

COLONIA DE HORMIGAS
FUNDAMENTACION TEÓRICA Y APLICACIÓN EN LA
OPTIMIZACIÓN DE SISTEMAS LOGÍSTICOS DE RUTEO
CON INTERVALOS DE RECEPCIÓN Y TIEMPO DE
ATENCIÓN MÁXIMO

JOSÉ LUÍS TOLOSA BARÓN

Trabajo de grado para optar el título de
Ingeniero Industrial

Director
Carlos Eduardo Díaz Bohórquez
Ingeniero Industrial

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE ESTUDIOS INDUSTRIALES Y EMPRESARIALES
BUCARAMANGA
2005

RESUMEN

TITULO¹: Colonia de hormigas, fundamentacion teórica y aplicación en la optimización de sistemas logísticos de ruteo con intervalos de recepción y tiempo de atención máximo.

AUTORES: TOLOSA BARON, Jose Luis**

PALABRAS CALVES: Optimización por colonia de Hormigas (OCH), Optimización multiobjetivo, Optimo pareto, frente pareto, heurísticas, feromona, Ruteo, Logistica.

DESCRIPCIÓN: El presente trabajo expone la utilización de la metodología conocida como Optimización por Colonia de Hormigas(OCH), en la construcción de un algoritmo para resolver un problema de ruteo multiobjetivo. Este trabajo no pretende ser una creación novedosa, sino una ejemplificación de la utilización de algoritmos de OCH en la construcción de algoritmos particulares de ruteo. El problema enfrentado plantea la necesidad de encontrar las rutas que debe seguir una flota de camiones para satisfacer las demandas particulares de un grupo de clientes que determina un intervalo de tiempo (ventana) para ser atendidos, mientras se optimizan cuatro funciones objetivo, numero de camiones, distancia recorrida, tiempo total empleado y tiempo de atención máximo. La metodología aplicada contempla una revisión bibliografica extensa de los problemas de ruteo, la creación de un modelo matemático, la creación de un algoritmo matemático, el estudio estadístico encaminado a determinar los parámetros que mas influyen en la calidad de las respuestas obtenidas por el algoritmo y la determinación de los valores óptimos para los cuales el desempeño del algoritmo es mayor. Como una nueva propuesta, el presente trabajo aplica conceptos del algoritmo Mejor/Peor hormiga y además integra conceptos de búsqueda multiobjetivo en una colonia de hormigas que construye soluciones en un tiempo virtual lineal (todas las hormigas parten a la vez). El algoritmo obtenido ha sido probado hasta el momento en una sola instancia con la cual fueron ajustados sus parámetros, produciendo resultados muy alentadores.

¹ Trabajo de investigación o Tesis.

** Facultad de Ingenierías Físicomecánicas, Escuela de Estudios Industriales y Empresariales, Director del proyecto: Ing. Carlos Diaz

ABSTRACT

TITLE²: Ant Colony, theoretical bases and application on logistics route systems optimization with time Windows and maximum service time

AUTHOR: TOLOSA BARON, Jose Luis**

KEY WORDS: Ant Colony Optimization, Multiobjective optimization, Pareto optimal, heuristic, pheromone, Routing, Logistic.

DESCRIPTION: The current study shows the application of a methodology known as Ant Colony Optimization (ACO), on the assemble of an algorithm to solve a multiobjective routing problem. This study do not attempt to be a new revolutionary design, but a exemplification of the use of OCH algorithms in the construction of particulars cases rout problems. The faced problem establish the necessity of founding a set of tours that must follow a group of trucks to satisfy the material request of a collection of clients that has reception, while four objectives are being optimize: truck number, distance traveled, total time used and maximum service time. The applied methodology includes a bibliographic extent review of the rout problems and ACO, the creation of a math model, the creation of a math algorithm, an statistic study focused to determine the most influential parameters on the quality of the answered solutions by the algorithm and the determination of optimal values which the performance of algorithm is better. As a new purpose, the current study uses the concept of Best/worst ant algorithm and also utilize multiobjective search concepts on an ant colony that builds solutions on linear virtual time (all ants leave de nest at the same time). The final algorithm has been tested yet on an only one instance, the same one that was used for parameter tuning, resulting on very promising results.

