evaluar en el espacio de los objetivos todas las soluciones factibles y comparar los valores resultantes de sus vectores objetivo, surgirá un subconjunto para el que no es posible definir cuál de ellos es menor o mayor que el otro.

Luego, para poder decidir cuándo una solución resulta de interés, es preciso una *relación de dominancia* entre la solución considerada y el resto de las soluciones. Se introduce así el orden inspirado en el concepto de eficiencia desarrollado por Vilfredo Pareto en 1896, conocido como “*Dominancia de Pareto*”⁹:

⁸ En adelante, siempre que se mencione el espacio de decisión o el espacio de los objetivos, se supondrá que nos referimos a la porción factible, salvo que se indique lo contrario.

⁹ Continuamos, a los fines expositivos, con el supuesto de que estamos minimizando en todos los objetivos.

Una solución $x^a = (x_1^a, x_2^a, \dots, x_n^a)$ domina (\succ) a una solución $x^b = (x_1^b, x_2^b, \dots, x_n^b)$ si y solo si para todo $i \in \{1, \dots, p\}$ se da $z_i^a \leq z_i^b$, donde $z_i^a = f_i(x^a)$ y $z_i^b = f_i(x^b)$; y $\exists i \in \{1, \dots, p\}$, tal que $z_i^a < z_i^b$.

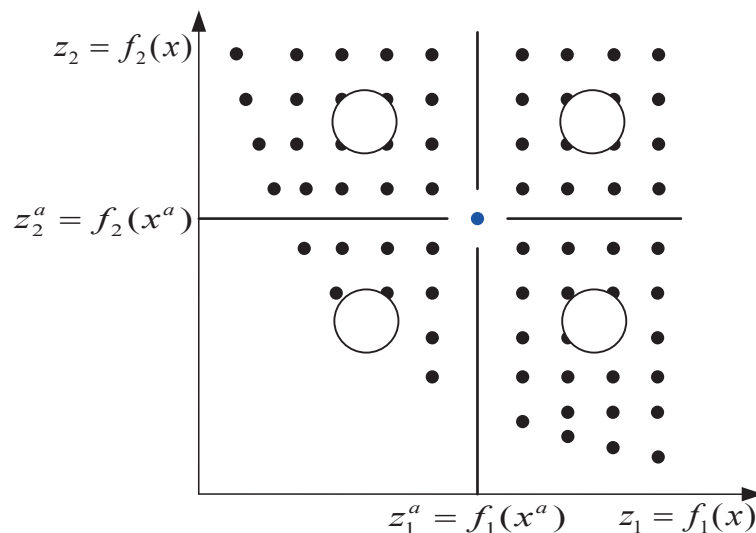
Se dice que x^a es Pareto-óptima, si y sólo si no existe otro vector x^b tal que $z^b = F(x^b)$ domine a $z^a = F(x^a)$.

En resumen, se puede decir que “el punto x^a es un óptimo de Pareto, si no existe un vector x^b , que haga mejorar alguno de los objetivos, sin que empeore de forma simultánea algún otro objetivo”.

Cuando a una solución z^a se le aplica el concepto de dominancia de Pareto, se pueden definir cuatro regiones (Méndez Babey, 2008).

En la Figura 11 se representan las cuatro regiones conformadas a partir de una solución x^a definida en el espacio de los objetivos como $z^a = F(x^a)$. Suponiendo que se busca minimizar ambos objetivos, puede verse que z^a domina a las soluciones pertenecientes a la región 1, es incomparable con las soluciones contenidas en las regiones 2 y 4. Y es dominada por las soluciones contenidas en la región 3.

Figura 11: Regiones de dominancia a partir de una solución determinada.



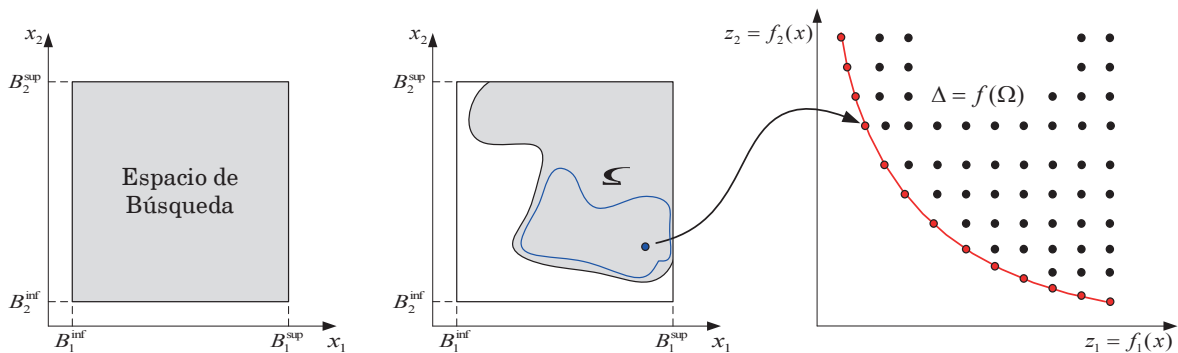
Fuente: Elaboración propia en base a Méndez Babey (2008)

Luego, el conjunto de soluciones óptimas al problema de optimización multi-objetivo se compone de los *vectores factibles no dominados*. Este conjunto se de-

nomina *Conjunto de Puntos Óptimos de Pareto* y está definido por: $P^* = \{x \in \Omega : \nexists x' \in \Omega \text{ para el que } F(x) \leq F(x')\}$

La región de puntos definida por el *Conjunto de Puntos Óptimos de Pareto* en el espacio de los objetivos, se conoce como *Frente de Pareto*. Así, el frente de Pareto queda definido por: $FP^* = \{u = (f_1(x), f_2(x), \dots, f_p(x)) : x \in P^*\}$

Figura 12: Representación frontera de Pareto



Fuente: Elaboración propia en base a Méndez Babey (2008)

En la Figura 12, se representa en el espacio de decisión factible el *Conjunto de Puntos Óptimos de Pareto* (en color azul) o *puntos eficientes*, denotado anteriormente como P^* . Subconjunto de soluciones que en general las técnicas de optimización multi-objetivo se centran en obtener.

En el espacio de los objetivos, se representa la imagen de P^* (en color rojo), denominado Frente de Pareto, FP^* . Y también se representan (en color negro) las imágenes de soluciones dominadas por las que están en el frente de Pareto.

4.3. Toma de Decisión

Como se describió anteriormente, el estudio del proceso de toma de decisiones, es el análisis de cómo elige un individuo, de entre un conjunto de acciones posibles, aquella acción que le conduce al mejor resultado dadas sus preferencias, en un contexto particular. En este apartado, se pretende presentar el enfoque desde el cual el agente realizará esta elección.

Como se vio en el apartado anterior, el resultado buscado al tratar de resolver un problema de optimización multi-objetivo, es el conjunto de soluciones que confor-

man el frente de Pareto completo. Y cuando se utiliza para ello un método no exacto como un algoritmo evolutivo, lo que se busca es *aproximar* el frente de Pareto completo, dado que se supone que en ausencia de preferencias todas las soluciones que se encuentran sobre el frente son equivalentes.

Pero en realidad, el agente decisor, está interesado en una única solución, por lo que en algún momento deberá revelar sus preferencias, para elegir la que a su juicio es la mejor de entre todas las soluciones que estén en el frente de Pareto. Es decir, el resultado final del proceso de toma de decisiones desde un enfoque multi-objetivo, depende tanto del proceso de optimización como del de decisión.

Entonces es relevante considerar en este punto, los métodos utilizados para articular las preferencias en el proceso de elección de la alternativa óptima. En este sentido, los métodos se clasifican en base al momento en que se introduce la información de las preferencias del agente (Van Veldhuizen & Lamont, 1998) (Hwang, Paidy, Yoon, & Masud, 1980) (Słowiński, 1984). Es decir, dependiendo de si las preferencias del decisor son expresadas antes, durante o después del proceso de optimización, los métodos para resolver este tipo de problemas se pueden clasificar en diferentes familias (ver Basseur, Talbi, Nebro y Alba (2006), Collette y Siarry (2011) y Marler y Arora (2004)).

4.3.1. Métodos *a posteriori*

Los *métodos a posteriori*, son aquellos en los que el decisor elige a partir de un conjunto de soluciones eficientes. Como se verá en el apartado 5 de este capítulo, la mayoría de las metaheurísticas evolutivas multi-objetivo tienen como objetivo generar simultáneamente un conjunto de soluciones que sean una buena aproximación del frente de Pareto, por lo que pueden ser clasificados como métodos *a posteriori*. Es decir, el algoritmo evolutivo aproxima el frente de Pareto, y el decisor debe elegir de acuerdo a sus preferencias una solución no dominada de todas las que conforman dicho frente. Algunos ejemplos de estos algoritmos son el *Multi-objective Genetic Algorithm* (MOGA) (Fonseca & Fleming, 1993), el *Strength Pareto Evolutionary Algorithm 2* (SPEA2) (Zitzler, Laumanns, & Thiele, 2001), o el

Non Dominated Sorting Genetic Algorithm II (NSGA-II) (Deb, Pratap, Agarwal, & Meyarivan, 2002).

Método de la combinación lineal de pesos

En este método el decisor combina los diferentes objetivos en una función escalar (Zadeh, 1963). Es decir, mediante la agregación de objetivos, se busca convertir un problema multi-objetivo en uno mono-objetivo. La forma más común es la agregación por combinación lineal de los diferentes objetivos.

Método de la distancia a un punto de referencia

Con este método las funciones objetivo del problema de optimización multiobjetivo, son combinadas en una sola función objetivo, la cual puede verse como una función de distancias que busca minimizar la distancia entre los puntos solución y un punto de referencia en el espacio de los objetivos (Yu & Leitmann, 1974) (Méndez, Salazar, & Galván, 2005).

Método de las ϵ -restricciones

Otra forma de transformar el problema de optimización multi-objetivo en otro mono-objetivo, consiste en convertir $k-1$ objetivos (siendo que hay k objetivos) del problema en restricciones, y minimizar o maximizar el objetivo restante (Ritzel, Eheart, & Ranjithan, 1994).

Métodos Interactivos o Progresivos

Los *métodos interactivos o progresivos*, son los métodos en los que la toma de decisión y la optimización se desarrolla en forma interactiva. Es decir, mediante un proceso secuencial de iteraciones en las que el decisor articula sus preferencias en cada etapa del proceso. Ejemplos de este tipo de método se pueden ver en Alves y Clímaco (2000), Miettinen y Mäkelä (2006) y Ojalehto, Miettinen y Mäkelä (2007).

4.3.2. Métodos a priori

Los *métodos de articulación de preferencias a priori*, son aquellos en los que el decisor tiene gran conocimiento del problema objeto de estudio, y ese conocimiento, en forma de preferencias, se utiliza en el proceso de búsqueda del conjunto de soluciones óptimas.

Programación por metas

La programación por metas puede verse de Charnes y Cooper (1957), Charnes, Cooper y Ferguson (1955), Ignizio (1976), Ignizio (1978) y Lee (1972). Este método presenta un cambio de lógica, de la optimización, a la búsqueda de soluciones satisfactorias. Lo que implica asumir que el decisor en vez de maximizar o minimizar una determinada función objetivo intenta que una serie de metas relevantes para su problema se aproximen lo más posible a unos niveles de aspiración fijados previamente. Es decir, el decisor busca soluciones razonables (satisfactorias) mediante el máximo cumplimiento posible de unos determinados niveles de aspiración.

El ordenamiento lexicográfico

El ordenamiento lexicográfico resuelve una secuencia de problemas monobjetivo de manera que cada problema considera una sola función objetivo sujeto tanto a las restricciones del problema, como a los valores de las soluciones obtenidas en la resolución de los problemas anteriores (Collette Y. &, 2011) (Miettinen, 2012).

Métodos de incorporación de preferencias parciales

Los *métodos de incorporación de preferencias parciales*, a los que pertenece la estrategia utilizada en este trabajo, suponen que el agente decisor, tiene una idea aproximada de qué soluciones pueden ser preferidas y puede especificar preferencias parciales. Este conocimiento, generalmente toma la forma de un punto de referencia, que guía la búsqueda hacia soluciones que son consideradas relevantes por el agente decisor. El objetivo de estos métodos no es encontrar una aproximación de todas las soluciones óptimas de Pareto, sino un pequeño conjunto de soluciones en-

tre las que está con un alto grado de probabilidad, la preferida por el agente decisor (Branke, 2008). En la literatura, se encuentran distintos enfoques para incorporar preferencias parciales, entre ellos se destacan los presentados en los trabajos de Molina, Santana, Hernández-Díaz, Coello y Caballero (2009) Branke (2008), Coello (2000), Rachmawati y Srinivasan (2006) y Coello (2002).

Finalmente, entre este reducido subconjunto del frente óptimo de Pareto, el decisor selecciona la alternativa de acción deseada.

Posteriormente, cuando se trate la herramienta utilizada para abordar el problema de decisión planteado en el presente trabajo, se profundizará tanto el mecanismo mediante el cual se realiza la introducción de preferencias parciales que guían la búsqueda del algoritmo inteligente durante la etapa de optimización, para que finalmente el decisor seleccione la alternativa factible, eficiente, y preferida por sobre todas las demás.

5. MÉTODOS DE RESOLUCIÓN

Los métodos utilizados en la literatura para resolver problemas ruteo de vehículos considerando múltiples objetivos, son diversos. Como fue mencionado anteriormente, las posibilidades de implementar un método u otro, difieren dependiendo de las características de la red de distribución considerada, del tamaño del problema y de las características consideradas respecto al proceso de toma de decisiones. En base a una revisión bibliográfica de las principales investigaciones en materia de problemas multi-objetivo de ruteo de vehículos, surgió que las técnicas más utilizadas son las meta-heurísticas (80% de los casos en las últimas dos décadas), en segundo lugar están las heurísticas (en el orden del 15%) y finalmente las exactas y otros métodos (en el orden del 5%).

Dentro de las *técnicas exactas*, (problemas “chicos” hasta 50 clientes aproximadamente), encontramos, por ejemplo, algoritmos de *ramificación y acotamiento* (Min, 1991) (Jozefowicz, Semet, & Talbi, 2007) (Riera-Ledesma & Salazar-González, 2005) (Riera-Ledesma & Salazar-González, 2012), *programación lineal entera mixta* (Riera-Ledesma & Salazar-González, 2005) (Lee & Ueng, A study of vehicle

routing problems with load-balancing., 1999), *programación lineal* (Sutcliffe & Boardman, 1990).

Entre las numerosas *técnicas heurísticas*, se encuentra la *búsqueda local* (Alabas-Uslu, 2008), *heurística de los ahorros* (Anbuudayasankar, Ganesh, Koh, & Ducq, 2012), *heurísticas en dos fases* (Chang & Wang, 2010), *heurísticas de inserción* (Campbell, 2004), *heurísticas de combinación* (Kritikos & Ioannou, 2010).

Y finalmente entre las *meta-heurísticas* encontramos los algoritmos basados en *colonias de hormigas* (Gong & Fu, 2010), *algoritmos evolutivos* (Frutos & Tohmé, 2012) (Ombuki, Ross, & Hanshar, 2006) (Ombuki-Berman, Runka, & Hanshar, 2007) (Tan, Chew, & Lee, 2006) (Tan, Cheong, & Goh, 2007) (Jozefowicz, Semet, & Talbi, 2009) (García-Nájera, Bullinaria, & Gutiérrez-Andrade, 2015), *búsqueda tabú* (Pacheco & Martí, 2006) (Chao I. M., 2002), *recocido simulado* (El-Sherbeny, 2001) (Ulungu, Teghem, Fortemps, & Tuyttens, 1999) (Baños, 2013) (Frutos M. , Tohmé, Delbianco, & Miguel, 2016) y *redes neuronales* (Somhom, Modares, & Enkawa, 1999) (Gomes & Von Zuben, 2002) entre otros.

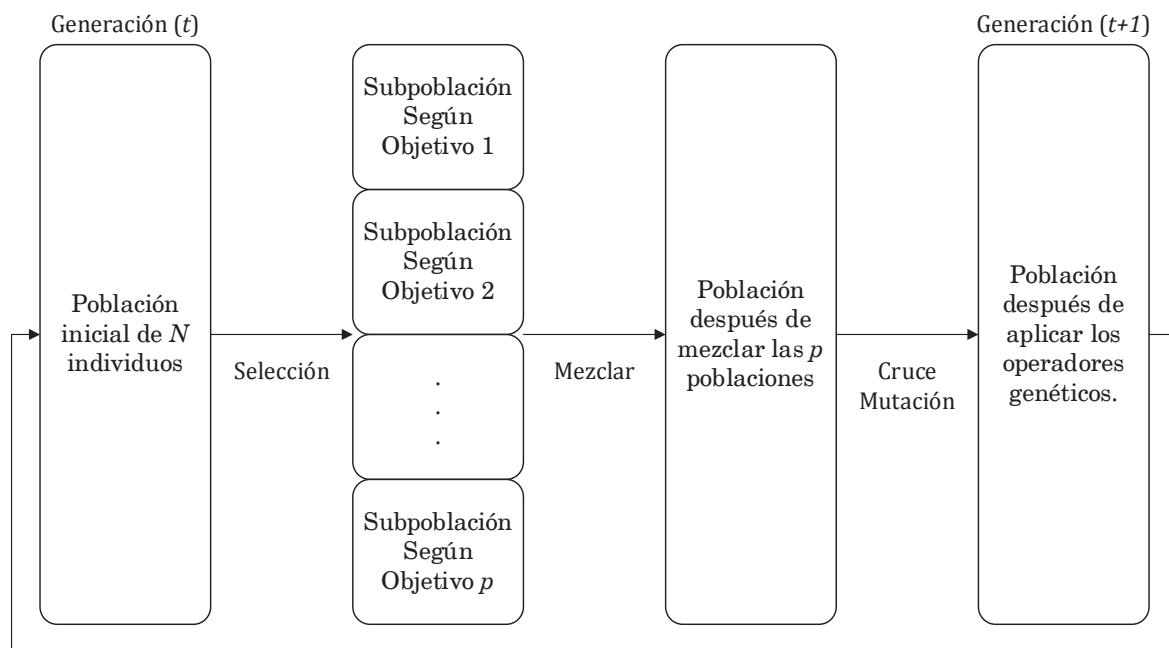
Una de las técnicas meta-heurísticas más utilizadas para resolver este tipo de problemas de optimización multi-objetivo, son los *Algoritmos Evolutivos Multi-objetivo* (MOEAs *Multi Objective Evolutionary Algorithms*), debido fundamentalmente a las falencias de los métodos clásicos para obtener soluciones alternativas múltiples, y además por la complejidad de estos problemas de optimización combinatoria multi-objetivo (grandes espacios de búsqueda, incertidumbre, ruido, curvas de Pareto disjuntas, etc.).

Por lo general, los métodos clásicos, resuelven el problema reduciendo el vector de objetivos a uno sólo con la ayuda de conocimiento específico del problema en cuestión. Como consecuencia, el agente decisor debe poseer un alto conocimiento del problema que está modelando antes de desarrollar la función objetivo. Mientras que los *algoritmos evolutivos multi-objetivo*, trabajan de manera simultánea con una población de soluciones, que buscan evolucionar para obtener aquellas que sean consideradas aceptables por parte del diseñador. Es decir, poseen la habilidad de encontrar y trabajar simultáneamente con múltiples soluciones del frente de Pareto, ya sea para espacios continuos, no continuos, discretos, no convexos, etc.

5.1. VEGA

Numerosos estudios destacan la potencialidad de los algoritmos evolutivos para el tratamiento de problemas multi-objetivo. Uno de los primeros autores en destacar sus ventajas, fue David Schaffer, a quien se le atribuye el primer algoritmo evolutivo multi-objetivo, como VEGA (*Vector Evaluated Genetic Algorithm*) (Schaffer, 1984). Como puede verse en la Figura 13, la diferencia del VEGA con un algoritmo genético simple fue la manera de efectuar la selección. Con k objetivos y una población de N individuos, se realiza una selección por torneo de N/k individuos para cada función objetivo. De esta manera, cada subpoblación k , contiene los mejores N/k individuos para la función objetivo k , que determina el nivel de aptitud de un individuo dentro de esa subpoblación. Luego se mezclan las subpoblaciones y se aplican los operadores genéticos.

Figura 13: Esquema de Funcionamiento de VEGA



Fuente: Elaboración propia en base a Schaffer (1984)

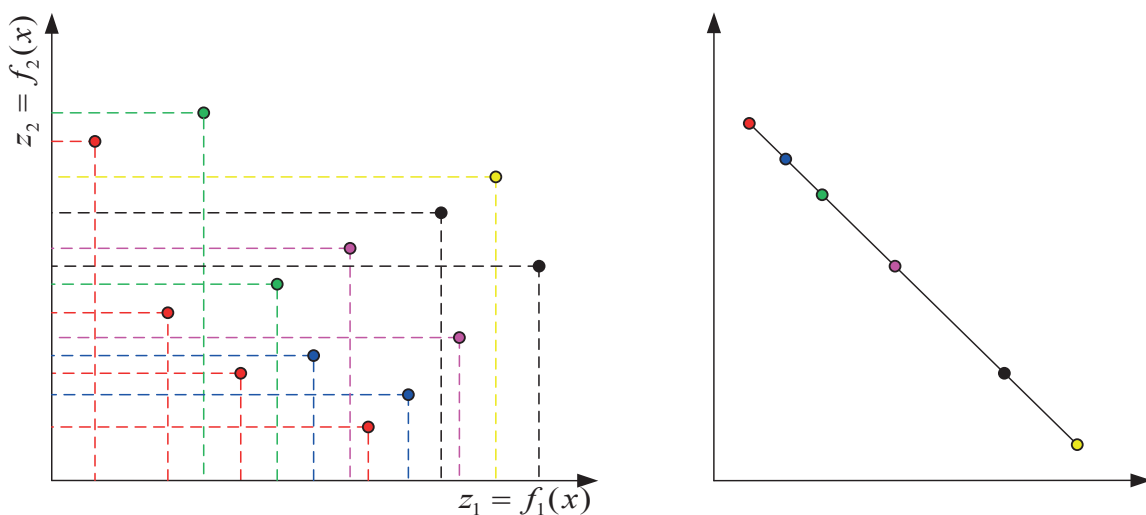
Luego vinieron los desarrollos de MOEAs basados en dominancia de Pareto, que tuvieron un desarrollo importante a partir de inicios de los 90's, surgieron como una extensión de los algoritmos evolutivos para resolver problemas mono-objetivo, utilizando fundamentalmente varios conceptos relacionados con el tratamiento de

funciones multimodales. Su característica principal, es que el conjunto de objetivos es optimizado en forma simultánea y que tienen la capacidad de encontrar múltiples soluciones en una sola ejecución. Para esto, usan el concepto de dominancia de Pareto, de forma que la población es clasificada en rangos de acuerdo a una regla de dominancia y luego a cada solución se le asigna su valor de aptitud en base a su rango en la población.

5.2. MOGA

Entre las formulaciones más destacadas, está el *Multi-objective Genetic Algorithm* (MOGA), de Fonseca y Fleming (1993), que se basa en la asignación de un rango a los individuos, respecto a su relación de dominancia con la población. Como se muestra en la Figura 14, si estamos minimizando en ambos objetivos, esta asignación de rangos a los individuos de la población, consiste en estimar el rango de un individuo a partir del número de individuos que dominan al individuo considerado. Es decir, los individuos no dominados tienen rango uno (en color rojo), los dominados por un solo individuo tienen rango dos (en color azul), y así sucesivamente. Luego se asigna la aptitud de cada individuo usando habitualmente una función lineal a partir del mejor hasta el peor rango, aunque pueden usarse otras.

Figura 14: Asignación de rangos MOGA



Fuente: elaboración propia en base a Fonseca & Fleming (1993)

Goldberg y Deb (1991) destacaron que este tipo de asignación fija de valor de aptitud puede producir una gran presión selectiva en la población, lo que puede dar lugar a una convergencia prematura en el proceso de búsqueda del algoritmo. Para evitar esto Fonseca y Fleming propusieron una estrategia de formación de nichos para distribuir la población sobre la región óptima de Pareto. Un nicho puede definirse como un subconjunto de la población actual, los cuales son agrupados por tener ciertas características entre si (la posición en el ranking en este caso).

El método que utilizan para preservar la diversidad es una técnica conocida como *Sharing* (o de participación). Este procedimiento consiste en disminuir el valor de aptitud de los individuos que se encuentran en un mismo nicho, penalizándolos al degradar su valor de aptitud mientras más concentrados estén, lo que significaría que hay poca diversidad.

Hay muchas formas de realizar esta disminución del valor de aptitud, la más utilizada es computar un valor que sea representativo de la falta de diversidad del nicho en cuestión, y luego disminuir el valor de aptitud de cada uno mediante alguna operación aritmética como resta o división con este valor.

La función de *Sharing* utilizada por Fonseca y Fleming es:

$$Sh(d_{i,j}) = \begin{cases} 1 - \left(\frac{d_{i,j}}{\sigma_{sharing}}\right)^\alpha & , si \ d_{i,j} < \sigma_{sharing} \\ 0 & , en \ otro \ caso \end{cases}$$

Con $\alpha = 1$ y el parámetro de *Sharing*, $\sigma_{sharing}$, que varía entre 0,01 y 0,1.

El valor de $Sh(d_{i,j})$, depende de la distancia entre dos soluciones y toma un valor comprendido entre [0,1]. La idea es que si la distancia entre dos soluciones $d_{i,j} = 0$, es decir que las dos soluciones son idénticas, entonces la función de *Sharing* tomaría el valor $Sh(d_{i,j}) = 1$. Si $d_{i,j} \geq \sigma_{sharing}$ (lo que significa que la distancia entre dos soluciones supera el umbral aceptable, $\sigma_{sharing}$), la función de *Sharing* toma el valor 0. Y para cualquier otra distancia $d_{i,j}$ entre dos individuos la función asumirá un valor positivo.

La distancia $d_{i,j}$ entre dos individuos se calcula como la distancia euclidiana normalizada en el espacio de los objetivos:

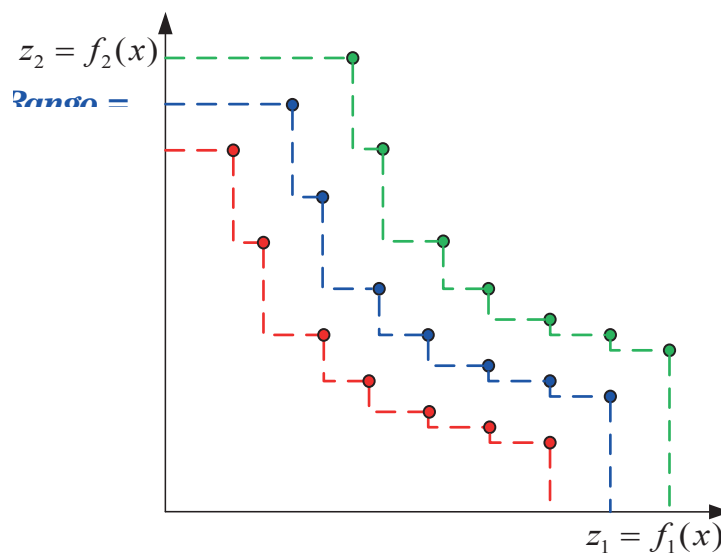
$$d_{i,j} = \sqrt{\sum_{r=1}^p \left(\frac{f_r(x_i) - f_r(x_j)}{f_r^{max} - f_r^{min}} \right)^2}$$

Siendo f_r^{min} y f_r^{max} los valores mínimo y máximo respectivamente para la función objetivo r-ésima.

