



Subsecretaría de Educación Superior
Dirección General de Educación Superior Tecnológica
Instituto Tecnológico de Ciudad Madero

INSTITUTO TECNOLÓGICO DE CIUDAD MADERO

División de Estudios de Posgrado e Investigación

PROYECTO DE DESARROLLO TECNOLÓGICO

Una Aplicación para la Solución de los Problemas de Enrutamiento, Asignación de Horarios y Distribución de Cargas

Usuario: GRUPO CONTINENTAL (CONTAL) DE SOCIEDAD INDUSTRIAL S.A. DE C.V

Participantes:

Dra. Laura Cruz Reyes (Líder del Proyecto)

Dra. Claudia Gpe. Gómez Santillán

Dr. Héctor J. Fraire Huacuja

Dr. Javier González Barbosa

Dr. José Antonio Martínez Flores

M.C. Guadalupe Castilla Valdez

Enero de 2014

Tabla de contenido

1. Introducción	1
2. Descripción de las embotelladoras de CONTAL	1
3. Logística de distribución y problemas de optimización en CONTAL	3
4. Estado del arte	4
5. Desarrollo tecnológico propuesto	5
5.1 Objetivos	
5.2 Metodología de solución propuesta	
5.5 Comercialización	
5.6 Formación de recursos	
5.7 Contribuciones del Cuerpo Académico	
5.8 Manuales y mantenimiento	
6. Reporte Técnico	15
Referencias	50
Anexos (Cartas de Usuario y Correos de Usuario)	

1. Introducción

Hoy en día, la gestión de transporte y distribución de productos representa uno de los mayores retos de la logística. Estudios recientes revelan que una adecuada planeación, daría como resultado ahorros significativos en el rango del 5 al 20% del costo total del producto [Toth, 2001]. Sin embargo, La mayoría de los esfuerzos de investigación han ido encaminados a entornos sumamente relajados, cubriendo parcialmente las características que se presentan en ambientes reales.

El presente proyecto plantea el desarrollo de una Metodología Integral basada en métodos Heurísticos y de aprendizaje automático para la Solución de instancias reales del problema de distribución de productos, capaz de encontrar una configuración adecuada para la asignación de rutas y cargas a vehículos repartidores, detalle de horarios de salida y arribo de los mismos, así como la distribución y acomodo del producto dentro de cada unidad móvil. Particularmente se resuelve la Logística de Distribución de productos para las embotelladoras de Grupo Continental (CONTAL), con apoyo de la compañía de desarrollo de software FORTIA.

El desempeño de los métodos heurísticos y de aprendizaje automático, que se desarrollan en este proyecto, son validados con un conjunto de problemas para los cuales existen casos de prueba estandarizados. Esto con la finalidad de contar con una base tecnológica robusta que permita al grupo de investigación abordar otros problemas de optimización altamente complejos que están presentes en entornos regionales y nacionales.

2. Descripción de las embotelladoras de CONTAL

Grupo Continental cuenta con un complejo formado por siete embotelladoras: Durango, Comarca Lagunera, Zacatecas, San Luis Potosí, Aguascalientes, Jalisco y Tecomán; la matriz se encuentra en Tampico, Tamaulipas. La topología de distribución de cada embotelladora del grupo corresponde con alguna de las regiones mostradas en la Figura 1. Una región representa un área de distribución formada por un conjunto de clientes asociado a uno o más almacenes. La Figura 1 también muestra un grafo de conexión típico de los tres tipos de regiones.

Previo al desarrollo del presente proyecto, el proceso de distribución en cada embotelladora lo realizaba un equipo de expertos humanos dedicados de tiempo completo a generar planes de

acomodo de productos en camiones y planes de rutas de vehículos. Como parte de las rutas generadas, se estiman los horarios de -salida y arribo- de los vehículos a los almacenes, así como a las instalaciones de los clientes. En la planificación de camiones se considera la posibilidad de rentar vehículos.

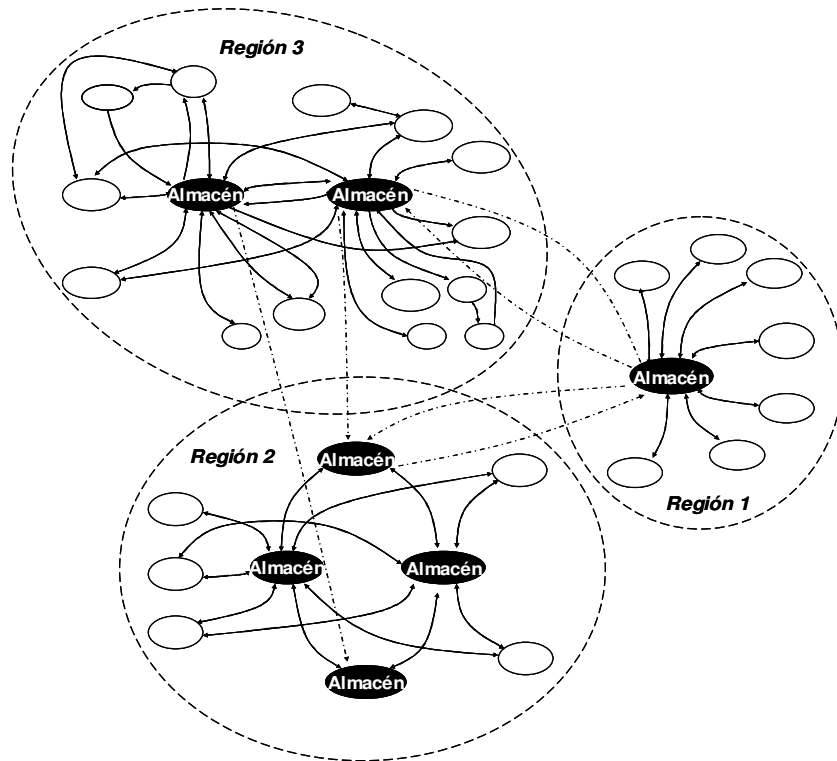


Figura 1 Grafo de conexión de las regiones de CONTAL

3. Logística de distribución y problemas de optimización en CONTAL

La logística de distribución de productos embotellados de CONTAL está conformada principalmente por los siguientes elementos.

- Un conjunto de *pedidos* u órdenes de compra a ser entregados en las localidades del cliente formadas por productos con diferentes atributos.
- Un conjunto de *clientes* con horarios de servicio independientes en cada localidad y una capacidad de atención finita en un determinado momento.
- Un conjunto de *almacenes* o centros de embarque con un horario de trabajo independiente en cada centro de embarque y con posibilidades de solicitar bienes a otros centros de embarque.

- d) Una flotilla de *vehículos* heterogénea para transportar los bienes, cada vehículo cuenta con un tiempo de servicio y un tiempo de atención en las localidades de los clientes de acuerdo a la capacidad del vehículo hacia una determinada localidad de los clientes y los recursos de cada localidad para realizar las operaciones de carga y descarga de los vehículos, asociados al tiempo de maniobra del vehículo en la localidad.
- e) Un conjunto de *caminos* que conectan a los centros de embarque con las localidades de los clientes, cada camino tiene un costo de transporte asignado y un tiempo de viaje del nodo al nodo, y cada uno cuenta con un límite de peso de los vehículos que circulan en las carreteras hacia un determinado destino.

El problema principal de la logística de distribución de productos de CONTAL busca minimizar el número de vehículos a ser utilizados y la distancia de recorrido, generando para ello rutas que inician y terminan en un determinado centro de embarque. También se busca satisfacer la entrega de los bienes en las localidades de los clientes; este objetivo implica acomodar las mercancías en el menos número de vehículos y optimizar los horarios de viaje.

Se definió RoSLOP (Routing-Scheduling-Loading Problem) con el fin de caracterizar y delinear el problema de transporte presente en el proceso de distribución y entrega de productos de situaciones reales, como las de CONTAL. Para tal propósito, la descripción general de RoSLOP se basa principalmente en la inclusión de diversas variantes de dos problemas ampliamente conocidos en las ciencias computacionales:

El Problema del Enrutado de Vehículos (VRP, Vehicle Routing Problem) [Dantzing, 1959]

El Problema del Empacado en Contenedores (BPP, Bin Packing Problem) [Coffman 2002]

La Tabla 1 presenta para cada problema, los parámetros de entrada, las salidas esperadas, así como la unidad de medición de la función objetivo de minimización. A cada problema se le ha antecedido la palabra Rich (enriquecido) para denotar, de acuerdo con la literatura [Toth, 2001], que el problema incluye una variedad de condiciones que están presentes en ambiente reales.

Como ya se mencionó, RoSLoP involucra tres subtarefas: asignación de rutas, asignación de horarios y asignación de cargas. Las dos primeras se pueden describir como un problema Rich-VRP [Pisinger, 2005], un entorno donde diversas variantes del VRP participan en la definición de una situación más compleja y cuyo objetivo es encontrar un conjunto de rutas que satisfagan las

demandas. Por otro lado, la asignación de cargas puede verse como un conjunto de variantes de Rich-BPP, donde se debe determinar una cantidad mínima de contenedores para distribuir los productos con la condición adicional del balanceo, ésta última modelada por el Problema de Planificación del Procesador. La Figura 2 muestra la relación de RoSLoP con dichos problemas y sus variantes.

Tabla 1. Problemas de optimización abordados

Problema	Entrada	Salida	Unidad de Medición
Rich-BPP	Características de los productos y de los contenedores	Acomodo de productos en contenedores.	Contenedores
Rich-VRP	Características de pedidos, flotillas, clientes, almacenes y caminos.	Rutas y horarios	Tiempo y número de camiones

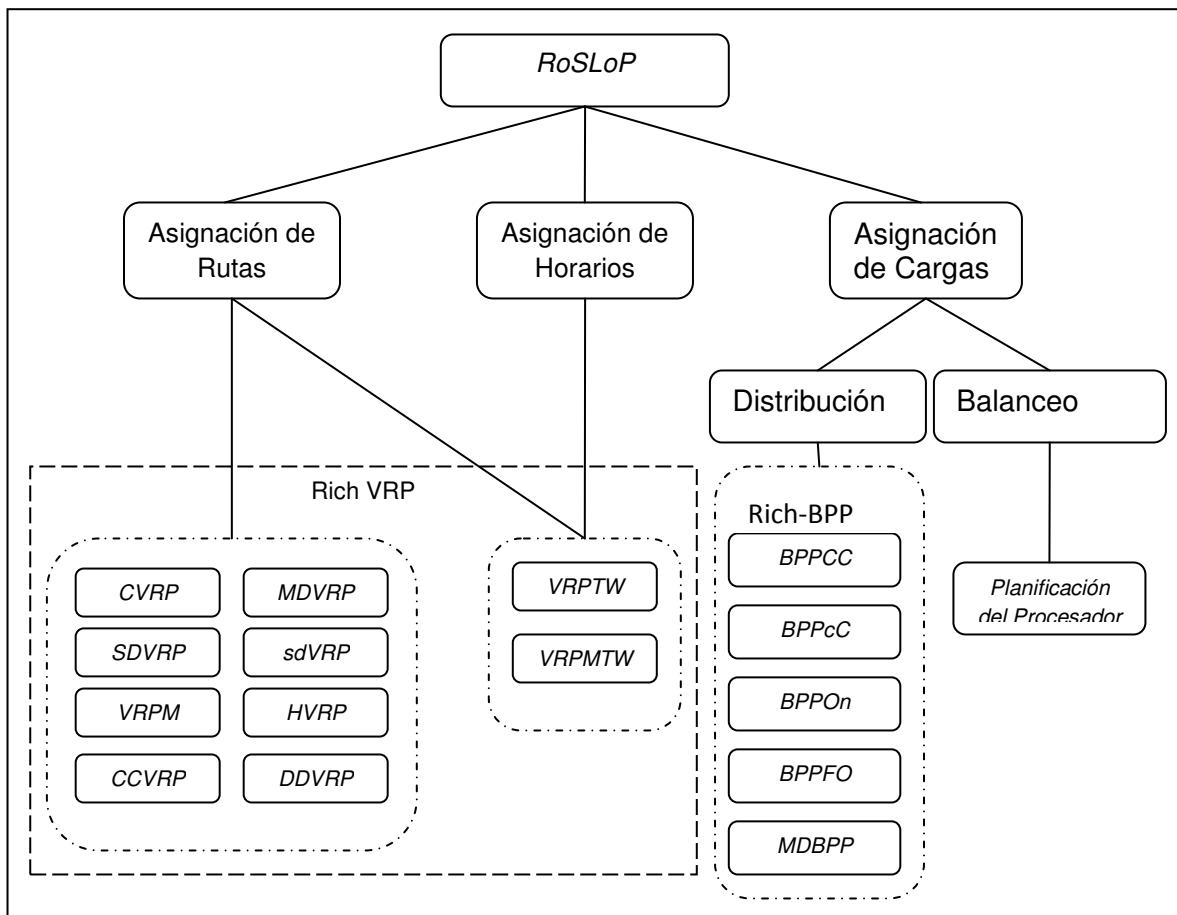


Figura 2. RosLoP y problemas de optimización involucrados

Con los elementos antes descritos, podemos precisar la definición de RoSLoP. Dado un conjunto de clientes, ordenes de productos, carreteras, centros de embarque, flotillas de vehículos repartidores circunscritos al proceso de distribución y entrega de productos, en el cuál se encuentra inmerso un conjunto de variantes tanto VRP como BPP que lo restringen y en forma optativa la función objetivo del Problema de Planificación del Procesador en el acomodo de la mercancía dentro de las unidades móviles, es necesario diseñar y asignar rutas, horarios y cargas a los vehículos de manera que:

- Se satisfagan las demandas de los clientes;
- El costo total del proceso sea minimizado; y
- Las restricciones impuestas por las variantes VRP y BPP sean satisfechas.

4. Estado del arte

VRP y BPP son problemas de optimización de naturaleza NP-duro, esto significa que el esfuerzo computacional requerido para resolverlo incrementa exponencialmente en relación al tamaño del mismo. Puesto que este tipo de problemas son intratables polinomialmente, no existe un algoritmo exacto capaz de resolver instancias relativamente grandes. Para tal situación es deseable obtener soluciones aproximadas y cercanas a lo óptimo, las cuales puedan ser encontradas en forma rápida y suficientemente adecuadas para cumplir con el objetivo planteado. Debido a que en la mayoría de las aplicaciones prácticas se prefiere reducción de tiempo sobre exactitud de la solución, dicho enfoque es abordado mediante el uso de métodos aproximados.

Los algoritmos aproximados se pueden clasificar en dos tipos principales: algoritmos constructivos y algoritmos de búsqueda local. Los primeros se basan en generar soluciones desde cero añadiendo componentes a cada solución paso a paso. Los segundos intentan repetidamente mejorar la solución actual con movimientos a soluciones vecinas. La investigación en este campo en las últimas dos décadas ha centrado su atención en el diseño de técnicas de propósito general para guiar la construcción de soluciones y la búsqueda local mediante distintas heurísticas. Estas técnicas se llaman comúnmente metaheurísticas y son ampliamente reconocidas como una de las mejores aproximaciones para atacar los problemas de optimización combinatoria.

Las metaheurísticas incorporan conceptos de diversos campos como la genética, la biología y la neurología, entre otras. Un campo prometedor para incrementar la eficiencia de la búsqueda heurística es la incorporación de estrategias de aprendizaje automático. Mediante la identificación de patrones, en la estructura del problema y el espacio de solución, es posible configurar dinámicamente la participación de las diferentes heurísticas que integran un algoritmo de solución.