² Research work.

** Facultad de Ingenierías Físicomecánicas, Escuela de Estudios Industriales y Empresariales, Project management: Ing. Carlos Diaz

TABLA DE CONTENIDO

INTRODUCCIÓN	12
OBJETIVOS	15
OBJETIVO GENERAL	15
OBJETIVOS ESPECÍFICOS.....	15
1. PROBLEMA OBJETO DE LA INVESTIGACIÓN	16
2. SISTEMAS LOGÍSTICOS	19
2.1. SISTEMAS LOGÍSTICOS	19
2.1.1. Sistema	19
2.1.2. Sistema logístico.	19
2.2. LA LOGÍSTICA COMO UNA VENTAJA COMPETITIVA.....	20
2.2.1. Principales criterios de la calidad logística.	21
2.3. DISTRIBUCIÓN FÍSICA.....	22
2.3.1. Medios de transporte.	23
2.3.3. Canales de distribución	24
2.4. INTEGRACIÓN DE LOS SISTEMA LOGÍSTICOS	26
2.4.1. Colaboración en operaciones.	27
2.4.2. Colaboración en Marketing.....	28
2.4.3. Colaboración con proveedores.....	29
2.4.5. Uso de intervalos de recepción (ventanas de tiempo).....	30
2.5. SISTEMAS DE INFORMACIÓN UTILIZADOS EN DISTRIBUCIÓN	31
2.6. TENDENCIAS DE LOS SISTEMAS LOGÍSTICOS	32
2.6.1. Respecto a los sistemas de operación.....	32
2.6.2. Respecto a la integración	32
3. EL PROBLEMA DEL RUTEO	34
3.1. DEFINICIÓN DE RUTEO Y SCHEDULING.....	34
3.2. FACTORES A CONSIDERAR EN SISTEMAS DE RUTEO	35
3.3. TIPOS DE PROBLEMAS DE RUTEO DE VEHÍCULOS.....	37
3.4. DESCRIPCIÓN DE PROBLEMAS DE RUTEO CON MISMO ORIGEN Y DESTINO.....	38

4.	PROBLEMAS DE DECISIÓN PRESENTES EN RUTEO	39
4.1.	PROBLEMAS DE DECISIÓN.....	39
4.2.	LA NATURALEZA DE LOS PROBLEMAS DE DECISIÓN.....	40
4.2.1.	Clasificación de los problemas de decisión.	40
4.3.	CLASES DE PROBLEMAS DE RUTEO.....	42
4.3.1.	Los Problemas de optimización.	42
4.4.	LOS PROBLEMAS CON RESTRICCIONES.....	43
4.5.	OPTIMIZACIÓN MULTIOBJETIVO	43
4.6.	ESFUERZO COMPUTACIONAL ASOCIADO A LOS PROBLEMAS DE RUTEO	45
4.7.	ENFOQUES UTILIZADOS PARA RESOLVER EL PROBLEMA DE RUTEO	45
5.	CONCEPTOS DE HEURÍSTICA Y ALGORITMOS HEURÍSTICOS	46
5.1.	DEFINICIÓN DE HEURÍSTICA.....	47
5.2.	APLICACIÓN DE LOS MÉTODOS HEURÍSTICOS	48
5.3.	PATRONES DE INFERENCIA PLAUSIBLE	48
5.3.1.	Patrones Inductivos.	48
5.3.2.	Verificación sucesiva de varias consecuencias	49
5.3.3.	Verificación de consecuencias improbables.	49
5.3.4.	Inferencia por Analogía.....	49
5.4.	SILOGISMO HEURÍSTICO (DE LAS SOLUCIONES HEURÍSTICAS).....	50
5.5.	METAHEURÍSTICAS	51
5.6.	HIPERHEURÍSTICAS	51
5.7.	TÉCNICAS HEURÍSTICAS DE BÚSQUEDA DE SOLUCIONES.....	53
5.7.1.	Métodos de búsqueda exhaustiva.....	53
5.7.2.	Métodos de búsqueda local.	54
5.7.3.	Métodos que usan soluciones parciales.	55
5.7.4.	Métodos de búsqueda global.....	56
6.	LOS SISTEMAS DE COLONIA DE HORMIGAS.....	58
6.1.	INTELIGENCIA DE ENJAMBRE Y COMPUTACIÓN EMERGENTE.....	58
6.2.	LAS HORMIGAS REALES Y SUS COLONIAS	60
6.2.1.	El camino entre la colonia y la comida.	60
6.2.1.	El uso de feromonas en la consecución del camino más corto.	61
6.2.2.	Un método no infalible.....	63
6.3.	PRINCIPIOS DE LAS HEURÍSTICAS CON MÚLTIPLES AGENTES BIOINSPIRADAS	64
6.4.	METAHEURISTICA DE SISTEMA DE COLONIA DE HORMIGAS	65
6.4.1.	Origen de la metaheurística de colonia de hormigas.	65
6.4.2.	Aspectos generales de los OCH.	65
6.5.	LAS HORMIGAS ARTIFICIALES	67
6.6.	SIMILITUDES Y DIFERENCIAS ENTRE LAS HORMIGAS NATURALES Y ARTIFICIALES	69