5.3. NSGA

Srinivas y Deb (1994), presentaron el algoritmo evolutivo multi-objetivo conocido como *Non Dominated Sorting Algorithm* (NSGA), que consistió en una implementación de la idea de Goldberg (1989) de tener múltiples niveles de clasificación de los individuos de la población a partir de los *frentes no dominados*. Plantean de esta manera la formación de *frentes no dominados* en cada generación, de forma que sean tratados como una subpoblación diferente y utilizan la técnica de *sharing* para penalizar la concentración de individuos dentro de cada uno de los nichos formados.

Figura 15: Asignación de rangos NSGA



Fuente: Elaboración propia en base a Goldberg (1989)

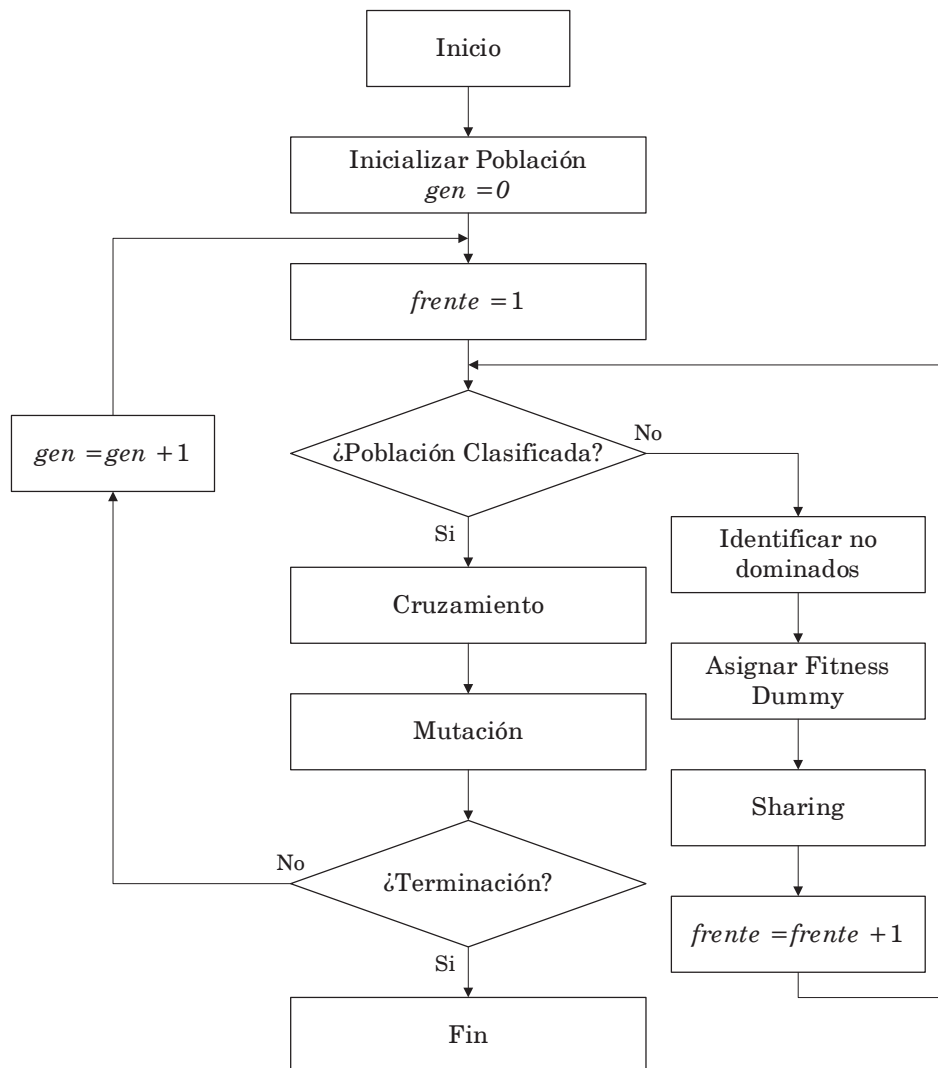
En la Figura 15 se representa la estrategia de formación de frentes mediante la asignación de rangos, que consiste en primero, asignar rango uno (en rojo) a todos los individuos no dominados de la población. Luego se seleccionan los no dominados de los que no tienen rango aún y se les asigna rango 2 (en azul). Posteriormente se seleccionan los no dominados de los que no tienen rango aún y se les asigna

rango 3 (en verde). Este proceso se repite sucesivamente hasta que todos los individuos de la población tengan asignado un rango.

Luego, a los individuos de cada nicho se les da un valor de aptitud ficticia (o *dummy*) que supone, les da la misma oportunidad de reproducción.

Luego, con el objetivo de mantener la dispersión entre los individuos de cada *frente no dominado*, se utiliza también la técnica de *Sharing*. La idea al igual que en el caso del MOGA, es degradar el valor de aptitud de un individuo, en función de la densidad de la población en su vecindad, buscando de esta manera reducir el grado de concentración de individuos que presentarán características similares en un frente determinado, penalizando la concentración y favoreciendo la diversidad.

Figura 16: Diagrama NSGA



Fuente: Elaboración propia en base a Srinivas y Deb (1994)

En la Figura 16 se presenta el diagrama de flujo del algoritmo NSGA, donde se muestra la estructura del algoritmo evolutivo con las variaciones implementadas por Srinivas y Deb.

5.4. NPGA

Otra formulación destacada de estos algoritmos evolutivos multi-objetivo de primera generación, fue el *Niched Pareto Genetic Algorithm* (NPGA), de Horn, Nafpliotis y Goldberg (Horn, Nafpliotis, & Goldberg, 1993), que básicamente utiliza como proceso de selección un esquema de torneo basado en dominancia de Pareto, en el cual dos individuos son seleccionados aleatoriamente de la población y comparados en términos de dominancia contra un subconjunto de la población también escogido al azar, llamado conjunto de comparación, usualmente conformado por el 10% de la población.

Luego, se comparan los dos individuos con los miembros del conjunto de comparación. Si uno de los individuos no es dominado por ninguno de estos y a su vez el otro individuo si lo es, el no dominado se declara ganador del torneo y es seleccionado. Si ambos individuos son dominados, entonces el ganador se decide mediante un proceso de *Sharing*.

En este caso el proceso de *Sharing* se efectúa considerando la cantidad de puntos que se encuentran a un radio denominado *SShare*, respecto a cada uno de los dos puntos. Luego, el individuo que posea la menor cantidad de puntos en este radio, es el ganador elegido. Es decir, gana el individuo que posea una menor concentración en su vecindario, priorizando así la diversidad.

Después de tener mucho éxito, los algoritmos de primera generación comenzaron a ser reemplazados por nuevos algoritmos con menor complejidad en el tiempo de ejecución. Aparece así una nueva generación de algoritmos evolutivos multi-objetivo, caracterizada principalmente por la búsqueda de la eficiencia computacional y por la introducción del concepto de elitismo, esto último realizado principalmente de dos maneras: una es en el proceso de selección y otra usando directamente una población secundaria.

5.5. SPEA

Entre los MOEAs de segunda generación más destacados, se encuentra el *Strength Pareto Evolutionary Algorithm* (SPEA), de Zitzler y Thiele (1999). Este método usa una población de tamaño fijo y un archivo externo para almacenar los individuos no dominados obtenidos en cada generación. Para asignar la aptitud de un individuo considera, tanto para cada individuo de la población actual, como también para aquellos del archivo externo, un índice llamado *raw fitness*, que representa a cuantos individuos domina y por cuantos individuos es dominado.

5.6. SPEA 2

Posteriormente, Zitzler, Laumanns y Thiele (2001) presentaron el *Strength Pareto Evolutionary Algorithm 2* (SPEA2), que se diferencia de su predecesor en algunos aspectos que permiten solucionar sus inconvenientes.

SPEA2 utiliza una estrategia de asignación de fitness que incorpora información adicional de densidad, para discriminar entre individuos con idéntico *raw fitness*. Cuando la mayoría de los individuos no se dominan entre sí, la técnica de estimación de densidad, (que es una adaptación del método del *k-ésimo* vecino más cercano), mejora el *raw fitness* y da mayor precisión al proceso de búsqueda. La técnica mide la densidad en términos de distancias Euclidianas (otras son posibles), de forma que se seleccionen aquellas soluciones con el mayor volumen hiper-esférico vacío alrededor de ellas.

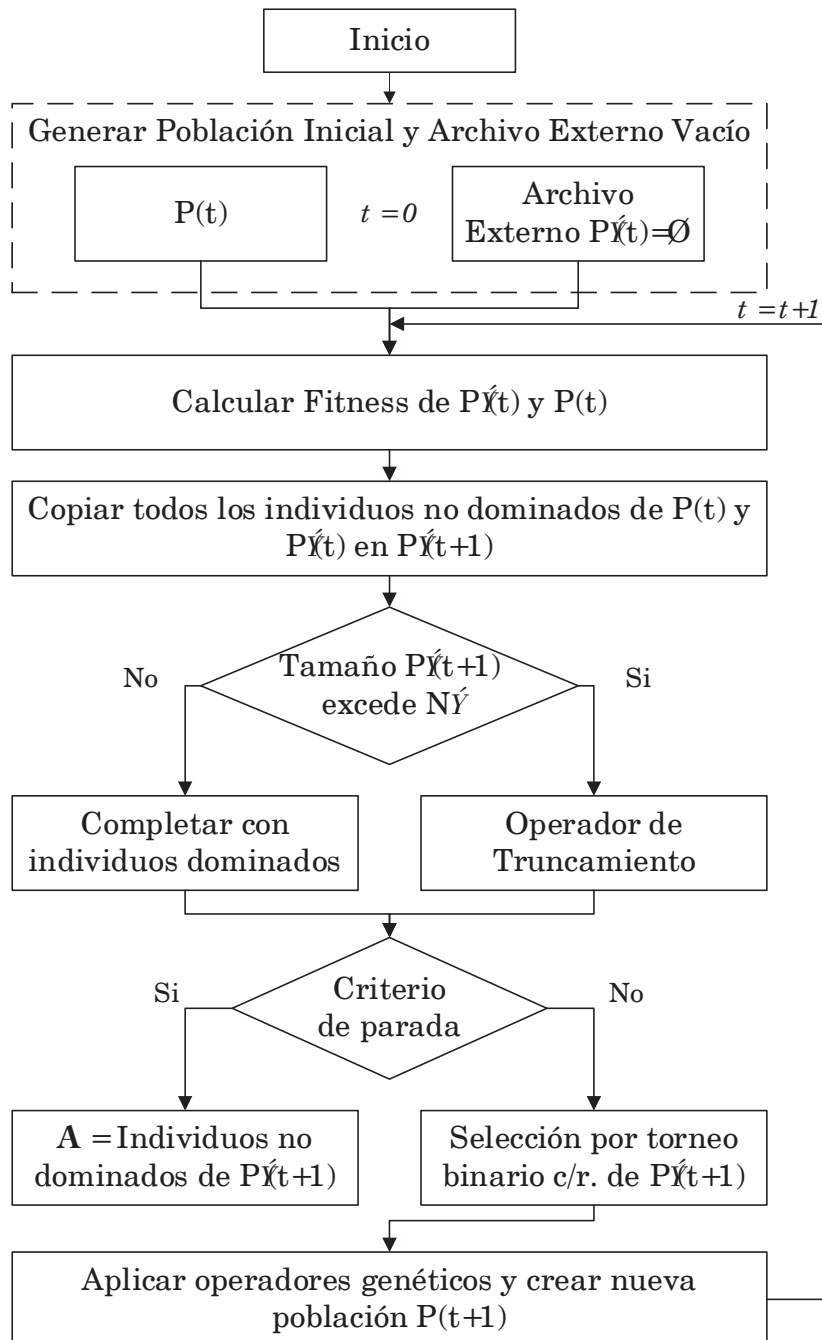
También, en SPEA 2, para el cálculo del *raw fitness*, se consideran individuos del archivo externo y de la población, a diferencia del SPEA, que solo consideraba miembros del archivo externo para esto.

Además, en SPEA 2 el tamaño del archivo externo para almacenar los individuos no dominados es fijo, es decir, cada vez que el número de individuos no dominados es menor que un tamaño predefinido, el archivo se llena con individuos dominados, mientras que en SPEA el tamaño del archivo podía variar a través de las generaciones. Además, la técnica de agrupación, que era ejecutada en SPEA cuando el frente no dominado excedía el tamaño del archivo externo, fue reemplazada por un método alternativo de truncamiento más eficiente.

Finalmente, otra diferencia con SPEA en que solo los miembros del archivo externo participan en el proceso de selección para reproducción.

En la Figura 17, se presenta el diagrama de flujo del algoritmo SPEA 2, donde se muestra la estructura del algoritmo con las variaciones implementadas por Zitzler, Laumanns y Thiele.

Figura 17: Diagrama SPEA 2 (Zitzler, Laumanns, & Thiele, 2001)



Fuente: Elaboración propia en base a Zitzler, Laumanns y Thiele (2001)

5.7. NSGA-II

De forma contemporánea a los autores anteriores, Deb, Agrawal y Pratap (2000) y (2002), publicaron el *Non Dominated Sorting Genetic Algorithm II*. Este algoritmo supero ampliamente a NSGA y es uno de los más utilizados actualmente, para resolver problemas de optimización multi-objetivo. Dado que el algoritmo utilizado para la resolución del problema objeto de estudio del presente trabajo, es una hibridación de este algoritmo, posteriormente se profundizará en forma detallada en su estructura e implementación. Pero aunque se verá en el capítulo IV, en este punto es importante destacar que con este algoritmo, estos autores lograron un ordenamiento no dominado más rápido que con NSGA, y redujeron su complejidad a $O(k \cdot N^2)$ ¹⁰, donde k es la cantidad de objetivos y N es el tamaño de la población. Además, utilizando la *distancia Crowding*, lograron mejorar la diversidad sobre el frente sin necesidad de especificar el parámetro de Sharing. Y mediante la utilización de *elitismo*, lograron mejorar la convergencia.

Otras formulaciones destacadas en estos algoritmos de segunda generación son el Niche Pareto Genetic Algorithm II (NPGA2), de Erickson, Mayer y Horn (2001), el Pareto Enveloped-Based Selection Algorithm (PESA), de Corne, Knowles y Oates (2000), el Pareto Enveloped-Based Selection 2 Algorithm (PESA2), de Corne, Knowles y Oates (2001), el Micro Genetic Algorithm (micro-GA), de Coello y Pulido (2001).

Finalmente, se requiere sumar las preferencias del decisor a esta fase de optimización evolutiva multi-objetivo embebida en un proceso de toma de decisiones, para poder concluir con el proceso decisional y seleccionar la alternativa preferida.

Como se vio anteriormente, en la literatura se han utilizado diferentes estrategias para articular las preferencias del decisor con el proceso de optimización llevado a cabo por los algoritmos evolutivos multi-objetivo, y así poder determinar el resultado final esperado del proceso de toma de decisiones multi-criterio.

¹⁰ Ver punto 3 del Capítulo III.

Una de las formas más habituales de proporcionar esta información, y que posteriormente se profundizará, dado que es la estrategia utilizada en el problema objeto de estudio del presente trabajo, es mediante la introducción de un punto de referencia.

El concepto de punto de referencia, hace alusión a unos niveles de aspiración, que reflejan los valores deseables para la función objetivo, es decir, un objetivo que el decisor espera (Branke, 2008).

El uso de esta técnica para guiar el proceso de búsqueda que sigue un algoritmo evolutivo multi-objetivo, fue propuesto por primera vez en 1993 por Fonseca y Fleming (1993), otras propuestas posteriores que siguen esta estrategia pueden verse en Deb et al. (1999), (2001), (2006), Tan et al. (1999), Molina et al. (2009) y Thiele et al. (2007).

CAPÍTULO III. MODELADO

1. INTRODUCCIÓN

1.1. Contexto operativo del problema

En este capítulo se realizará la definición formal y modelado del problema de distribución física de mercaderías, en el contexto del servicio logístico objeto de estudio de la presente investigación. En el marco teórico se vio una descripción de diferentes caracterizaciones de esta problemática, en base a las características físicas del problema y de la información con la que se cuenta al llevar adelante el proceso de toma de decisiones. De esta manera, se identificaron diferentes tipos de problemas tradicionales, llamados habitualmente en la literatura problemas de ruteo de vehículos.

Como se verá con mayor detalle en el capítulo siguiente, el problema estudiado y modelado, se inspiró un problema real de planificación, que enfrenta una empresa que se desempeña como operador especializado en un servicio independiente (2PL: *Second Party Logistics*), de distribución de mercadería a carga fraccionada, con volúmenes reducidos y plazos de entrega diarios. La empresa presta este servicio a diversos clientes ubicados en mercados regionales donde posee depósitos desde donde realiza la distribución a los minoristas. El ámbito geográfico es la ciudad de Buenos Aires y el Gran Buenos Aires.

Considerando lo expuesto en el marco teórico, la problemática de la distribución urbana de mercaderías planteada, puede catalogarse como la combinación de tres de los problemas tradicionales (ver Figura 18): el Problema de ruteo de vehículos capacitados con múltiples depósitos (*MD-CVRP: Multi Depot Vehicle Routing Problem*), el problema de ruteo de vehículos capacitados sujeto a restricciones de ventanas de tiempo (*CVRP-TW: Vehicle Routing Problem with Time Windows*), y el problema de ruteo de vehículos capacitados con dependencia temporal (*TD-CVRP: Time Dependent Vehicle Routing Problem*).

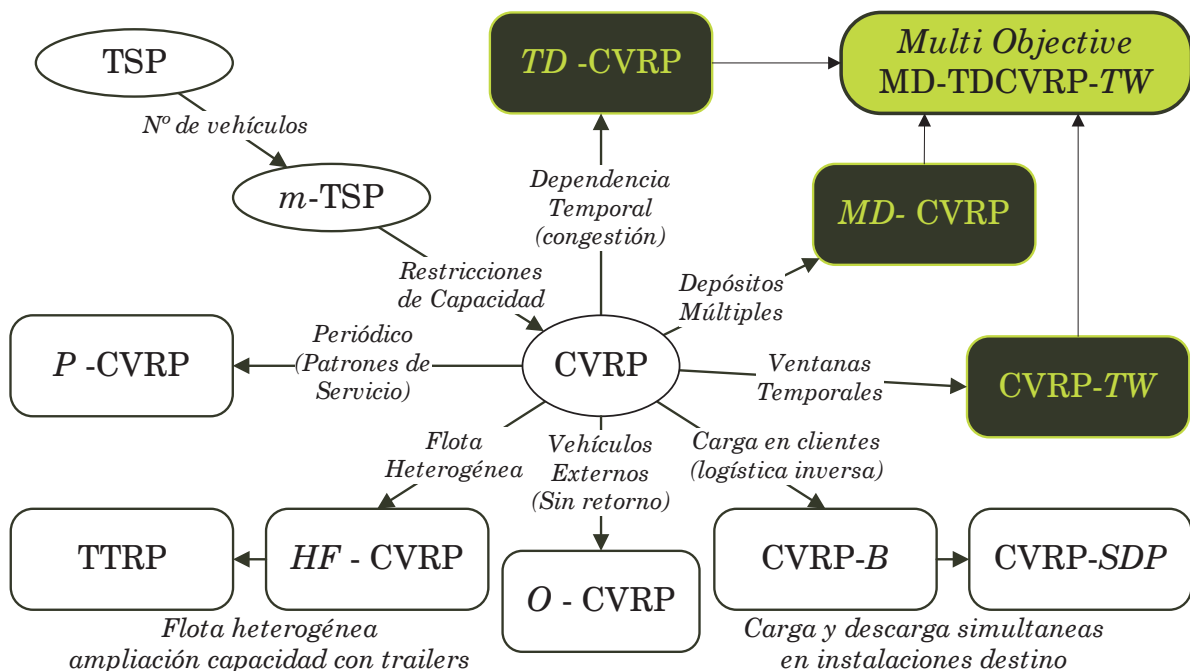
Además, se considera un abordaje multi-objetivo de este problema compuesto. Esto responde a que, como se mencionó anteriormente, es una forma más realista de estudiar este tipo de procesos de toma de decisiones en ambientes logísticos reales,

aunque mucho más compleja de modelar y de resolver aunque sea de forma aproximada.

Por otro lado, cabe mencionar que es muy reducida la cantidad de trabajos que tratan este tipo de problemas desde un abordaje multi-objetivo. Montoya et al. (2015) realizaron una completa revisión bibliográfica de los trabajos que estudian el problema de ruteo de vehículos con múltiples depósitos, y encontraron que solamente en el 11.5% de los trabajos se consideraban múltiples objetivos, de los 17 trabajos que usan el enfoque multi-objetivo, solamente dos consideran simultáneamente ventanas temporales y restricciones de capacidad, pero no consideran la dependencia temporal (Weise, Podlich, & Gorldt, 2009) y (Adelzadeh, Asl, & Koosha, 2014). Y los dos trabajos que sí consideran la dependencia temporal (Lau, y otros (2009) y Lau, Chan, Tsui, & Pang (2010)), no trabajan con ventanas temporales ni restricciones de capacidad.

Es decir, desde el punto de vista de las restricciones impuestas por el contexto del problema, y tratadas simultáneamente, el modelo es original.

Figura 18: Problema compuesto



Fuente: Elaboración propia en base a Weise, Podlich y Gorldt (2009)

En la Figura 18 se muestra el problema multi-objetivo de distribución urbana de mercaderías objeto de estudio, como una combinación de los tres problemas tradicionales descritos en el marco teórico, el *MD-CVRP*, el *TD-CVRP* y el *CVRP-TW*. A este problema lo denominaremos *MO-MDTDCVRPTW*, por las siglas de los nombres en inglés de las variantes combinadas, esto es, *Multi-Objective Multi-Depot Time Dependent Capacitated Vehicle Routing Problem with Time Windows*.

1.2. Objetivos perseguidos por el decisor

Hace décadas que los investigadores realizan grandes esfuerzos en la búsqueda de mejores técnicas y herramientas de ayuda a la toma de decisión, tanto a nivel estratégico como operativo. Persiguiendo mejorar la eficiencia y reducir los costos de las actividades relacionadas a la distribución urbana de mercadería, dado su significativo impacto financiero y sobre la competitividad de la organización.

Sin embargo, la atomización de los operadores que prestan este tipo de servicios, y la complejidad implícita de la gestión en este nivel de la cadena de suministros, requiere de esfuerzos colaborativos dirigidos a considerar intereses de terceras partes, en el diseño del proceso de toma de decisiones.

Como se describió al principio de este trabajo, si el proceso decisional únicamente persigue la minimización de los costos, las decisiones surgidas de este tipo de estrategia llevarán al deterioro de las relaciones con los demás agentes intervinientes en la prestación del servicio, con su correspondiente pérdida de calidad y competitividad. Por lo que es indispensable abordar esta problemática, desde un enfoque multi-objetivo que considere estos aspectos.

Por esta razón, en este trabajo, se propone un enfoque que le permite al agente decisor, guiar la búsqueda de la herramienta computacional hacia una región deseable del espacio de los objetivos. Y así contar rápidamente con un conjunto reducido de distintas opciones de programas de distribución, eficientes, y de su interés. Que le permitan, evaluar diferentes niveles de compromiso entre, la búsqueda de la reducción de costos (dependientes del tiempo de cada programa), y de la generación de cargas de trabajo equilibradas, para los distintos operadores del servicio.

Es importante profundizar en este punto, que los procesos de toma de decisiones que se basan en una estrategia que persigue solo la minimización de costos, generalmente consideran que los costos mantienen una relación directa con la distancia recorrida por cada vehículo. Este tipo de planteo presenta dos falencias fundamentales:

- Llevan a decisiones del tipo todo o nada. Es decir, generan programas de distribución muy desequilibrados, con operadores trabajando al 10% o 20% de su capacidad, mientras que otros reciben una carga de trabajo del 90% o hasta el 100% de su capacidad. En entornos urbanos, esto afecta mucho la variabilidad en el cumplimiento de la puntualidad de las entregas, con su correspondiente pérdida de calidad del servicio. Además de la desconformidad de los distintos operadores.
- Llevan a decisiones coherentes, si la distribución de los tiempos de servicio es homogénea. Al considerar distribuciones heterogéneas en los tiempos de servicio en las instalaciones del cliente (tiempos que incluyen actividades de maniobra en las instalaciones del minorista, de chequeo de documentación, de inspección de la mercadería, de descarga, de estiba, etc.) la solución que es óptima en términos de distancia, puede diferir significativamente de la que es óptima cuando consideramos los tiempos de servicio. Esto se agrava al considerar diferentes velocidades medias según la zona a la que pertenece cada arco del grafo de conectividades entre clientes, debido a puntos de congestión con velocidades promedio muy bajas asociadas a los arcos de zonas congestionadas.

Esto permite entender la relevancia de considerar el tiempo y no la distancia recorrida por los vehículos, como objetivo a minimizar en el contexto de la distribución diaria de mercaderías en entornos urbanos. Por un lado, porque en entornos congestionados cobran relevancia factores determinantes del costo relacionados al costo de oportunidad, como bajos factores de utilización de la capacidad del vehículo y de su dotación de personal, factores que se ven muy afectados por las bajas velocidades de circulación promedio, por la vuelta en lastre y la menor capacidad de carga útil de los vehículos (Sánchez, Cortés, Peralta, & Díaz, 2007), en compa-

ración con las características del transporte de larga a media distancia, donde presentan mayor incidencia los costos de insumos variables como el combustible, por ejemplo. Por otro lado, porque al considerar en la programación el tiempo en lugar de la distancia, la heterogeneidad en los tiempos de servicio en las instalaciones de los clientes, y los tiempos de espera por haber llegado de forma anticipada a las instalaciones de un cliente (derivados de las ventanas temporales), cambia substancialmente el programa óptimo de distribución, al compararlo con el que surge considerando como objetivo la minimización de distancias recorridas.

Esta es una contribución original presente en el modelado de este problema, al considerar objetivos perseguidos por el agente decisor, que no están presentes de forma simultánea en otros trabajos de la literatura especializada en problemas de ruteo de vehículos con múltiples depósitos y ventanas temporales.

Se considera entonces, que el agente decisor persigue dos criterios simultáneamente y en conflicto. El criterio económico, presente mediante los tiempos acumulados en los programas de distribución de cada operador. Y el criterio social, que en este contexto implica la generación de programas de trabajo equilibrados entre los diferentes operadores. Es decir, los costos económicos incluyen los costos de transporte desde los centros de distribución a los clientes minoristas de la red de distribución, y en nuestro caso, se consideran proporcionales al tiempo, dadas las condiciones de distribución urbana consideradas. El equilibrio en los programas de distribución, se busca mediante el equilibrio en la carga de trabajo de cada distribuidor (operador), definida en relación al tiempo requerido de cada distribuidor para cumplir con su programa. A continuación, se presentan las expresiones generales de estos criterios, y en el apartado siguiente se detallará su explicación y la de cada una de las variables involucradas (ver ecuaciones 1.1 y 1.2 del modelo).

Criterio Económico: Minimizar el costo total del programa de distribución, determinado por el tiempo total acumulado:

$$\min f_1 = \sum_{\substack{i \in V \\ i \neq j}} \sum_{j \in V} t_{ij} \cdot x_{ij}$$

Criterio Social: Balancear en tiempo las rutas:

$$\min f_2 = L_{\max} - L_{\min}$$

Donde:

L_{\max} : Tiempo acumulado de la ruta más larga.