Debido al alto nivel de relajación con que cuentan las definiciones de VRP y BPP, se han desarrollado a lo largo del tiempo diversas variantes de éstos problema, cada una de las cuales propone una restricción adicional al planteamiento, incrementando al mismo tiempo tanto la dificultad como la cercanía con un modelo de la realidad.

La Tabla 2 presenta algunos trabajos de investigación que han abordado Rich-VRP. En general, la mayoría de los investigadores científicos se han centrado en estudiar una, dos o hasta tres variantes de VRP en conjunto; la única excepción encontrada es [Pisinger, 2005], quien aborda cinco variantes con una misma técnica. Por lo que, los métodos de solución desarrollados y publicados en la comunidad científica resuelven problemáticas bastante alejadas de la realidad. Por otro lado, la mayoría de las herramientas comerciales resuelven cuatro variantes del VRP o menos; sólo CATRIN y STARTS abordan cinco y seis de ellas respectivamente. Una situación similar se identificó en Rich-BPP.

Tabla 2. Estado del arte de trabajos de investigación sobre variantes rich VRP

Variantes que aborda	CVRP	VRPTW	VRPMTW	OVRP	VRPPD	MDVRP	PVRP	SDVRP	sdVRP	VRPM	HVRP	CCVRP	DDVRP	rdVRP
Autor														
[Vidal, 2011]		✓				✓	✓		✓					
[Goel, 2006]	✓	✓								✓	✓			
[Hasle, 2007]	✓	✓	✓								✓			
[Pisinger, 2005]	✓	✓		✓		✓			✓					
[Cano, 2005]	✓	✓						✓		✓	✓			
[Rangel, 2005]	✓	✓				✓		✓		✓	✓			
[Herrera, 2006]	✓	✓	✓			✓		✓	✓	✓	✓	✓	✓	✓

[Delgado, 2007]	✓	✓	✓	✓		✓		✓	✓	✓	✓	✓	✓	✓
[Cruz, 2008]														

Se realizó un estudio profundo y detenido del sistema de transportación de las embotelladoras de CONTAL. Se observó que la planeación de estas tareas está sujeta a las restricciones impuestas por nueve de las variantes de VRP de la Tabla 2. De igual manera, el proceso de distribución de CONTAL impone seis variantes de BPP.

5. Desarrollo tecnológico propuesto

Esta sección describe el desarrollo tecnológico propuesto para la solución de la logística de distribución de productos embotellados del grupo CONTAL. La propuesta de solución se obtuvo con la colaboración de FORTIA, particularmente en el levantamiento de requerimientos.

5.1. Objetivos

Objetivo general

Desarrollar un Modelo Integral y su metodología de solución que permita satisfacer las demandas de clientes de una distribuidora de productos que sea representativa de las empresas de transportación del sector nacional.

Objetivos específicos

- a) Formalizar los requisitos del problema de distribución de productos en términos de variantes de problemas clásicos de las ciencias computacionales: programación y enrutamiento de vehículos (VRP, Vehicle Routing Problem) y problema de empacado de objetos en contenedores (BPP, Bin Packing Problem).
- b) Formular el problema de distribución de productos mediante un modelo de programación matemática.
- c) Construir una solución heurística eficiente que determine rutas, horarios y cargamento de una flotilla de vehículos de tal manera que se minimice el costo total del transporte de productos. La solución óptima debe satisfacer los requerimientos especificados como variantes VRP y BPP

- d) Construir una plataforma robusta de algoritmos heurísticos y de aprendizaje automático que facilite la solución de problemas complejos que están presentes en ambientes reales, y para los cuales existen casos de prueba estandarizados.

5.2 Metodología de Solución propuesta

Se propone una metodología integral basada en heurísticas y aprendizaje automático para la solución de instancias reales y complejas del Problema de Asignación de Rutas, Horarios y Cargas con un enfoque de optimización. Dicha metodología se conforma por un Sistema de Colonia de Hormigas (ACS, Ant Colony System) que resuelve las dos primeras tareas; y DiPro un algoritmo aproximado, ad-hoc y determinista que aborda la última. La metodología permite encontrar una configuración adecuada para la especificación de rutas y cargas de vehículos repartidores, detalle de horarios de salida y arribo de los mismos, así como la distribución y acomodo del producto dentro de cada unidad móvil.

La metodología de solución se obtuvo mediante la aplicación de las siguientes etapas:

- a) Modelado Matemático del Problema de Distribución.
Este problema incluye tres subproblemas: enrutamiento, programación de horarios y carga de vehículos; por lo que se ha denominado RoSLoP (Routing, Scheduling and Loading Problem). Se desarrolló un modelo matemático para la tarea de enrutamiento y carga.
- b) Recopilación de casos de prueba de RoSLoP.
Para la evaluación parcial y final de los algoritmos desarrollados se recolectaron casos de prueba de las embotelladoras de CONTAL. Los algoritmos de base fueron comparados con respecto a casos estándar reconocidos por la comunidad científica.
- c) Desarrollo de métodos heurísticos robustos.
En esta etapa se realizó el diseño, implementación y evaluación de métodos heurísticos. Para determinar la veracidad de los resultados se utilizaron técnicas de control de la varianza. Está en desarrollo la extensión de estos algoritmos a la solución del problema de enrutamiento de consultas semánticas.
- d) Desarrollo de métodos de aprendizaje automático
En esta etapa se realizó el diseño e implementación de un método de agrupación para la formación dinámica de grupos. Este método se incorporó en el algoritmo de enrutamiento para adaptarse a las características de las instancias.

- e) Desarrollo de una arquitectura de solución del problema de distribución RoSLoP.
 Se formalizaron los requerimientos de RoSLoP en términos de variantes de los problemas clásicos VRP y BPP. A partir de esta formalización se diseñó e implementó una arquitectura de solución, la cual está basada principalmente en métodos heurísticos y de manera secundaria en métodos de aprendizaje automático.
- f) Pruebas de funcionamiento del sistema de distribución.
 La evaluación del Sistema de distribución se realizó en un ambiente real, comparando los resultados obtenidos computacionalmente contra los utilizados por dicho entorno, permitiendo efectuar una proyección de la disminución de los costos de operación. Como trabajo futuro queda la ampliación del registro histórico de datos para incorporar condiciones de mayor complejidad.

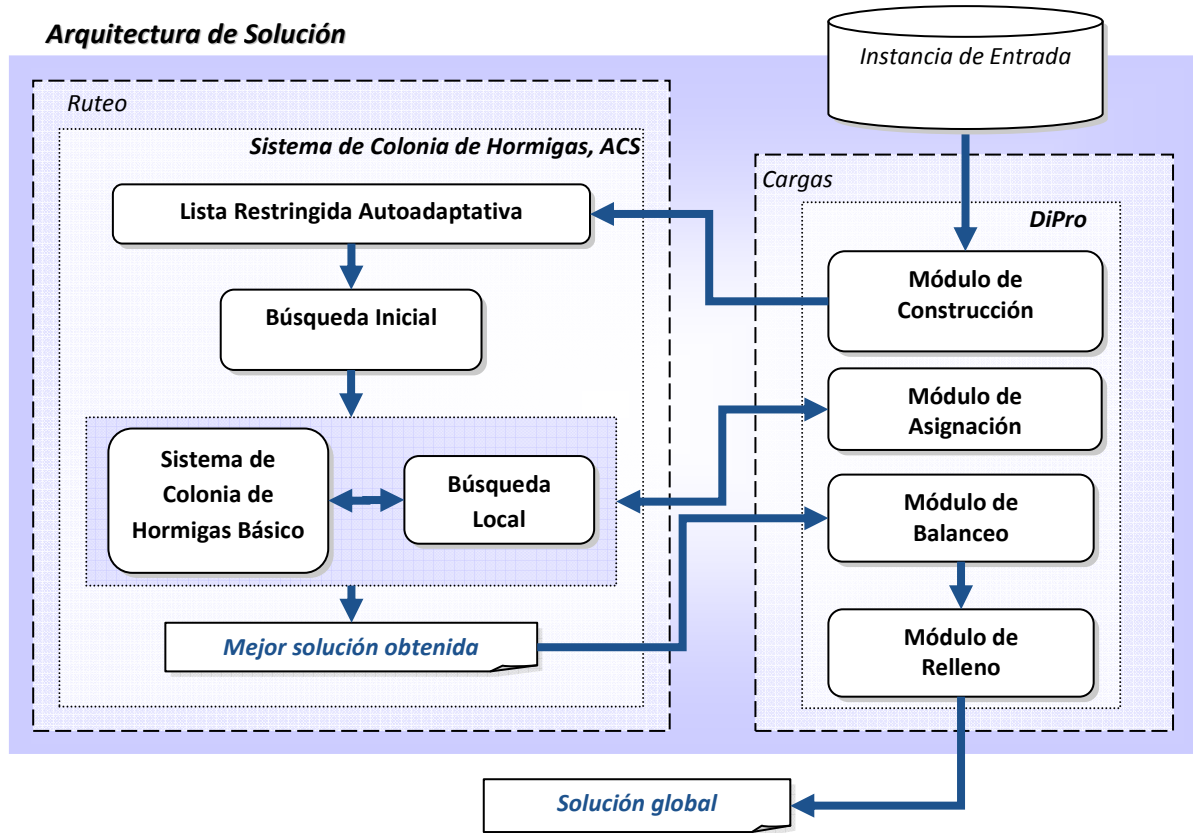


Figura 3. Arquitectura de solución de RoSLoP

La Figura 3 muestra la arquitectura propuesta para la solución integral de RoSLoP. La arquitectura consta de dos algoritmos heurísticos: DiPro (Distribution of the Products) y ACS (Ant Colony

System) que interactúan sistemáticamente, y sacan ventaja de información proporcionada por métodos estadísticos y de aprendizaje automático.

La forma en que interactúan entre si todos los componentes para concretar la construcción de la solución es la siguiente. Antes de inicializar el proceso de búsqueda de la ruta óptima se invoca el Módulo Construcción que transforma los pedidos en tarimas homogéneas e incompletas, reduciendo el problema de empaqueo a una dimensión. Después se crea la Lista Restringida Autoadaptativa para reducir el espacio de búsqueda utilizando información de la instancia y un método de agrupación. Posteriormente se encuentra una primera solución mediante la Búsqueda Inicial, seguida por la construcción de soluciones a través del ACS básico y de la mejora de éstas mediante la Búsqueda Local. El Módulo Asignación es invocado durante el proceso de construcción de la ruta, cuando se visita un cliente, para mantener actualizada la información tanto de la carga que transporta cada unidad móvil como de la mercancía ya entregada. Cuando la búsqueda de soluciones culmina y se obtiene la mejor solución generada, esta es completada a través del Módulo Balanceo y Módulo Relleno, ambos ejecutados fuera de línea, administrando de esta manera eficazmente el tiempo de cómputo.

Los resultados de la experimentación muestran que ambos algoritmos disminuyen el número de vehículos y satisfacen la demanda dentro de los límites establecidos por la compañía caso de estudio (5 minutos). Todas las instancias de prueba tomadas de situaciones reales provienen de usuarios que realizan manual y cotidianamente planes de transportación.

5.5 Comercialización

El presente desarrollo tecnológico ha sido concebido para CONTAL en el marco de acuerdos de colaboración. El análisis de requerimientos ha corrido a cargo del ITCM y de FORTIA. En este acuerdo, los investigadores del ITCM son responsables de analizar y optimizar el proceso de Logística de Distribución de productos embotellados, guardando los aspectos de confiabilidad que le dan ventaja competitiva a la empresa. FORTIA es responsable de desarrollar un sistema de información que integre los procesos administrativos de los almacenes de las embotelladoras. Por su

parte, CONTAL se compromete a proporcionar los datos y el acceso a las instalaciones y al personal relacionado con la Logística de Distribución de productos embotellados.

La colaboración trajo beneficios para todos los participantes. El ITCM ha publicado todo el trabajo científico desarrollado y FORTIA mejoró su posicionamiento ante CONTAL y organismos certificadores. Por su parte, CONTAL incrementó la satisfacción del usuario y obtuvo una reducción importante de los gastos de operación al minimizar el número de unidades utilizadas para la transportación en un 9.6% y el tiempo de viaje de las mismas en un 9.1%

La solución de RosLoP actualmente está orientada hacia una empresa. Sin embargo, la metodología seguida es general y apegada a estándares, por lo cual puede ser extendida para su futura comercialización a empresas de transportación del sector nacional. Además, muchos de los requisitos que impone la distribución de mercancías están presentes en otros problemas de distribución. En este sentido, el contar con soluciones robustas facilitaría su aplicación a problemas de otros dominios.

5.6 Formación de recursos y publicaciones

En este proyecto participaron estudiantes de maestría del ITCM:

1. **Nelson Rangel Valdez** con la tesis “Análisis de los Problemas de Asignación de Rutas, Horarios y Cargas en una Distribuidora de Productos”.
2. **Arturo Herrera Ortiz** con la tesis “Desarrollo de una Metodología basada en Heurísticas para la Solución Integral de Problemas de Asignación de Rutas, Horarios y Cargas en el Proceso de Distribución y Entrega de Productos”.
3. **José Francisco Delgado Orta con la tesis** “Modelado matemático para el análisis y mejoramiento de los métodos aproximados para la solución de los problemas de enrutamiento, asignación de horarios y distribución de cargas: una aplicación a la entrega de productos embotellados”.
4. **Bárbara Abigail Arrañaga Cruz** con la tesis “Búsqueda Local Mejorada para la Solución del Problema VRPTW en un Sistema de Colonia de Hormigas”.
5. **Diana Maritza Nieto** con la tesis “Hibridación de Algoritmos Metaheurísticos para Problemas de Bin Packing”.

6. **Ricardo Cámara Covarrubias** con la tesis “Impacto de la búsqueda local aplicada al Problema de Rutas de Vehículos con Ventanas de Tiempo”.
7. **Juan José Martínez Ponce** con la tesis “Análisis de factores que afectan el desempeño de metaheurísticos aplicados a VRPTW”

En cuanto a publicaciones, se obtuvieron 2 capítulos de libro, 2 artículo indizado y 5 artículos en revista arbitrada.

5.7 Contribuciones del cuerpo académico

El sistema de solución de RosLoP ha transitado por tres versiones, así como con cambios en algunos profesores participantes. La mayor participación fue de los seis profesores listados en la Tabla 3, junto con las actividades realizadas. Los profesores pertenecen a dos programas en Ciencias de la Computación (maestría y doctorado) del ITCM.

Tabla 3. Contribuciones del cuerpo académico

Nombre del Participante	Actividades realizadas
Dra. Laura Cruz Reyes (Líder del proyecto)	Supervisión del desarrollo de la investigación. Comunicación con el sector industrial. Formulación de requerimientos. Diseño de la arquitectura de solución. Formulación del 1er modelo matemático de RosLoP
Dra. Claudia G. Gómez Santillán	Caracterización de instancias de VRP. Desarrollo del método de construcción de la lista restringida de ACO basado en las características de las instancias.
Dr. Héctor Fraire Huacuja	2a versión del modelo matemático de RoSLoP. Desarrollo del Método de aprendizaje por niveles para el algoritmo de Colonia de hormigas ACO.
Dr. Juan Javier González Barbosa	Análisis del estado del arte en métodos heurísticos para la variante VRPTW. Desarrollo de un método de búsqueda local.
Dr. José Antonio Martínez Flores	Diseño de protocolos para la construcción del software científico como librería portable y estandarizada. Pruebas de funcionamiento al sistema heurístico de solución en ambiente real.
M.C. Guadalupe Castilla Valdez	Análisis del estado del arte en hibridación de métodos heurísticos. Desarrollo de un algoritmo híbrido para BPP.