6.7.	LA METAHEURÍSTICA DE OPTIMIZACIÓN BASADA EN COLONIAS DE HORMIGAS	71
6.8.	MODO DE FUNCIONAMIENTO DE UN ALGORITMO DE OCH	72
6.9.	ESTRUCTURA GENÉRICA DE UN ALGORITMO DE OCH	73
6.10.	DIFERENCIAS ENTRE LAS METAHEURÍSTICA OCH, LOS ALGORITMO OCH Y LOS ALGORITMOS DE HORMIGAS	75
6.11.	MODELOS DE ALGORITMOS OCH	76
6.12.	IMPLEMENTACIÓN DE UN ALGORITMO OCH EN UN HABIENTE PARALELO DE CÓMPUTO	78
7.	FUNDAMENTOS MATEMÁTICOS DE LOS SISTEMAS DE COLONIAS DE HORMIGAS	79
7.1.	TIPOS DE PROBLEMAS RESOLUBLES POR LA OCH	79
7.2.	MODELOS DE ALGORITMOS OCH	80
7.2.1.	El Sistema de Hormigas.....	80
7.2.2.	El Sistema de Colonias de Hormigas (SCH).....	82
7.2.3.	El Sistema de Hormigas Max-Min (SHMM).....	86
7.2.4.	El Sistema de Hormigas con Ordenación (SH ordenación).....	87
7.2.5.	El Sistema de la Mejor - Peor Hormiga (SMPH).....	89
8.	METODOLOGÍA DE SOLUCIÓN DE RUTEO UTILIZANDO OCH	92
9.	CONSTRUCCIÓN DEL MODELO MATEMÁTICO PARA RESOLVER EL PROBLEMA PLANTEADO	94
9.1.	FORMULACIÓN MATEMÁTICA DEL PROBLEMA.....	94
9.2.	DEFINICIÓN DE COSTOS	95
9.3.	DEFINICIÓN DEL OBJETIVO	97
9.4.	METODOLOGÍA DE SOLUCIÓN AL PROBLEMA MULTIOBJETIVO.....	98
10.	CONSTRUCCIÓN DEL ALGORITMO PROPUESTO	101
10.1.	LA VISIÓN MULTIOBJETIVO	102
10.1.1.	Estrategia Heurística	103
10.1.2.	Estrategias memorísticas (Feromona).....	104
10.2.	REGLAS DE TRANSICIÓN.....	104
10.3.	ITERACIONES, CICLOS Y SUBDIVISIONES	105
10.4.	ACTUALIZACIÓN DE LA FEROMONA	106
10.5.	INICIACIÓN DE FEROMONA.....	107
10.5.1.	Matriz de feromona Constante.....	107
10.5.2.	Escalamiento de la matriz de feromonas.....	107
10.6.	REINICIACIÓN DE FEROMONA.....	108
10.7.	OPERADOR DE MUTACIÓN	109
10.8.	BÚSQUEDA LOCAL.....	110
10.9.	ALGORITMO PROPUESTO.....	111
10.9.1.	Bloque de iniciación.....	111
10.9.2.	Bloque de actividades de creación de soluciones.	112