L_{\min} : Tiempo acumulado de la ruta más corta.

Tenga en cuenta que, el objetivo de minimizar $f_2 = L_{\max} - L_{\min}$ fuerza a L_{\max} tomar el valor más bajo posible (duración de la ruta más larga) y a L_{\min} a tomar el mayor valor posible (duración de la ruta más corta). Además, en el modelo, al objetivo de minimización de tiempos, se le agregarán los tiempos por llegar a las instalaciones de cada cliente minorista antes del horario previsto de descarga, y los tiempos previstos para maniobra y del servicio en las instalaciones del cliente.

Se considera adicionalmente que los programas de distribución, deben cumplir con restricciones de ventanas temporales “*tipo hard*” para la prestación del servicio a los clientes, y que existen restricciones a la cantidad de horas de trabajo de los conductores. Además de las restricciones de capacidad de los vehículos y restricciones de factibilidad en la conectividad vial por infraestructura requerida para el paso de los vehículos de carga.

2. MODELADO DEL MO-MDTDCVRPTW

En este apartado se presenta la descripción formal del modelo. La formulación responde a una adaptación multi-objetivo y más realista del modelo originariamente presentado por Báez Viezca (2014), permitiendo la asignación de vehículos a cada depósito con la posibilidad de que tengan diferentes capacidades, y ventanas temporales “*tipo hard*”.

El modelo formalizado es el del problema compuesto, nombrado como MO-MDTDCVRPTW (*Multi-objective Multi-Depot Time Dependent Capacitated Vehicle Routing Problem with Time Windows*), descrito en la introducción de este capítulo, en la Figura 18. Y presentado y caracterizado detalladamente en el punto 2.2 del capítulo siguiente.

El MO-MDTDCVRPTW, se puede representar en un grafo $G = (V, E)$, donde V es el conjunto de nodos y E es el conjunto de arcos. En el presente trabajo se divide el conjunto de nodos en dos subconjuntos, uno con L depósitos y el otro con R clientes minoristas, los clientes compiten entre sí por usar la capacidad limitada de las rutas de los vehículos. Entonces $V_L = \{1, 2, \dots, L\}$ es el conjunto de los L depósitos, y $V_R = \{L + 1, L + 2, \dots, L + R\}$ es el conjunto de los R clientes. Luego, $V = V_L \cup V_R$. Para cada cliente i se tiene una demanda b_i y un tiempo de servicio ts_i , donde $ts_i = 0 \forall i \in V_L$. Además cada cliente tiene una ventana temporal $[e_i, l_i]$. Estas ventanas se consideran de “*tipo hard*”, es decir, el vehículo puede llegar antes del momento e_i al nodo i , pero no será atendido antes, deberá esperar a que llegue el momento e_i . Si el vehículo llegara al nodo i después del momento l_i , habrá perdido la oportunidad para realizar el servicio, lo que no es aceptable.

Para cada arco $(i, j) \in E$, se cuenta con el tiempo tr_{ij} , que incluye el tiempo esperado, t_{ij} , para ir desde el nodo i al nodo j , más el tiempo de servicio en el nodo de llegada. Se tiene flota de vehículos S , con base en cada depósito. Cada vehículo, s , cuenta con una capacidad de carga Q . Se tiene también un tiempo máximo de duración de la ruta de cada vehículo r_k .

2.1. Variables Binarias

Se utilizan las siguientes variables binarias:

- x_{ij}^{sl} igual a 1 si el vehículo s va del nodo i al j , habiendo salido del depósito l y cero en otro caso.
- z_i^s igual a 1 si el vehículo s llega al cliente i antes del inicio e_i de su ventana de tiempo, y 0 en otro caso.

2.2. Variables Continuas

También se definen las variables continuas:

- w_{ij}^{sl} indica el volumen de carga del vehículo s al ir del nodo i al j habiendo salido ese vehículo del depósito l ;
- t_i^s indica el momento en que el vehículo s llega al cliente i .

- u_i^s indica la demora por llegar antes de la apertura de la ventana del cliente i .

2.3. Funciones Objetivo:

Como se vio en la introducción de este capítulo, en el problema planteado se busca minimizar simultáneamente el tiempo total de realizar toda la distribución (proporcional al costo económico en entornos urbanos) y el balanceo de la carga de trabajo asignada a los distintos vehículos.

$$\min_{x_{ij}^{sl}} F(x_{ij}^{sl}) = (f_1, f_2) \quad (1)$$

La expresión (1) hace referencia a la minimización de dos objetivos: f_1 y f_2 .

La función objetivo 1, f_1 , no sólo minimiza el tiempo total insumido en todas las rutas si no también el costo por incumplimiento de las ventanas de tiempo de los clientes, este costo es igual al tiempo de incumplimiento cuando el vehículo haya llegado antes del inicio de la ventana.

$$f_1: \sum_{\substack{l \in V_L \\ s \in S}} \sum_{\substack{i, j \in V_R \\ i \neq j}} (tr_{ij} \cdot x_{ij}^{sl}) + \sum_{i \in V_R} [z_i^s \cdot (e_i - t_i^s)] \quad (1.1)$$

La función objetivo 2, f_2 , busca lograr el objetivo de balanceo de carga en términos de tiempo, para generar programas de trabajo equilibrados entre los diferentes operadores.

$$f_2: \max_{\forall s \in S} \left(\sum_{l \in V_L} \sum_{\substack{i, j \in V_R \\ i \neq j}} (tr_{ij} \cdot x_{ij}^{sl}) + \sum_{i \in V_R} [z_i^s \cdot (e_i - t_i^s)] \right) - \min_{\forall s \in S} \left(\sum_{l \in V_L} \sum_{\substack{i, j \in V_R \\ i \neq j}} (tr_{ij} \cdot x_{ij}^{sl}) + \sum_{i \in V_R} [z_i^s \cdot (e_i - t_i^s)] \right) \quad (1.2)$$

2.4. Restricciones:

La restricción (2) implica que el número de vehículos utilizados no debe exceder al número de vehículos disponibles para cada depósito.

$$\sum_{s \in S_l} \sum_{j \in V_R} x_{lj}^{sl} \leq |S| \quad \forall l \in V_L \quad (2)$$

La restricción (3) controla el cumplimiento de la capacidad de los vehículos, la carga total de la ruta no debe exceder a la capacidad del vehículo.

$$\sum_{r \in V_R} b_r \sum_{j \in V_R} x_{rj}^{sl} \leq Q^l \quad \forall s \in S, \forall l \in V_L \quad (3)$$

La restricción (4) asegura que cada vehículo sale y regresa al mismo depósito.

$$\sum_{j \in V_R} x_{lj}^{sl} = \sum_{j \in V_R} x_{jl}^{sl} \leq 1 \quad \forall s \in S, \forall l \in V_L \quad (4)$$

La restricción (5) preserva el flujo, es decir si el vehículo s llega al cliente r también debe salir de él.

$$\sum_{\substack{i \in V_R, s \in S, \\ l \in V_L}} x_{ir}^{sl} = \sum_{\substack{i \in V_R, s \in S, \\ l \in V_L}} x_{ri}^{sl} = 1 \quad \forall r \in V_R \quad (5)$$

La restricción (6) evita que un vehículo vaya de un depósito a otro.

$$\sum_{j \in V_L} x_{jl}^{sl} = \sum_{j \in V_L} x_{lj}^{sl} = 0 \quad \forall s \in S, \forall l \in V_L \quad (6)$$

Las restricciones (7) y (8) tratan la carga del vehículo y las demandas de los clientes. La (7) indica que la carga del vehículo cuando atraviesa el arco (i, j) , no debe de exceder la capacidad del mismo, si es que ese vehículo atraviesa ese arco.

$$0 \leq w_{ij}^{sl} \leq Q \cdot x_{ij}^{sl} \quad \forall i, j \in V, \forall s \in S, \forall l \in V_L \quad (7)$$

La (8) indica que la carga que se deja en el cliente r debe ser igual a la demanda de ese cliente r .

$$\sum_{j \in V_R} \sum_{s \in S} x_{jr}^{sl} \cdot w_{jr}^{sl} - x_{rj}^{sl} \cdot w_{rj}^{sl} = b_r \quad \forall r \in V_R, \forall l \in V_L \quad (8)$$

La restricción (9) establece que la duración total de la ruta no debe exceder al máximo tiempo permitido para la misma.

$$\sum_{\substack{i,j \in V_R \\ i \neq j}} x_{ij}^{sl} \cdot (tr_{ij} + u_i^s) \leq r_k \quad \forall s \in S, \forall l \in V_L \quad (9)$$

La restricción (10) marca que los tiempos de salida (o demoras) de los depósitos, deben de ser igual a cero.

$$\sum_{s \in S} t_l^s = \sum_{s \in S} u_l^s = 0 \quad \forall l \in V_L \quad (10)$$

La restricción (11) indica que si el cliente j es atendido por el vehículo s perteneciente al depósito l , después del cliente i ; entonces el tiempo de llegada al cliente j debe ser mayor que el de salida desde el cliente i .

$$\sum_{\substack{i \in V_R \\ i \neq j}} x_{ij}^{sl} \cdot (t_i^s + tr_{ij} + u_i^s) \leq t_j^s \quad \forall j \in V_R, \forall l \in V_L, \forall s \in S \quad (11)$$

Las restricciones (12 a), (12b) y (13) verifican el cumplimiento del cierre de las ventanas de tiempo.

$$e_i - \sum_{s \in S} t_i^s \leq l_i \cdot \sum_{s \in S} z_i^s \quad \forall i \in V_R \quad (12a)$$

$$\sum_{s \in S} t_i^s - e_i \leq l_i \cdot \left(1 - \sum_{s \in S} z_i^s\right) \quad \forall i \in V_R \quad (12b)$$

$$\sum_{s \in S} t_i^s - l_i \leq r_k \quad \forall i \in V_R \quad (13)$$

Por último, la restricción (3.14) es de la naturaleza de las variables.

$$x_{ij}^{sl} \in \{0,1\}, \quad z_i^{s1} \in \{0,1\}, \quad t_i^s \in R, \quad u_i^s \in R, \quad w_{ij}^{sl} \in R \quad (14)$$

3. COMPLEJIDAD COMPUTACIONAL

Desde la programación matemática a los problemas de ruteo de vehículos se los clasifica como problemas de *Optimización Combinatoria* y desde la teoría de la complejidad computacional se los clasifica como *NP-Hard* (Garey & Johnson, 1979). En este apartado se revisan estos conceptos a los fines de explicar por qué para resolver el problema objeto de estudio se ha seleccionado la técnica que se presentará en el Capítulo IV.

En los problemas de optimización combinatoria, las variables de decisión son enteras y, por lo general, el espacio de soluciones está formado por ordenaciones o subconjuntos de números naturales. Este tipo de problemas se presenta con frecuencia en la toma de decisiones inherentes a la distribución física y el transporte. La importancia práctica de estos problemas ha hecho que fueran objeto de importantes esfuerzos de investigadores para hallar un método de solución eficiente. Por esto, el área de optimización combinatoria resulta tan atractiva, ya que cualquier contribución en este ámbito tiene repercusiones directas en la industria.

El objetivo de la teoría de la complejidad computacional es clasificar los problemas de acuerdo a su tratabilidad, tomando el o los algoritmos más eficientes para resolverlos. En este sentido, los problemas se clasifican de acuerdo a los recursos indispensables para su solución, en *Tratables* o *Intratable*.

El grado de tratabilidad es subjetivo (varía mucho de acuerdo al modelo computacional, los recursos disponibles, las variantes de las estructuras de datos, etc.). Por lo tanto, un objetivo primario del estudio de la complejidad es definir cuáles problemas son tratables, y cuáles no. Pero para lograrlo primero debemos entender el concepto de complejidad computacional.

La complejidad de un algoritmo se mide en función del uso de dos recursos, el *tiempo de ejecución* (medido en cantidad instrucciones elementales) y, el *espacio o memoria de almacenamiento* (medido en cantidad de variables elementales), que necesita el algoritmo para obtener una solución de un problema para una entrada cualquiera del mismo. Mientras que la complejidad de un problema es equivalente a la complejidad del mejor algoritmo que resuelve ese problema (Talbi, 2009).

Por lo tanto, la medida de complejidad será una función del tamaño del problema. Por ejemplo, tomemos el caso del problema de búsqueda secuencial de un elemento determinado dentro de un vector con n elementos. En este caso el tamaño del problema, está dado por la cantidad de elementos del vector, o sea n . Y el algoritmo consistirá básicamente en comparar cada elemento del vector con el elemento buscado. Luego, en el peor de los casos el algoritmo hará n operaciones de comparación, esto sería cuando el elemento buscado está en la última posición o no está. Entonces informalmente podemos decir que la complejidad de este algoritmo en el peor de los casos es n , porque ese es el número de instrucciones elementales (comparaciones) que hay que realizar para resolver el problema cuando su tamaño es n . Luego, si para $n=100$ un ordenador determinado tarda 8 unidades de tiempo en resolver este problema con este algoritmo, podemos deducir que el tiempo que tardará en resolverlo, para $n=1000$, es de 80 unidades de tiempo. Es decir, que el tiempo de ejecución es una función lineal de tamaño del problema o, en otras palabras, que el algoritmo presenta una *complejidad lineal*.

Al agrupar los algoritmos con *complejidad lineal*, por ejemplo $5n + 1$ o $22n + 7$, o los que tienen una *complejidad cuadrática*, por ejemplo $5n^2 + 1$ o $22n^2 + n + 7$, a cada uno de estos grupos se los denomina *orden de complejidad*, y normalmente al hacer referencia a la complejidad de un algoritmo, nos referimos al orden de complejidad al que pertenece. Por ejemplo, si la complejidad del algoritmo es $22n + 7$ decimos que el mismo pertenece al *orden n* , o más formalmente, a $O(n)$. Mientras que si la complejidad del algoritmo es $22n^2 + n + 7$ decimos que el mismo pertenece al *orden n^2* , o más formalmente, a $O(n^2)$. En este segundo caso hablamos de un nivel de complejidad mayor que en el primero, dado que el tiempo de ejecución depende de forma cuadrática del tamaño del problema.

En general, cuando se hace referencia a uno de estos ordenes de complejidad, se lo generaliza como de *orden polinomial* o, $O(n^c)$, es decir si $c = 2$ es de *orden cuadrático*, si $c = 3$ es de *orden cúbico*.

Otros ordenes de complejidad muy conocidos, pero mucho más complejos que los mencionados son el *exponencial* $O(c^n)$, el *factorial* $O(n!)$, y el *combinatorio* $O(n^n)$.

La evidencia empírica ha permitido acordar que, si un problema admite una solución en tiempo de ejecución polinomial (o sea, pertenece a un *orden de complejidad polinomial*) entonces se trata de un problema *tratable*; en caso contrario se lo considera *intratable*. Es decir, el problema se considera intratable, si es tan difícil que ningún algoritmo determinístico puede resolverlo en tiempo polinomial (Garey & Johnson, 1979). Por ejemplo, si el tiempo de ejecución requerido para resolverlo es n^{10000} o $100.000.000.000 \cdot n$ sería *tratable*; mientras que si fuera $1,00001^n$ o $n^{\log_2 n}$ sería *intratable*.

Para visualizar la diferencia entre un problema tratable y uno intratable, en la tabla 1 se presentan diferentes tamaños de problemas y su *tiempo de ejecución* para distintos órdenes de complejidad.

Tabla 1: Tiempos de ejecución para distintos

$O()$ \ n	$n = 10$	$n = 50$	$n = 100$	$n = 300$	$n = 1000$
	<i>Tiempo de Ejecución</i>				
$5n$	50	250	500	1500	5000
$n \log_2 n$	33	282	665	2469	9966
n^2	100	2.500	10000	90000	un millón (7 díg.)
n^3	1.000	125.000	un millón (7 díg.)	27 millones (8 díg.)	mil millones (10 díg.)
2^n	1.024	16 díg.	31 díg.	91 díg.	302 díg.
$n!$	3.628.800 (7 díg.)	65 díg.	161 díg.	623 díg.	inimaginable
n^n	11 díg.	85 díg.	201 díg.	744 díg.	inimaginable

Fuente: adaptado de Garey y Johnson (1979).

En la tabla se presenta la idea de la tratabilidad o no de un problema, al mostrar la explosión combinatoria en los tiempos de ejecución de los órdenes de complejidad no polinomiales (últimas tres filas), en comparación con los órdenes polinomiales (primeras cuatro filas). Para dejar clara la diferencia, tomemos por ejemplo, el orden polinomial más grande de la tabla 1, este sería $O(n^3)$, y el orden exponencial $O(2^n)$, y supongamos que se pueden ejecutar un millón de instrucciones por segundo (1 mips). En la tabla 2 se expone la magnitud de esta explosión en el tiempo necesario para resolver los problemas no polinomiales.

Tabla 2: Comparación entre complejidad polinomial y exponencial

$O()$ \ n	$n = 10$	$n = 50$	$n = 100$	$n = 300$	$n = 1000$
	<i>Tiempo (sup. 1 millón de operaciones por segundo)</i>				
n^3	0,001 segundos	0,125 segundos	1 segundo	27 segundos	17 minutos
2^n	0,001 segundos	35,7 años	402 billones de siglos	---	----

Fuente: elaboración propia.

Para un tamaño de problema de $n=10$, ambos ordenes de complejidad tienen el mismo tiempo de resolución, pero al aumentar el tamaño de los problemas, por ejemplo para un tamaño de $n = 100$, mientras que el de orden polinomial demora 1 segundo, el exponencial demora 402 billones de años en resolverse, lo que evidencia la intratabilidad de este último tipo de problemas.

Un aspecto importante en este punto es clasificar a los problemas en base a sus clases de complejidad. Una clase de complejidad representa el conjunto de todos los problemas que se pueden resolver usando una cantidad dada de recursos computacionales. Siguiendo a Talbi (2009), a grandes rasgos destacan dos clases importantes de problemas: los *Clase P* y los *Clase NP*.

La *clase de complejidad P* representa el conjunto de todos los problemas que pueden ser resueltos por un algoritmo determinista en tiempo polinomial. Es decir, son los problemas para los que existe un algoritmo conocido con tiempo de ejecución polinomial para resolver el problema.

La *clase de complejidad NP* representa el conjunto de todos los problemas que solo pueden resolverse en tiempo polinomial de forma aproximada mediante un algoritmo no determinista. Un problema es *NP-completo* si todos los demás problemas de la clase NP se reducen polinomialmente a este problema, es decir, la clase de los problemas *NP-completos* contiene a los problemas más difíciles en NP. Y los problemas *NP-duros* (o *NP-hard*) son problemas de optimización cuyos problemas asociados son NP-completos. Son un subconjunto de la clase *NP-Completo*, y son los más difíciles de resolver dentro de esa clase.

Entonces, los algoritmos con orden de complejidad polinomial constituyen una buena base para identificar los problemas tratables, por lo que se identifica gene-

ralmente la *Clase P* como una aproximación razonable del concepto de tratabilidad. Para mayor detalle de este tema, se recomienda Garey y Johnson (1979).

El problema básico del TSP presentado en el capítulo 2, es un problema no tratable. Lenstra y Kan (1981) demostraron que el VRP y sus variantes pertenecen al tipo *NP-duro*, incluso prueban que el *L-Depot Vehicle Scheduling Problem* (formulación similar al MDVRPTW mono-objetivo), es intratable y abierto. Como describen Kohl et al. (1999), Laporte y Nobert (1987) y Laporte (1992) los algoritmos deterministas sólo pueden resolver problemas relativamente pequeños de este tipo, dado que al aumentar el tamaño de las instancias del problema, el tiempo de ejecución crece de forma explosiva.

El VRPTW es considerado un problema *NP-duro* en sentido fuerte, pues contiene como un caso particular al VRP. Lo que indica que no es solucionable de forma óptima por métodos exactos a partir de instancias pequeñas como $n=40$ o 50 (Kindervarter & Savelsbergh, 1997) (Olivera, 2004). El problema MDCVRP-TW en su versión mono objetivo, también es considerado perteneciente a la clase *NP-Duro* (Polacek, Hartl, Doerner, & Reimann, 2004).

Luego, dada la intratabilidad de los problemas de ruteo de vehículos, entre los que se encuentra el problema objeto de estudio del presente trabajo, se requiere de algoritmos no deterministas para resolverlo en tiempos polinomiales. Es abundante la literatura que demuestra el éxito de los algoritmos evolutivos obteniendo soluciones para este tipo de problemas pertenecientes a la clase *NP-duros* (Baaj & Mahmassani, 1995). Por esta razón, la técnica propuesta en este trabajo pertenece a esta familia de técnicas meta-heurísticas englobados bajo el término Computación Evolutiva descriptas en el capítulo dos.

CAPÍTULO IV. RESOLUCIÓN E IMPLEMENTACIÓN

En este capítulo se presenta la técnica para resolver el problema de decisión objeto de estudio. Como fue mencionado en el Capítulo II, se utilizará un algoritmo evolutivo multi-objetivo, que es una hibridación del conocido NSGA-II (Deb, Agrawal, Pratap, & Meyarivan, 2000). Se utiliza una estrategia de introducción de preferencias parciales para dirigir la búsqueda del algoritmo. Se especifica un criterio de codificación original, usando una representación con cadenas de enteros con cuatro genomas. Se presentan operadores evolutivos cerrados y especializados, introduciendo un operador de recombinación original. Y finalmente se presenta los resultados de la validación y el testeo del algoritmo propuesto y se realiza el análisis de la esta experiencia y sus resultados.

1. OPTIMIZADOR PROPUESTO

1.1. Introducción

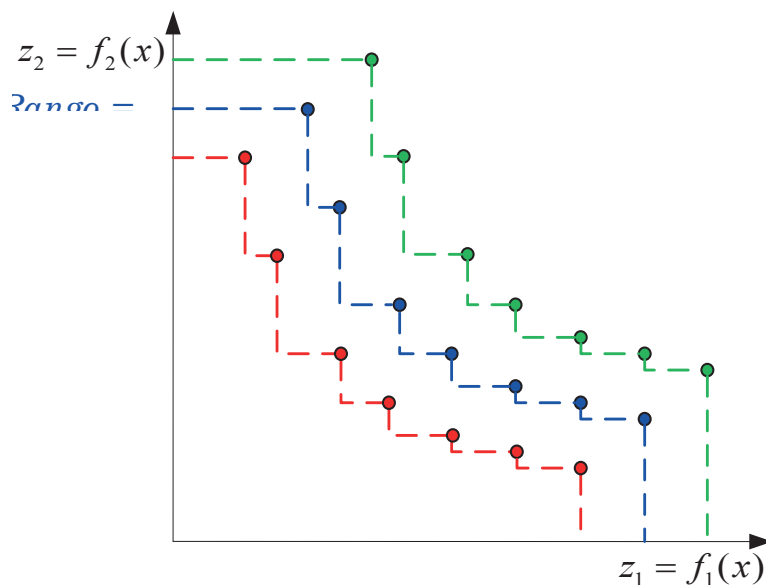
Como fue descrito en el marco teórico, para buscar soluciones a problemas de ruteo de vehículos desde un abordaje multi-objetivo, las técnicas más utilizadas en la literatura son las metaheurísticas. Dada la intratabilidad de este tipo de problemas, estas técnicas representan el estado del arte en para su resolución, esto se refleja por ejemplo en que de los trabajos publicados en ruteo de vehículos con múltiples objetivos y con múltiples depósitos, en el 100% de los casos en que se trabaja con problemas con un tamaño superior a los 20 nodos, se utilizan procedimientos heurísticos o metaheurísticos, y estos últimos representa el 83% de los casos (Montoya-Torres, Franco, Isaza, Jiménez, & Herazo-Padilla, 2015).

El algoritmo implementado para llevar adelante el proceso de optimización, pertenece a la familia de las metaheurísticas, y dentro de esta a los algoritmos evolutivos multi-objetivo. Es una variante hibridada del conocido algoritmo genético de ordenamiento no dominado con elitismo (NSGA-II: *Non-dominated Sorting Genetic Algorithm*), propuesto por Deb, Agrawal, Pratap y Meyarivan (2000). Y se utiliza el concepto de g-dominancia (Molina, Santana, Hernández-Díaz, Coello, & Caballero, 2009) para la introducción de preferencias parciales del decisor, que guíen la búsqueda del algoritmo evolutivo multi-objetivo.

1.2. Estructura NSGA-II

El NSGA-II, es un algoritmo que usa el concepto de dominancia de Pareto, es decir, que la población se clasifica en *rangos* acorde a una regla de dominancia y luego a cada solución se le asigna su valor de aptitud en base a su rango en la población. El esquema de clasificación utilizado está basado en la idea de Goldberg (1989), que consiste en asignar el rango 1 a todos los individuos no dominados de la población. Luego sin considerar los individuos ya ranqueados, se repite el proceso asignando rango 2 a los individuos no dominados. Esto continúa hasta que todos los individuos de la población tengan rango asignado. Pero estos autores mejoraron este procedimiento mediante uno llamado *rápido ordenamiento no dominado* (*fast nondominated sorting*) llegando al mismo resultado pero reduciendo la complejidad desde $O(NM^3)$ a $O(MN^2)$, ver detalladamente en Deb, Agrawal, Pratap, Agarwal y Meyarivan (2002), pág. 184.

Figura 19: Ordenamiento no dominado de NSGA-II



Fuente: Adaptado de Goldberg (1989)

Este método de asignación de aptitud basado en rangos, tiende a favorecer la convergencia hacia el frente óptimo de Pareto. Pero puede tener problemas con la diversidad de las soluciones, tanto en el espacio de las decisiones como en el de los objetivos. Por esta razón es que el algoritmo utiliza como método de mantenimiento de la diversidad, la *distancia crowding* (o distancia de hacinamiento). Este mé-

todo consiste en evaluar la densidad de soluciones presentes alrededor de una solución i . Disminuyendo las posibilidades de supervivencia de las soluciones presentes a una región con mucha densidad, es decir, donde hay varias soluciones conocidas. De esta manera, a las soluciones más aisladas se les asigna una distancia grande, para darle prioridad en la exploración a estas zonas y preservar la diversidad en el frente de Pareto.

Además, al igual que los demás algoritmos de segunda generación es un algoritmo *elitista* (Coello C. C., 2006). Es decir, que en cada nueva generación se conservan los mejores individuos encontrados, lo que acelera significativamente la convergencia del AG al prevenir la pérdida de buenas soluciones.

Estructura explicada del NSGA-II:

1. Inicialización:

1.1. Población inicial de padres, P_t , de tamaño N :

Se crea de forma aleatoria la población inicial de padres¹¹.

***1.2. Evaluar* individuos de P_t mediante las funciones objetivo.**

1.3. Aplicar Ordenamiento No Dominado a P_t (ver paso 2.4)

2. Bucle Principal:

2.1. Generar población de descendientes, Q_t , de tamaño N :

Se utiliza la población de padres, P_t , para crear una nueva población de descendientes, Q_t , mediante:

2.1.1. Selección:

Selección por Torneo binario con distancia crowding de la población de *Progenitores*, a partir de P_t : este operador compara de a dos soluciones y elige un ganador del torneo (*torneo binario*).

Se basa en:

¹¹ En la hibridación propuesta en el presente trabajo, la población inicial se genera mediante un proceso de clustering aleatorio, que genera solo individuos factibles.

- *Ranking*: nivel de no dominancia del Frente al que pertenece cada uno de los dos individuos seleccionados al azar de la P_t .
- *Distancia Crowding*: Si el anterior no es concluyente y tenemos más de un ganador, tomamos el que tiene mayor distancia crowding (ver paso 2.6.1).