5. Manuales y mantenimiento

Para el sistema de solución de RosLoP se elaboraron los correspondientes manuales técnico y del usuario. Por razones de confiabilidad y de extensión no se presentan en este documento. Dada la naturaleza compleja del problema, después de su implantación total en 2008, en 2009 se dieron mantenimientos periódicos para asegurar la robustez del sistema. A la fecha, debido a una reestructuración de CONTAL, por fusión, han aparecido nuevos requerimientos que anteriormente no fueron considerados (ver hoja con correo).

6. Reporte Técnico

DEFINICIONES

Aquí se describe la terminología necesaria para el desarrollo de este trabajo.

COMPLEJIDAD DE LOS PROBLEMAS

En los problemas de tipo combinatorio el método tradicional para encontrar una solución óptima es realizar una búsqueda exhaustiva en el conjunto de soluciones posibles, es decir generar todas las configuraciones factibles, calcular su costo y elegir aquella que ofrezca mejores resultados [Herrera, 2006]. Esta metodología no es eficiente debido al crecimiento exponencial del tiempo de cálculo en función de varios factores como la complejidad de la instancia del problema. Stephen Cook en 1971 [Garey, 1979] propuso los fundamentos para lo que hoy se conoce como la teoría NP-Complejidad.

PROBLEMAS DE DECISIÓN Y ANÁLISIS DE ALGORITMOS

Un problema de decisión $\pi = (D, Y)$ es una pareja formada por un conjunto de instancias D , las cuales se obtienen a partir de una instancia genérica que se especifica en términos de varios componentes: conjuntos, funciones, números, etc.; y un subconjunto de instancias-sí $Y \subseteq D$. Una instancia $i \in Y$, si y sólo si la respuesta a la cuestión del problema es sí para esa instancia.

Un problema de decisión se asocia con un lenguaje formal usando algún mecanismo de codificación y un algoritmo con una máquina de Turing. Un algoritmo se dice que resuelve un problema de decisión si y sólo si el lenguaje aceptado por la máquina de Turing es el

subconjunto de todas las cadenas asociadas con las instancias-sí del problema. En las siguientes definiciones, si no se señala otra cosa, la palabra problema se usa como sinónimo de problema de decisión.

Clase P

Es el conjunto de todos los problemas de decisión que pueden ser resueltos en tiempo polinomial por un algoritmo determinista. A los problemas que pertenecen a esta clase se les denomina tratables.

PROBLEMAS INTRATABLES

Son todos los problemas de decisión para los que no existe un algoritmo determinista de tiempo polinomial que los resuelva, es decir, son todos los problemas comprendidos en P^c .

CLASE NP

La clase NP es el conjunto de todos los problemas de decisión que se pueden verificar en tiempo polinomial con un algoritmo no determinista.

RELACIÓN ENTRE P Y NP

Como toda máquina determinista es un caso particular de una máquina no determinista, se tiene entonces que $P \subseteq NP$.

TRANSFORMACIÓN POLINOMIAL

Se dice que un problema de decisión $\pi_1 = (D_1, Y_1)$ se puede transformar polinomialmente en el problema de decisión $\pi_2 = (D_2, Y_2)$, si y sólo si existe una función $f: D_1 \rightarrow D_2$ que satisface las siguientes dos condiciones:

1. f es computable con un algoritmo determinista de tiempo polinomial.
2. Para toda instancia $i \in D_1$, $i \in Y_1$, si y sólo si $f(i) \in Y_2$.

En tal caso se dice que $\pi_1 \alpha_p \pi_2$.

PROBLEMAS NP-COMPLETOS Y NP-DUROS

Un problema π es *NP-Completo* si y sólo si, $\pi \in NP$ y $\forall \phi \in NP, \phi \leq_p \pi$. Al conjunto de todos los problemas *NP-Completo* se les denota por *NPC*. Un problema de optimización se dice que es *NP-Duro*, si y sólo si, su versión de decisión es *NP-Completo*. Para probar que $\pi \in NPC$, es suficiente y necesario probar que $\pi \in NP$ y que existe $\pi^* \in NPC$ tal que $\pi^* \alpha_p \pi$. Una propiedad importante de los problemas *NP-Completo* es que si $\pi \in NPC$, entonces $\pi \in P$, si y sólo si $P=NP$.

PROBLEMA DE RUTAS DE VEHÍCULOS Y SU VARIANTE CON VENTANAS DE TIEMPO

El ruteo de vehículos ha sido de gran interés por la comunidad científica en los últimos cincuenta años. Sin embargo, aún quedan preguntas por responder debido a su complejidad [Thangiah, 2003]. VRP, definido por [Dantzig, 1959], es un clásico problema de optimización combinatoria. Consiste en satisfacer un conjunto de clientes usando una flota de vehículos, respetando restricciones de vehículos, clientes, etc. Dentro de las variables más importantes del VRP se encuentra el VRPTW [Kallehauge, 2006]. Este problema consiste en encontrar la ruta óptima con una flota de vehículos desde un almacén a un número de clientes que deben ser visitados dentro de un intervalo de tiempo determinado, llamado ventana de tiempo. Un dígrafo $G = (V, A, c, t, a, b, d, q)$ restringido por tiempo y capacidad es definido por un conjunto de nodos $V = V_* \cup \{0, n+1\}$, donde $V_* = \{1, \dots, n\}$ es un conjunto de nodos cliente, y los nodos 0 y $n+1$ son el almacén inicial y el almacén final respectivamente.

El conjunto de arcos $A = A_* \cup \delta^+(0) \cup \delta(n+1)$, donde $A_* = A(V_*)$ es el conjunto de arcos tal que $(i, j) \in V_*$, $\delta^+(0) = \{(0, i) \mid i \in V_*\}$ es el conjunto de arcos dejando el almacén inicial, y $\delta(n+1) = \{(i, n+1) \mid i \in V_*\}$ es el conjunto de arcos que entran al almacén final. Cada arco (i, j) tiene un costo $c_{ij} \in \mathbb{Z}^A$, tal que $c_{ij} \leq c_{ik} + c_{kj}$ y tiene una duración de $t_{ij} \in \mathbb{N}^A$, tal que $t_{ij} \leq t_{ik} + t_{kj}$, para $i, j, k \in V$.

Los tiempos de inicio y final de las ventanas de tiempo en los nodos son denotados por $a, b \in \{\mathbb{Z}_+ \cup \{+\infty\}\}^V$, donde $a_0 = a_{n+1} = 0$, $b_0 = b_{n+1} = +\infty$, $a_i \geq t_{0i}$, y $b_i \geq a_i$ para $i \in V_*$ y $b_j \geq a_j$

+ t_{ij} para $(i, j) \in A^*$. Cada nodo i tiene una demanda $d_i \in \mathbb{Z}_+^V$, donde $d_0 = d_{n+1} = 0$, y una capacidad de carga $q \in \mathbb{Z}_+$ donde $q \geq d_i$ para $i \in V^*$ y $q \geq d_i + d_j$ para $(i, j) \in A^*$.

Para cada camino $P = (v_1, \dots, v_k)$ en G , los tiempos de llegada del conjunto de nodos $V(P)$ del camino se representan por el vector $s \in \mathbb{Z}_+^{V(P)}$ definido por: $s_{v_1} = a_{v_1}$, $s_{v_i} = \max_{i=2, \dots, k} \{s_{v_{i-1}} + t_{v_{i-1}v_i}, a_{v_i}\}$. La demanda del camino se representa por $d(V(P))$ y su costo por $c(A(P))$.

Se dice que un camino $P = (v_1, \dots, v_k)$ en G es factible si $s_{v_i} \leq b_{v_i}$ para $i \in V(P)$ y $d(V(P)) \leq q$.

Una ruta factible R desde 0 a $n+1$ en G es definida como $R(0, v_2, \dots, v_{k-1}, n+1)$. Se denota por \mathcal{R} el conjunto de todas las rutas factibles desde 0 a $n+1$ en G . Dado un dígrafo G restringido por tiempo y capacidad, encontrar una ruta con mínimo costo. VRPTW se expresa por la fórmula 1:

$$\min_{R \in \mathcal{R}} \{c(A(R))\} \quad (1)$$

VRPTW es clasificado como un problema multiobjetivo, donde primeramente se intenta minimizar el número de vehículos utilizados y después la distancia recorrida y tiempos de espera requeridos para satisfacer a todos los clientes.

Un grupo de instancias reconocido por la comunidad científica para evaluar los algoritmos que abordan el VRPTW es el benchmark de [Solomon, 1987]. Este benchmark contiene instancias de tamaño 25, 50 y 100 clientes, además de caracterizarse por su distribución geográfica según la tabla 4.

COMPLEJIDAD DEL VRPTW

Siendo el VRP una generalización del problema TSP [Gómez, 2012], puede decirse que el primero posee la dificultad combinatoria del segundo [Kallehauge, 2006]. Suponiendo que la capacidad del vehículo es infinita, un VRP se simplifica al hecho de encontrar un circuito hamiltoniano, lo cual es la definición del TSP. Por otro lado, si la capacidad de los vehículos es finita, de tal forma que es imposible satisfacer a todos los clientes en un solo recorrido, el VRP puede representarse como un TSP Múltiple (MTSP, Multiple Traveling Salesman Problem), una instancia de MTSP puede ser transformada en su equivalente TSP añadiendo al grafo $k-1$ (siendo k el número de rutas) copias adicionales del nodo centro de

embarque y sus arcos incidentes (no existiendo arcos entre los $k-1$ nodos y el centro de embarque) [Gutin, 2002].

Tabla 4. Caracterización de las instancias de Solomon por su distribución geográfica.

Prefijo	Posfijo	Clase	Descripción
C	1	C1	Cientes distribuidos en cúmulos con ventanas de tiempo cortas y horizonte de calendarización corto.
C	2	C2	Cientes distribuidos en cúmulos con ventanas de tiempo amplias y horizonte de calendarización largo.
R	1	R1	Cientes generados aleatoriamente uniformemente distribuidos con ventanas de tiempo cortas y horizonte de calendarización corto.
R	2	R2	Cientes generados aleatoriamente uniformemente distribuidos con ventanas de tiempo amplias y horizonte de calendarización largo.
RC	1	RC1	Cientes semi-agrupados (combinación de cúmulos y distribución aleatoria) con ventanas de tiempo cortas y horizonte de calendarización corto.
RC	2	RC2	Cientes semi-agrupados (combinación de cúmulos y distribución aleatoria) con ventanas de tiempo amplias y horizonte de calendarización largo.

Dado que TSP es NP-Duro [Garey, 1979], por extensión VRP también pertenece a la clase NP-Duro. Si el VRP clásico es NP-Duro, se induce que toda variante del VRP también lo es, igual complejidad tendría el VRPTW [Kallehauge, 2006]. Por lo tanto, se determina que el VRPTW pertenece a la clase NP-Duro.

ENFOQUES DE SOLUCIÓN

Existen básicamente dos tipos de enfoque en la solución de problemas de optimización: los métodos exactos y los heurísticos. A continuación se brinda una breve explicación de cada uno.

MÉTODOS EXACTOS

Los algoritmos exactos de solución para problemas de optimización más utilizados son la búsqueda exhaustiva, el método simplex de Dantzing, el método elipsoidal de Khachian y el método Karmarkar; el primero se basa en la creación y evaluación de todas las

soluciones factibles, haciéndolo sumamente costoso, en términos computacionales, para problemas complejos, por lo que su aplicación se limita a instancias relativamente simples; el segundo realiza una búsqueda en los puntos adyacentes del cerco convexo del espacio de soluciones factibles del problema; el tercero crea un elipsoide que modifica en cuanto a volumen y posición basándose en las restricciones quebrantadas hasta que el centro del elipsoide converge a la región de soluciones factibles, en ese punto el centro elipsoidal constituye la solución óptima; el último método, considerado actualmente como el más eficiente de los cuatro, inicia en el centro del espacio de soluciones factibles y se mueve por los puntos internos de dicha región hacia la solución óptima.

A pesar de que todos estos métodos obtienen el resultado óptimo, su desempeño resulta inefectivo en problemas del mundo real. Debido a lo anterior se emplea una alternativa como los métodos heurísticos que si bien no siempre encuentran la mejor solución, son de gran ventaja por su bajo costo computacional.

MÉTODOS HEURÍSTICOS

Una heurística es un procedimiento simple, generalmente basado en el sentido común, que permite de manera perspicaz abordar un problema [Díaz, 1996]. Los algoritmos aproximados o heurísticos se pueden clasificar en dos tipos principales: algoritmos constructivos y algoritmos de búsqueda local. Los primeros se basan en generar soluciones desde cero añadiendo componentes a cada solución paso a paso. Un ejemplo bien conocido son las heurísticas voraces. Su gran ventaja es la velocidad: normalmente son muy rápidas y, además, a menudo devuelven soluciones razonablemente buenas. Sin embargo, no puede garantizarse que dichas soluciones sean óptimas con respecto a pequeños cambios a nivel local.

En consecuencia, una mejora típica es refinar la solución obtenida por la heurística voraz utilizando una búsqueda local. Los algoritmos de búsqueda local intentan repetidamente mejorar la solución actual con movimientos a soluciones vecinas. El caso más simple son los algoritmos de mejora iterativos.

Desafortunadamente, los algoritmos de mejora iterativos pueden estancarse en soluciones de baja calidad. Para permitir una mejora adicional en la calidad de las soluciones, la investigación en este campo en las últimas dos décadas ha centrado su atención en el diseño de técnicas de propósito general para guiar la construcción de soluciones y la búsqueda local mediante distintas heurísticas. Estas técnicas se llaman comúnmente metaheurísticas y consisten en conceptos generales empleados para definir métodos heurísticos. Dicho de otra manera, una metaheurística puede verse como un marco de trabajo general referido a algoritmos que puede aplicarse a diversos problemas de optimización combinatoria con pocos cambios significativos. De hecho, las metaheurísticas son ampliamente reconocidas como una de las mejores aproximaciones para abordar problemas de optimización combinatoria.

ESTRATEGIAS DE SOLUCIÓN

ALGORITMOS METAHEURÍSTICOS : SISTEMA DE COLONIA DE HORMIGAS (ACS)

En [Dorigo, 1991] se introduce por primera vez el metaheurístico de optimización basado en Colonia de Hormigas (ACO, Ant Colony Optimization), éste se inspira en el funcionamiento de las hormigas reales, las cuales encuentran el camino de su hormiguero a la comida por medio de una sustancia química llamada “feromona”. Las hormigas se mueven aplicando decisiones locales estocásticas que utilizan los rastros de la feromona (se denota por la letra τ) e información heurística (se denota por la letra η).

Los rastros de feromona τ son actualizados cuando una hormiga ha construido la solución, depositando cantidades de sustancia química por el camino que eligió, sin embargo dicho rastro también puede actualizarse después de haber recorrido el trayecto de un nodo a otro. Por otro lado, la información heurística η mide la predilección por recorrer el camino comprendido entre v_i y v_j . Ambos componentes dirigirán la búsqueda de futuras hormigas.