10.9.3. Bloque de actualización de la feromona.	114
10.9.4. Bloque de Mutación.	114
10.9.6. Bloque de evaluación de las soluciones multiobjetivo.	116
10.10. PSEUDOCÓDIGO DEL ALGORITMO PROPUESTO	118
10.11. DESARROLLO DEL ALGORITMO COMPUTACIONAL.....	121
10.11.1. Lenguaje Utilizado.....	121
10.11.2. Instancias.....	122
10.12. ANÁLISIS ESTADÍSTICO DE LOS PARÁMETROS	123
10.13. DESEMPEÑO DEL ALGORITMO	127
 BIBLIOGRAFÍA	 129
 ANEXOS	 137

LISTA DE FIGURAS

Figura 1. Uso de feromona para la consecución de la ruta mas corta.....	63
Figura 2. Espacio de las soluciones factibles	100
Figura 3. Esquema de valores escalado para la matriz de feromona	108
Figura 4. Esquema Bloque de iniciación	112
Figura 5. Esquema Bloque de actividades de creación de soluciones	113
Figura 6. Esquema bloque de actualización de feromona (Actualización).....	114
Figura 7. Esquema bloque de actualización de feromona (Reiniciación).....	114
Figura 8. Esquema bloque de mutación	115
Figura 9. Esquema bloque de optimización local	116
Figura 10. Esquema de evaluación de soluciones	117
Figura 11. Interfase del algoritmo computacional creado.....	121
Figura 12. Esquema de instancia requerido	122

LISTA DE TABLAS

Tabla 1. Parámetros constantes utilizados en la ANOVA	124
Tabla 2. Parámetros variables utilizados en la ANOVA.....	124
Tabla 3. Resultados de la prueba ANOVA	126
Tabla 4. Mejores valores para los parámetros según prueba ANOVA.....	126
Tabla 5. Mejores resultados obtenidos para el problema C101	127
Tabla 6. Comparación de resultados obtenidos con el algoritmo propuesto,.....	127

LISTA DE ANEXOS

Anexo 1. Método Heurístico de Kart & Thompson	12438
Anexo 2. Método Heurístico de Barrido (sweep) para los problemas de ruteo con múltiples vehículos	12441

INTRODUCCIÓN

El crecimiento económico ha traído cambios importantes en la cadena de suministro de materias primas y productos terminados tanto para los productores como para sus clientes, en diferentes espacios y condiciones.

El sistema logístico de una empresa pasó de ser un paso más en su operación a ser una ventaja competitiva y estratégica en el desarrollo y crecimiento de la empresa.

El nacimiento de grandes operadores logísticos encargados de manejar la distribución de múltiples productos, evidencia las ventajas de contar con sistemas eficientes y la especialización, profundidad y dificultad que representa para las empresas el manejo logístico de distribución

La variedad de productos y materias primas que una empresa utiliza es cada vez más compleja haciéndose necesario una administración estricta en los puntos de recepción de la mercancía. Con el fin de agilizar el proceso de recepción de materiales y la descongestión de los puertos de desembarque, cada vez son más las empresas que asignan a sus proveedores un horario o intervalo de tiempo para realizar el proceso de entrega en un lugar específico. Las empresas transportadoras ahora deben incluir en sus planes de distribución, las medidas necesarias para evitar faltar a uno de estos intervalos de recepción o el llegar demasiado temprano y generar costos adicionales por esperas.