De esta manera se promueve como progenitores de la siguiente generación los individuos que ocupan *los mejores frentes* (ranking) y *los más diversas* (distancia crowding).

2.1.2. Aplicar operadores evolutivos a la población de *Progenitores*:

La población de descendientes, Q_t , se genera a partir de la aplicación de los operadores de *cruzamiento* y *mutación* sobre la población de *Progenitores*.

2.2. ***Evaluar individuos*** de Q_t , mediante las funciones objetivo.

2.3. ***Combinación de poblaciones***: Nueva población de padres y descendientes, R_t , tiene tamaño $2N$:

$$R_t = P_t \cup Q_t$$

2.4. ***Ordenamiento no dominado*** de R_t :

Se clasifica los individuos en diferentes frentes de Pareto, de acuerdo a su *rango de no dominancia*.

2.4.1. *Fitness*:

Se asigna a cada individuo un *fitness* igual a su *rango de no dominancia*. Si está en el 1º Frente de Pareto (en adelante, 1ºFP) tiene rango 1, \therefore fitness = 1. Si está en el 2º FP tendrá rango 2, \therefore fitness = 2. Y así sucesivamente. (Se asume minimización).

2.4.2. *Elitismo*:

Está asegurado debido a que todos los miembros de la población anterior y actual son incluidos en R_t . Y las soluciones que quedan en el primer frente, son las mejores soluciones de la población R_t .

2.5. Elección de la población de la nueva generación, P_{t+1} , de tamaño N :

Después de ordenar R_t , se eligen los mejores N individuos, para formar la nueva población P_{t+1} . Es decir, miembros del **1^o FP**, si es necesario del **2^o FP**, luego del **3^o FP**, y así hasta un último ***L*-ésimo FP** que complete los N lugares de P_{t+1} .

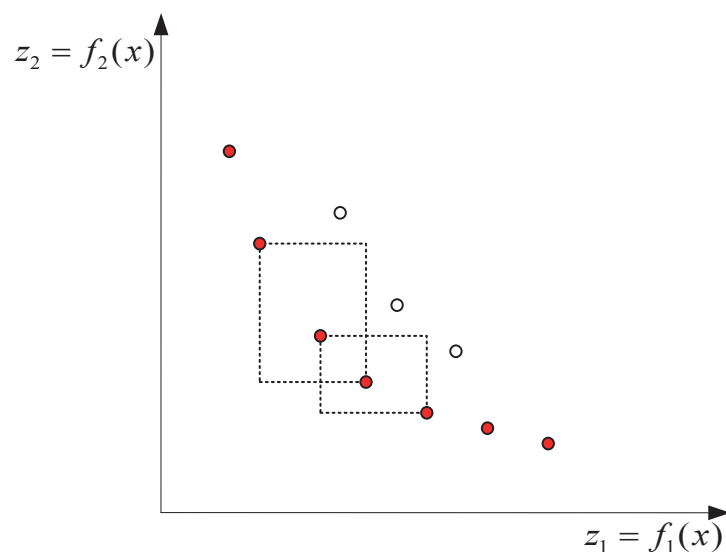
2.5.1. Si en ese último ***L*-ésimo FP** hay más miembros que lugares disponibles en P_{t+1} , se realiza un 2^o ordenamiento de ese *L*-ésimo frente por *distancia-crowding*, para tomar del último frente, los mejores individuos que faltan para completar P_{t+1} .

Ordenamiento por Distancia Crowding:

La *distancia crowding*, es la medida del espacio de búsqueda alrededor de la solución evaluada, que no está ocupado por otra solución en la población. Es decir, se busca tomar del último frente (puntos rojos de Figura 20), los individuos que están en regiones menos pobladas, favoreciendo la diversidad.

Su cálculo consiste en evaluar la densidad de soluciones presentes alrededor de una solución i , mediante el cálculo de la distancia d_i , basada en el perímetro del hipercubo formado por los vértices de las soluciones más próximas a i , en el espacio de los objetivos.

Figura 20: Distancia Crowding en NSGA-II



Fuente: adaptado de Deb, Agrawal, Pratap, Agarwal y Meyarivan (2002)

Puede verse en la Figura 20, que al comparar la solución i con la $i + 1$, el hipercubo de la solución i tiene un mayor perímetro que el de la solución $i + 1$, por lo que la solución i quedaría mejor clasificada y tendría menos probabilidades de ser eliminada que la solución $i + 1$.

3. Finalizaciyn

Este bucle se repite hasta alcanzar el criterio de finalización de tipo MAX. Es decir, se detiene al alcanzar una cantidad máxima de generaciones permitida.

En la Tabla 3, se presenta el pseudocódigo del NSGA-II, en el que se ve la estructura descrita anteriormente.

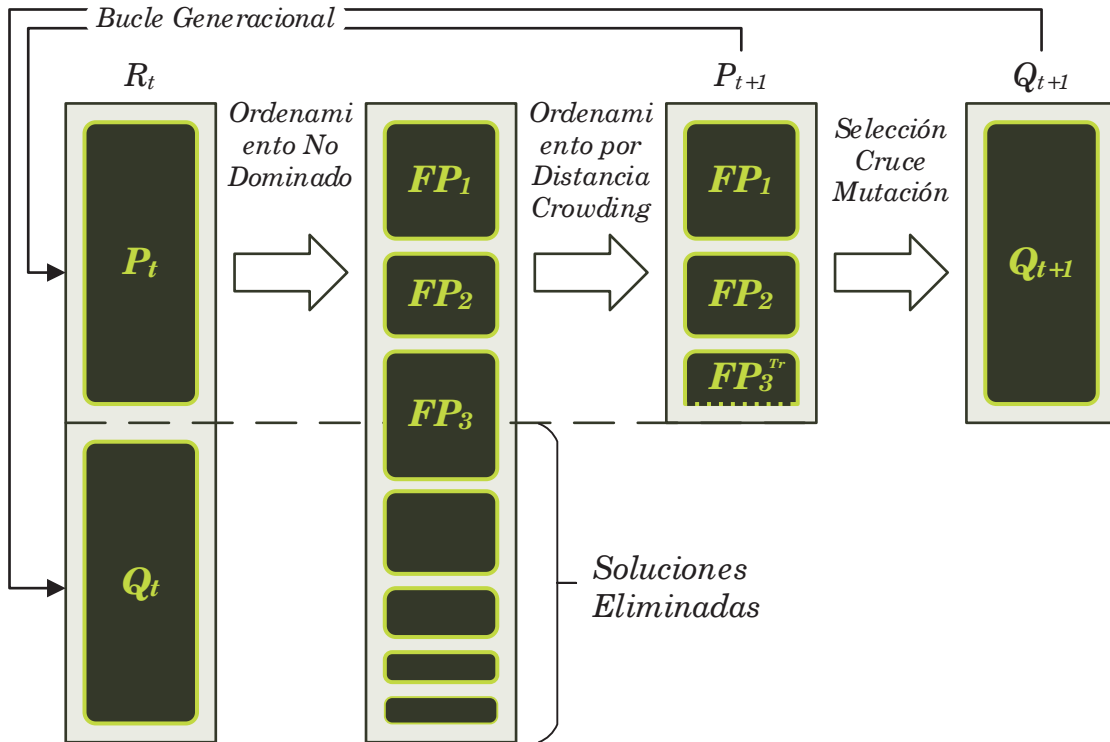
Tabla 3: Pseudocódigo HG-NSGA-II

1:	Inicio	
2:	$t = 0$	<i>% Generación</i>
3:	$P_0 \leftarrow$ Generar aleatoriamente la <i>Población Inicial</i>	
4:	Evaluar Individuos de P_t	
5:	$P_t = (FP_1, FP_2, \dots) \leftarrow$ Aplicar <i>Ordenamiento No Dominado</i> a P_t	
6:	Para $\forall FP_i \in P_0$ hacer:	
7:	<i>Calcular Distancia Crowding</i> a FP_i	
8:	Fin-Para	
9:	$t \leftarrow 0$	
10:	Mientras No sea alcanzado el <i>Criterio de Parada</i> hacer	
11:	<i>Progenitores</i> \leftarrow Aplicar <i>Operador de Selección por Torneo</i> a P_t	
12:	$Q_t \leftarrow$ Aplicar <i>Operadores Evolutivos</i> a <i>Progenitores</i>	<i>% Descendientes</i>
13:	Evaluar <i>Descendientes</i> , Q_t	
14:	$R_t = P_t \cup Q_t$	
15:	$R_t = (FP_1, FP_2, \dots) \leftarrow$ Aplicar <i>Ordenamiento No Dominado</i> a R_t	
12:	$i \leftarrow 1$	<i>% número de frente</i>
13:	$P_{t+1} \leftarrow \emptyset$	
14:	Mientras $ P_{t+1} + FP_i < N$ hacer: <i>% N, número de individuos en P_t</i>	
15:	<i>Calcular Distancia Crowding</i> a FP_i	
16:	$P_{t+1} \leftarrow P_{t+1} \cup FP_i$	
17:	$i \leftarrow i + 1$	
18:	Fin-Mientras	
19:	$FP'_i \leftarrow$ ordenar por distancia crowding a FP_i	
20:	$P_{t+1} \leftarrow P_{t+1} \cup FP'_i[N - P_{t+1}]$	
21:	$t \leftarrow t + 1$	
22:	Fin-Mientras	

Fuente: Adaptado en base a Deb, Agrawal, Pratap, Agarwal y Meyarivan (2002)

En la Figura 21, se presenta esquemáticamente el mecanismo de promoción de individuos.

Figura 21: Mecanismo de promoción de individuos NSGA-II



Fuente: Adaptado de Deb, Agrawal, Pratap, Agarwal y Meyarivan (2002)

1.3. Mejoras propuestas

Como fue presentado anteriormente, la herramienta computacional desarrollada para llevar adelante el proceso de optimización en el contexto modelado, es una hibridación del algoritmo NSGA-II visto en el apartado anterior (punto 1.2), dada su demostrada adaptación a este tipo de problemas. Las mejoras propuestas al NSGA-II consistieron en la introducción de información específica mediante un criterio de representación diseñado para este tipo de problemas, en el uso de un operador reproductivo de cruce (desarrollado específicamente para este problema) y en la introducción de preferencias parciales (*g-dominancia*). A este procedimiento se lo llamó HG-NSGA-II y a continuación, se presentan las mejoras propuestas en este algoritmo:

1.3.1. Representación

Respecto al criterio de representación, se hibrida el algoritmo siguiendo la estrategia de Michalewicz (1992), mediante la incorporación directa de conocimiento específico en la representación y en los operadores genéticos.

Se usa una representación con *un cromosoma* compuesto por *cuatro genomas*, para la etapa de calibración, y solo tres para la etapa de optimización. El *primer genoma* contiene secuencia de visitas o programas de cada operador. Esta representación se conoce en la literatura como “*path representation*”, y consiste en una permutación de números naturales. Los *genomas 2 y 3* contienen información respecto a la secuencia del genoma 1. Y el *genoma 4* puede contener información de parámetros a evolucionar en la etapa de calibración.

Este tipo de representación con un alfabeto de mayor cardinalidad, permite lograr mayores niveles de eficiencia en este tipo de problemas que la codificación binaria original de Holland (1975). Al no ser tan rígida y facilitar la consideración de restricciones no triviales del espacio de búsqueda. Pero tiene como consecuencia, la necesidad de utilizar operadores específicos para esta representación, adaptables al dominio.

A continuación, se describen estos cuatro genomas y se presenta un ejemplo de un individuo que representa una solución, para un problema con veinte clientes minoristas, tres depósitos y cuatro vehículos por depósito:

Genomas:

G1: Secuencia de nodos, de todas las rutas de todos los depósitos.

G2: Secuencia de índices de finalización de cada ruta.

G3: Secuencia de cantidades de rutas en cada depósito.

G4: Parámetros a evolucionar en la etapa de calibración.

Cromosoma = [**G1**, **G2**, **G3**, **G4**];

Dimensiones:

G1: 1 x n° de clientes (en el ejemplo 20)

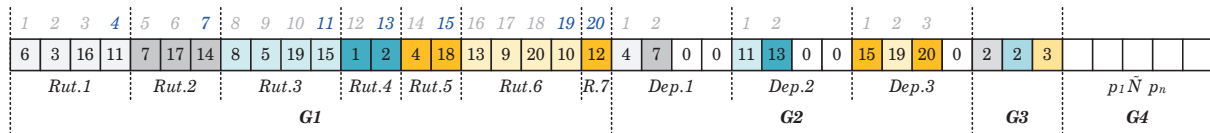
G2: 1 x n° de vehículos (en el ejemplo 4) por depósito

G3: 1 x n° de depósitos

G4: 1 x n° de parámetros a evolucionar en la etapa de calibración

En la Figura 22 puede se presenta esquemáticamente un cromosoma, utilizando la representación propuesta.

Figura 22: Esquema del Cromosoma



Fuente: elaboración propia

El primer genoma ($G1$) se compone de los primeros veinte lugares, los que corresponden a los nodos clientes presentes en el programa de distribución del ejemplo. Cada permutación posible es a una secuencia de visita para todos los clientes. El segundo genoma ($G2$) se compone de los siguientes doce lugares, que indican el índice donde terminan las secuencias de cada ruta que parte de cada uno de los tres depósitos. Por ejemplo, el primer lugar (o *locus*) del segundo genoma, indica que la primera ruta termina en el índice 4 (también resaltado en el esquema en color azul encima del cromosoma) y del primer genoma se observa que la secuencia de esta primera ruta es 6, 3, 16 y 11. Los ceros del genoma dos, son rutas vacías (no utilizadas) en cada depósito. Los tres lugares del genoma tres, indican la cantidad de rutas presentes en cada depósito. Y los lugares presentes en el genoma 4, son para cargar parámetros a evolucionar durante la etapa de calibración del método.

1.3.2. Operador de recombinación

El operador de cruce propuesto, al que llamamos *ERX-MD*, está inspirado en el conocido operador de cruce por recombinación de arcos (*ERX, Edge Recombination Operator*) de Whitley, Starkweather y Fuquay (1989), en el que la descendencia se genera combinando los cromosomas de dos progenitores a partir de la lista de conexiones factibles de sus genes. Ese operador, está diseñado específicamente para problemas de optimización combinatoria, con representación del tipo permutación de enteros.

Se construye el descendiente, a partir de la lista de conexiones factibles de los progenitores. Para esto:

1°. **Lista:**

Se realiza una lista de arcos factibles para el subgrafo de cada cluster (cluster de nodos clientes asignados a un depósito determinado), a partir de la información genética de los progenitores seleccionados.

2°. **Re-clustering:**

Se realiza un reagrupamiento, para los nodos que figuran en clusters distintos en sus progenitores (sobre asignados), buscando maximizar la cantidad de conexiones hacia adelante que tienen estos nodos en cada cluster. Es decir, para cada nodo sobre asignado se comprueba la lista de conexiones un movimiento hacia adelante dentro de cada cluster al que pertenece y se lo deja en el que se generen las mayores posibilidades de conexión.

3°. **Ruteo:**

Se construye el descendiente, a partir de la lista de arcos factibles, y de la lista de conexiones hacia adelante a partir de la posición actual. Para esto se utiliza una variación de la heurística de inserción más cercana de Sólon (Sólon, 1987), que consiste en sucesivas inserciones clientes en las rutas, buscando minimizar una función de costos en base al siguiente procedimiento:

Un nodo j se inserta después del nodo actual i , si tiene el menor costo ϑ , que surge de la ponderación entre tres medidas:

tr_{ij} : mide la *cercanía temporal* entre los nodos i y j , calculada en términos de la velocidad promedio para recorrer el arco en una zona determinada. Prioriza al que se llega antes.

δ_j : cuenta la cantidad de conexiones hacia adelante que tiene el nodo j . Prioriza el que tiene más posibilidades de quedar aislado.

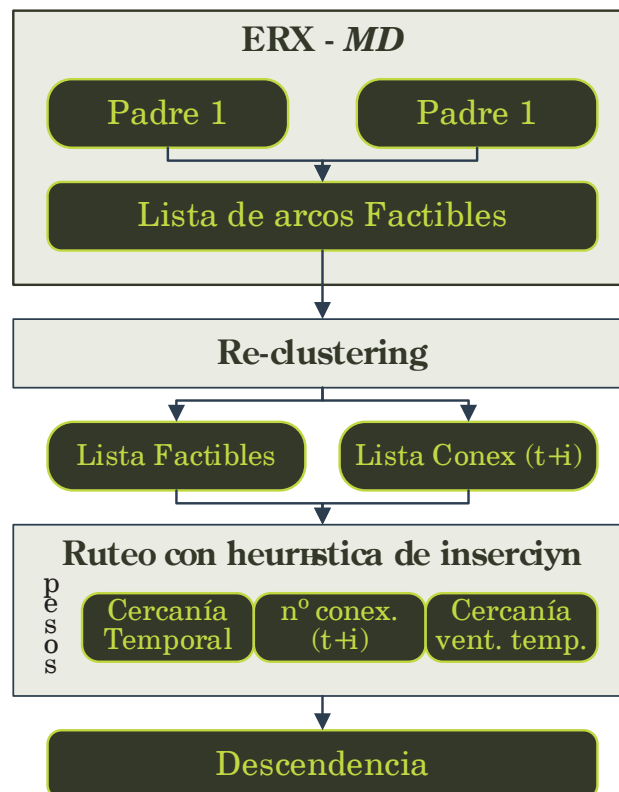
φ_j : mide la cercanía en la apertura y el cierre de la ventana temporal del nodo j . Es decir, la urgencia para insertarlo en la ruta. $\varphi_j \leq 0$ implica que j ya no puede insertarse. φ_j cercanos a 0 , indican que dentro de “pocas” iteraciones, el

cliente j no podrá ser insertado en esta ruta. Prioriza el que tenga el cierre más próximo de la ventana de servicio.

Luego, para elegir el nodo siguiente entre los nodos conectados al nodo i , se comparan los distintos nodos conectados (es decir, $j, j+1, j+2, \dots$) utilizando la siguiente combinación lineal $\vartheta_{ij} = \alpha_1 \cdot tr_{ij} + \alpha_2 \cdot \delta_j + \alpha_3 \cdot \varphi_j$.

4°. En el caso del primer nodo de una ruta de distribución, la heurística cambia las prioridades y busca nodos con cierre de ventana temporal cercana a la apertura de la ventana del depósito que se esté programando.

Figura 23: Esquema del Operador de Recombinación (ERX-MD)

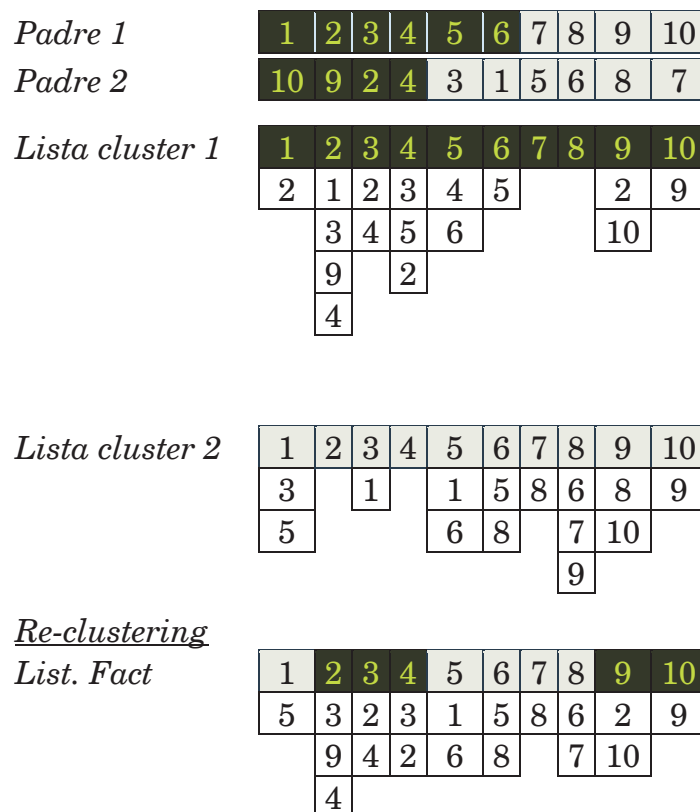


Fuente: elaboración propia

En la Figura 23 se presenta el esquema del operador de recombinación descrito anteriormente.

En la Figura 24 se presenta la idea básica del re-clustering mediante un ejemplo muy sencillo en el que se toman dos padres con diez nodos cada uno, dos clusters (*cluster 1* color verde y *cluster 2* color gris), y se supone además que hay una sola ruta por cluster.

Figura 24: Ejemplo simplificado Re-clustering



Fuente: elaboración propia.

En este ejemplo sencillo, una vez armadas las listas de arcos factibles de cada cluster, puede verse que el nodo 6 pertenece a ambos clusters (ya que en el padre 1 estaba en el cluster 1 mientras que en el padre 2, estaba en el cluster 2. Después del re-clustering, el nodo 6 queda asignado al cluster 2, dado que tiene más conexiones (nodos 5 y 8) que en el cluster 1 (nodo 5), quedando así con posibilidades de conectarse al nodo 5 y al 8.

En este ejemplo simplificado ya en el primer movimiento se puede ver que, si se hubiese dejado el nodo 6 en el cluster 1, este nodo hubiera quedado aislado, dado

que el nodo 5 (también sobre asignado), fue asignado al cluster 2 porque es en el que tiene mayores posibilidades de conexión. Luego, esto tiene como consecuencia la reducción del número de conexiones del sub-grafo del cluster 2

A continuación, se presenta la lista modificada de arcos factibles después del re-clustering de la Figura 24, en la que el nodo 6 se asigna al cluster 1 y queda aislado:

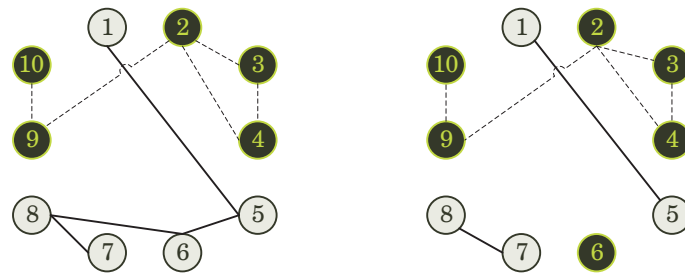
Re-clustering

Lista Fact

1	2	3	4	5	6	7	8	9	10
5	3	2	3	1	×	8	7	2	9
	9	4	2	×	×		×	10	
	4								

En la Figura 25 se presentan los arcos factibles en cada cluster, según se asigne el nodo 6 al cluster 1 o al cluster 2.

Figura 25: sub-grafos de cluster 1 y 2 según la asignación del nodo 6



Fuente: Elaboración propia

Puede verse que al asignar el nodo 6 al cluster 1 (sub-grafo de la derecha de la Figura 25) no solo queda aislado este nodo dentro del cluster 1, sino que también se reducen las conexiones factibles de los nodos 5 y 8 del cluster 2, dejando el cluster 2 con solo dos arcos factibles.

Es importante destacar que la estrategia del *ERX-MD* garantiza de una forma económica, en términos de recursos computacionales, que un nodo insertado en una ruta, cumpla con todas las restricciones y que el operador sea cerrado. Es decir, que los individuos que forman la descendencia, cumplan con todas las restricciones impuestas a este tipo de problemas.

En la literatura dedicada a resolver problemas de ruteo de vehículos que consideran simultáneamente múltiples depósitos y ventanas temporales, se encuentran dos tipos

de estrategias. Por un lado, están las estrategias que implican heurísticas en las que primero se determinan las rutas y después se asignan las rutas a cada depósito. Y, por otro lado, están las que primero asignan clientes a un depósito y luego determinan las rutas para esos clientes asignados a un depósito.

El *ERX-MD*, utiliza una combinación de estrategias. Toma la información respecto a los clusters proveniente de los progenitores, luego reagrupa los nodos sobreasignados mediante el proceso de re-clustering descrito y finalmente determina las rutas para cada cluster. El re-clustering dentro del proceso de cruce, permite mejorar la conectividad dentro del subgrafo de cada cluster (nodos asignados a un depósito), sin necesidad de utilizar costosos procedimientos de reparación.

Respecto al operador de mutación, no hay ningún aporte en este proceso, se busca introducir diversidad mediante una adaptación del conocido operador de Inserción (Fogel D. B., 1988). Consistente en seleccionar un gen perteneciente al genoma uno e insertarlo en otro lugar factible del mismo genoma, elegido aleatoriamente del mismo individuo. Los cambios provocados se pueden dar tanto intra-ruta, como inter-ruta e inter-depósito.

1.3.3. Introducción de Preferencias Parciales (*g*-dominancia)

Como se presentó en el punto 4.3 del Capítulo II al describir el proceso de toma de decisiones, el agente decisor debe elegir de entre un conjunto de acciones factibles y eficientes, aquella acción que le conduce al mejor resultado dadas sus preferencias.

Y se supuso que, dado el contexto decisional objeto de estudio, el agente decisor ¹² tiene una idea aproximada de qué soluciones pueden ser preferidas y puede especificar preferencias parciales. Luego, estas preferencias son utilizadas para conducir el proceso de búsqueda hacia la región de interés en la que se encuentra la alternativa que será finalmente elegida.

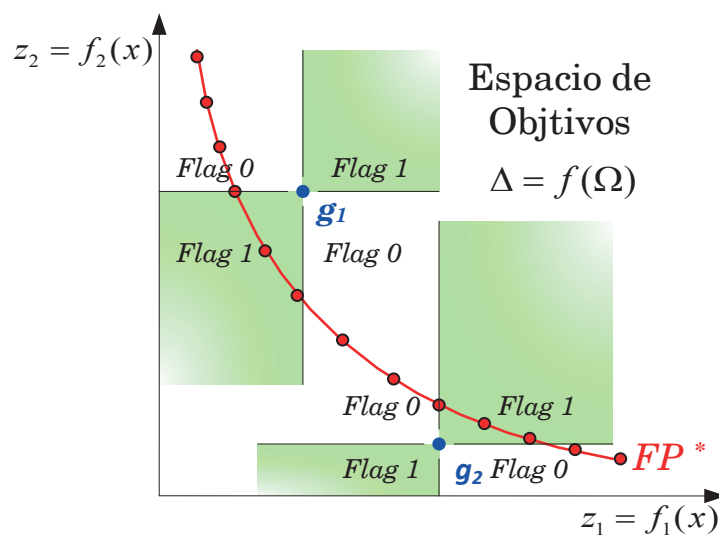
Como método de incorporación de preferencias parciales, se utiliza la idea de *g*-dominancia de Molina, Santana, Hernández-Díaz, Coello y Caballero (2009), que consiste en recoger las preferencias como un punto de referencia *g*, y a partir de

¹² Responsable de logística encargado de programar la distribución diaria de mercadería.

este punto, se prefieren aquellas soluciones que lo dominan o aquellas que son dominadas por él. Esto se muestra en la Figura 26, donde hay dos puntos de referencia señalados por un decisor, y para un frente determinado, g_1 es factible y g_2 no. Luego, la g -dominancia implica que se prefieren las soluciones pertenecientes a las regiones sombreadas (soluciones que dominan o son dominadas por los puntos g) al resto de las soluciones (Incomparables con g).