La figura 4 muestra el pseudocódigo de ACS general, en donde además de las actividades propias de las hormigas de recorrer los caminos; el ACS incluye dos procedimientos adicionales: evaporación y acciones demonio. La evaporación consiste en que la feromona

depositada disminuya con el tiempo. Esto se dirige a evitar una rápida convergencia y favorecer la exploración de nuevas áreas del espacio de solución. Mientras que las acciones demonio son opcionales, siendo un ejemplo de ellas el procedimiento de búsqueda local [Glover, 2003].

Figura 4. Pseudocódigo de algoritmo metaheurístico de ACS general [Glover, 2003]

1	Procedimiento de metaheurístico ACS
2	Agenda de Hormigas
3	Administrar_actividad_hormigas(τ, η, β)
4	Evaporar_feromona (τ)
5	Acciones_demonio () { opcionales}
6	Fin Agenda de Hormigas.
7	Fin Procedimiento de metaheurístico ACS

En [Dorigo, 1996] se presenta una mejora para ACO, llamada Sistema de Colonia de Hormigas (ACS, Ant Colony System), ver figura 1, cuyas principales diferencias con su antecesor son:

- se modifica la regla de selección de clientes de tal manera que la importancia relativa de la información heurística denotada por β proporciona una manera directa de balance entre la exploración de nuevos arcos y la explotación a priori del conocimiento acumulado del problema.
- la regla de actualización global se aplica de forma general, también para aquellos arcos que pertenecen al mejor *tour* de la hormiga.
- mientras que las hormigas construyen una solución, es aplicada una regla de actualización local de la feromona.

ALGORITMOS GENÉTICOS

Método de búsqueda que imita la teoría de la evolución biológica de Darwin para la resolución de problemas. Para ello, se parte de una población inicial de la cual se seleccionan los individuos más capacitados para luego reproducirlos y mutarlos para finalmente obtener la siguiente generación de individuos que estarán más adaptados que la anterior generación [Arranz, 2010].

Entre los operadores para este tipo de algoritmos se encuentran:

- Selección: elegir los individuos más capacitados para que éstos sean los que se reproduzcan de acuerdo con la teoría de Darwin, en la cual los más capacitados son los que deben sobrevivir y crear una nueva descendencia más facultada. Por lo tanto, una vez evaluado cada cromosoma y obtenida su puntuación, se crea la nueva población teniendo en cuenta los rasgos de los mejores individuos de la población actual.
- Cruza: consiste en el intercambio de material genético entre dos cromosomas. El objetivo del cruce es conseguir que el descendiente mejore la aptitud de sus padres. Algunos ejemplos de este operador son:
 - De un punto: Se copia la información genética de uno de los padres desde el inicio hasta el punto de cruce y el resto se copia del otro progenitor.
 - Uniforme: cada gen de la descendencia se crea copiando el correspondiente gen de uno de los dos padres, escogido de acuerdo a una "máscara de cruce" generada aleatoriamente. Cuando existe un 1 en la "máscara de cruce", el gen es copiado del primer padre, mientras que cuando exista un 0 en la "máscara de cruce", el gen se copia del segundo padre.
- Mutación: consiste en modificar ciertos genes de forma aleatoria con una probabilidad determinada.

BÚSQUEDA LOCAL

[Herrera, 2006] sostiene que a pesar de lo sofisticadas y aparentemente adecuadas que pueden resultar algunas metaheurísticas constructivas como ACS o los algoritmos genéticos; es altamente efectivo alternar una heurística de búsqueda local a partir de la última o mejor solución producida, en vez de ejecutar iterativamente sólo el método constructivo; por lo tanto, la posibilidad de añadir una búsqueda local resulta una alternativa prometedora. Tal tipo de mecanismos forman una clase general de heurísticas aproximadas o subóptimas, basadas en el concepto de mejora iterativa mediante la exploración de vecindades. La generación de un vecino se logra a través de un esquema κ que cambia un atributo o combina características de la solución actual.

La búsqueda local efectúa una exploración miope, pues sólo acepta soluciones secuenciales originadas por su diseño; la Figura 5 generaliza este concepto, sea κ el esquema de vecindad, s una solución factible y χ es la función de aptitud de la solución.

Figura 5. Procedimiento de Búsqueda Local

Procedimiento <i>Búsqueda Local</i> ()	
1	Crear solución factible s
2	Sea K el esquema de vecindad.
3	Repeat
4	Buscar en el vecindario, $K(s)$
5	if se encuentran s' con $f(s') < f(s)$ then
6	$s \leftarrow s'$
7	Endif
8	until no se encuentren soluciones s' factibles con menor costo

BÚSQUEDA K -OPT

Para introducir mayor flexibilidad al modificar un ciclo hamiltoniano, se puede considerar dividirlo en k partes, en lugar de dos, y combinar los caminos resultantes de la mejor manera posible. Se le llama movimiento k -opt a tal modificación.

Al aumentar k de igual manera aumenta el tamaño del entorno y el número de posibilidades a examinar en el movimiento, tanto por las posibles combinaciones para eliminar las aristas del ciclo, como por la reconstrucción posterior [Martí, 2006].

ANÁLISIS DE DATOS

Existen dos enfoques principales para el estudio visual de los datos y el análisis de la información obtenida. Esto se lleva a cabo a través de las técnicas exploratorias o bien con técnicas confirmatorias [Parra, 2002].

[Vilar, 2005] El objetivo en el primer tipo de técnicas es la organización, representación y descripción de los mismos, así como los pasos necesarios para extraer la información que contienen. Las técnicas más utilizadas son las siguientes:

- Diagrama de tallo y hojas (método semigráfico).
- Tablas resumen de 3 y 5 números (índices descriptivos).
- Gráfico de caja (método gráfico).

Ante un conjunto de datos, el enfoque exploratorio recomienda iniciar con gráficos que permitan visualizar su estructura. Para datos cuantitativos es aconsejable comenzar con un histograma [Sánchez, 1990]. Las técnicas de carácter confirmatorio utilizan estadísticos numéricos de resumen generados a partir del empleo de un modelo, definido *a priori*, para confirmar o no una hipótesis.

MODELOS ESTADÍSTICOS

Un modelo estadístico es una distribución de probabilidad construida para hacer inferencias y tomar decisiones a partir de datos [Davison, 2003]. Del mismo modo puede definirse como una formalización de las relaciones entre variables en forma de ecuaciones matemáticas [Ader, 2008]. Un modelo estadístico es un caso especial del modelo estocástico donde los parámetros son desconocidos, dichos parámetros se estiman a partir de datos empíricos obtenidos a través de procedimientos estadísticos tales como diseño de experimentos o estudios de observación [Merli, 2008].

CLASIFICACIÓN DE LOS MODELOS ESTADÍSTICOS

En la literatura estudiada se presentan diversas clasificaciones de los modelos estadísticos. Por lo tanto aquí son mostradas las siguientes clasificaciones, con el objetivo de identificar las características de dichos modelos.

- Tipos de efectos: La distinción entre efectos está relacionada con la naturaleza de los parámetros desconocidos del modelo.
 - Fijos: El modelo de efectos fijos se aplica a situaciones en las que el experimentador ha sometido al grupo o material analizado a varios factores, cada uno de los cuales le afecta sólo a la media, permaneciendo la "variable respuesta"

con una distribución normal. Este modelo se utiliza cuando el investigador se interesa únicamente por los niveles del factor presentes en el experimento, por lo que cualquier variación observada en las puntuaciones se deberá al error experimental.

- Aleatorios: estos modelos se usan para describir situaciones en que ocurren diferencias incomparables en el material o grupo experimental. El ejemplo más simple es el de estimar la media desconocida de una población compuesta de individuos diferentes y en el que esas diferencias se mezclan con los errores del instrumento de medición. Este modelo se supone cuando el investigador está interesado en una población de niveles, teóricamente infinitos, del factor de estudio, de los que únicamente una muestra al azar (t niveles) están presentes en el experimento.

 - Mixtos: algunos efectos son fijos y otros aleatorios.
- Número de variables de respuesta: como su nombre lo dice, la cantidad de variables que son utilizadas para interpretar el resultado del modelo.
- Univariados: Muestran aisladamente característica por característica, exhiben una única respuesta.

 - Multivariados: Presentan las relaciones entre conjuntos de variables dependientes y los individuos para los cuales se han medido dichas variables. A partir de una muestra de variables observadas en una muestra de elementos de una población, se busca:
 - Resumir la información en pocas variables
 - Analizar dependencias e independencias
 - Agrupar y clasificar

➤ Tipo de modelo obtenido

- Descriptivos: el objetivo de estos modelos es la generación de hipótesis. Los modelos exploratorios son un ejemplo de ellos.
- Analíticos: tienen como finalidad la confirmación de hipótesis. Los modelos predictivos y explicativos (causales) son ejemplos de modelos analíticos.

Como se observa en tabla 5, el análisis causal es una alternativa que brinda explicaciones para dar solución a un problema determinado, éste hace uso de herramientas como diagramas y reglas de asociación para dar solución al problema.

Tabla .5. Clasificación de multivariados por método obtenido.

Técnica	Objetivo	Carácter
Componentes principales Factores comunes Agrupación Correlación	Reducir Reducir Reducir Reducir	Descriptivo
Regresión lineal Regresión logística Análisis discriminante Análisis causal	Relacionar-Predecir Relacionar-Predecir Clasificar Explicar	Analítico

CORRELACIÓN

Número que indica el grado de relación que existe entre dos variables. En términos de probabilidad se expresa que una de las variables se produzca (o no) cuando ocurra (o no ocurra) la otra. La correlación no indica que una variable cause la otra. Dos variables presentan correlación si una de ellas se produce de modo más (o menos) frecuente al producirse la otra. La correlación es positiva si ambas variables tienden a aumentar juntas o a disminuir juntas; es negativa si una variable tiende a aumentar cuando la otra disminuye, y es cero si no hay una conexión evidente [Stassen, 2007].

En el mismo sentido, [Moore, 2005] la expresa como el coeficiente que mide la fuerza y la dirección lineal entre dos variables cuantitativas. Se simboliza con la letra r . Suponer que se tienen datos de dos variables x e y para n individuos. Los valores del primer individuo se denotan por x_1 e y_1 , para el segundo por x_2 e y_2 . Las medias y desviaciones estándar de las dos variables son \bar{x} y s_x para los valores de x , \bar{y} y s_y para los valores de y . La correlación entre x e y es:

$$r = \frac{1}{n-1} \sum \left(\frac{x_i - \bar{x}}{s_x} \right) \left(\frac{y_i - \bar{y}}{s_y} \right) \quad (2.2)$$

ANÁLISIS DE VARIANZA Y FACTORIAL

Desarrollada por R. Fisher, el análisis de varianza es una técnica que permite se realicen inferencias sobre las medias poblacionales, cuando éstas se ven afectadas por diversos factores. El interés al aplicar el análisis de varianza es explicar los valores de la variable dependiente en términos de las condiciones experimentales, considerando un rango de error. Se pretende determinar bajo qué condición experimental difieren las medias de la variable dependiente [Rutherford, 2012].

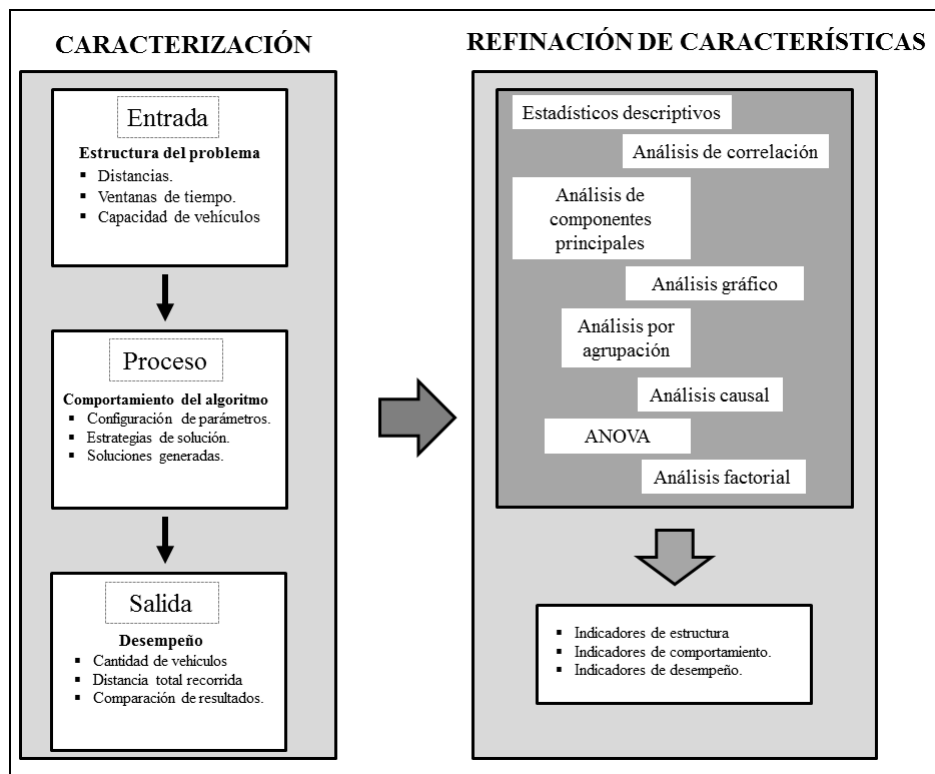
Por otro lado, el análisis factorial es una técnica matemática y estadística basada en correlación. En ella se evalúa un conjunto inicial de atributos de entrada con el objetivo de identificar relaciones entre ellos y para conseguir un número reducido de factores en términos de los valores originales que mejor representen a los datos [Nettleton, 2012].

En este tipo de análisis, toda combinación posible entre los niveles de las variables es evaluada en cada réplica del experimento. Cuando las variables son estructuradas en diseño factorial, se les llama variables cruzadas, mientras que el efecto de una variable se define como el cambio en respuesta por la modificación del nivel de dicha variable [Montgomery, 2008].

METODOLOGÍA

La metodología consta de dos etapas (ver figura 6), una de ellas es la *caracterización*, que tiene como objetivo principal identificar, en cada una de las fases del proceso de optimización los factores de desempeño relevantes y factibles de medición. Estos factores son caracterizados a través de funciones de caracterización (métricas) que proporcionan información útil para describir el desempeño algorítmico.