Este tipo de problema no es nuevo en el estudio de la investigación de operaciones, ya que las condiciones actuales de distribución han estimulado a múltiples autores a crear diferentes modelos para solucionar el problema que representa encontrar la ruta mas adecuada para distribuir sus productos de la forma más económica. Los diferentes instrumentos matemáticos utilizados hasta hace unos años han visto cómo los problemas cada vez mas grandes en número de requisitos y de variables los han relegado a un segundo plano dando paso a nuevas técnicas que combinan de manera ingeniosa estructuras de la naturaleza con métodos computacionales complementarios.

El presente proyecto pretende incursionar en el ámbito de los nuevos modelos matemáticos inspirados en comportamientos de la naturaleza y caracterizados por combinar aspectos exploratorios con aspectos de aprendizaje. El lector podrá encontrar una fundamentación teórica de los nuevos algoritmos inspirados en comportamientos de algunos animales que realizan tareas complicadas siendo especies que individualmente carecen de inteligencia notable. Estos comportamientos aparentemente inteligentes sirven de base para la aplicación de métodos heurísticos fundamentados en razonamiento plausible conocidos desde hace varios años en la filosofía y recientemente utilizados en métodos de inteligencia forzada.

El proyecto profundiza en la creación de un modelo matemático que permita afrontar problemas de tipo actual, satisfaciendo los requerimientos de los clientes de una forma medianamente confiable y eficaz mediante la utilización de métodos heurísticos conocidos como Optimización por Colonia de Hormigas los cuales pretenden emular el comportamiento de las hormigas en su búsqueda del camino más corto para la consecución de alimentos. El proyecto presenta una adecuada documentación del proceso seguido para crear algunos modelos genéricos difundidos y todo el proceso llevado a cabo para transformar al caso particular un algoritmo conocido como Mejor/Peor Hormiga, el cual hasta el momento ha producido los mejores resultados de la metodología³.

Con la finalidad de lograr una verdadera creación particular de un modelo matemático y el algoritmo derivado de este, se utilizan por primera vez varias técnicas utilizadas en otras áreas de la investigación como los algoritmos genéticos, para introducir procesos con nuevas variaciones. También se acudió a utilizar técnicas de diferentes tipos de algoritmos de colonias de hormigas con el ánimo de comparar mediante pruebas de desempeño el comportamiento de dos métodos excluyentes como son los tipos de mutación y los tipos de iniciación de la feromona.

Para incursionar de una manera novedosa y cumplir con los objetivos propuestos es necesario el desarrollo de una problemática más profunda en cuanto a la satisfacción de los requerimientos múltiples de la industria del transporte. El problema que se resuelve en este proyecto utiliza conceptos de optimización multiobjetivo, según los cuales no solo se persigue mejorar un aspecto sino que

³ ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

varios aspectos son mejorados en forma paralela con la ejecución del algoritmo. De esta manera no solamente los problemas que tradicionalmente preocupan a las empresas de transporte (Tamaño de la flota y distancias a recorrer) son tenidos en cuenta, sino que aspectos como la calidad del servicio y los costos secundarios, pueden ser manejados en un solo algoritmo a la vez.

La metodología aplicada contempla además un estudio estadístico encaminado a determinar los parámetros que mas influyen en la calidad de las respuestas obtenidas por el algoritmo, y a la determinación de los valores óptimos para los cuales el desempeño del algoritmo es mayor.

Los resultados obtenidos mediante experimentación una vez finalizado el algoritmo son muy dicentes y satisfactorios ya que sin ser este un estudio profundo, se logro obtener resultados muy cercanos al optimo para el problema estudiado, sirviendo este algoritmo como punto de partida para programas realmente orientados a usuarios logísticos.

El potencial demostrado durante el proyecto por esta metodología, enmarca el primer paso en lo que seguramente serán algoritmos mas rápidos y precisos en la solución de problemas de transporte y enrutamiento.

OBJETIVOS

OBJETIVO GENERAL

Desarrollar un modelo general para la optimización de sistemas logísticos de ruteo con intervalos de recepción y control del tiempo de atención máximo basado en la metodología heurística de colonia de hormigas como una muestra del potencial de esta metodología en la solución de problemas de transporte y enrutamiento.