Para la implementación de la g -dominancia, se penaliza en la función objetivo vectorial, las soluciones no pertenecientes a las regiones de preferencia. Por ejemplo para el punto de referencia g_1 , se penaliza con un valor M muy grande¹³ a los valores de la función objetivo de las soluciones con $flag_{g_1}(F(x)) = 0$, para que sean dominadas por cualquier solución con $flag_{g_1}(F(x)) = 1$. Entonces, modificando la función de evaluación del MOEA mediante la inclusión de esta estrategia de penalización, se logra la implementación de la g -dominancia.

Figura 26: Concepto de g -dominancia



Fuente: adaptado de Molina y otros (2009)

Tabla 4: Pseudocódigo g -dominancia

1:	Inicio
2:	Evaluar $f_i(x), i=1, \dots, p$
3:	Calcular $flag_g(F(x))$

¹³ Siempre suponiendo que estamos minimizando.

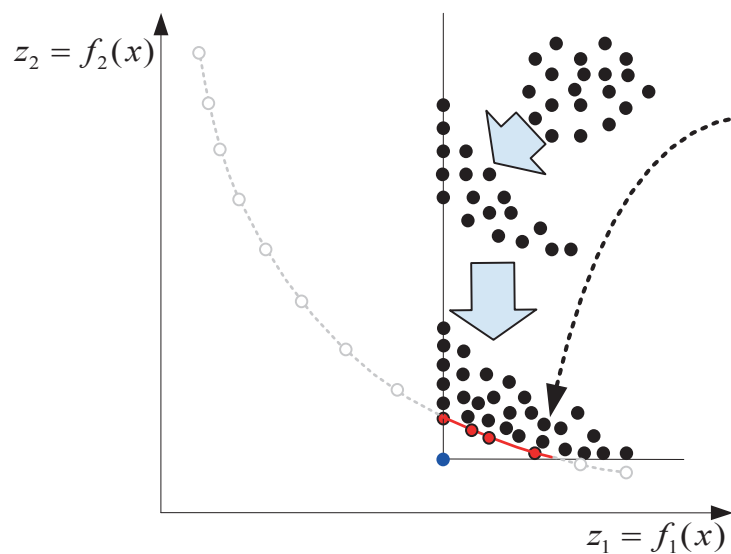
- 4: **Si** $flag_g(F(x)) = 0$ **Entonces**
5: $f_i(x) = f_i(x) + M, i=1, \dots, p$
6: **Fin-Si**
7: **Fin**
-

Fuente: Molina, Santana, Hernández-Díaz, Coello y Caballero (2009)

Esta modificación hace posible usar g-dominancia, con cualquier meta-heurística multi-objetivo.

En la Figura 27, se muestra esquemáticamente el efecto de la aplicación de esta estrategia sobre evolución durante el proceso de optimización.

Figura 27: Evolución generacional con g-dominancia



Fuente: elaboración propia.

En la parte superior de la Figura 27 puede verse una población inicial de individuos que al ir evolucionando con el paso de las generaciones se acerca a la frontera de Pareto. En la representación, la evolución de los individuos se concentra en la región de interés del agente decisor. Esto hace que el proceso de optimización evolucione más rápidamente hacia la porción de interés del frente, que si se tratara de encontrar el frente de Pareto completo.

2. VALIDACIÓN Y TESTEO

En este apartado se presenta la validación del modelo y el testeo de la herramienta algorítmica implementada para la resolución del problema objeto de estudio. Dadas las características originales del problema planteado, no se conocen resultados publicados comparables, para testear las mejoras propuestas en el algoritmo, ya sea en base a listados de problemas académicos o no. De manera que se realizará la validación y testeo del desempeño sobre la información de una instancia real, comparando la propuesta realizada en el presente trabajo, contra el obtenido de la aplicación de dos algoritmos evolutivos muy conocidos y utilizados en la resolución de problemas de ruteo de vehículos con múltiples objetivos, uno es el NSGA-II de Deb, Agrawal, Pratap y Meyarivan (2000) y (2002) sin las mejoras propuestas en el presente trabajo, y otro es el SPEA2 de Zitzler, Laumanns y Thiele (2001). Los operadores evolutivos usados en los algoritmos implementados en su versión estándar usaron como operador de cruce un *Order Crossover* propuesto originalmente por Davis (1985), para la mutación un Operador de Mutación por Intercambio o *Swap Mutation*, y se utilizó un operador de reparación cíclico. La experimentación computacional consistió en realizar de a treinta corridas independientes con cada método, con diferentes puntos de referencia g (ver *g-dominancia*). Luego se realiza la comparación de los resultados obtenidos por cada método, en base a indicadores descriptos posteriormente en ésta capítulo.

En la etapa de calibración de parámetros de los algoritmos, para etapa experimental, se utilizó una estrategia de control de parámetros auto-adaptativa, en la que los valores de los parámetros evolucionan junto con los individuos, codificados como parte de los cromosomas (ver *Representación*) y sujetos a las mismas reglas de variación y selección de la población. Como resultado se obtuvieron como parámetros óptimos para realizar los experimentos, una probabilidad de cruce de 0.8, una probabilidad de mutación de 0.01, un número máximo de generaciones de 1000 y un tamaño de la población de 500.

2.1. Caracterización de la red de distribución

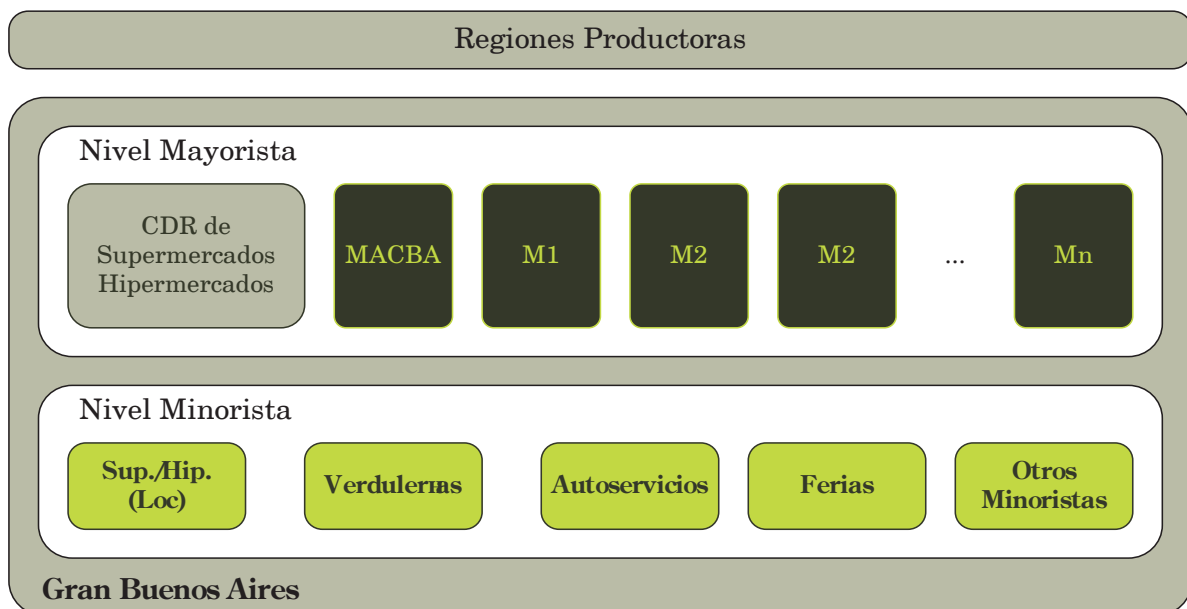
Como se comentó anteriormente, el problema de distribución sobre el que se busca validar el modelo y testear el algoritmo propuesto para su resolución, está inspira-

do en información proporcionada por un operador logístico bonaerense especializado en la distribución urbana de mercaderías. La información se recabó mediante entrevistas a responsables de la empresa y de otras empresas referentes del sector, que operan en distintos niveles de la etapa de comercialización de la cadena de suministros de frutas en fresco. Con el objetivo de caracterizar la red de distribución bajo estudio y la problemática que enfrentan los distintos operadores, también se recopiló información relevante al objeto de estudio, como informes y datos de diferentes organismos nacionales como INDEC, SENASA, FUN-BAPA y de la base de datos del Mercado Central.

La red de distribución de la que participa el operador, tiene como *nivel superior* el Mercado Central de Buenos Aires (MCBA) junto a otros mercados regionales como, el Mercado de Abasto 3 de Febrero (M1), el Mercado Consorcio de Beccar (M2), el Mercado de abasto de Quilmes (M3), entre otros. Y tiene como nivel inferior distintos puntos de venta minorista.

A continuación, se presenta un esquema simplificado de los niveles de distribución mayorista y minorista de la cadena de suministro en cuestión.

Figura 28: Niveles de distribución



Fuente: elaboración propia.

En la Figura 28, se muestra que la empresa participa del canal indirecto de comercialización de fruta en fresco, en el que desde las regiones productoras se vende a través de los mercados concentradores y luego expendedores minoristas de dife-

rentes tipos. Una vez en los mercados concentradores, los puestos mayoristas hacen su distribución a diversos canales minoristas, como verdulerías, autoservicios, mercados minoristas y otros (ferias, puestos callejeros, etc.) para su venta a consumidor final. Mientras que la mercadería ingresada al canal supermercados/hipermercados es enviada en su totalidad a las bocas de expendio minoristas de las cadenas, para su venta a consumidor final.

2.2. Caracterización del problema de planificación modelado

Como se comentó anteriormente, el problema estudiado y modelado, se inspiró un problema real de planificación, que enfrenta una empresa que se desempeña como operador especializado en un servicio independiente (*Second Part Logistics*, 2PL), de distribución de mercadería a carga fraccionada, con volúmenes reducidos y plazos de entrega diarios.

La empresa realiza servicios de distribución de forma coordinada, a varios clientes mayoristas con presencia en los cuatro principales mercados regionales ($L = 4$) que abastecen el área geográfica del Gran Buenos Aires (GBA).

Tabla 5: Coordenadas Depósitos

l	V_l	Lat.: X_i	Long.: Y_i
1	Mercado Central de Buenos Aires	34°42'36.37"S	58°30'4.74"O
2	Mercado de Abasto 3 de Febrero	34°37'14.62"S	58°33'24.54"O
3	Mercado Consorcio de Beccar	34°27'44.95"S	58°32'49.81"O
4	Mercado de abasto de Quilmes	34°43'55.73"S	58°16'47.88"O

Fuente: elaboración propia.

Para la validación y experimentación computacional, se realiza sobre información de pedidos realizados en un día de operatoria habitual de la empresa. Se considera la información correspondiente a ($R = 200$) clientes minoristas ubicados en Ciudad de Buenos Aires y del Gran Buenos Aires (*Ver Tabla 10 del Anexo*).

Luego, el conjunto de nodos $V = V_L \cup V_R$, que incluye los L depósitos y los R clientes minoristas, posee una cardinalidad de 204 nodos.

En entornos urbanos como el considerado, hay zonas o trayectos que tienen una velocidad promedio durante el día más bajas que otros, lo que implica un tiempo por kilómetro mayor. Esta dependencia temporal de la programación de la distribución, se considera en los tiempos, t_{ij} , para ir del nodo i al nodo j en una zona determinada, esto sería el peso de cada $arco(i, j)$. Estos tiempos fueron estimados a partir de la base de datos del *GIS (Geographic Information System)*, utilizado por la empresa para el seguimiento satelital de los vehículos.

La flota de vehículos, S , que es gestionada por la empresa, corresponde en parte a vehículos propios y otros contratados (como agente de cargas). Los vehículos no poseen características homogéneas en cuanto a las condiciones de frío ni en cuanto a su relación peso potencia. La capacidad de los vehículos, está restringida por la normativa para poder circular dentro de zonas urbanas. Por lo que a los fines de la modelización se consideró una flota homogénea, de camiones con 6.000 kg de carga útil ($Q = 300$ bultos). En este tipo de servicio, la unidad de carga habitual, son empaques homogéneos en forma de cajas de cartón corrugado apilables de dos piezas (telescopicas) o cajones de madera, con un peso de 20kg, largo 50 cm, ancho 32 cm y alto 30 cm.

Cada cliente i , tiene una demanda b_i determinada. La demanda individual, depende mucho de las características del cliente y de la época del año. En base a la información obtenida de la empresa, se utilizaron en la simulación computacional demandas individuales entre 10 y 30 unidades diarias. Pero este es un dato que el agente decisor, posee con certeza al momento de la programación de la distribución. Además, cada cliente estipula previamente un horario para la recepción de la mercadería. Estas ventanas temporales, generalmente responden a la disponibilidad de personal o a restricciones de circulación o aparcamiento del vehículo en horarios determinados. Por lo que no se permite en la programación que el vehículo llegue después del horario establecido por el cliente. Sí puede llegar antes, pero se considera en los tiempos de la ruta programada, el tiempo de espera hasta que se pueda realizar el servicio.

Los tiempos de servicio al cliente i , ts_i , deben considerar una serie de actividades que se llevan a cabo desde el momento en que el vehículo llega a las instalaciones

del cliente, o a los lugares habilitados para carga y descarga en la vía pública, hasta que parte nuevamente hacia su próximo destino. Por ejemplo, además de las actividades de maniobra, cuya duración dependerán de las condiciones de acceso con las que cuente el cliente. Hay una serie de actividades de chequeo de la documentación, de descarga, de inspección, etc., con tiempos que varían en función del personal y del equipamiento, que posee cada cliente. En la simulación computacional, se trabajó en base a estimaciones de estos tiempos suministradas por la empresa.

No se consideraron relaciones de preferencia o asignaciones predeterminadas de clientes, a algún depósito en particular en el modelado, dado que esas políticas implican considerar grupos de nodos ya asignados sin necesidad de agrupar de forma óptima, por lo que resultan en una simplificación del problema y se busca abordar el caso más general.

El agente decisor, en este caso, el responsable encargado de determinar el programa de distribución, conoce las regiones de interés en las que la herramienta computacional debe buscar. Luego, el problema consiste en encontrar los programas óptimos de distribución, dentro de la región de interés del agente. Esto es determinar las secuencias óptimas de clientes a visitar por cada vehículo, para realizar la distribución diaria de mercaderías a doscientos clientes minoristas, a partir de cuatro mercados regionales donde posee sus depósitos. Buscando minimizar simultáneamente, el tiempo total necesario para cumplir con la distribución, y a la vez balancear la carga de trabajo entre los vehículos utilizados para llevarla a cabo. Cumpliendo con las restricciones de capacidad de cada vehículo, con las ventanas temporales de los clientes en las instalaciones del cliente, con la restricción de que la duración total de la ruta no debe exceder al máximo tiempo permitido para la misma y que el número de vehículos utilizados no debe exceder al número de vehículos disponibles.

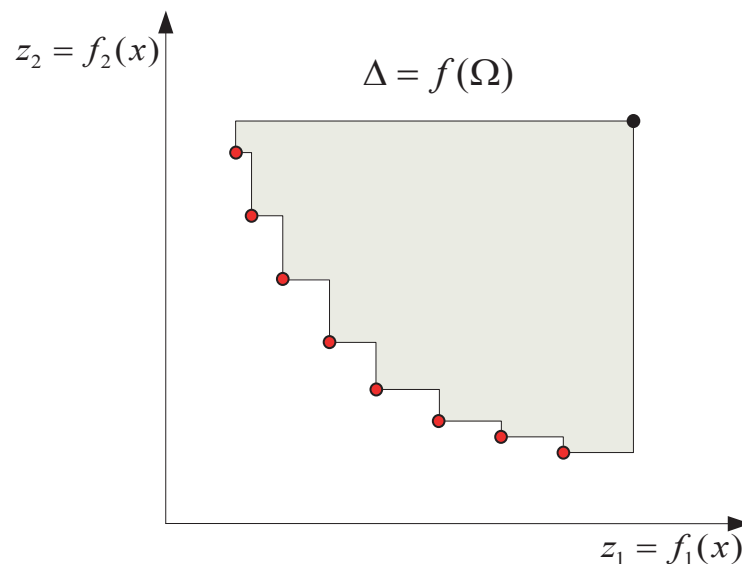
De esta manera el decisor o responsable de determinar el programa óptimo de distribución, primero guía a la herramienta computacional para que busque dentro de cierta zona de interés. Y luego selecciona la opción deseada, entre las alternativas óptimas encontradas en la región de interés.

2.3. Experiencia y evaluación de resultados

Para evaluar los resultados de problemas de optimización multi-objetivo, obtenidos con métodos estocásticos, como es el caso de los algoritmos evolutivos utilizados en el presente trabajo, debe considerarse varias ejecuciones del método e incluso diferentes métodos, dado que no siempre se obtienen los mismos resultados y, por lo tanto, la comparación de la eficacia de estos métodos requiere poder comparar distintos resultados (frentes de Pareto).

En la literatura se han propuesto diferentes indicadores, que buscan comparar la calidad de los frentes de Pareto obtenidos teniendo en cuenta distintos aspectos, como el acercamiento al frente real, la amplitud del frente y la uniformidad de la distribución de las soluciones. Knowles y otros (2006) y Fonseca y otros (2005), afirman que básicamente existen dos enfoques en la valoración de las eficacias de este tipo de métodos. El primer enfoque es estadístico, y el segundo enfoque utiliza indicadores de medidas cuantitativas de la eficacia.

Figura 29: Hipervolumen H del frente de Pareto A



Fuente: Elaboración propia en base a Knowles y otros (2006)

En este trabajo se ha seguido el enfoque cuantitativo para la valoración de la eficacia del optimizador estocástico multi-objetivo propuesto, ya que como ha sido probado en la literatura, estos procedimientos cuantitativos son útiles para decidir el

rendimiento relativo de diferentes optimizadores multi-objetivo en un problema dado (Knowles, Thiele, & Zitzler, 2006) (Fonseca, Knowles, Thiele, & Zitzler, 2005), especialmente cuando se desconoce el frente de Pareto real de dicho problema.

Específicamente, en este trabajo, se ha utilizado el *Indicador del Hipervolumen* (o *métrica S*) presentado por Zitzler y Thiele (1999), dado que este indicador no solo considera dentro de la medida cuantitativa de la eficacia, la convergencia hacia el frente de Pareto real, sino que también considera la distribución de las soluciones sobre el frente (Roudenko, 2004). En otras palabras, está diseñado para medir convergencia y diversidad en un frente dado (Van Veldhuizen, 1999).

Como se muestra en la Figura 29, este indicador mide el hipervolumen, H (en el espacio de los objetivos), de la región dominada por el frente de Pareto, FP_A , y limitada por algún punto de referencia Q . Luego, al comparar dos frentes diferentes, aquel con el mayor hipervolumen será el preferido.

Matemáticamente, para cada solución $i \in FP_A$ se construye un hipercubo v_i con el punto límite Q y la solución i como la diagonal de los extremos del mismo.

El punto Q se puede obtener simplemente con los peores valores de las funciones objetivo. Es decir, el punto Q debe ser dominado por todas las soluciones incluidas en el frente de Pareto, y en el caso de que se estén comparando diferentes frentes, el punto Q debe ser dominado por todas las soluciones incluidas en los frentes que se estén comparando.

Luego, la unión del hipervolumen de estos hipercubos, es el *Indicador de Hipervolumen*, H :

$$H = \text{volumen} \left(\bigcup_{i=1}^{|FP_A|} v_i \right)$$

El indicador H tiene una propiedad deseable: siempre que el frente A sea una mejor aproximación del frente real que el frente B , resultará que $H(A) > H(B)$; con la

condición de que el punto límite Q esté estrictamente dominado por todos los puntos en A y B .

Por lo tanto, si se da que $H(A) > H(B)$ uno puede inferir que B no puede ser mejor que A . “*El indicador de hipervolumen es el único indicador unario que es capaz de detectar que B no es mejor que A* ” (Knowles, Thiele, & Zitzler, 2006).

Es importante mencionar que se ha demostrado que el cálculo del indicador de hipervolumen pertenece a un orden exponencial en el número de objetivos y polinomial en el número de puntos en el conjunto de aproximación.

2.3.1. Resultados del Análisis Gráfico

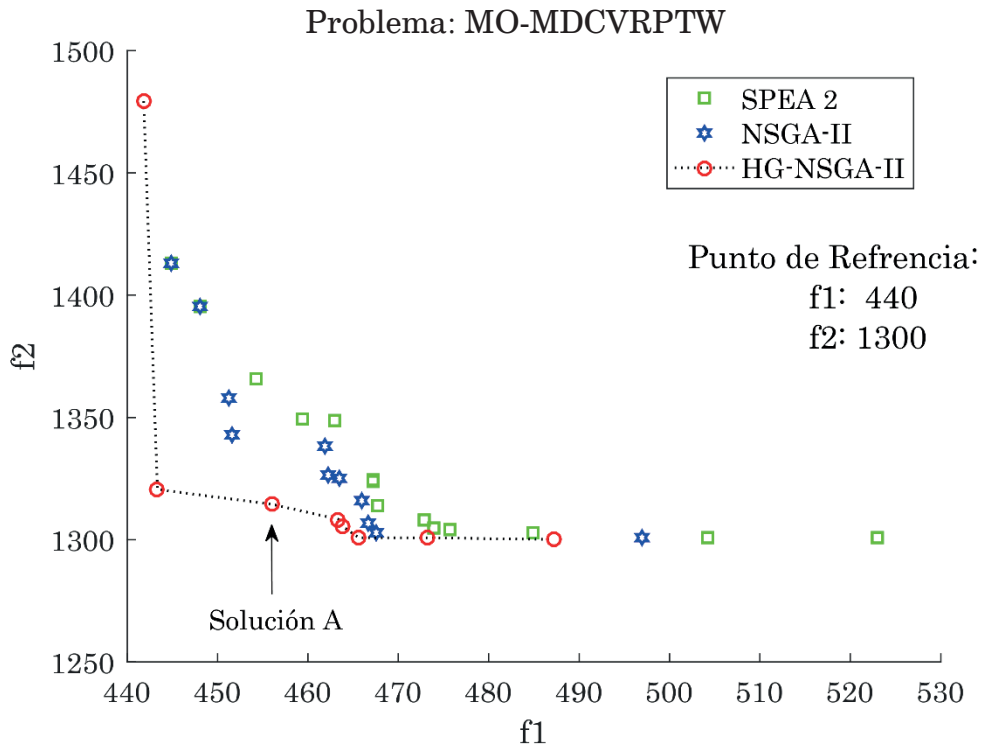
A continuación, se presentan gráficamente los resultados de los frentes de Pareto alcanzados por los tres algoritmos. Se trabajó con la estrategia de introducción de preferencias parciales de una región de interés representativa a los fines del testeo de los algoritmos.

Se realizó el análisis de la búsqueda en los extremos y en el centro de la zona de referencia. Estos fueron los puntos (440, 1300), (450, 1150) y (500, 1000). (el objetivo 1, está expresado en horas y el objetivo 2 en minutos).

En las figuras siguientes se presentan los resultados obtenidos mediante los tres algoritmos, HG-NSGA-II, NSGA-II y SPEA2, utilizando la estrategia de g-dominancia para la introducción de las preferencias parciales, implementada como penalización de las funciones objetivo.

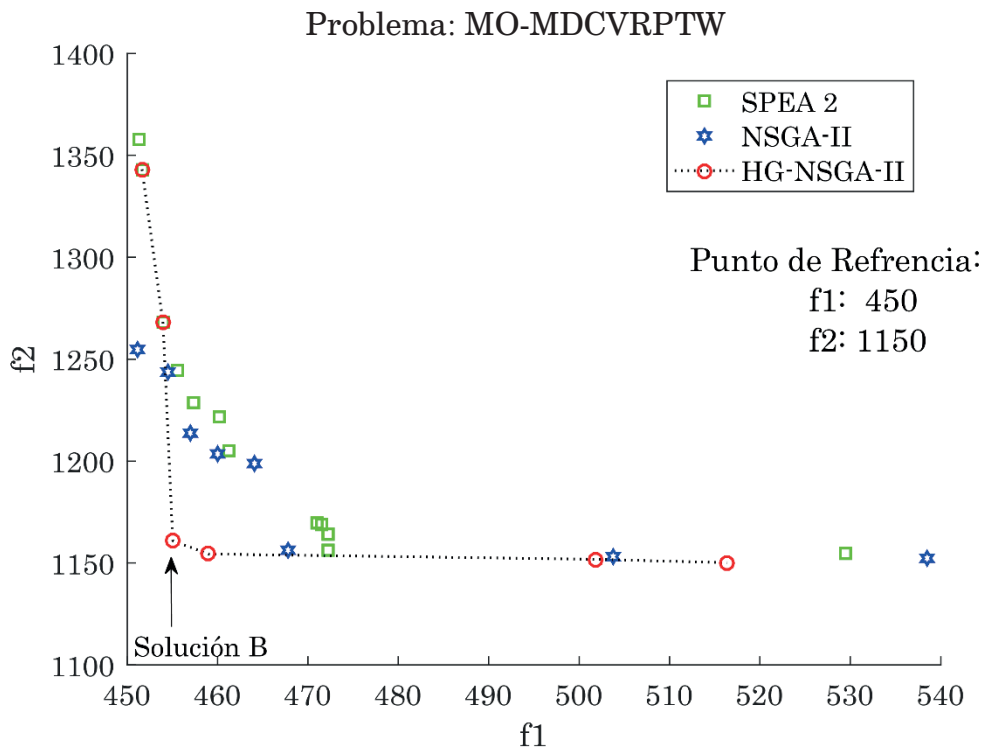
Para los tres algoritmos, el tamaño de la población fue fijado en 500, con probabilidades de cruce y mutación de 0.8 y 0.01 respectivamente. Y el número máximo de generaciones en 1000.

Figura 30: Punto de referencia superior izquierdo, G1



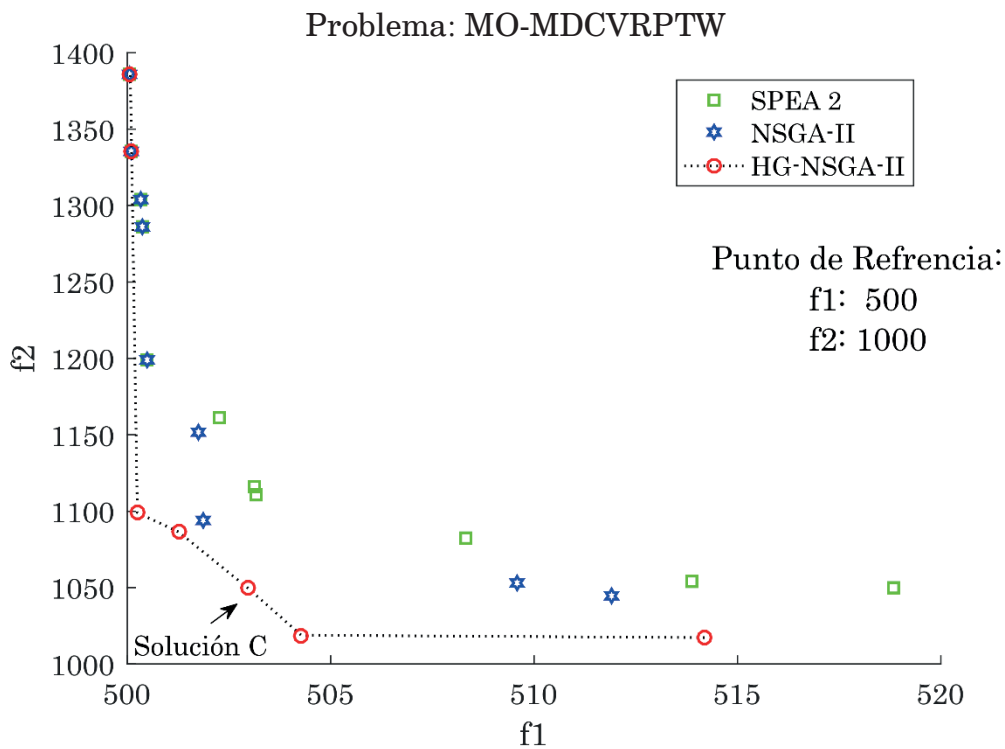
Fuente: Elaboración Propia.