Figura 6: Esquema de metodología propuesta

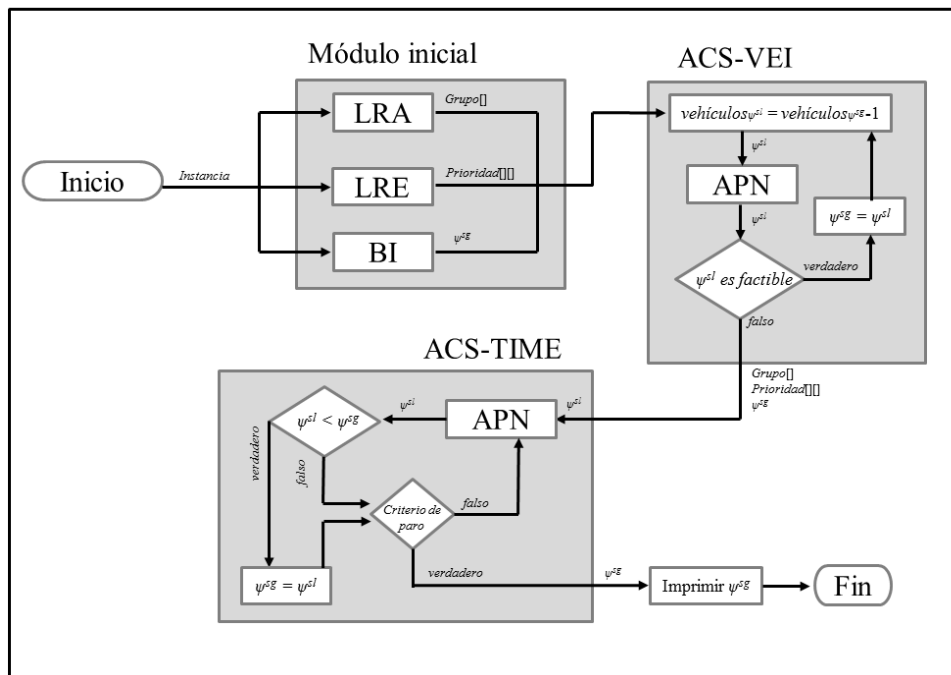


La segunda etapa en figura 6 corresponde a la *refinación de características*, las métricas definidas en la etapa de caracterización son analizadas mediante técnicas exploratorias con la finalidad de descartar variables redundantes o irrelevantes. En caso de ser necesario, nuevas variables son definidas mediante el uso de técnicas de análisis multivariado.

CARACTERIZACIÓN DEL COMPORTAMIENTO DEL ALGORITMO

Los algoritmos metaheurísticos incluyen estrategias que tratan de ajustar el proceso de solución al problema de optimización que resuelven. Para estos algoritmos, la medición de funciones de caracterización debe estar ligada directamente con las técnicas heurísticas utilizadas [Quiroz, 2009]. En esta investigación se continuó con lo realizado por [Cámara, 2011], donde se emplea el algoritmo MACS-VRPTW que coordina las actividades de dos colonias [Gambardella, 1999]. El objetivo de la primera colonia, ACS-VEI disminuye el número de vehículos usados, mientras una segunda colonia ACS-TIME optimiza la distancia recorrida encontrada por ACS-VEI. Ambas colonias usan rastros de feromona independiente pero colaboran al compartir una variable de solución global administrada por el MACS-VRPTW. El aspecto general del algoritmo puede observarse en la figura 7.

Figura 7 Esquema general de solución de [Cámara, 2011].



Dentro de ambas colonias se llama al módulo *APN* (Aprendizaje por niveles), el cual se observa en la figura 8 y dentro de dicho modulo se llama a otro modulo llamado *Construcción de $\psi^{iE}_{numgnumh}$* mostrado en la figura 9.

La figura 8 muestra cómo se guarda la información del rastro de la feromona a través del proceso de construcción de la solución parcial, mientras que la figura 9 presenta de forma general como la hormiga construye la solución parcial, además del momento en que se calcula la información heurística, se selecciona el siguiente cliente y se ejecuta la búsqueda local durante el algoritmo.

Figura 8. Modulo APN de esquema general de solución de [Cámara, 2011].

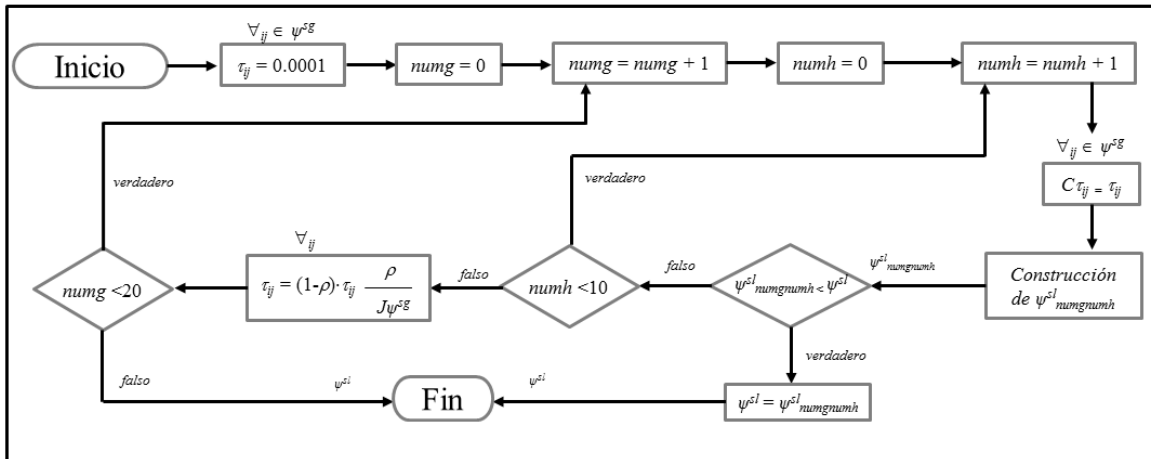
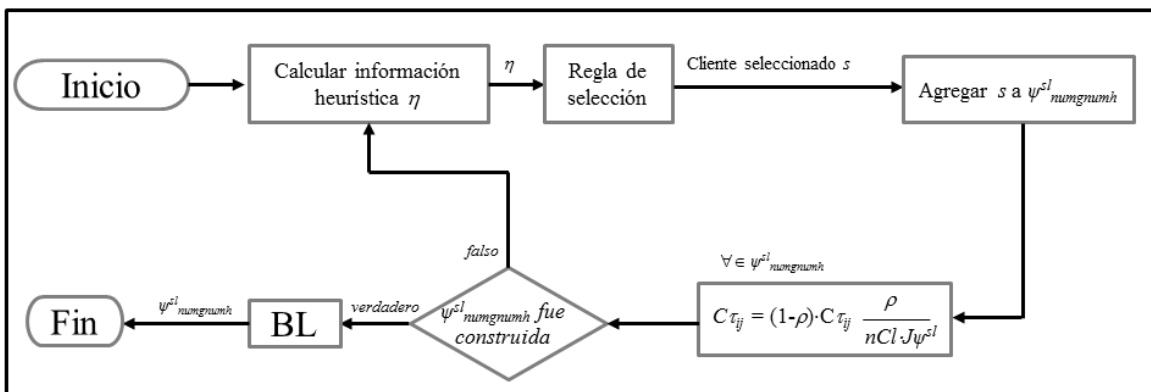
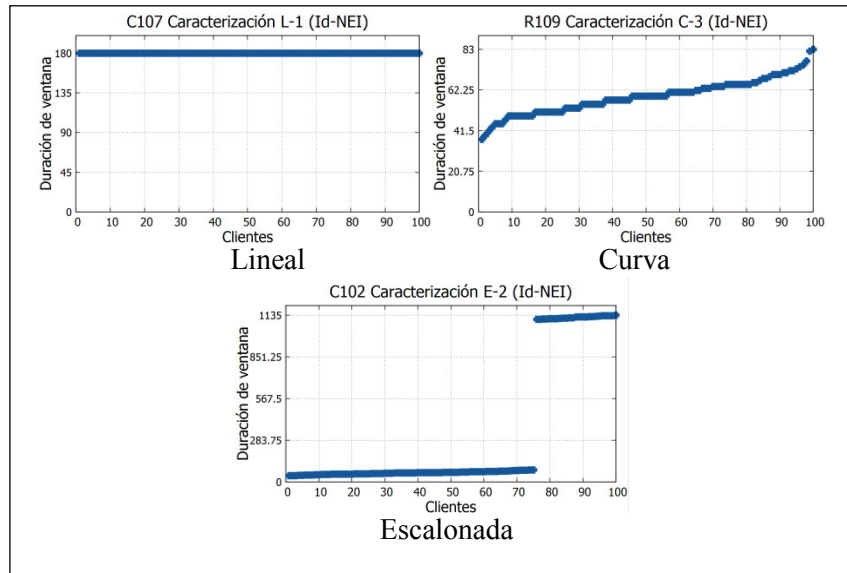


Figura 9. Modulo construcción de $\psi^{sl}_{numgnumh}$ de esquema general de solución de [Cámara, 2011].



Además del algoritmo, las instancias también fueron caracterizadas por [Cámara, 2011] por medio de su distribución de ventanas de tiempo. Dicha caracterización arrojó lo siguiente:

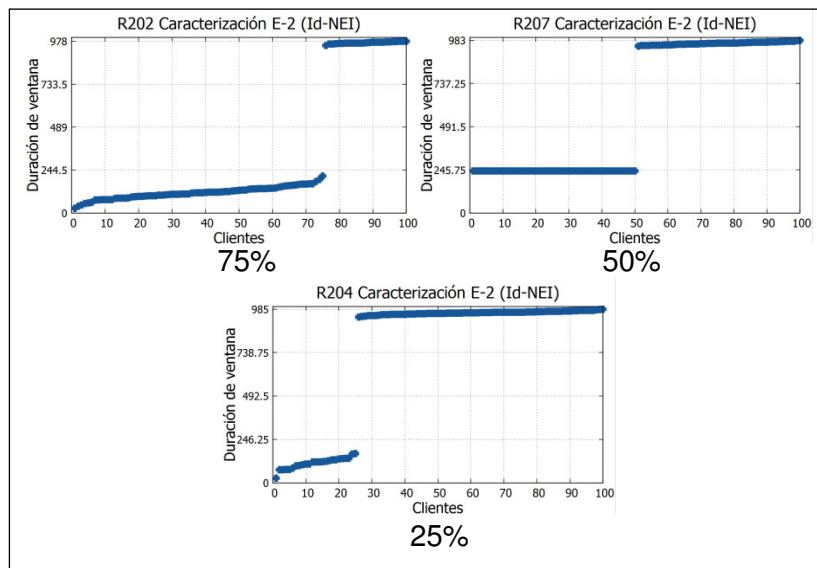
Figura 10. Caracterización de las instancias de [Solomon, 1987] en base a su ventana de tiempo.



[Cámara, 2011] también reporta una sub división de las instancias escalonadas mostrada a continuación:

- Escalonada al 25%, Escalonada al 50% y Escalonada al 75%

Figura 11. Caracterización de las instancias escalonadas de [Cámara, 2011].



BÚSQUEDA INICIAL (BI)

MACS-VRPTW emplea dos enfoques para realizar la construcción de soluciones, la explotación, que hace uso de la información adquirida por las hormigas, y la exploración, un procedimiento semi-aleatorio basado en una distribución de probabilidad proporcional con base en la preeminencia de la feromona e información heurística. Debido a que el enfoque explotativo es sumamente similar a una búsqueda avara, la posibilidad de usarlo como una búsqueda inicial redundante en soluciones de buena calidad. Dicha búsqueda se realiza mediante:

- Búsqueda de Vecinos Cercanos NNS: dado un conjunto S de n puntos en un espacio métrico X , la tarea es pre procesar dichos punto tal que, desde un punto $q \in X$, sea obtenido el punto más cercano a q [Arya,1994].

HEURISTICAS DE CARACTERIZACIÓN

Como se observa en figura 7, son utilizadas diversas heurísticas para auxiliar en el proceso de selección de clientes.

LISTA RESTRINGIDA ADAPTATIVA (LRA)

De acuerdo con [Dorigo, 1996], el uso de una Lista Restringida de Candidatos por parte de las hormigas en el proceso constructivo, resulta ventajoso debido a la condición restrictiva en la creación de soluciones factibles y las distintas posibilidades de distribución de los clientes. Una Lista Restringida de Candidatos limita la población global del momento en el proceso de selección del siguiente nodo, a un subconjunto que cumple con cierto criterio previamente especificado. Debido a las variaciones de la densidad geográfica en diferentes puntos del espacio de una misma instancia, la definición de regiones con mínima distancia entre los puntos pertenecientes a dicha región permite restringir la selección del siguiente punto de visita. Para construir dicha lista fueron utilizados algoritmos basados en técnicas de agrupación.

[Herrera, 2006] en su tesis define la agrupación jerárquica a través de los siguientes seis pasos:

- Se genera un árbol de expansión mínimo (MST) que incluya todos los clientes y el almacén de la instancia. Este MST es un sub grafo que conecta todos los clientes a un costo mínimo.
- Desviación estándar σ es calculada de tc (costo de árbol), donde cada tc_{ij} es la distancia del cliente i al cliente j , $(i,j) \in MST$.
- Porcentaje de variabilidad θ de los costos asociados de cada arco perteneciente a MST es calculado por la fórmula 2, la cual normaliza σ

$$\theta = \frac{\sigma}{2(\max\{tc_{ij}\} - \min\{tc_{ij}\})}, (i,j) \in MST \quad (2)$$

En caso que $\theta < 0.1$ (muy baja variabilidad), la localización de los clientes de la instancia se aproxima a una distribución uniforme, entonces la población constituye un solo grupo. En sentido contrario si $\theta \geq 0.1$, esto revela la posible existencia de regiones de diferente densidad.

- La formación del conjunto de grupos H se lleva a cabo si se satisface la regla de agrupamiento $\theta \geq 0.1$, en caso contrario se considera que los clientes pertenecen a un solo grupo $|H| = 1$.
- Satisfecha la regla de agrupamiento, cada cliente comienza en su propio grupo; pares de grupos son unidos iterativamente con objetivo de encontrar un conjunto de clientes H apropiado, basándose para dicho proceso en un umbral de aceptación ω . El valor inicial para ω se calcula con la ecuación 3

$$\omega = 2 \cdot \max\{tc_{ij}\}, (i,j) \in MST \quad (3)$$

Para cada par de grupos (h_i, h_j) , donde $h_i, h_j \in H$, se calcula la distancia de Mahalanobis dM .

Esta se usa para la regla: si $dM(h_i, h_j) < \omega$ y $h_i \neq h_j$, entonces $h_i \cup h_j$. Dicha regla determina si

dos grupos son mezclados en uno solo.

- La formación de grupos continua mientras operaciones de mezcla existan con el mismo valor de ω , en caso contrario ω es modificado por $\omega = \omega \cdot (1 + \theta)$, finalizando si no hubo operaciones de mezcla con este nuevo valor de ω .

LISTA RESTRINGIDA EXTENDIDA (LRE)

Esta heurística se ejecuta sólo si la distribución de ventana de tiempo de la instancia resulta escalonada. Para este tipo de instancias, se lleva a cabo una clasificación de clientes, tal que; cuando las hormigas hagan la búsqueda de una nueva solución, los clientes con ventana de tiempo corta tengan mayor prioridad de visita. [Cámara, 2011] define la heurística mediante los siguiente cinco pasos:

- Calcular la longitud de ventana de tiempo para cada cliente de la instancia. Ver fórmula 4.

$$tw_i = b_i - a_i \quad (4.3)$$

donde tw_i es la longitud de ventana de tiempo del cliente i , a_i y b_i , son inicio y fin de ventana respectivamente.

- Calcular media (μ) (fórmula 5) y la desviación estándar (σ) (fórmula 6) de la duración de las ventanas de tiempo de la instancia.

$$\mu = \sum_{i=1}^{|V|} tw_i \quad (5)$$

$$\sigma = \sqrt{\frac{1}{|V|} \cdot \sum_{i=1}^{|V|} (tw_i - \mu)^2} \quad (6)$$

- En caso de que fórmula 7 sea verdadera, se identifica a la instancia como escalonada, por lo que procede con el siguiente paso. En caso contrario finaliza el algoritmo.

$$\sigma > \mu \quad (7)$$

- La importancia de cada cliente se denota por vt_i . Dicha importancia se define por la fórmula (8).

$$vt_i = \begin{cases} 1, & \text{si } tw_i < \mu \\ 0, & \text{otro caso} \end{cases} \quad (8)$$

Se asume con importancia alta a un cliente cuyo tamaño de ventana sea menor a la media (μ), y por lo tanto asignando $vt_i = 1$. En caso contrario, el cliente se asume con importancia baja y se asigna con $vt_i = 0$.

- Determinar $priority_{lm}$ para cada par (l, m) , donde l es el cliente actual y clientes m y k son clientes aún no asignados a la solución (camino). Ver fórmula (9)

$$priority_{lm} = \begin{cases} 3, & \text{si } vt_l = 1, vt_m = 1, wait_{lm}^1 < \lambda_1 \\ 2, & \text{si } vt_l = 0 \parallel vt_m = 0, wait_{lm}^1 \geq \lambda_1, \\ & wait_{lm}^2 < \lambda_1, de_{lk} < \lambda_2 \\ 0, & \text{otro caso} \end{cases} \quad (9)$$

donde $wait_{lm}^2 = a_m - (a_l + s_l + de_{lk} + s_k)$, $wait_{lm}^1 = (a_m - a_l + s_l) - de_{lm}$, s_l es el tiempo de servicio del cliente l , de_{lm} es la distancia euclidiana entre l y m mientras que λ_1 y λ_2 son umbrales de espera. Los niveles de prioridad dos y tres son únicamente para instancias escalonadas. Nivel tres es para un par de clientes (l, m) ambos con importancia alta ($vt_l = 1$ y $vt_m = 1$) y con un tiempo de espera aceptable $wait_{lm}^1$ delimitado por λ_1 entre los clientes l y m . Por otro lado, los pares de clientes con nivel dos tienen un tiempo de espera aceptable $wait_{lm}^2$ también delimitado por λ_1 , además de una distancia euclidiana aceptable de_{lk} entre cliente l y cliente intermedio k delimitada por λ_2 .

BÚSQUEDA LOCAL (BL)

Para la incorporación de la Búsqueda Local al ACS se eligieron los esquemas de intercambio de ejes: 3-opt [Bock, 1958] y Cross-Exchange [Taillard, 1997], operando sobre una y dos rutas respectivamente, ambos incluyen implícitamente otros operadores más sencillos, el primero contiene por naturaleza al 2-opt [Croes, 1958], el segundo permite el uso de segmentos vacíos y comprende movimientos de tipo 2-opt*, relocalización e

intercambio, haciéndolo sumamente versátil [Potvin, 1995] [Prosser, 1996]. Las Figuras 12 y 13 muestran gráficamente cada uno de los operadores de búsqueda local.

Figura 12 Esquema 3-opt, se muestran las ocho posibles combinaciones que ofrece el operador 3-opt, dicho operador sólo puede actuar sobre una misma ruta.

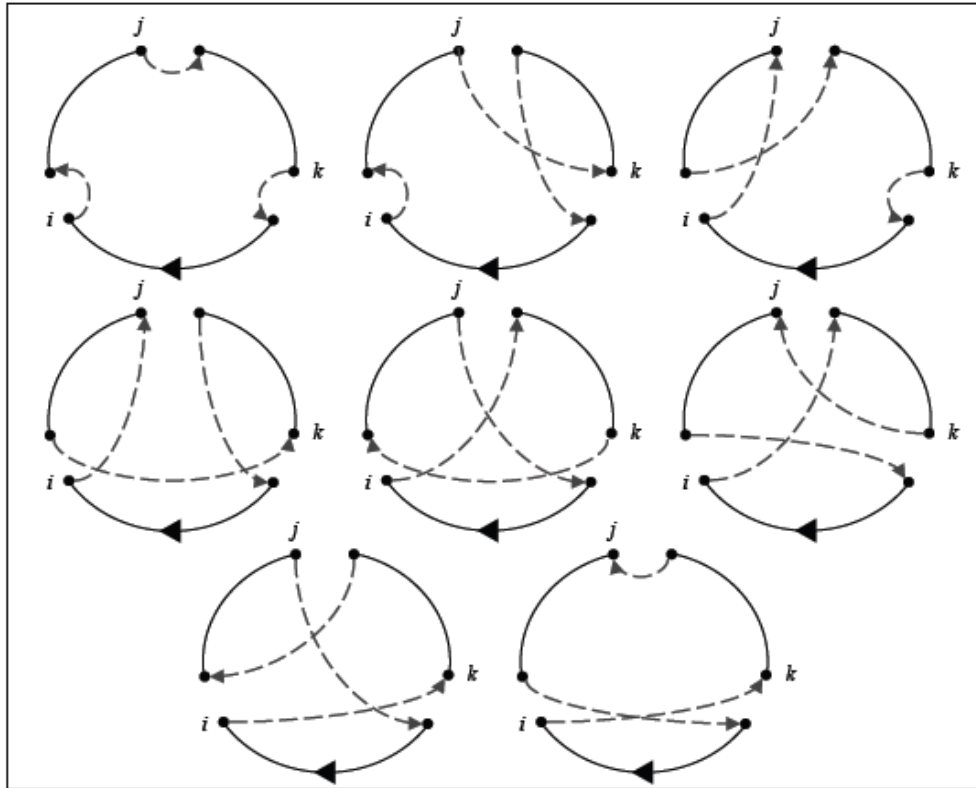
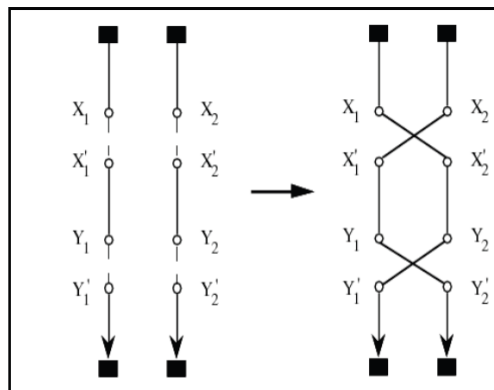


Figura 13..Esquema Cross-Exchange, éste operador actúa sobre dos rutas, donde la longitud de los segmentos que interviene de ambas rutas puede variar desde cero hasta tres nodos, este enfoque añade una gran versatilidad al esquema.



APRENDIZAJE POR NIVELES (APN)

[Arrañaga, 2007] propone una heurística de actualización de valores para la tabla de feromonas utilizadas por el ACS, esta es parte de la estrategia de actualización de valores del algoritmo DQL propuesto por [Mariano, 2001] donde se observan dos niveles de conocimiento por parte del algoritmo.

- En un primer nivel se tienen los valores originales de la tabla de feromona que solamente contienen las mejores soluciones obtenidas por las hormigas y sus valores son modificados con la actualización global de feromona natural del ACS, ver fórmulas (10) y (11).

$$\tau_{ij} = (1 - \rho) * \tau_{ij} \quad (10)$$

Posteriormente procede actualizar las $n - 1$ parejas de la solución en τ .

$$\tau_{ab} = \frac{\rho}{fDist} + \tau_{ab} \quad (11)$$

Donde (a, b) es una pareja de cliente a con cliente b , la cual pertenece a la solución actual, $fDist$ es la distancia recorrida por la mejor solución.

- En un segundo nivel se tiene una copia de la tabla que contiene los valores locales de feromona ya que solo es modificada por la actualización local de las hormigas. Ver fórmula (12)

$$C\tau_{ab} = (1 - \rho) * C\tau_{ab} + \frac{\rho}{nCl * fDist} \quad (12)$$

Donde $C\tau$ una copia de τ , a es cliente actual, b es cliente siguiente, $fDist$ es la distancia recorrida en la mejor solución, y nCl es número de clientes visitados hasta el momento por la solución actual.

De igual forma esta heurística conserva los valores de τ a través de las generaciones, lo cual permite a las hormigas iniciar desde un estado previo de búsqueda y no desde un estado inicial.

SISTEMAS DE COLONIA DE HORMIGAS

SELECCIÓN DE CANDIDATOS

Una vez que la membresía de cada cliente a un grupo fue asignada según su distribución topológica (LRA) y la prioridad de cada cliente con respecto del resto de clientes fue definida según la distribución de ventana de tiempo. Se procede a la selección de candidatos según la regla de selección en fórmula (13). La cual actualiza tanto la tabla de feromonas τ como la información heurística η .

$$s = \begin{cases} \operatorname{argmax} \{ \tau_{ij} \cdot \eta_{ij}^\beta \}, & \text{si } (q \leq q_0) \\ S, & \text{en otro caso} \end{cases} \quad (13)$$

$$S = f(\operatorname{prob}(i, j)) \text{ donde } (\operatorname{prob}(i, j)) = \frac{\tau_{ij} \cdot \eta_{ij}^\beta}{\sum_{k \in N(i)} \tau_{ik} \cdot \eta_{ik}^\beta} \quad (14)$$

Donde β es la importancia relativa de la información heurística, $N(i)$ es el conjunto de clientes vecinos disponibles y S es la variable aleatoria dada por la fórmula (14), la cual determina la probabilidad con que una hormiga en cliente i se dirige al cliente j . ACS-VEI y ACS-TIME utilizan el mismo proceso de selección de candidatos, sin embargo cada una mantiene su tabla de feromonas de manera independiente.

DEFINICIÓN DEL CÁLCULO DE LA INFORMACIÓN HEURÍSTICA

Los valores de la información heurística η son asignados según la fórmula (15). Si el cliente fuente v_i y el cliente destino v_j pertenecen a diferente grupo ($h_l \neq h_m$) se aplica un factor de corrección dado por el tamaño del conjunto de clientes V y el tamaño del conjunto de grupos H , de lo contrario se aplica la prioridad establecida por la longitud de ventana de tiempo.

$$\eta_{ij} = \begin{cases} \eta_{ij} \cdot |H| / |V|, & \text{si } h_i \neq h_m \& v_i \in h_i \& v_j \in h_m \\ \eta_{ij} \cdot \text{priority}_{ij}, & \text{otro caso} \end{cases} \quad (15)$$

Donde $h_i, h_j \in H$ (conjunto de grupos), $v_i, v_j \in V$ (conjunto de clientes) y *priority* denota la información de ventana de tiempo.

MODIFICACIONES FINALES

Aquí se describen al algoritmo MACS-VRPTW. Para ello, se analizaron las instancias, y el algoritmo.

Tabla 6.. Modificaciones realizadas al algoritmo MACS-VRPTW

Elemento	Modificaciones
LRE: Lista Restringida Extendida	<ul style="list-style-type: none"> - Recalculo después de selección de cliente. - Activación por intervalos - Considerar Inicio de ventana para generar información de ventanas.
LRA: Lista Restringida Adaptativa	<ul style="list-style-type: none"> - Agrupación semi-jerárquica. - Agrupación alterna.
BI: Búsqueda Inicial	<ul style="list-style-type: none"> - Búsqueda aleatoria. - Método Lin-Kernighan.
BL: Búsqueda Local	<ul style="list-style-type: none"> - Método Lin-Kernighan.
APN: Aprendizaje por niveles	<ul style="list-style-type: none"> - Desactivado. - Tabla única.

La tabla 6 presenta información general de lo que ha sido modificado en el algoritmo MACS-VRPTW.

LISTA RESTRINGIDA EXTENDIDA (LRE)

En este punto es necesario recordar que esta heurística provee información de las ventanas de tiempo contenida en las instancias. Previo a este trabajo se utilizaron las unidades estadísticas media y desviación estándar para caracterizar las instancias; lo cual generaba confusión entre curvas y escalonadas, además de únicamente calcularse esta información al principio del algoritmo. A continuación se describe lo propuesto para esta heurística.

CARACTERIZACIÓN POR INTERVALOS

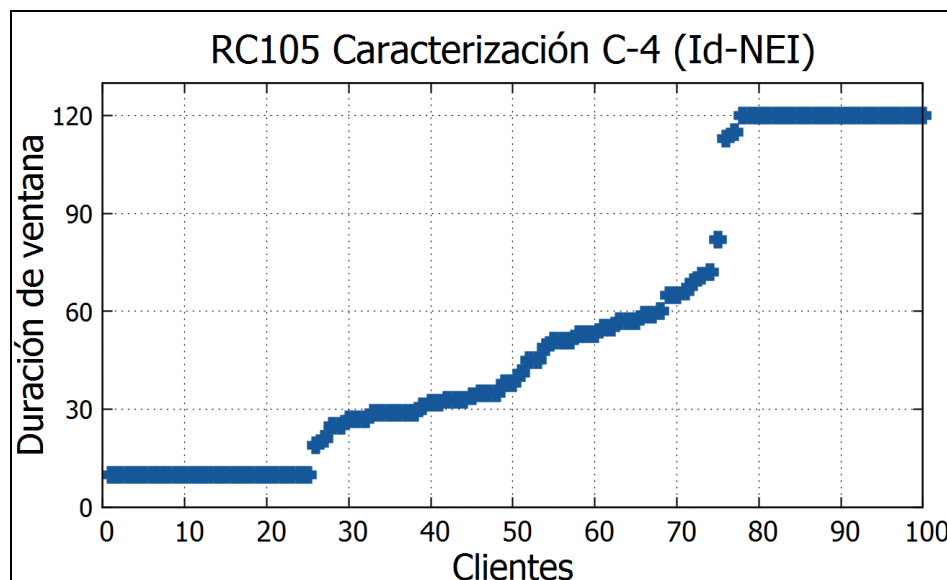
En la versión de [Cámara, 2011], el tipo al que una instancia pertenecía era determinado en base a la observación de su gráfica, siendo la activación de la heurística el único aspecto automático. Las instancias ahora son caracterizadas en base al valor de la variable NEI (número de intervalos no vacíos), ver tabla 7.

Tabla 7. Caracterización por intervalos según ventana de tiempo.

Tipo	Id	Descripción
Lineal	L	Todas las ventanas se encuentran en un sólo rango, NEI = 1.
Escalonada	E	Las ventanas pueden encontrarse en tan sólo 2 rangos diferentes, NEI = 2.
Curva	C	Las ventanas se encuentran en 3 o más rangos diferentes, NEI > 2.

Si la figura 14 mostrara una instancia en donde NEI = 2, dicha instancia se consideraría escalonada, y por lo tanto se procedería con el llenado de la tabla prioridad de la heurística LRE.

Figura 14. Gráfica de ventanas de tiempo de instancia RC105. Existe al menos una ventana de tiempo en cada uno de los rangos, por lo que NEI = 4, y por ende no se ejecuta la heurística LRE.



A continuación se describe el algoritmo para identificar las instancias de acuerdo a esta caracterización.

ACTIVACIÓN POR INTERVALOS

Los siguientes pasos sustituyen del segundo al quinto paso del algoritmo de LRE.