Figura 31: Punto de referencia central, G2



Fuente: Elaboración Propia.

Figura 32: Punto de referencia inferior derecho, G3

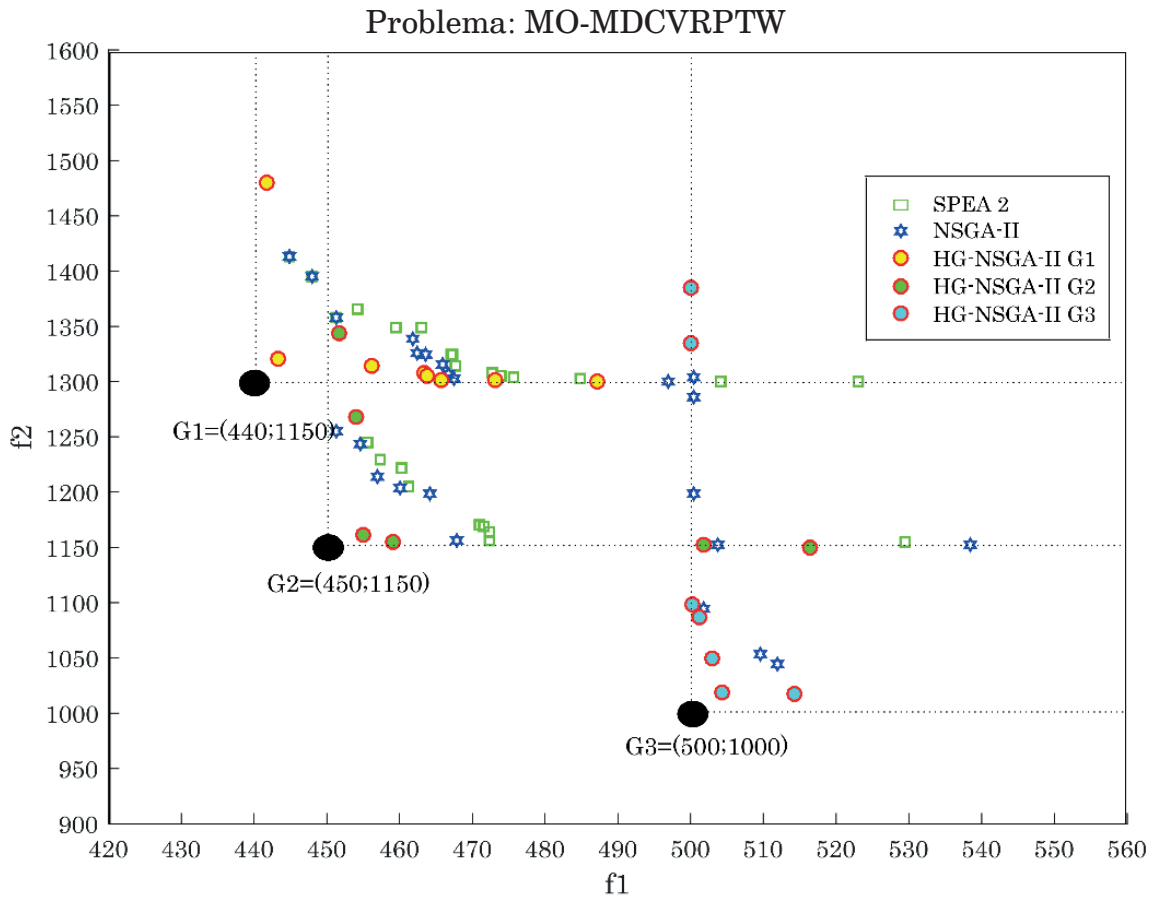


Fuente: Elaboración Propia.

En la Figura 33, se presentan los resultados obtenidos por los tres algoritmos, en una sola figura para los tres puntos de referencia. En la que se puede apreciar claramente el direccionamiento de la búsqueda de los algoritmos hacia los puntos de referencia indicados.

Dada la complejidad del problema, se desconoce el frente real de Pareto, por lo que no es posible comparar los resultados obtenidos, con dicho frente. Pero del análisis gráfico de los resultados obtenidos a una corrida, para cada punto de referencia, de cada algoritmo, puede verse que los resultados obtenidos con el algoritmo propuesto, son comparables con los de los conocidos NSGA-II y SPEA2.

Figura 33: Frentes aproximados para tres puntos de referencia



Fuente: elaboración propia.

En las primeras tres figuras, dado el efecto del zoom realizado sobre la zona de cada uno de los puntos de referencia, puede verse de forma gráfica que en estas soluciones, el HG-NSGA-II propuesto parece obtener un buen posicionamiento del sub-frente de Pareto del problema, al compararlo con los obtenidos con los otros dos algoritmos. A continuación, se realiza el análisis cuantitativo en el que se busca generalizar este análisis y corroborar esta afirmación.

2.3.2. Resultados del Análisis cuantitativo

Como se describió anteriormente, el análisis cuantitativo se realizó usando la medida del hipervolumen del frente de Pareto conseguido por cada algoritmo al introducirse los puntos de referencia dados por el decisor.

Para el cálculo de los valores de H (métrica S), el punto Q considerado para el cálculo fue $f_1 = 1.000$, $f_2 = 2.000$, el cual es dominado por todas las soluciones generadas duran-

te la evolución de los algoritmos. A continuación, se presenta las tablas con las medidas resumen del cálculo del hipervolumen para 30 corridas independientes de los tres algoritmos y para cada uno de los tres puntos de referencia considerados.

Tabla 6: Punto de Referencia G1: ($f_1 = 440, f_2 = 1300$)

MOEA	\bar{H}	H_σ	H_{mejor}
HG-NSGA-II	3,26914	0,80691	3,33452
NSGA-II	3,23066	0,69513	3,26297
SPEA2	3,20125	0,96238	3,28326

Fuente: elaboración propia.

Tabla 7: Punto de Referencia G2: ($f_1 = 450, f_2 = 1150$)

MOEA	\bar{H}	H_σ	H_{mejor}
HG-NSGA-II	3,51303	0,84130	3,65355
NSGA-II	3,44164	0,72433	3,47606
SPEA2	3,40040	1,05312	3,53642

Fuente: elaboración propia.

Tabla 8: Punto de Referencia G3: ($f_1 = 500, f_2 = 1000$)

MOEA	\bar{H}	H_σ	H_{mejor}
HG-NSGA-II	3,26182	0,64618	3,29444
NSGA-II	3,13516	0,65403	3,26057
SPEA2	3,11142	0,98243	3,27365

Fuente: elaboración propia.

De la información resumida en las tablas de resultados, puede verse que el algoritmo propuesto presenta un mayor hipervolumen medio al de los otros algoritmos para las 30 corridas independientes de cada uno de los tres puntos de referencia considerados, por lo que consigue mayor convergencia en términos comparativos. Es decir, podemos afirmar que el HG-NSGA-II produce las mejores aproximaciones de los sub-frentes de cada punto de referencia, en base a este indicador. Mientras que los niveles de variabilidad, captados por el desvío estándar del hipervolumen, son similares a los de los otros algoritmos y en algunos casos superiores.

También es importante destacar que las mejoras más grandes se dan en los puntos de referencia que exigen mayor equilibrio en los programas de trabajo. Es decir,

puntos de referencia que den mayor preferencia al equilibrio en las cargas de trabajo que al tiempo total acumulado en los programas de distribución. De las tablas 6, 7 y 8 puede verse que, la diferencia entre el hipervolumen medio obtenido con HG-NSGA-II y los obtenidos con NSGA-II y SPEA2, es mayor (mejora mayor) para los puntos de referencia más desplazados hacia el extremo derecho del frente, como el caso del punto G3.

2.3.3. Conclusiones de la experimentación computacional

El algoritmo presenta un buen desempeño. Del análisis gráfico puede verse que encuentra muy buenas soluciones al compararse los frentes encontrados por el HG-NSGA-II propuesto con los frentes logrados por los otros algoritmos, aunque en términos de hipervolumen medio estas diferencias no parezcan ser tan significativas. Es decir, el algoritmo propuesto es superior en términos de la medida del hipervolumen del frente encontrado, para los distintos puntos de referencia seleccionados, pero la diferencia no parece ser tan grande como se presenta en el análisis gráfico a una corrida.

Esto responde a que la densidad de soluciones sobre los frentes encontrados por el algoritmo propuesto es igual o un poco más baja que la de los frentes encontrados por los otros algoritmos. Esto es lo que explica que las diferencias en términos de hipervolumen no sean tan grandes en las comparaciones de los promedios.

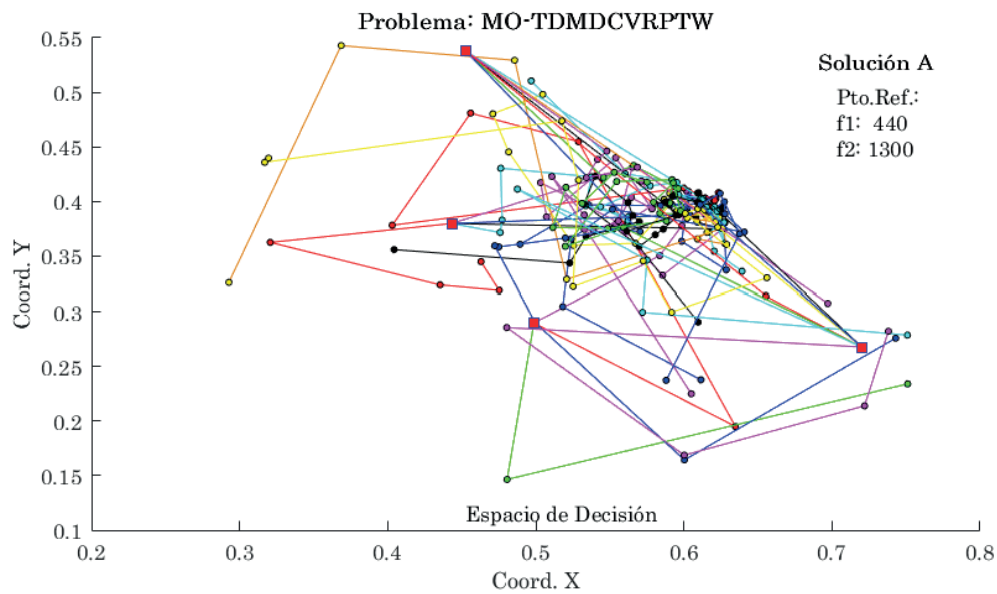
A continuación, se presentan las figuras con tres soluciones seleccionadas por un agente decisor, marcadas en el análisis gráfico del punto 2.3.1 para cada punto de referencia en el espacio de decisión. Se toma de la Figura 30 la Solución A ($f_1 = 456,1$; $f_2 = 1314$), de la Figura 31 la Solución B ($f_1 = 455,1$; $f_2 = 1161$), y de la Figura 32 la Solución C ($f_1 = 503,1$; $f_2 = 1050$).

Para cada solución se presentan dos figuras. La primera (figuras 34, 36 y 38) corresponde a la representación habitual en este tipo de problemas, considerando las dos dimensiones del espacio de decisión. Se puede ver el conjunto de nodos dispersos en el área de la capital federal y el conurbano bonaerense transformado del sistema de coordenadas esféricas (coordenadas geográficas) al sistema de coordenadas cartesianas para $x \in [0,1]$ e $y \in [0,1]$ y los arcos (con un mismo color) presentes en cada ruta de la solución óptima seleccionada. Y una segunda figura (figuras

35, 37 y 39), consistente en agregar una dimensión temporal a la representación anterior. Es decir, se agrega en la representación el tiempo de viaje acumulado (en minutos) en cada ruta de la solución seleccionada.

En la Figura 34, se ve la Solución A seleccionada (ver Figura 30), al considerar el primer punto de referencia.

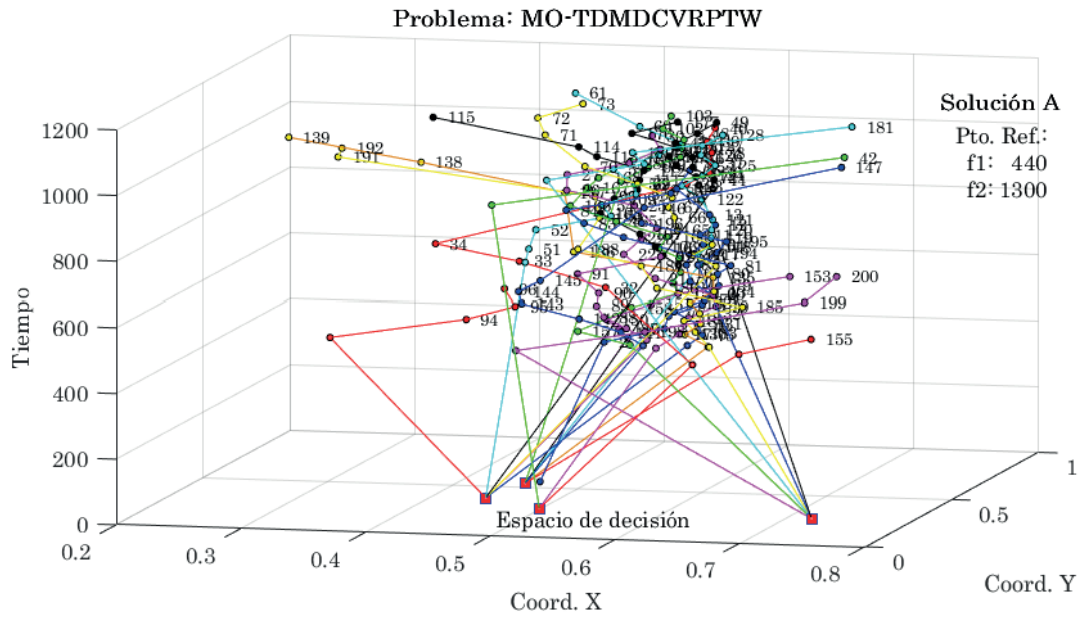
Figura 34: Solución A en representación 2D. Pto. Ref.: (440; 1300)



Fuente: elaboración propia.

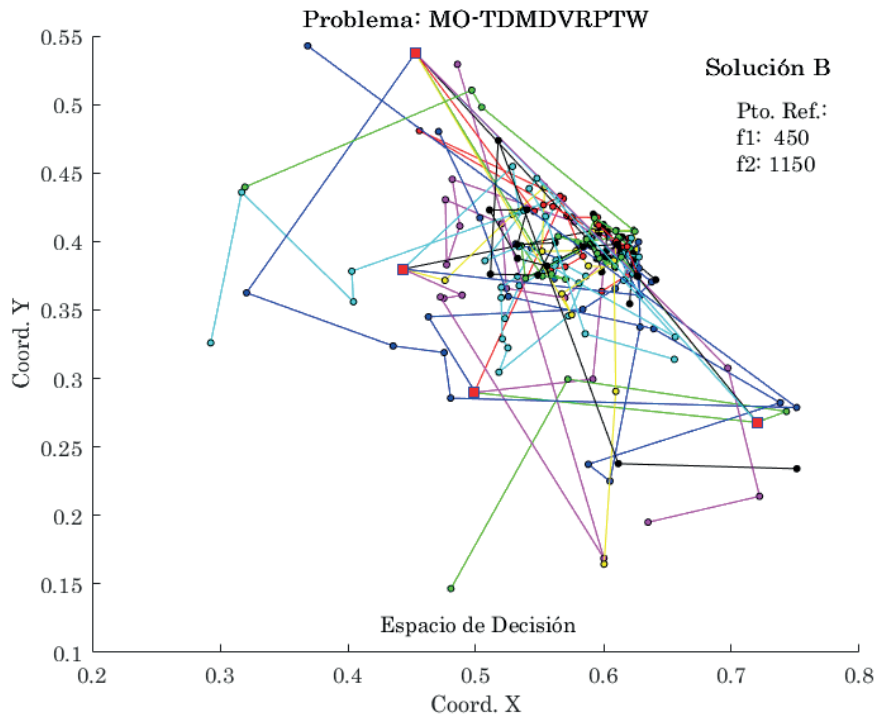
En la Figura 35, se representa el espacio de decisión incluyendo en la representación la dimensión temporal (en minutos), para la Solución A seleccionada.

Figura 35: Solución A en representación 3D. Pto. Ref.: (440; 1300)



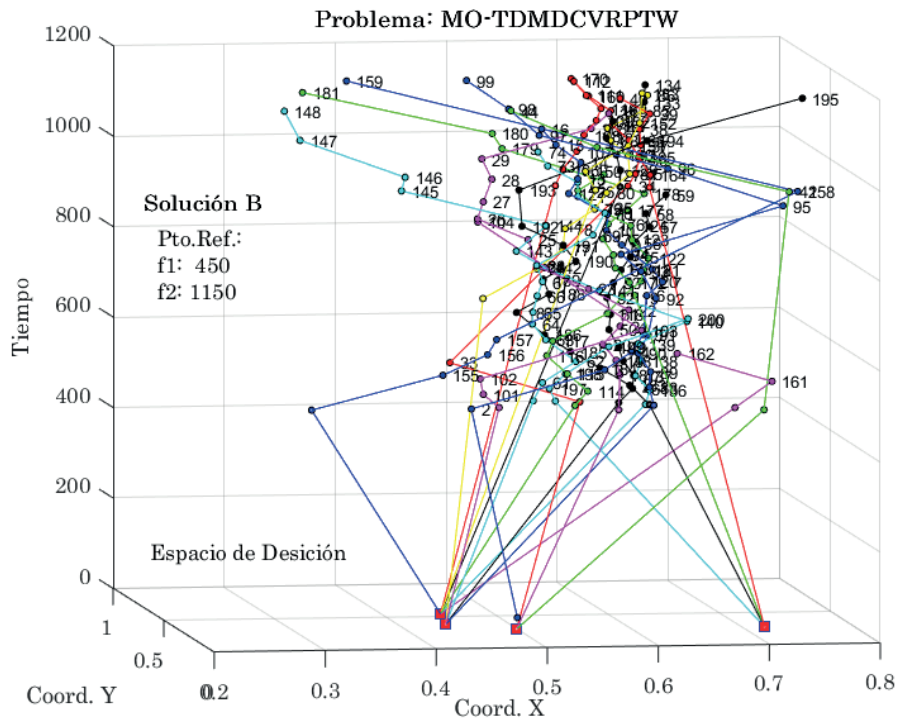
Fuente: elaboración propia.

Figura 36: Solución B en representación 2D. Pto. Ref.: (450; 1150)



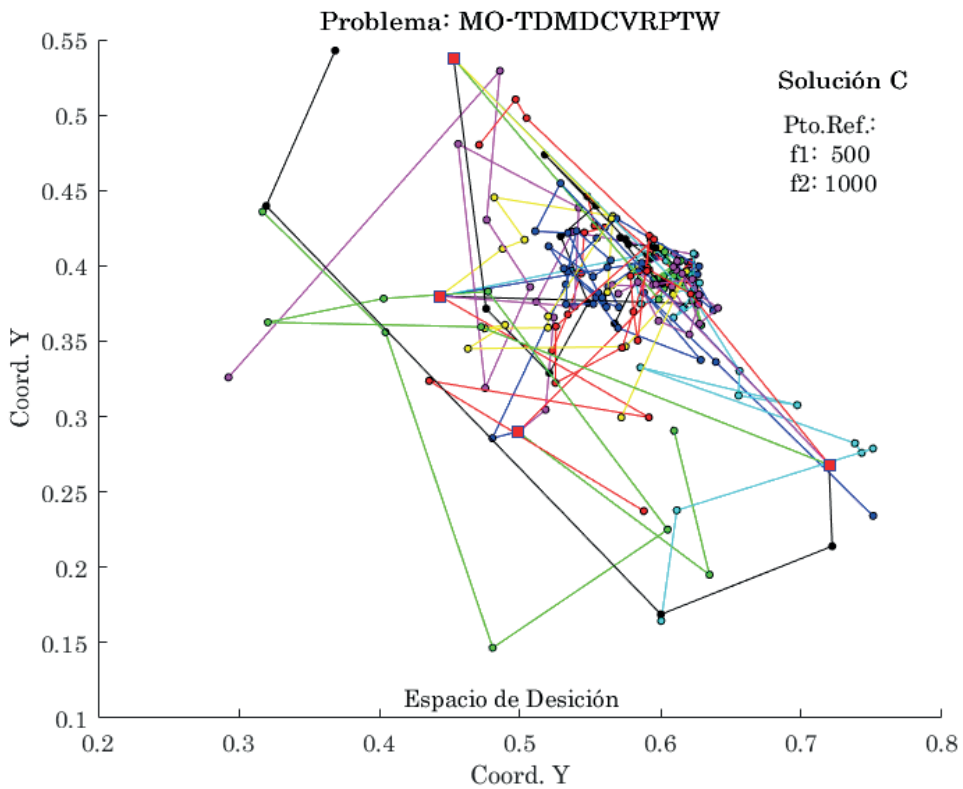
Fuente: elaboración propia.

Figura 37: Solución B en representación 3D. Pto. Ref.: (450; 1150)



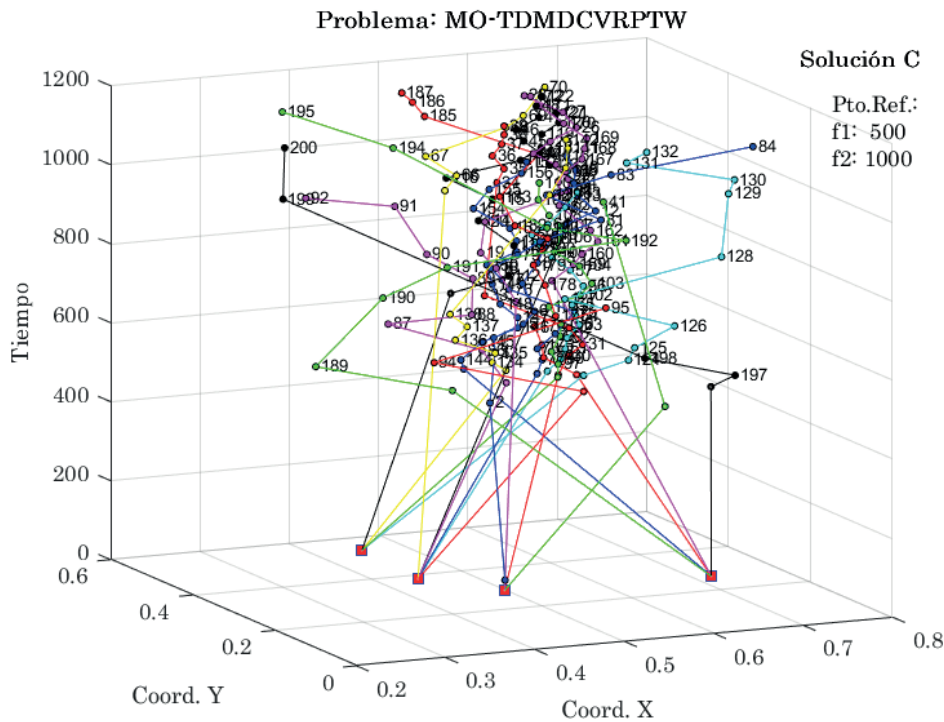
Fuente: elaboración propia.

Figura 38: Solución C en representación 2D. Pto. Ref.: (500; 1000)



Fuente: elaboración propia.

Figura 39: Solución C en representación 3D. Pto. Ref.: (500; 1000)



Fuente: elaboración propia.

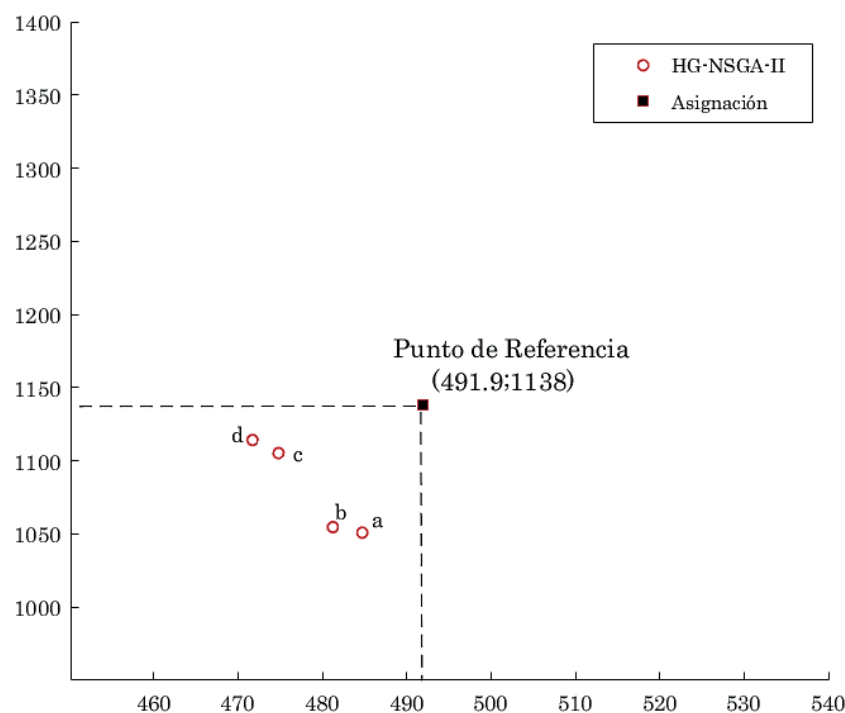
Del análisis gráfico de las soluciones en el espacio de decisión (representación con dos dimensiones), puede verse que la consideración de tiempos de servicio y ventanas temporales heterogéneos, se reflejan en rutas óptimas que no siguen patrones del tipo de gotas de agua en torno a los depósitos (cuadrados de color rojo), como en el caso de las soluciones de problemas de ruteo de vehículos que no consideran estos aspectos temporales. Por otro lado, al observar las soluciones en el espacio de decisión incluyendo en la representación la dimensión temporal, puede verse que las rutas siguen en su secuencia óptima, un patrón creciente en base a los tiempos requeridos para recorrer cada arco y al cumplimiento de las ventanas temporales de cada nodo. Ubicando al inicio de la ruta los nodos con ventanas con una apertura más temprana y al final, a los que presentan una apertura más tardía.

Finalmente, se realizó la comparación con información de programas de distribución adoptados por el operador bajo condiciones similares a las que se han considerado en el modelo.

En cada ensayo se tomó una asignación adoptada por el operador como punto de referencia y se aplicó el HG-NSGA-II, para determinar el desempeño obtenido, a una sola corrida, por el algoritmo propuesto.

En la Figura 40, se presenta una asignación tomada como punto de referencia y las soluciones encontradas sobre el frente por HG-NSGA-II.

Figura 40: Punto de referencia interior



Fuente: elaboración propia

En la tabla 9 se resumen los resultados de este ensayo, con el punto de referencia ($f1 = 491,86$; $f2 = 1138,12$).

Tabla 9: Mejoras comparación con solución adoptada.

Sol.	f1	f2	Mejoras en los objetivos		Mejoras en N° de Vehículos
			$-\Delta f1$	$-\Delta f2$	
a	484,74	1050,90	1%	8%	-3
b	481,26	1054,67	2%	7%	0
c	474,81	1105,23	3%	3%	-3
d	471,73	1114,20	4%	2%	-3
Pto. Ref.	491,86	1138,12			

Fuente: elaboración propia.