- Definir cuatro intervalos de tiempo: $Int_k = ((k-1) \cdot IntLength, k \cdot IntLength)$, $k = 1, \dots, 4$. Donde $IntLength$ denota la longitud del intervalo. Ver fórmula 16.

$$IntLength = \frac{\max(tw_i)}{4} \quad (16)$$

- Para cada intervalo Int_k , determinar el número de clientes O_k cuya ventana se encuentre en dicho intervalo, con el objetivo de clasificar la instancia según su distribución de ventana de tiempo. Ver fórmulas (17) y (18).

$$O_k = \sum_{i=1}^{|V|} g(i, k) \quad (17)$$

donde

$$g(i, k) = \begin{cases} 1, & \text{si } tw_i \in Int_k \\ 0, & \text{otro caso} \end{cases} \quad (18)$$

- Si el número de intervalos no vacíos NEI es igual a 2, ver fórmula (19) y (20), entonces llenar la tabla de prioridad de MACS-VRPTW. Esta tabla de prioridad de tamaño $[n, n]$, donde n es el número de clientes, se construye de acuerdo a los valores de prioridad de visita entre los clientes.

$$NEI = \sum_{k=1}^4 g(k) \quad (19)$$

$$g(k) = \begin{cases} 1, & \text{si } O_k > 0 \\ 0, & \text{otro caso} \end{cases} \quad (20)$$

- Determinar $priority_{lm}$ para cada par (l, m) , donde l es el cliente actual y m y k son clientes aún no asignados a la solución (camino). Ver fórmula (21)

$$priority_{lm} = \begin{cases} 3, & \text{si } NEI = 2, tw_l \& tw_m \in Int_{\min\{i|O_i=1, i=1,2,3\}}, wait_{lm}^1 < \lambda_1 \\ 2, & \text{si } NEI = 2, wait_{lm}^1 \geq \lambda_1, wait_{lm}^2 < \lambda_1, de_{lk} < \lambda_2 \\ 0, & \text{otro caso} \end{cases} \quad (21)$$

donde $wait_{lm}^2 = a_m - (a_l + s_l + de_{lk} + s_k)$, $wait_{lm}^1 = (a_m - a_l + s_l) - de_{lm}$, s_l es el tiempo de servicio del cliente l , de_{lm} es la distancia euclidiana entre l y m mientras que λ_1 y λ_2 son umbrales de espera.

Los niveles de prioridad dos y tres son únicamente para instancias escalonadas. Nivel tres es para un par de clientes (l, m) ambos con ventanas de tiempo corta (pertenecientes al primer intervalo) y con un tiempo de espera aceptable $wait_{lm}^1$ delimitado por λ_1 entre los clientes l y m . Por otro lado, los pares de clientes con nivel dos tienen un tiempo de espera aceptable $wait_{lm}^2$ también delimitado por λ_1 , además de una distancia euclidiana aceptable de_{lk} entre cliente l y cliente intermedio k delimitada por λ_2 .

INICIO DE VENTANA

Hasta este punto, únicamente se consideró el tamaño de la ventana de tiempo para realizar el llenado de la tabla prioridad de la heurística LRE. Entonces, se propuso incluir el inicio de ventana. Esto tiene como objetivo que aquellos clientes con ventana corta y cuya apertura se presente de forma temprana tengan la prioridad más alta.

Tomando en cuenta que aquí se utilizó la activación por intervalos, para poder añadir el inicio de ventana fue necesario realizar las siguientes modificaciones.

- Al comienzo del algoritmo, definir cuatro intervalos de inicio de tiempo: $InitInt_k = ((k-1) \cdot InitIntLength, k \cdot InitIntLength)$, $k = 1, \dots, 4$. Donde $InitIntLength$ denota la longitud del intervalo de inicio de tiempo. Ver fórmula 22.

$$InitIntLength = \frac{\max(a_i)}{4} \quad (22)$$

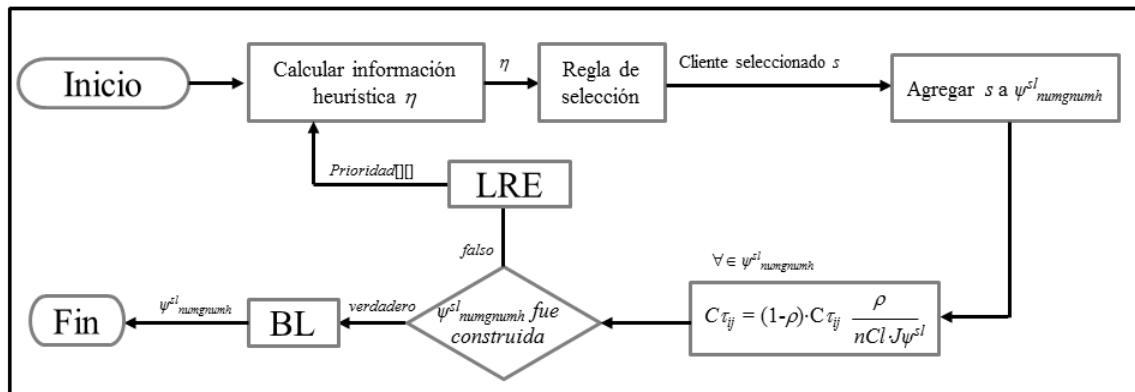
- Determinar $priority_{lm}$ para cada par (l, m) , donde l es el cliente actual y m y k son clientes aún no asignados a la solución (camino). Ver fórmula (23)

$$priority_{lm} = \begin{cases} 7 & , NEI = 2, tw_l \& tw_m \in Int_{\min\{i|O_i=1, i=1,2,3\}}, wait_{lm}^1 < \lambda_1, \\ & , a_l \in InitInt_1, a_k \in InitInt_2 \\ 6 & , NEI = 2, tw_l \& tw_m \in Int_{\min\{i|O_i=1, i=1,2,3\}}, wait_{lm}^1 < \lambda_1, \\ & , a_l \& a_k \in InitInt_2 \\ 5 & , NEI = 2, tw_l \& tw_m \in Int_{\min\{i|O_i=1, i=1,2,3\}}, wait_{lm}^1 < \lambda_1, \\ & , a_l \parallel a_k \in InitInt_{\{i|i=3,4\}} \\ 4 & , NEI = 2, wait_{lm}^1 \geq \lambda_1, de_{lk} < \lambda_2, a_l \in InitInt_1, a_k \in InitInt_2 \\ 3 & , NEI = 2, wait_{lm}^1 \geq \lambda_1, de_{lk} < \lambda_2, a_l \& a_k \in InitInt_2 \\ 2 & , NEI = 2, wait_{lm}^1 \geq \lambda_1, de_{lk} < \lambda_2, a_l \parallel a_k \in InitInt_{\{i|i=3,4\}} \\ 0 & , \text{otro caso} \end{cases} \quad (23)$$

LRE ADAPTATIVO

LRE adaptativo representa el recalculo de la información de ventanas de tiempo, esto quiere decir que a medida que el tiempo progresa, el tamaño de la ventana para cada cliente puede verse modificada. La figura 15 muestra el lugar del algoritmo donde se vuelve a calcular la información de ventanas de tiempo es cuando la hormiga construye la solución parcial.

Figura 15 Modulo construcción de $\psi^{sl}_{numgnumh}$ de esquema adaptativo.



LISTA RESTRINGIDA ADAPTATIVA (LRA)

Esta heurística se emplea para extraer información topológica de las instancias, donde a cada cliente se le da una zona geográfica de pertenencia llamada grupo, y así propiciar que sea más probable visitar a otro cliente entre más cercano se encuentre. Previamente se extraía la información mediante el algoritmo de agrupación jerárquica, el cual requería la construcción del árbol de expansión mínimo (MST) de los clientes, sin embargo fue hecho de forma incompleta. Por esta razón fueron propuestos dos algoritmos de agrupación.

AGRUPACIÓN SEMI JERÁRQUICA

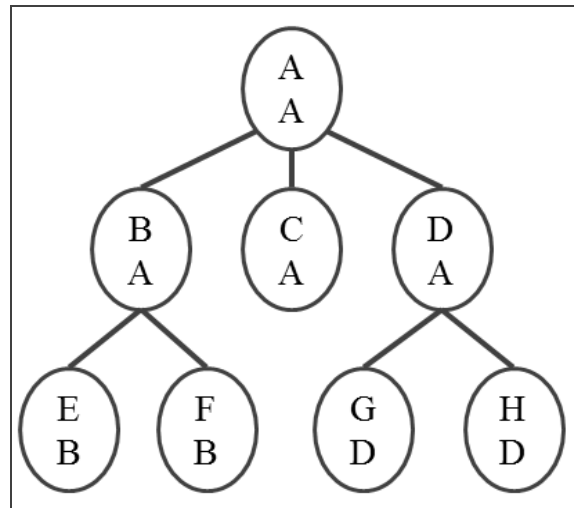
Se utilizó la implementación de la generación del árbol de expansión mínimo (MST) para adaptarla como agrupación semi-jerárquica, el pseudocódigo se puede observar en la figura 16

```
1  G = (V,E)
2  Ordenar_Aristas_MenMay();
3  Altura [n]= {0};
4  Grupo [n]= {desde i=1 hasta n};
5  cont_aristas = 0;
6  k=0;
7  While(cont_aristas<= num_clientes -1)
8     Pequeña = ordenado[k];
9     v1 = Pequeña.0;
10    v2 = Pequeña.1;
11    If(v1.padre != v2.padre)
12       If (altura [v1] == altura [v2])
13           Incrementar altura [v1];
14           Grupo [v2] = v1;
15       Fin de If
16       Else
17           If (altura [v1] > altura [v2])
18               Grupo [v2] = v1;
19           Fin de If
20           Else
21               Grupo [v1] = v2;
22           Fin de Else
23       Fin de Else
24   Fin de If
25   Incrementar k;
26 Fin de While.
```

Figura 16, Algoritmo de agrupación semi-jerárquica. La información que indica el grupo al que pertenece cada nodo es almacenada en el vector llamado "Grupo"

El diagrama de figura 17 ejemplifica el árbol que resulta de la agrupación semi - jerárquica. En cada nodo hay dos letras: la etiqueta de cliente y el grupo al que pertenecen. Únicamente el nodo raíz pertenece a sí mismo. El resto de los clientes pertenecerán al mismo grupo siempre y cuando sean hermanos en el árbol.

Figura 17, Árbol resultante del algoritmo de agrupación semi-jerárquica.



AGRUPACIÓN ALTERNA

Como otra opción a los algoritmos de agrupación anteriormente mencionados, se diseñó la agrupación alterna.

Figura 18, Algoritmo de agrupación alterna. La información que indica a qué grupo pertenece cada grupo es almacenada en el vector llamado “Grupo” y el número de grupos se almacena en “grupo_actual”.

1	Definir $Deuc[][]$ // distancias euclidianas entre clientes
2	Definir $cmax$ // capacidad máxima de distancia para un grupo.
3	Definir $dmax$ // distancia máxima para que dos clientes pertenezcan al mismo grupo.
4	Definir a // cliente inicial para iniciar agrupación.
5	Inicializar $Grupo[] = 0$ // guarda a qué grupo pertenece cada cliente.
6	$Agrupado[]$ // guarda los clientes agrupados.
7	$No_Agrupado[]$ // guarda los clientes no agrupados.
8	Inicializar $grupo_actual = 1$ // grupo actual al que se agrupará siguiente cliente.
9	Inicializar $cliente_previo = a$
10	Inicializar $dacum = 0$ // acumulador de distancia para $grupo_actual$.

```

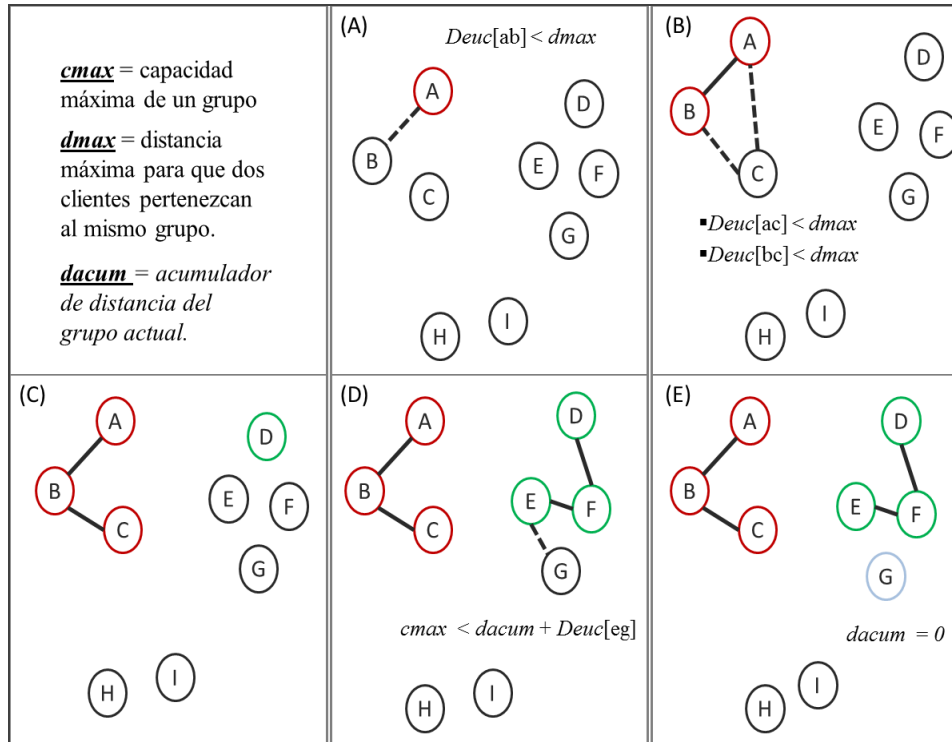
11  Grupo[a] = grupo_actual
12  Agrupado.add (a )
13  No_Agrupado.add ( todos excepto a)
14  While (Agrupado.HaySinAgrupar() )
15  Cliente_siguiete = -1
16      Inicializar fMinD = Float.Maxnum // guarda la menor distancia promedio
17      para clientes del grupo actual respecto de los no agrupados.
18  Fori en No_Agrupado
19      Inicializar fDaux= 0 // acumulador de distancia para clientes del grupo actual
20      respecto de los no agrupados.
21      Inicializar nSameAcum = 0 //acumulador de clientes agrupados al grupo_actual.
22  Forjen Agrupado
23  If (Grupo[j] == grupo_actual)
24      If (dmax>= Deuc[i,j])
25      fDaux+= Deuc[i,j];
26      nSameAcum++;
27      Fin de If
28      Fin de If
29      Fin de For
30  fDaux /= nSameAcum;
31  If (fDaux> 0) // si fDaux = 0 significa que cliente i no puede agruparse a grupo_actual
32  If (fDaux<fMinD)
33      fMinD = fDaux;
34      cliente_siguiete = i;
35      Fin de If
36      Fin de For
37  If (cliente_siguiete != -1)
38      dacum += Deuc[cliente_previo,cliente_siguiete];
39  If (dacum>cmax)
40      dacum = 0;
41      grupo_actual++;
42      Fin de If
43      Grupo[cliente_siguiete] = grupo_actual;
44  Fin de If
45  Else
46      cliente_siguiete = No_Agrupado[0];
47      Grupo[cliente_siguiete] = ++grupo_actual;

```

48	<i>dacum</i> = 0;
49	Fin de Else
50	<i>No_Agrupado.Remove(cliente_siguiete);</i>
51	<i>Agrupado.Add(cliente_siguiete);</i>
52	<i>Cliente_previo = cliente_siguiete;</i>
53	Fin de While

En la figura 19, en el cuadro (A) se comienza por elegir un nodo inicial, el cual pertenecerá al primer grupo. Desde tal nodo seleccionado como inicial, ahora perteneciente al grupo actual, se obtiene el nodo más cercano a él y se le incluye al grupo actual, siempre y cuando d_{max} sea mayor a la distancia euclidiana entre los dos nodos involucrados. Para el cuadro (B), de nuevo se obtiene el nodo sin agrupar más cercano respecto a los nodos del grupo actual □ el inicial y el elegido previamente □ siempre y cuando d_{max} sea mayor a la distancia euclidiana entre los nodos del grupo actual y el nodo más cercano a ellos. En cuadro (C), si respecto de todos nodos del grupo actual, no hay ningún nodo que cumpla con d_{max} , entonces se elige otro nodo --- el primero de los no agrupados--, asignándose al siguiente grupo, reinicializando *dacum*, la cual hasta el momento tenía el valor de la distancia euclidiana de nodo *A* al nodo *B* más la distancia euclidiana del nodo *B* al nodo *C*. Para cuadro (D), otra restricción para asignar un nodo al grupo actual es que la capacidad acumulada del grupo actual más la distancia del nodo más cercano no sea mayor a *cmax*. Para este caso se supone que al agregar el nodo *G*, la capacidad acumulada ha superado la capacidad máxima *cmax*. En el cuadro (E), se asigna el nodo *G* a un grupo distinto, y se reinicializa la capacidad acumulada *dmax* a cero.