En la tabla 9, pueden verse las cuatro soluciones alternativas sobre el frente, que mejoran la adoptada por el operador.

En promedio, se obtienen mejoras en el orden del 5% al 10% respecto de los dos objetivos. Además, se obtuvieron reducciones importantes en el número de vehículos requeridos para realizar la distribución. Por lo que podemos afirmar que las soluciones obtenidas mediante el HG-NSGA II, mejoran substancialmente las adoptadas por la empresa.

CAPÍTULO V. CONCLUSIONES Y TRABAJO FUTURO

El objetivo general de este libro ha sido el desarrollo de una herramienta de ayuda a la toma de decisiones, que mejore la eficiencia de la planificación operativa de vehículos y la programación de cargas, en la distribución urbana de mercaderías, desde un enfoque multi-objetivo.

Se logró comprender y caracterizar la problemática enfrentada por un agente decisor, responsable de la programación de carga y ruteo de vehículos de un operador logístico, que desarrolla sus actividades en una red de distribución urbana de mercaderías, perteneciente a una cadena de suministros de alimentos perecederos, con múltiples centros regionales desde los que se abastecen una cantidad de clientes minoristas. Se abordó su análisis, desde el enfoque de la tecnología y dirección de operaciones, en el marco de la gestión de la cadena de suministros.

Se modeló formalmente, una variante de un problema de programación de cargas y ruteo de vehículos desde un enfoque multi-objetivo, que considera explícitamente la dependencia temporal en la optimización. Este es un aspecto relevante en contextos de distribución urbana de mercaderías, dado que cambia substancialmente los programas óptimos de distribución, en comparación con el abordaje habitual desde el que se considera como principal factor determinante de los costos la distancia recorrida. También se considera el objetivo de balanceo de carga de trabajo entre los transportistas. Esto busca evitar conflictos en las asignaciones de carga entre los distintos agentes de transporte, dado que en los modelos que solo consideran como determinante del objetivo económico la distancia recorrida, generan programas desequilibrados, que son muy eficientes, en términos de distancia recorrida, pero que implican factores de utilización de la capacidad del 90% para algunos operadores mientras que otros trabajan al 10% o 20%. En el modelado se consideran distintos aspectos de la infraestructura de la red y del contexto urbano de distribución, como el hecho de que la red está compuesta por múltiples centros de distribución que operan de forma coordinada, la existencia de restricciones en los horarios para poder realizar la descarga en las instalaciones de los clientes (ya sea por restricciones a la circulación en distintas zonas urbanas, por disponibilidad del personal para la realización del servicio, etc.), las restricciones de capacidad de

los vehículos en un contexto urbano, la limitación a la duración del programa de distribución de cada vehículo, etc.

El problema modelado fue identificado como el MO-TDMDCVRPTW (*Multi-objective Time Dependent Multi-Depot Capacitated Vehicle Routing Problem with Time Windows*). Y se adapta a las características de la problemática que enfrenta el agente responsable de la programación de la distribución del operador logístico considerado, desde un abordaje multi-objetivo, integrando tres problemas NP-duros conocidos en la literatura. El MD-CVRP: *Multi Depot Vehicle Routing Problem*, el CVRP-TW: *Vehicle Routing Problem with Time Windows*, y el TD-CVRP: *Time Dependent Vehicle Routing Problem*.

Dada la intratabilidad del problema MO-TDMDCVRPTW (de clase NP-duro), se abordó la búsqueda de soluciones desde un enfoque meta-heurístico, técnicas que representan el estado del arte en la resolución de este tipo de problemas. Específicamente, se trabajó con una de las técnicas meta-heurísticas más utilizadas para resolver este tipo de problemas de optimización multi-objetivo en el contexto del transporte automotor de carga, los Algoritmos Evolutivos Multi Objetivo (MOEAs). Se desarrolló una hibridación del conocido algoritmo evolutivo multi-objetivo NSGA-II, y las mejoras propuestas en este algoritmo consistieron en: Primero, la incorporación directa de conocimiento específico del problema mediante el uso de una original estrategia de representación, en la que se utiliza una estructura cromosómica de cuatro genomas cuya codificación se basa en ordenaciones de números naturales, que resulta muy eficiente para este tipo de problemas basados en el orden. Segundo, la utilización de un novedoso procedimiento de recombinación, que permite el clustering y el ruteo de forma simultáneas, al que se lo llamó *ERX-MD*. Este procedimiento fue diseñado específicamente para este problema de optimización combinatoria con representación mediante cadenas de números naturales, y garantiza de una forma económica en términos de recursos computacionales que los individuos que forman la descendencia cumplan con todas las restricciones impuestas en este problema, sin necesidad de utilizar costosos procedimientos de reparación. Tercero, en la utilización de una estrategia conocida como *g-dominancia*. Estrategia que está inspirada en la idea de racionalidad limi-

tada de H. A. Simons, y consiste en la introducción de preferencias parciales por parte del agente decisor (en nuestro caso el responsable de la planificación de la distribución) que posteriormente son utilizadas para conducir el proceso de búsqueda del algoritmo hacia la región de interés en la que se encuentra la alternativa óptima que será finalmente seleccionada. Esta estrategia, por un lado, permite que la herramienta algorítmica potencie las capacidades computacionales del agente decisor, y por otro, que éste mejore el desempeño del algoritmo evolutivo, mediante la introducción de información de su experiencia relacional, guiando el proceso de búsqueda del algoritmo para que el proceso de optimización evolucione más rápidamente hacia la región de interés del agente.

Esta herramienta de ayuda a la toma de decisiones permite al agente decisor analizar múltiples soluciones dentro de una zona de interés, que contemplan los objetivos contrapuestos del MO-TDMDCVRPTW mejorando la eficiencia de la planificación operativa en este contexto.

Dadas las características originales del problema planteado, no se conocen resultados publicados comparables, para testear las mejoras propuestas en el algoritmo, ya sea en base a listados de problemas académicos o no. De manera que se realizó la validación del modelo y el testeo del algoritmo, sobre la información de una instancia real, comparando el desempeño de la herramienta desarrollada contra el desempeño obtenido de la aplicación de dos algoritmos evolutivos muy conocidos y utilizados en la resolución de problemas de ruteo de vehículos con múltiples objetivos, el NSGA-II de Deb, Agrawal, Pratap y Meyarivan sin las mejoras propuestas en el presente trabajo, y otro el SPEA2 de Zitzler, Laumanns y Thiele.

De los resultados del análisis gráfico de los experimentos en los que se realiza la comparación del desempeño de los tres algoritmos a una corrida, se observa que el algoritmo desarrollado presenta un buen desempeño, produciendo las mejores aproximaciones del frente de Pareto al compararse con los frentes logrados por los otros algoritmos para distintos puntos de referencia. A partir de los resultados obtenidos del análisis cuantitativo de los experimentos computacionales en los que se realiza la comparación del desempeño de los tres algoritmos a 30 corridas independientes para distintos puntos de referencia, la herramienta computacional

desarrollada obtiene la mayor eficacia en cuanto a convergencia hacia el frente de Pareto real y en cuanto a la distribución de las soluciones sobre el frente, utilizando como medida cuantitativa de la eficacia, el indicador del hipervolumen (o métrica S de Zitzler y Thiele, 1999). Es decir, que el HG-NSGA-II mejora significativamente los resultados obtenidos por otros algoritmos conocidos utilizados en la resolución de este tipo de problemas de optimización combinatoria.

Finalmente, en los ensayos utilizando programas de distribución del operador, como puntos de referencia y aplicando el HG-NSGA-II a una corrida, se obtuvieron mejoras en el orden del 5% al 10% respecto de los dos objetivos al comparar las asignaciones. Además, se obtuvieron reducciones importantes en el número de vehículos requeridos en los programas de distribución.

A partir de este trabajo, se presentan como líneas futuras de investigación, en primer lugar y como extensión natural a las investigaciones realizadas, la incorporación de mayor nivel de detalle en el modelo del sistema real, utilizando diferentes técnicas de simulación. También se contempla trabajar con problemas de decisión en el ámbito de la dirección de operaciones logísticas considerando una mayor cantidad de objetivos y desde una óptica que integre distintos niveles decisionales de las operaciones logísticas.

También se proyecta trabajar en el desarrollo de herramientas de ayuda a la toma de decisiones en tiempo real, contemplando el desarrollo de algoritmos híbridos que consideren otros paradigmas de la inteligencia computacional, como la lógica difusa, las redes neuronales, de manera de poder abordar la resolución de la problemática mencionada en el párrafo anterior.

ANEXO

Tabla 10: Resumen información de clientes

i	x	y	d_i	TW		s_i
				a_i	b_i	
1	0,55	0,42	24	16:00	20:00	15
2	0,63	0,37	21	16:00	20:00	15
3	0,6	0,41	22	8:00	12:00	15
4	0,57	0,42	20	16:00	20:00	15
5	0,56	0,38	23	8:00	12:00	15
6	0,61	0,39	22	12:00	16:00	15
7	0,6	0,38	25	8:00	12:00	15
8	0,53	0,45	26	8:00	12:00	15
9	0,48	0,37	25	12:00	16:00	15
10	0,61	0,4	28	16:00	20:00	15
11	0,6	0,39	27	12:00	16:00	15
12	0,62	0,4	21	16:00	20:00	15
13	0,61	0,4	23	12:00	16:00	15
14	0,63	0,39	27	12:00	16:00	15
15	0,62	0,41	23	16:00	20:00	15
16	0,6	0,4	25	12:00	16:00	15
17	0,61	0,41	22	16:00	20:00	15
18	0,62	0,38	25	8:00	12:00	15
19	0,6	0,39	21	8:00	12:00	15
20	0,62	0,38	23	12:00	16:00	15
21	0,6	0,39	25	16:00	20:00	15
22	0,49	0,41	23	16:00	20:00	15
23	0,6	0,41	29	8:00	12:00	15
24	0,62	0,38	30	8:00	12:00	15
25	0,63	0,37	25	8:00	12:00	15
26	0,59	0,39	27	16:00	20:00	15
27	0,6	0,41	25	16:00	20:00	15
28	0,62	0,37	23	8:00	12:00	15
29	0,54	0,4	29	16:00	20:00	15
30	0,59	0,4	29	8:00	12:00	15
31	0,58	0,4	30	8:00	12:00	15
32	0,64	0,37	26	12:00	16:00	15
33	0,58	0,4	24	16:00	20:00	15
34	0,61	0,37	23	8:00	12:00	15
35	0,58	0,4	28	12:00	16:00	15
36	0,63	0,36	21	12:00	16:00	15
37	0,6	0,36	22	16:00	20:00	15
38	0,63	0,36	24	8:00	12:00	15
101	0,63	0,38	24	8:00	12:00	15
102	0,62	0,4	27	12:00	16:00	15
103	0,62	0,4	28	8:00	12:00	15
104	0,6	0,39	30	8:00	12:00	15
105	0,62	0,39	27	16:00	20:00	15
106	0,63	0,39	26	16:00	20:00	15
107	0,62	0,39	23	8:00	12:00	15
108	0,61	0,4	29	12:00	16:00	15
109	0,62	0,4	25	16:00	20:00	15
110	0,32	0,36	24	8:00	12:00	15
111	0,61	0,4	21	8:00	12:00	15
112	0,62	0,39	22	8:00	12:00	15
113	0,63	0,4	22	8:00	12:00	15
114	0,61	0,4	28	12:00	16:00	15
115	0,61	0,4	25	12:00	16:00	15
116	0,61	0,39	22	8:00	12:00	15
117	0,61	0,4	30	16:00	20:00	15
118	0,62	0,39	26	12:00	16:00	15
119	0,56	0,38	27	8:00	12:00	15
120	0,63	0,38	27	12:00	16:00	15
121	0,58	0,42	23	16:00	20:00	15
122	0,59	0,4	25	8:00	12:00	15
123	0,59	0,41	30	8:00	12:00	15
124	0,63	0,37	25	16:00	20:00	15
125	0,59	0,41	24	16:00	20:00	15
126	0,55	0,42	26	8:00	12:00	15
127	0,6	0,41	28	16:00	20:00	15
128	0,74	0,28	29	16:00	20:00	15
129	0,64	0,34	28	16:00	20:00	15
130	0,64	0,37	20	12:00	16:00	15
131	0,59	0,33	29	8:00	12:00	15
132	0,62	0,41	26	12:00	16:00	15
133	0,51	0,38	30	12:00	16:00	15
134	0,59	0,38	26	12:00	16:00	15
135	0,62	0,4	30	16:00	20:00	15
136	0,62	0,39	28	12:00	16:00	15
137	0,59	0,4	30	8:00	12:00	15
138	0,59	0,42	25	8:00	12:00	15

39	0,58	0,41	27	16:00	20:00	15
40	0,58	0,37	26	12:00	16:00	15
41	0,57	0,43	28	8:00	12:00	15
42	0,57	0,38	20	16:00	20:00	15
43	0,56	0,38	28	16:00	20:00	15
44	0,57	0,39	22	16:00	20:00	15
45	0,56	0,4	27	12:00	16:00	15
46	0,57	0,37	20	8:00	12:00	15
47	0,56	0,4	27	12:00	16:00	15
48	0,56	0,39	21	12:00	16:00	15
49	0,56	0,38	21	16:00	20:00	15
50	0,56	0,37	23	16:00	20:00	15
51	0,57	0,36	20	12:00	16:00	15
52	0,58	0,35	29	8:00	12:00	15
53	0,55	0,38	27	12:00	16:00	15
54	0,57	0,43	24	16:00	20:00	15
55	0,55	0,37	24	8:00	12:00	15
56	0,55	0,43	20	16:00	20:00	15
57	0,54	0,37	23	8:00	12:00	15
58	0,54	0,4	21	8:00	12:00	15
59	0,55	0,44	20	12:00	16:00	15
60	0,53	0,4	29	12:00	16:00	15
61	0,54	0,42	30	12:00	16:00	15
62	0,53	0,4	21	12:00	16:00	15
63	0,53	0,4	30	16:00	20:00	15
64	0,53	0,37	30	16:00	20:00	15
65	0,55	0,45	21	8:00	12:00	15
66	0,52	0,37	23	8:00	12:00	15
67	0,52	0,41	30	12:00	16:00	15
68	0,52	0,37	25	12:00	16:00	15
69	0,52	0,36	26	8:00	12:00	15
70	0,53	0,36	22	16:00	20:00	15
71	0,6	0,41	24	16:00	20:00	15
72	0,51	0,42	23	8:00	12:00	15
73	0,51	0,39	29	12:00	16:00	15
74	0,5	0,42	25	16:00	20:00	15
75	0,59	0,41	24	12:00	16:00	15
76	0,63	0,39	21	8:00	12:00	15
77	0,52	0,33	28	12:00	16:00	15
78	0,53	0,32	24	12:00	16:00	15
79	0,58	0,35	29	16:00	20:00	15
80	0,49	0,36	26	8:00	12:00	15
81	0,59	0,37	30	12:00	16:00	15
82	0,6	0,41	23	12:00	16:00	15
83	0,48	0,36	28	8:00	12:00	15
139	0,66	0,33	22	8:00	12:00	15
140	0,57	0,43	29	8:00	12:00	15
141	0,46	0,48	30	8:00	12:00	15
142	0,62	0,39	28	16:00	20:00	15
143	0,61	0,4	25	8:00	12:00	15
144	0,6	0,39	24	12:00	16:00	15
145	0,61	0,38	20	12:00	16:00	15
146	0,6	0,39	30	12:00	16:00	15
147	0,59	0,4	21	12:00	16:00	15
148	0,59	0,4	25	16:00	20:00	15
149	0,59	0,3	25	8:00	12:00	15
150	0,62	0,37	27	16:00	20:00	15
151	0,59	0,4	26	12:00	16:00	15
152	0,62	0,35	27	16:00	20:00	15
153	0,59	0,42	27	12:00	16:00	15
154	0,58	0,39	30	12:00	16:00	15
155	0,57	0,36	28	16:00	20:00	15
156	0,56	0,43	21	16:00	20:00	15
157	0,55	0,38	30	12:00	16:00	15
158	0,54	0,42	26	12:00	16:00	15
159	0,53	0,42	26	8:00	12:00	15
160	0,54	0,44	26	8:00	12:00	15
161	0,53	0,42	23	12:00	16:00	15
162	0,52	0,34	30	16:00	20:00	15
163	0,48	0,38	24	12:00	16:00	15
164	0,48	0,29	28	8:00	12:00	15
165	0,63	0,34	25	12:00	16:00	15
166	0,48	0,32	27	8:00	12:00	15
167	0,52	0,47	29	12:00	16:00	15
168	0,66	0,31	27	8:00	12:00	15
169	0,57	0,3	30	16:00	20:00	15
170	0,61	0,29	29	16:00	20:00	15
171	0,7	0,31	20	8:00	12:00	15
172	0,44	0,32	30	8:00	12:00	15
173	0,46	0,34	28	8:00	12:00	15
174	0,48	0,43	22	12:00	16:00	15
175	0,52	0,3	23	12:00	16:00	15
176	0,48	0,45	22	16:00	20:00	15
177	0,74	0,28	30	12:00	16:00	15
178	0,5	0,5	20	16:00	20:00	15
179	0,47	0,48	26	16:00	20:00	15
180	0,5	0,51	27	16:00	20:00	15
181	0,72	0,27	23	8:00	12:00	15
182	0,61	0,24	20	16:00	20:00	15
183	0,75	0,28	25	16:00	20:00	15

Anexo

84	0,61	0,39	20	16:00	20:00	15	184	0,59	0,24	25	12:00	16:00	15
85	0,59	0,4	27	16:00	20:00	15	185	0,5	0,29	23	8:00	12:00	15
86	0,47	0,36	24	8:00	12:00	15	186	0,6	0,23	21	12:00	16:00	15
87	0,53	0,37	26	8:00	12:00	15	187	0,49	0,53	21	12:00	16:00	15
88	0,53	0,39	26	8:00	12:00	15	188	0,4	0,38	26	12:00	16:00	15
89	0,59	0,41	25	16:00	20:00	15	189	0,75	0,23	29	16:00	20:00	15
90	0,59	0,4	24	16:00	20:00	15	190	0,57	0,35	22	12:00	16:00	15
91	0,61	0,41	25	12:00	16:00	15	191	0,63	0,2	27	8:00	12:00	15
92	0,6	0,42	30	16:00	20:00	15	192	0,72	0,21	20	8:00	12:00	15
93	0,63	0,38	30	8:00	12:00	15	193	0,4	0,36	21	16:00	20:00	15
94	0,6	0,39	23	8:00	12:00	15	194	0,6	0,17	23	8:00	12:00	15
95	0,61	0,39	24	8:00	12:00	15	195	0,6	0,16	27	16:00	20:00	15
96	0,63	0,38	23	16:00	20:00	15	196	0,32	0,44	21	16:00	20:00	15
97	0,6	0,38	27	8:00	12:00	15	197	0,32	0,44	21	16:00	20:00	15
98	0,58	0,39	23	16:00	20:00	15	198	0,37	0,54	20	16:00	20:00	15
99	0,55	0,39	23	12:00	16:00	15	199	0,48	0,15	27	16:00	20:00	15
100	0,61	0,38	25	16:00	20:00	15	200	0,29	0,33	21	16:00	20:00	15

REFERENCIAS

- Adelzadeh, M., Asl, V. M., & Koosha, M. (2014). A mathematical model and a solving procedure for multi-depot vehicle routing problem with fuzzy time window and heterogeneous vehicle. *The International Journal of Advanced Manufacturing Technology*, 75(5), 793-802.
- Aguilar, J. (2007). La Programación Evolutiva en la Identificación de Sistemas Dinámicos a Eventos Discretos. *Revista IEEE Latinoamerica Transactions*, 5(5), 301-310.
- Alabas-Uslu, C. (2008). A self-tuning heuristic for a multi-objective vehicle routing problem. *Journal of the Operational Research Society*, 59, 988-996.
- Alves, M. J., & Clímaco, J. (2000). An interactive method for 0-1 multiobjective problems using simulated annealing and tabu search. *Journal of Heuristics*, 6(3), 385-403.
- Anbuudayasankar, S. P., Ganesh, K., Koh, S. L., & Ducq, Y. (2012). Modified savings heuristics and genetic algorithm for bi-objective vehicle routing problem with forced backhauls. *Expert Systems with Applications*, 39(3), 2296-2305.
- Antonioni, A., & Lu, W. S. (2007). *Practical optimization: algorithms and engineering applications*. Springer Science & Business Media.
- Antún, J. P. (2013). *Distribución urbana de mercancías: Estrategias con centros logísticos*. Inter-American Development Bank.
- Baaj, M. H., & Mahmassani, H. S. (1995). Hybrid route generation heuristic algorithm for the design of transit networks. *Transportation Research Part C: Emerging Technologies*, 3(1), 31-50.
- Báez Viezca, S. B. (2014). Problema de rutas con múltiples depósitos y ventanas de tiempo. *Doctoral dissertation, Universidad Autónoma de Nuevo León*.
- Ballou, R. H. (2004). *Logística. Administración de la cadena de suministro* (5ta ed.). México: Pearson Educación.

Referencias

- Baños, R. O. (2013). A Simulated Annealing-based parallel multi-objective approach to vehicle routing problems with time windows. . *Expert Systems With Applications*, 5(40), 1696-1707.
- Barchini, G. E. (2005). Métodos “I + D” de la Informática. *Revista de Informática Educativa y Medios Audiovisuales*, 2(5), 16-24.
- Basseur, M., Talbi, E. G., Nebro, A., & Alba, E. (2006). *Metaheuristics for multiobjective combinatorial optimization problems: Review and recent issues*. . National Institute of Research in Informatics and Control (INRIA).
- Bektas, T. (2017). *Freight Transport and Distribution: Concepts and Optimisation Models*. New York: CRC Press.
- Belenguer, J., Benavent, E., Labadi, N., Prins, C., & Reghioui, M. (2010). Split delivery capacitated arc routing problem: Lower bound and metaheuristic. *Transportation Science*, 44, 206–220.
- Beyer, H. G., & Schwefel, H. P. (2002). Evolution strategies - A comprehensive introduction. *Natural computing*, 1(1), 3-52.
- Bonome, M. G. (2010). *La Racionalidad en la Toma de Decisiones: Análisis la Teoría de la Decisión de Herbert A. Simón*. España: Netbiblo.
- Bowersox, D. J., Closs, D. J., & Cooper, M. B. (2007). *Administración y logística en la cadena de suministros* (2da ed.). México, D.F.: McGraww-Hill/Intermaericana.
- Branke, J. (2008). Consideration of partial user preferences in evolutionary multiobjective optimization. En *Multiobjective optimization* (págs. 157-178). Springer Berlin Heidelberg.
- Buckles, B. P., & Petry, F. E. (1992). *Genetic Algorithms*. IEEE Computer Society Press.
- Cabrero, J., & Camacho, J. P. (2002). Fundamentos de genética de poblaciones. En *Evolución: la base de la biología* (págs. 83-126). Granada: Proyecto Sur.
- Campbell, A. M. (2004). Efficient insertion heuristics for vehicle routing and scheduling problems. *Transportation science*, 38(3), 369-378.

Referencias

- Carpaneto, G., Dell'Amico, M., Fischetti, M., & Toth, P. (1989). A branch and bound algorithm for the multiple depot vehicle scheduling problem. *Networks*, 19(5), 531-548.
- Chang, T. S., & Wang, S. D. (2010). Multi-trip vehicle routing and scheduling problems. *The 40th International Conference on Computers and Industrial Engineering (CIE)*.
- Chao, I. (2002). A tabu search method for the truck and trailer routing problem. *Computers and Operations Research*, 29(1), 33 - 51.
- Chao, I. M. (2002). A tabu search method for the truck and trailer routing problem. *Computers & Operations Research*, 29(1), 33-51.
- Charnes, A., & Cooper, W. (1957). Programming, Management Models and Industrial Applications of Linear. *Management Science*, 4(1), 38-91.
- Charnes, A., Cooper, W. W., & Ferguson, R. O. (1955). Optimal estimation of executive compensation by linear programming. *Management science*, 1(2), 138-151.
- Chase, R. B., Jacobs, F. R., & Aquilano, N. J. (2010). *OPERATIONS AND SUPPLY MANAGEMENT*. McGraw-Hill/Irwin.
- Chopra, S., & Meindl, P. (2008). *Administración de la cadena de suministro. Estrategia, planeación y operación*. México: Pearson Educación.
- Coello Coello, C. A., & Toscano Pulido, G. (2001). A micro-genetic algorithm for multiobjective optimization. *Evolutionary multi-criterion optimization. Springer Berlin Heidelberg*, 126-140.
- Coello, C. A. (2002). Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer methods in applied mechanics and engineering*, 191(11), 1245-1287.
- Coello, C. A., & Zacatenco, C. S. (2015). *Introducción a la computación evolutiva*. México, DF.: CINVESTAV-IPN, Departamento de Ingeniería Eléctrica, Sección de Computación.

Referencias

- Coello, C. C. (2000). Handling preferences in evolutionary multiobjective optimization: A survey. En *Evolutionary Computation* (págs. 30-37). IEEE.
- Coello, C. C. (2006). Evolutionary multi-objective optimization: a historical view of the field. *IEEE computational intelligence magazine*, 1(1), 28-36.
- Collette, Y. &. (2011). *Optimisation multiobjectif: Algorithmes*. Editions Eyrolles.
- Collette, Y., & Siarry, P. (2011). *Optimisation multiobjectif: Algorithmes*. Editions Eyrolles.
- Comer, D. E., Gries, D., Mulder, M. C., Tucker, A., Turner, A. J., Young, P. R., & Denning, P. J. (1989). Computing as a discipline. *Communications of the ACM*, 32(1), 9-23.
- Contreras, H. (2012). *Teoria de la Computacion para Ingenieria de Sistemas: Un enfoque practico*.
- Cordeau, J., Laporte, G., & Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 52, 928-936.
- Corne, D. W., Knowles, J. D., & Oates, M. J. (2000). The Pareto envelope-based selection algorithm for multiobjective optimization. En *Parallel problem solving from nature PPSN VI* (págs. 839-848). Berlin Heidelberg: Springer.
- Corne, D. W., Knowles, J. D., & Oates, M. J. (2001). The Pareto envelope-based algorithm for multiobjective optimization. *Proceedings of the sixth international conference on parallel problem solving from nature*, (págs. 839-848). Lect Notes Comput Sci.
- Correa Espinal, A., & Gómez Montoya, R. A. (2009). Tecnologías de la información en la cadena de suministro. *Dyna*, 76(157), 37-48.
- Crainic, T. G., Ricciardi, N., & Storchi, G. (2004). Advanced freight transportation systems for congested urban areas. *Transportation Research Part C: Emerging Technologies*, 12(2), 119-137.
- Davis, L. (1985). Job Shop Scheduling with Genetic Algorithms. En J. J. Grefenstette, *Proceedings of the First International Conference on Genetic*

Referencias

- Algorithms* (págs. 136–140). Hillsdale, New Jersey: Lawrence Erlbaum Associates.
- Davis, L. (1991). *Handbook of genetic algorithms*. New York: Van Nostrand Reinhold.
- Deb, K. (1999). Solving goal programming problems using multi-objective genetic algorithms. En *Evolutionary Computation*. Proceedings of the 1999 Congress on IEEE.
- Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons.
- Deb, K., & Sundar, J. (2006). Reference point based multi-objective optimization using evolutionary algorithms. En *Proceedings of the 8th annual conference on Genetic and evolutionary computation* (págs. 635-642). ACM.
- Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. *Proceedings of the 6th international conference on parallel problem solving from nature* (págs. 849-858). Berlin Heidelberg: Springer.
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on evolutionary algorithms*, 6, 182-197.
- Denby, B., & Le Hégarat-Masclé, S. (2003). Swarm intelligence in optimisation problems. *ACAT2002*, 502, 364-368.
- Dorigo, M., Maniezzo, V., & Coloni, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics*, 26(1), 1-13.
- Eiben, A., & Smith, J. (2015). *Introduction to Evolutionary Computing*. Springer.
- El-Sherbeny, N. (2001). *Resolution of a vehicle routing problem with multi-objective simulated annealing method*. Mons, Belgique: Faculté Polytechnique de Mons.

Referencias

- Engelbrecht, A. P. (2007). *Computational intelligence: an introduction*. John Wiley & Sons.
- Erickson, M., Mayer, A., & Horn, J. (2001). The niched pareto genetic algorithm 2 applied to the design of groundwater remediation systems. *Evolutionary Multi-Criterion Optimization. Springer Berlin Heidelberg*, 681-695.
- Fogarty, D. W., Blackstone, J. H., & Hoffmann, T. R. (2007). *Administración de la producción e inventarios*. México: GRUPO EDITORIAL PATRIA.
- Fogel, D. B. (1988). An evolutionary approach to the traveling salesman problem. *Biological Cybernetics*, 60(2), 139-144.
- Fogel, D. B. (1995). *Evolutionary Computation. Toward a New Philosophy of Machine Intelligence*. New York: The Institute of Electrical and Electronic Engineers.
- Fogel, L. J., Owens, A. J., & Walsh, M. J. (1966). *Artificial intelligence through simulated evolution*. Chichester, UK: Wiley.
- Fonseca, C. M., & Fleming, P. J. (1993). Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization. *ICGA*, 93, 416-423.
- Fonseca, C. M., Knowles, J. D., Thiele, L., & Zitzler, E. (2005). A tutorial on the performance assessment of stochastic multiobjective optimizers. *Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, (pág. p. 240). Guanajuato, Mexico.
- Fonseca, M., & Fleming, P. J. (1993). Genetic Algorithms for Multi-Objective Optimization: Formulation, Discussion and Generalization. *Fifth International Conference on Genetic Algorithms*. San Mateo, USA.
- Frazelle, E., & Sojo, R. (2007). *Logística de Almacenamiento de Clase Mundial*. Grupo editorial Noma.
- Frutos, M., & Tohmé, F. (2012). A New Approach to the Optimization of the CVRP through Genetic Algorithms. *American Journal of Operations Research*, 2(4), 495.

Referencias

- Frutos, M., Tohmé, F., & Miguel, F. (2016). Chapter 8: A genetic algorithm's approach to the optimization of capacitated vehicle routing problems. En *Handbook of research on modern optimization algorithms and applications in engineering and economics*. (págs. 209-235). Hershey: Engineering Science Reference.
- Frutos, M., Tohmé, F., Delbianco, F., & Miguel, F. (2016). An alternative hybrid evolutionary technique focused on allocating machines and sequencing operations. *International Journal of Industrial Engineering Computations*, 7.
- García-Nájera, A., Bullinaria, J. A., & Gutiérrez-Andrade, M. A. (2015). An evolutionary approach for multi-objective vehicle routing problems with backhauls. *Computers & Industrial Engineering*, 81, 90-108.
- Garey, M. R., & Johnson, D. (1979). *Computers and intractability. A Guide to the theory of NPCompleteness*. New York: W.H.Freeman and Company.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. New York: Addison-Wesley Publishing Company.
- Goldberg, D. E., & Deb, K. (1991). A comparative analysis of selection schemes used in genetic algorithms. *Foundations of genetic algorithms*, 69-93.
- Gomes, L., & Von Zuben, F. J. (2002). A neuro-fuzzy approach to the capacitated vehicle routing problem. *International Joint Conference on Neural Networks*.
- Gong, W., & Fu, Z. (2010). ABC-ACO for perishable food vehicle routing problem with time windows. *International Conference on Computational and Information Sciences (ICCIS)*.
- Gulczynski, D., Golden, B., & Wasil, E. (2011). The multi-depot split delivery vehicle routing problem: An integer programming-based heuristic, new test problems, and computational results. *Computers & Industrial Engineering*, 61, 794 - 804.

Referencias

- Gulczynski, D., Golden, B., & Wasil, E. (2011). The period vehicle routing problem: New heuristics and real-world variants. *Transportation Research Part E*, 47, 648 - 668.
- Hernández Sampieri, R., Fernández Collado, C., & Baptista, L. (2014). *Metodología de la investigación*. México: Mc Graw Hill.
- Hoffman, A. (1989). *Arguments on Evolution: A Paleontologist's Perspective*. New York: Oxford University Press.
- Holland, J. H. (1962 a). Concerning efficient adaptive systems. En *Self-Organizing Systems* (págs. 215-230). Washington, D.C.: Spartan Books.
- Holland, J. H. (1962 b). Outline for a logical theory of adaptive systems. *Journal of the Association for Computing Machinery (JACM)*, 9(3), 297-314.
- Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press.
- Holland, J. H. (1992). *Adaptation in Natural and Artificial Systems*. Cambridge: MIT Press.
- Horn, J., Nafpliotis, N., & Goldberg, D. E. (1993). *Multiobjective optimization using the niched pareto genetic algorithm*. IlliGAL Report No. 93005.
- Hugos, M. (2003). *Essentials of Supply Chain Management*. New Jersey, USA: John Wiley & Sons, Inc.
- Hwang, C. L., Paidy, S. R., Yoon, K., & Masud, A. S. (1980). Mathematical programming with multiple objectives: A tutorial. *Computers & Operations Research*, 7(1), 5-31.
- Ignizio, J. P. (1976). *Goal programming and extensions*. Lexington Books.
- Ignizio, J. P. (1978). A review of goal programming: A tool for multiobjective analysis. *Journal of the Operational Research Society*, 29(11), 1109-1119.
- Irnich, S., Toth, P., & Vigo, D. (2014). The Family of Vehicle Routing Problems. En *Vehicle routing: problems, methods, and applications* (págs. 1-33). Philadelphia: Society for Industrial and Applied Mathematics and the Mathematical Optimization Society.

Referencias

- Jozefowicz, N., Semet, F., & Talbi, E. G. (2007). The bi-objective covering tour problem. *Computers & operations research*, 34(7), 1929-1942.
- Jozefowicz, N., Semet, F., & Talbi, E. G. (2009). An evolutionary algorithm for the vehicle routing problem with route balancing. *European Journal of Operational Research*, 195(3), 761-769.
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of the 1995 IEEE International Conference on Neural Networks*. Piscataway, NJ, United States.
- Kindervarter, G. A., & Savelsbergh, M. W. (1997). chapter Vehicle Routing: handling edge exchanges. En *Local Search in Combinatorial Optimization* (págs. 337 - 360). England: John Wiley & Sons.
- Knowles, J. D., Thiele, L., & Zitzler, E. (February de 2006). A tutorial on the performance assessment of stochastic multiobjective optimizers. *TIK-Report No. 214*, págs. 1-35.
- Kohl, N., Desrosiers, J., Madsen, O. B., Solomon, M. M., & Soumis, F. (1999). 2-path cuts for the vehicle routing problem with time windows. *Transportation Science*, 33(1), 101-116.
- Krajewski, L. J., Ritzman, L. P., & Malhotra, M. J. (2007). *Operations management: process and value chains* (8th ed.). México: Pearson Education.
- Kritikos, M. N., & Ioannou, G. (2010). The balanced cargo vehicle routing problem with time windows. *International Journal of Production Economics*, 123(1), 42-51.
- Kulkarni, R. V., & Bhave, P. R. (1985). Integer programming formulations of vehicle routing problems. *European Journal of Operational Research*, 20(1), 58-67.
- Laporte, G. (1992). The Vehicle Routing Problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59, 345 - 358.

Referencias

- Laporte, G., & Nobert, Y. (1987). Exact algorithms for the vehicle routing problem. *North-Holland Mathematics Studies*, 132, 147-184.
- Laporte, G., Nobert, Y., & Taillefer, S. (1988). Solving a family of multi-depot vehicle routing and location-routing problems. *Transportation science*, 22(3), 161-172.
- Lau, H. C., Chan, T. M., Tsui, W. T., & Pang, W. K. (2010). Application of genetic algorithms to solve the multidepot vehicle routing problem. *IEEE Transactions on Automation Science and Engineering*, 7(2), 383-392.
- Lau, H. C., Chan, T. M., Tsui, W. T., Chan, F. T., Ho, G. T., & Choy, K. L. (2009). A fuzzy guided multi-objective evolutionary algorithm model for solving transportation problem. *Expert Systems with Applications*, 36(4), 8255-8268.
- Lee, S. M. (1972). *Goal Programming for Decision Analysis*. Philadelphia: Auerbach Publishers.
- Lee, T. R., & Ueng, J. H. (1999). A study of vehicle routing problems with load-balancing. *International Journal of Physical Distribution & Logistics Management*, 29(10), 646-657.
- Lee, T. R., & Ueng, J. H. (1999). A study of vehicle routing problems with load-balancing. *International Journal of Physical Distribution & Logistics Management*, 29(10), 646-657.
- Lenstra, J., & Rinnooy KAN, A. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, 11, 221 - 228.
- Li, F., Golden, B., & Wasil, E. (2007). A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem. *Computers & Operations Research*, 34, 2734 - 2742.
- Li, F., Golden, B., & Wasil, E. (2007). The open vehicle routing problem: Algorithms, large-scale test problems, and computational results. *Computers & Operations Research*, 34, 2918 - 2930.

Referencias

- Lim, C. P., & Jain, L. C. (2009). Advances in swarm intelligence. En *Innovations in Swarm Intelligence* (págs. 1-7). Heidelberg: Springer Berlin.
- Lin, C. K., & Kwok, R. C. (2006). Multi-objective metaheuristics for a location-routing problem with multiple use of vehicles on real data and simulated data. *European Journal of Operational Research*, 175(3), 1833-1849.
- Malandraki, C., & Dial, R. (1994). A restricted dynamic programming heuristic algorithm for the time dependent traveling salesman problem. *European Journal of Operational Research*, 90, 45-55.
- Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26(6), 369-395.
- Mauleón, M. (2003). *Sistemas de almacenaje y picking*. España: Díaz de Santos.
- Mayr, E. (1982). Processes of speciation in animals. *Progress in clinical and biological research*, 96, 1-19.
- McCarthy, J., Minsky, M. L., Rochester, N., & Shannon, C. E. (31 de 08 de 1955). A proposal for the dartmouth summer research project on artificial intelligence. 27(4). AI magazine.
- Méndez Babey, M. J. (2008). *Algoritmos evolutivos y preferencia del decisor aplicados a problemas de optimización multiobjetivos discretos*. Las Palmas de Gran Canaria: ULPGC Departamento de Ingeniería Mecánica.
- Méndez, M., Salazar, D., & Galván, B. (2005). Optimización evolutiva multiobjetivo del diseño de Sistemas de Seguridad: Comparando el NSGAI y el criterio del tomador de decisiones. In *Actas del IV Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB 2005)*. Granada.
- Michalewicz, Z. (1992). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag.
- Miettinen, K. (2012). *Nonlinear multiobjective optimization*. Springer Science & Business Media.

Referencias

- Miettinen, K., & Mäkelä, M. M. (2006). Synchronous approach in interactive multiobjective optimization. . *European Journal of Operational Research*, 170(3), 909-922.
- Miguel, F., Frutos, M., & Tohmé, F. (2014). Modelado del BPP/CVRP y su resolución a través de una meta-heurística evolutiva. En *Memorias del COINI 2013 UTN FRSR* (págs. 01-13). Ciudad Autónoma de Buenos Aires: edUTecNe.
- Miguel, F., Frutos, M., & Tohmé, F. (2015). Programación de Operaciones en Sistemas Logísticos de Transporte y Distribución mediante Algoritmos Evolutivos. *ANDAP, Asociación Argentina de Docentes de Administración de la Producción*, 3.
- Min, H. (1991). A multiobjective vehicle routing problem with soft time windows: the case of a public library distribution system. *Socio-Economic Planning Sciences*, 25(3), 179-188.
- Molina, J., Santana, L. V., Hernández-Díaz, A. G., Coello, C. A., & Caballero, R. (2009). g-dominance: Reference point based dominance for multiobjective metaheuristics. *European Journal of Operational Research*, 197(2), 685-692.
- Molina, J., Santana, L. V., Hernández-Díaz, A. G., Coello, C. A., & Caballero, R. (2009). g-dominance: Reference point based dominance for multiobjective metaheuristics. *European Journal of Operational Research*, 197(2), 685-692.
- Montoya-Torres, J. R., Franco, J. L., Isaza, S. N., Jiménez, H. F., & Herazo-Padilla, N. (2015). A literature review on the vehicle routing problem with multiple depots. . *Computers & Industrial Engineering*, 79, 115-129.
- Mula, J., Peidro, D., Diaz-Madronero, M., & Vicens, E. (2010). Mathematical programming models for supply chain production and transport planning. *European Journal of Operational Research*., 204(3), 377-390.

Referencias

- Ojalehto, V., Miettinen, K., & Mäkelä, M. M. (2007). Interactive software for multiobjective optimization: IND-NIMBUS. *WSEAS Transactions on Computers*, 6(1), 87-94.
- Olivera, A. (2004). *Heurísticas para problemas de ruteo de vehículos (Reportes Técnicos)*. Uruguay: UR. FI - INCO.
- Ombuki, B., Ross, B. J., & Hanshar, F. (2006). Multi-objective genetic algorithms for vehicle routing problem with time windows. *Applied Intelligence*, 24(1), 17-30.
- Ombuki-Berman, B. M., Runka, A., & Hanshar, F. (2007). Waste collection vehicle routing problem with time windows using multi-objective genetic algorithms. *Proceedings of the Third IASTED International Conference on Computational Intelligence*. ACTA Press.
- Ombuki-Berman, B., & Hanshar, F. T. (2009). Using genetic algorithms for multi-depot vehicle routing. En *Bio-inspired algorithms for the vehicle routing problem* (págs. 77-99). Berlin: Springer Berlin Heidelberg.
- Osyczka, A. (1985). Multicriteria optimization for engineering design. En *Design Optimization* (págs. 193-227). New York: Academic Press.
- Pacheco, J., & Martí, R. (2006). Tabu search for a multi-objective routing problem. *Journal of the Operational Research Society*, 57, 29-37.
- Paletta, F. C., & Dias Vieira Junior, N. (2008). Information technology and communication and best practices in it lifecycle management. *Journal of technology management & innovation*, 3(4), 80-94.
- Palma Méndez, J. T., & Morales, R. M. (2008). *Inteligencia Artificial. Técnicas, métodos y aplicaciones*. Madrid, España: McGraw-Hill.
- Payne, A. (2006). *Handbook of CRM: achieving excellence in customer management*. Great Britain: Elsevier.
- Pazos, A., Pedreira, N., Rabuñal, J. R., & Pereira, J. (2007). Inteligencia Artificial y Computación Avanzada. En *Inteligencia Artificial y Computación Avanzada* (págs. 9-34).

Referencias

- Pérez, J. M. (2010). *Inteligencia computacional inspirada en la vida*. SPICUM Universidad de Málaga.
- Polacek, M., Hartl, R. F., Doerner, K., & Reimann, M. (2004). A variable neighborhood search for the multi depot vehicle routing problem with time windows. *Journal of heuristics*, 10(6), 613-627.
- Professionals Council of Supply Chain Management. (2015). *CSCMP- Council of Supply Chain Management Professionals*. (CSCMP) Obtenido de <https://cscmp.org/about-us/supply-chain-management-definitions>
- Puche Regaliza, J. C., Costas Gual, J., & Arranz Val, P. (2016). Simulación como herramienta de ayuda para la toma de decisiones empresariales. Un caso práctico. *Revista de Métodos Cuantitativos para la Economía y la Empresa*, 21, 188-204.
- Rachmawati, L., & Srinivasan, D. (2006). Preference incorporation in multi-objective evolutionary algorithms: A survey. *International Conference on Evolutionary Computation*, (págs. 3385-3391). Los Alamitos.
- Ridley, M. (1993). *Evolution*. Boston: Blackwell.
- Riera-Ledesma, J., & Salazar-González, J. J. (2005). The biobjective travelling purchaser problem. *European Journal of Operational Research*, 160(3), 599-613.
- Riera-Ledesma, J., & Salazar-González, J. J. (2012). Solving school bus routing using the multiple vehicle traveling purchaser problem: A branch-and-cut approach. *Computers & Operations Research*, 39(2), 391-404.
- Ritzel, B. J., Eheart, W. J., & Ranjithan, S. (1994). Using genetic algorithms to solve a multiple objective groundwater pollution containment problem. *Water Resources Research*, 30(5), 1589-1603.
- Ropke, S., & Pisinger, D. (2006). A unified heuristic for a large class of Vehicle Routing Problems with Backhauls. *European Journal of Operational Research*, 171, 750 - 775.

Referencias

- Roudenko, O. (2004). *Application of evolutionary algorithms to multi-objective optimization problems with constraints*. Doctoral dissertation, Ecole Polytechnique X.
- Russell, S. J., & Norvig, P. (2004). *Inteligencia Artificial. Un Enfoque Moderno*. Madrid: Pearson Educación.
- Salazar González, J. J. (2001). *Programación Matemática*. Tenerife, España: Ed. Diaz de Santos.
- Salazar, D., Carrasquero, N., & Galván, B. (2005). Exploiting Comparative Studies Using Criteria: Generating Knowledge From An Analyst's Perspective. *International Conference on Evolutionary Multi-Criterion Optimization* (págs. 221-234). Berlin: Springer.
- Sánchez, J., Cortés, A., Peralta, E., & Díaz, S. (2007). *El Transporte Automotor de Cargas en la Argentina*. Ciudad Autónoma de Buenos Aires: edUTecNe.
- Schaffer, J. D. (1984). Multiple objective optimization with vector evaluated. . En *Genetic Algorithms Ph. D. Dissertation*. Vanderbilt University.
- Schwefel, H. P. (1995). *Evolution and Optimization Seeking*. New York: John Wiley & Sons.
- Simon, H. A. (1972). Theories of Bounded Rationality. En *Decision and Organization*. Amsterdam: North-Holland.
- Simon, H. A. (2003). Racionalidad limitada en Ciencias Sociales: Hoy y mañana. En *Racionalidad, historicidad y predicción en Herbert A. Simon* (págs. 97-110).
- Słowiński, R. (1984). Review of multiple objective programming methods. *Przeegląd Statystyczny*, 31, 47-63.
- Sólomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2), 254-265.

Referencias

- Somhom, S., Modares, A., & Enkawa, T. (1999). Competition-based neural network for the multiple travelling salesmen problem with minmax objective. *Computers & Operations Research*, 26(4), 395-407.
- Srinivas, N., & Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3), 221–248.
- Sutcliffe, C., & Boardman, J. (1990). Optimal solution of a vehicle-routeing problem: transporting mentally handicapped adults to an adult training centre. *Journal of the Operational Research Society*, 41(1), 61-67.
- Talbi, E. G. (2009). *Metaheuristics: from design to implementation*. John Wiley & Sons.
- Tan, K. C., Cheong, C. Y., & Goh, C. K. (2007). Solving multiobjective vehicle routing problem with stochastic demand via evolutionary computation. *European Journal of operational research*, 177(2), 813-839.
- Tan, K. C., Chew, Y. H., & Lee, L. H. (2006). A hybrid multi-objective evolutionary algorithm for solving truck and trailer vehicle routing problems. *European Journal of Operational Research*, 172(3), 855-885.
- Tan, K. C., Chew, Y. H., & Lee, L. H. (2006). A hybrid multiobjective evolutionary algorithm for solving vehicle routing problem with time windows. *Computational Optimization and Applications*, 34(1), 115-151.
- Tan, K. C., Lee, T. H., & Khor, E. F. (1999). Evolutionary algorithms with goal and priority information for multi-objective optimization. En *Proceedings of the 1999 Congress on Evolutionary Computation*. IEEE.
- Tavakkoli-Moghaddam, R., Safaei, N., Kah, M., & Rabbani, M. (2007). A new capacitated vehicle routing problem with split service for minimizing fleet cost by simulated annealing. *Journal of the Franklin Institute*, 344, 406–425.
- Taylor, G. D. (2007). *Logistics engineering handbook*. New York: CRC press.

Referencias

- Thiele, L., Miettinen, K., Korhonen, P. J., & Molina, J. (2007). A preference-based interactive evolutionary algorithm for multiobjective optimization. En *Technical Report W-412*. Helsinki, Finland: Helsinki School of Economics.
- Ulungu, E. L., Teghem, J. F., Fortemps, P. H., & Tuyttens, D. (1999). MOSA method: a tool for solving multiobjective combinatorial optimization problems. *Journal of multicriteria decision analysis*, 8(4), 221.
- Urquhart, N., Hart, E., & Scott, C. (2010). Building low CO₂ solutions to the vehicle routing problem with Time Windows using an evolutionary algorithm. *IEEE Congress on Evolutionary Computation*, (págs. 1-6). Barcelona.
- Van Veldhuizen, D. A. (1999). *Multiobjective evolutionary algorithms: classifications, analyses, and new innovations*. PhD tesis, Graduate School of Eng. of the Air Force Inst. of Technol.
- Van Veldhuizen, D. A., & Lamont, G. B. (1998). *Multiobjective evolutionary algorithm research: A history and analysis*. Ohio: Wright-Patterson AFB.
- Vecchiato, P. (2012). *Lean Distribution*. Padova: Padova Digital University Archive. Obtenido de <http://paduaresearch.cab.unipd.it/4708/>
- Weise, T., Podlich, A., & Gorltdt, C. (2009). Solving real-world vehicle routing problems with evolutionary algorithms. En *Natural intelligence for scheduling, planning and packing problems* (págs. 29-53). Berlin: Springer Berlin Heidelberg.
- Wetzel, A. (1983). *Evaluation of the effectiveness of genetic algorithms in combinatorial optimization*. Pittsburgh, Philadelphia, USA: University of Pittsburgh.
- Whitley, L. D. (1989). The GENITOR Algorithm and Selection Pressure: Why Rank-Based Allocation of Reproductive Trials is Best. En *Proceedings of the Third International Conference on Genetic Algorithms* (págs. 116-121). California: Morgan Kaufmann Publishers.
- Whitley, L. D., Starkweather, T., & Fuquay, D. (1989). Scheduling Problems and Traveling Salesmen: The Genetic Edge Recombination Operator. En

- Proceedings of the 3rd International Conference on Genetic Algorithms* (págs. 133-140). San Francisco: Morgan Kaufmann Publishers Inc.
- Wolpert, D. H., & Macready, W. G. (1995). No free lunch theorems for search. *Technical Report SFI-TR-95-02-010*, 10.
- Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *Evolutionary Computation, IEEE Transactions*, 1(1), 67-82.
- Yu, P. L., & Leitmann, G. (1974). Compromise solutions, domination structures, and Salukvadze's solution. *Journal of Optimization Theory and Applications*, 13(3), 362-378.
- Zadeh, L. (1963). Optimality and non-scalar-valued performance criteria. *IEEE transactions on Automatic Control*, 8(1), 59-60.
- Zapata Cortés, J. A., Arango Serna, M. D., & Adarme Jaimes, W. (2010). Herramientas tecnológicas al servicio de la gestión empresarial. *Avances en Sistemas e Informática*, 7(3), 87-102.
- Zheng, B. (2010). Multi-objective vehicle routing problem in hazardous material transportation. *ICLEM 2010: Logistics for Sustained Economic Development-Infrastructure, Information, Integration-. Proceedings of the 2010 International Conference of Logistics Engineering and Management*. Chengdu.
- Zitzler, E., & Thiele, L. (1999). Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE transactions on Evolutionary Computation*, 3(4), 257-271.
- Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. En *Evolutionary Methods for Design, Optimisations and Control* (págs. 19-26). Barcelona, Spain: CIMNE.

RESUMEN

Este libro se enmarca en el campo de las Operaciones, un área de las Ciencias de la Administración. El objetivo es el desarrollo y la validación de herramientas tecnológicas para la toma de decisiones en la rama logística de la gestión de la cadena de suministro de la industria alimentaria. Más específicamente, presentamos un novedoso procedimiento inteligente híbrido que ayuda a los responsables de la toma de decisiones a optimizar en un entorno multi-objetivo. Examinamos los modelos habituales de la distribución física de los bienes, clasificándolos según sus principales características. También presentamos los algoritmos evolutivos multi-objetivo que generalmente brindan las soluciones a esos modelos. Nuestro enfoque es introducir una nueva variante multi-objetivo del problema de distribución de bienes en un área urbana. Para modelar la red de centros de distribución, al servicio de los puntos de venta finales de productos, agregamos franjas horarias y otros requisitos. Consideramos la dependencia temporal de los programas óptimos de distribución, a diferencia del caso de los sistemas logísticos de media y larga distancia, para los cuales la distancia es el criterio clave. También agregamos el objetivo de equilibrar cargas entre las diferentes unidades operativas. Una hibridación del algoritmo evolutivo multi-objetivo NSGA-II es nuestra elección de herramienta computacional, junto con el concepto de g-dominación para preferencias parciales, que proporciona la guía informativa en el espacio de búsqueda. Las fases de validación y prueba de este algoritmo utilizan datos del mundo real, comparando sus resultados con los resultados de otros procedimientos evolutivos multi-objetivo utilizados para la solución de problemas complejos de distribución. La información fue proporcionada por un operador logístico, especializado en el transporte y la distribución de cargas fraccionarias. En todos los casos examinados, nuestro algoritmo se desempeñó mejor que los habituales.

RESUMEN

Este libro se enmarca en el campo de las Operaciones, un área de las Ciencias de la Administración. El objetivo es el desarrollo y la validación de herramientas tecnológicas para la toma de decisiones en la rama logística de la gestión de la cadena de suministro de la industria alimentaria. Más específicamente, presentamos un novedoso procedimiento inteligente híbrido que ayuda a los responsables de la toma de decisiones a optimizar en un entorno multi-objetivo. Examinamos los modelos habituales de la distribución física de los bienes, clasificándolos según sus principales características. También presentamos los algoritmos evolutivos multi-objetivo que generalmente brindan las soluciones a esos modelos. Nuestro enfoque es introducir una nueva variante multi-objetivo del problema de distribución de bienes en un área urbana. Para modelar la red de centros de distribución, al servicio de los puntos de venta finales de productos, agregamos franjas horarias y otros requisitos. Consideramos la dependencia temporal de los programas óptimos de distribución, a diferencia del caso de los sistemas logísticos de media y larga distancia, para los cuales la distancia es el criterio clave. También agregamos el objetivo de equilibrar cargas entre las diferentes unidades operativas. Una hibridación del algoritmo evolutivo multi-objetivo NSGA-II es nuestra elección de herramienta computacional, junto con el concepto de g-dominación para preferencias parciales, que proporciona la guía informativa en el espacio de búsqueda. Las fases de validación y prueba de este algoritmo utilizan datos del mundo real, comparando sus resultados con los resultados de otros procedimientos evolutivos multi-objetivo utilizados para la solución de problemas complejos de distribución. La información fue proporcionada por un operador logístico, especializado en el transporte y la distribución de cargas fraccionarias. En todos los casos examinados, nuestro algoritmo se desempeñó mejor que los habituales.