Figura 19. Proceso de agrupación alterna.



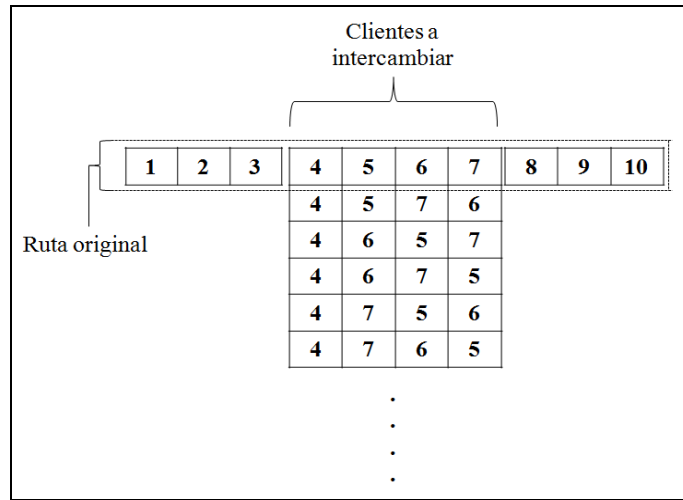
BÚSQUEDA INICIAL (BI) Y LOCAL (BL)

Cámara en [Cámara,2011] realizó una búsqueda inicial mediante el método de vecinos cercanos y una búsqueda local por esquema 3-opt y Cross-Exchange. En este trabajo se realizó lo siguiente:

- Búsqueda Aleatoria: como su nombre lo dice, dado un cliente actual q , elige al azar el cliente siguiente $q1$. La búsqueda sigue si $q1$ fue visitado anteriormente.
- Algoritmo Lin-Kernighan: El algoritmo se especifica en términos de intercambios que pueden convertir una ruta en otra. Dada una ruta factible, el algoritmo realiza intercambios repetidamente con objeto de reducir la longitud de la ruta, hasta que se alcance una ruta para la que cualquier intercambio desde ella no repercuta en mejora [Helsgaun, 2000].

En la figura 20 sólo es mostrado cuando el vértice etiquetado con 1 es el primer vértice, restándolos otros 3 vértices por ser el primer vértice, lo cual hace un total de 24 permutaciones para $k = 2$. El algoritmo de Lin-Kernighan también fue aplicado como búsqueda local dentro de las colonias de hormigas de algoritmo MACS-VRPTW.

Figura 20. Ejemplo de permutaciones en movimiento para un entero k de valor 2.



APRENDIZAJE POR NIVELES (APN)

El algoritmo utilizado por [Cámara, 2011] utiliza dos niveles de conocimiento, uno global y uno local, almacenando cada uno de ellos en estructuras diferentes, τ y $C\tau$ respectivamente. Mientras que los algoritmos propuestos son los siguientes:

- APN desactivado: Consiste en utilizar el primer nivel de conocimiento de aprendizaje por niveles y descartar el segundo nivel, es decir la actualización local.
- Tabla única: En este se almacenan ambos niveles de conocimiento en una única estructura.

Referencias

- [Cano, 2005] I. Cano, I. Litvinchev, R. Palacios and G. Naranjo, “Modeling Vehicle Routing in a Star-Case Transportation Network,” Proc. XVI Congreso Internacional de Computación (CIC), 2005 pp. 373-377.
- [Cruz, 2008] Laura Cruz Reyes, José Francisco Delgado Orta, Juan Javier González Barbosa, José Torres-Jimenez, Héctor Joaquín Fraire Huacuja, Bárbara Abigail Arraño Cruz: An Ant Colony System Algorithm to Solve Routing Problems Applied to the Delivery of Bottled Products. Lectures Notes in Computer Science, Vol. 4994. Springer-Verlag, Berlin Heidelberg New York (2008) 329-338. ISSN 0302-9743, ISBN 978-3-642-02318-7

- [Coffman 2002] Coffman, J.E., Courboubetis, C., Garey, M.R., Johnson, D.S., Shor, P.W., Weber, R.R.: Perfect Packing Theorems and the Average Case Behavior of Optimal and Online Bin Packing, SIAM Review Vol. 44, pp. 95-108, 2002
- [Dantzing, 1959] G. B. Dantzig and R.H. Ramser. "The Truck Dispatching Problem". Management Science 6, 80–91. 1959.
- [Goel, 2006] Goel, A.; Gruhn, V. Solving a Dynamic Real-Life Vehicle Routing Problem. Haasis, H.-D.; Kopfer, H.; Schönberger, J. (Ed.): Operations Research Proceedings 2005, Bremen, Deutschland, 7.9.2005 - 9.9.2005, 367-372. Springer ISBN: 3-540-32537-9
- [Hasle, 2007] Hasle G., O. Kloster, E. J. Nilssen, A. Riise, T. Flatberg,. "Dynamic and Stochastic Vehicle Routing in Practice." Chapter in book Dynamic Fleet Management, series Operations Research/Computer Science Interfaces Series, Springer, Vol. 38, pp 45-68, 2007
- [Herrera, 2006] A. Herrera, "Desarrollo de una Metodología Basada en Heurísticas para la Solución Integral de Problemas de Asignación de Rutas, Horarios y Cargas en el Proceso de Distribución y Entrega de Productos". Masters's thesis, Posgrado en Ciencias de la Computación, Instituto Tecnológico de Ciudad Madero, 2006.
- [Pisinger, 2005] D. Pisinger and S. Ropke, A General Heuristic for Vehicle Routing Problems, tech. report, Dept. of Computer Science, Univ. Copenhagen, 2005.
- [Rangel, 2005] N. Rangel, "Análisis de los Problemas de Asignación de Rutas, Horarios y Cargas en una Distribuidora de Productos," master's thesis, Posgrado en Ciencias de la Computación, Instituto Tecnológico de Ciudad Madero, 2005.
- [Toth, 2001] P. Toth and D. Vigo, "The Vehicle Routing Problem," Proc. SIAM Monographson Discrete Mathematics and Applications, 2002.
- [Vidal, 2001] T Vidal, TG Crainic, M Gendreau, "A Hybrid Genetic Algorithm with Adaptive Diversity Management for a Large Class of Vehicle Routing Problems with Time Windows", Technical Report CIRRELT-2011-61, 2011



SOCIEDAD INDUSTRIAL, S. A. DE C.V.

Tampico, Tam. a 17 de diciembre de 2011

A quien corresponda:

Hago constar que los profesores del Instituto Tecnológico de Ciudad Madero

Dra. Laura Cruz Reyes (Líder del proyecto)
Dra. Claudia Guadalupe Gómez Santillán
Dr. Juan Javier González Barbosa
Dr. Héctor J. Fraire Huacuja
Dr. José Antonio Martínez Flores
M.C. Guadalupe Castilla Valdéz

Realizaron durante el periodo enero 2005 a agosto de 2008 el proyecto:

Una Aplicación para la Solución de los Problemas de Enrutamiento, Asignación de Horarios y Distribución de Cargas

Como producto de este proyecto se obtuvo un software científico que fue incorporado en el sistema logístico de almacenes. La adaptación del software para su implantación en tres tipos de regiones se realizó en el periodo enero a agosto de 2008.

Para el 2012 se tiene contemplando la ampliación del software para dar solución a aspectos complejos derivados de una reestructuración de la empresa.

Aprovecho la ocasión para agradecer al Cuerpo Académico de Optimización Inteligente, dirigido por la Dra. Cruz, por su valiosa colaboración en el desarrollo de este importante proyecto para las embotelladoras del grupo.

Se extiende la presente a petición de los interesados y para su uso ante las autoridades educativas que a ellos convenga en la Ciudad de Tampico, Tamaulipas a los 17 días del mes de diciembre del año 2011.

Cordialmente,

Ing. Basilio González Rocha

Av. Hidalgo 2303 • C.P. 89140 Tampico, Tamaulipas, México
Apartado Postal 664 • C.P. 89000 Tampico, Tamaulipas, México
Teléfono: (833) 241.2500 • Fax: (833) 241.2577



SOCIEDAD INDUSTRIAL, S.A. DE C.V.

Tampico, Tam., a 18 de Septiembre de 2006

A quien corresponda:

Hacemos constar que a partir del 24 de noviembre de 2004, iniciamos en conjunto con el Instituto Tecnológico de Ciudad Madero (ITCM), el desarrollo de un programa de cómputo cuyo propósito es facilitar la logística de distribución de producto desde los Centros de Producción a los Centros de Distribución de nuestro Grupo.

El trabajo de parte del ITCM, consistió en la elaboración de un algoritmo y su programación, que realiza los siguientes cálculos:

1. Programación de viajes con horarios de salida, llegada y maniobra
2. Optimización de la flotilla, con el mínimo número de vehículos para atender la demanda
3. Reducción de los tiempos de espera de los vehículos para ser atendidos por los Centros de Producción y Distribución
4. Distribución de tarimas de producto a bordo de los vehículos de carga para maximizar su capacidad

Actualmente este programa está operando satisfactoriamente en nuestra embotelladora ubicada en la ciudad de Durango.

Posteriormente, se instalará en el resto de las embotelladoras, ubicadas en los territorios de Comarca Lagunera, Zacatecas, San Luis Potosí, Aguascalientes, Jalisco y Tecomán.

MS

Av. Hidalgo 2303 • C.P. 89140 Tampico, Tamaulipas, México
Apartado Postal 664 • C.P. 89000 Tampico, Tamaulipas, México
Teléfono: (833) 241.2500 • Fax: (833) 241.2577

Empresa subsidiaria de Grupo Continental, S.A.

SOCIEDAD INDUSTRIAL, S.A. DE C.V.

Los profesores-investigadores del ITCM que participan en este proyecto, son los siguientes:

Responsables:

- Dra. Laura Cruz Reyes
- M.C. Apolinar Ramírez Saldívar

Colaboradores:

- Dr. Juan Javier González Barbosa
- Dr. Héctor Fraire Huacuja
- M.C. Ana Guadalupe Vélez Chong
- M.C. José Antonio Martínez Flores

Aprovecho la ocasión para enviarles un sincero agradecimiento por su valiosa colaboración en el desarrollo de este importante proyecto para las embotelladoras de Grupo Continental, S.A.

Se extiende la presente a petición de los interesados y para su uso ante las autoridades educativas que a ellos convenga.

Cordialmente,



Ing. Sergio Serrano Vázquez
Director de Sistemas de información
Grupo Continental, S.A.

SSV'bfo
1949'S

Av. Hidalgo 2303 • C.P. 89140 Tampico, Tamaulipas, México
Apartado Postal 664 • C.P. 89000 Tampico, Tamaulipas, México
Teléfono: (833) 241.2500 • Fax: (833) 241.2577

Empresa asociada de Grupo Continental, S.A.



Cd. Madero, Tam., a 9 de septiembre de 2005

A quien corresponda:

El suscrito Director General de FORTIA TECHNOLOGY, S.A. de C.V., hace constar que a partir del 24 de noviembre de 2004, inició el desarrollo de un Sistema de Cómputo para la Administración de la Transportación de productos embotellados, con la participación de un equipo interdisciplinario formado por ingenieros de software de la compañía a mi cargo; profesores y estudiantes del área de optimización del Instituto Tecnológico de Ciudad Madero (ITCM) y, expertos en la administración de almacenes de la Compañía Coca-Cola.

El producto desarrollado se implementará en tres regiones. Actualmente, la primera versión del producto se está implementando en la zona conurbana de Durango.

Los profesores-investigadores del ITCM que participan en este desarrollo se enlistan a continuación:

Responsables:

Dra. Laura Cruz Reyes
Fis. Mat., Apolinar Ramírez Saldivar

Colaboradores:

Dr. Héctor Fraire Huacuja
M.C. Juan Javier González Barbosa

A petición de los interesados y para los usos legales que a los mismos convengan, se extiende la presente en la ciudad de Madero, Tamaulipas a los nueve días del mes de septiembre de dos mil cinco.


ATENTAMENTE
Ing. Ponciano Betancourt Vicencio
Director General
FORTIA TECHNOLOGY

Av. Hidalgo 5004 1er. Piso Desp. 6 Col. Sierra Morena C.P. 89210, Tampico, Tam., México
Tel. (1)213-78-90. Fax (1)213-74-90
www.fortia.com.mx

RE: Algoritmo

De: emmanuel.juarez@arcacontal.com
Enviado: jueves, 15 de diciembre de 2011 02:20:15 p.m.
Para: lauracruzreyes@hotmail.com

Doctora, buenos días. Le pedí a Joselo que me apoya para reactivar el contacto con su equipo de Investigación pues han surgido nuevos requerimientos para el algoritmo de fiteo en el cual nos apoyaron para la Investigación y desarrollo.

¿Qué día pudiera comunicarme con usted para platicar con más detalle estos nuevos requerimientos? La Idea sería buscar un esquema de trabajo que a las dos partes nos parezca conveniente.

Quedo a sus órdenes para cualquier duda o aclaración al respecto.

Saludos



Emmanuel J. Juárez Gándara

Tecnologías de Información

emmanuel.juarez@arcacontal.com | www.arcacontal.com

Conn.+52(81) 8151-1400 Ext. 11829

De: Jose L. Vargas Carretero [mailto:jose.vargas@fortia.com.mx]
Enviado el: miércoles, 14 de diciembre de 2011 06:52 p.m.
Para: Juárez Gándara, Emmanuel (MXCOT)
CC: launcruzreyes@hotmail.com
Asunto: Algoritmo

Estimados Emmanuel,

Anexo el correo de la Dra. Laura Cruz, ella forma parte del equipo del equipo de investigación del Tec de Madero y ha estado desde un inicio en el proyecto de la generación del algoritmo para la logística de los camiones.

Comentamos brevemente entre nosotros que tienes necesidades y requerimientos para el funcionamiento del algoritmo, por lo que los pongo en contacto directo para no triangular y veamos si puede haber algún apoyo por parte del Tecnológico.

Sin más por el momento quedo de tí y sigo pendiente del tema para cualquier situación.

saludos

José Luis Vargas Carretero

Fortia Technology

Teléfono Oficina: + 52 (833) 241.80.80 Ext. 601

www.fortia.com.mx