OBJETIVOS ESPECÍFICOS

- Identificar la dinámica de operación de los sistemas logísticos de ruteo con intervalos de recepción.
- Analizar el sistema de toma de decisión presente en el problema de enrutamiento.
- Analizar el comportamiento particular de las colonias de hormigas para encontrar la ruta mas corta entre la fuente de alimento y la colonia y su utilización en modelos heurísticos.
- Presentar la fundamentación teórica de los sistemas matemáticos de colonias de hormigas.
- Modificar los modelos matemáticos actuales para afrontar el problema de ruteo con intervalos de recepción y control del tiempo de atención máximo.
- Desarrollar una plataforma de software que permita ilustrar la aplicación del modelo matemático.

1. PROBLEMA OBJETO DE LA INVESTIGACIÓN

Los avances mostrados por los sistemas logísticos a nivel mundial reflejan una tendencia de las empresas a reducir costos y aumentar la satisfacción del cliente mediante un gerenciamiento logístico respaldado en sistemas cada vez más complejos de soporte para la toma de decisiones.

En el entorno logístico, el transporte de mercancías en camiones es una actividad típica, utilizada para la distribución de productos en ciudades y un problema común a la mayoría de las organizaciones que deben atender un número elevado de clientes.

Con el fin de agilizar el proceso de recepción de materiales y la descongestión de los puertos de desembarque, cada vez son más las empresas que asignan a sus proveedores un horario o intervalo de tiempo⁴ (ventanas de tiempo) para realizar el proceso de entrega en un lugar específico. Las empresas transportadoras ahora deben incluir a sus planes de distribución, las medidas necesarias para evitar faltar a uno de estos intervalos de recepción o el llegar demasiado temprano y generar costos adicionales por esperas.

El problema que se enfrenta en este proyecto se plantea como un depósito central que cuenta con una flota de m vehículos de capacidad limitada y que debe atender a un conjunto de n clientes geográficamente distribuidos que se conectan por una red de vías simétrica⁵. El objetivo es entregar artículos a este conjunto de clientes con demandas individuales y conocidas, cumpliendo con unos intervalos de tiempo conocidos para cada cliente en el que debe ser atendido, cada cliente debe ser atendido una sola vez por un solo vehículo y todos los clientes deben ser atendidos. Además cada cliente requiere de un tiempo de servicio en el cual el vehículo descargan los artículos. Se debe aclarar que los vehículos no vuelven a salir una vez han regresado a la bodega.

⁴ Léase el numeral 1.4.5. *Uso de intervalos de recepción (ventanas de tiempo)* para una descripción mas detallada.

⁵ En la forma simétrica de un problema de ruteo, la única ruta directa entre dos clientes permite moverse tanto en una dirección como en otra. Esta forma es la predilecta para realizar análisis por permitir una serie importante de optimizaciones al momento de hacer experimentos prácticos.

Se busca crear m rutas cada una con k clientes atendidos en un orden específico, las cuales minimicen a la vez el número de vehículos, la distancia total recorrida, el tiempo total en que se incurre para terminar todas las rutas y el tiempo máximo de servicio⁶. El problema puede definirse entonces como un problema de ruteo⁷ de optimización⁸ con intervalos de recepción (conocido en la literatura por sus siglas en inglés VRPTW⁹) de tipo multiobjetivo¹⁰.

En un medio en el que mas clientes exigen horarios de entrega, las flotas de entrega crecen, las entregas son mas frecuentes y las peticiones, horarios y condiciones cambian diariamente, la necesidad de herramientas que permitan una asignación óptima de los recursos de una forma rápida, hace que los métodos convencionales de planeación que se caracterizan por ser de difícil escalado¹¹, sean poco eficientes, o inaplicables por el esfuerzo computacional asociado para encontrar la solución¹².

Se trata de un problema tipo NP-Duro¹³, lo que garantiza solamente una solución determinística en un tiempo exponencial. Por tanto la solución debe ser un algoritmo heurístico que pueda ser ejecutado por un computador personal en un tiempo aceptable para ser útil.

Los sistemas creados con algoritmos basados en Optimización por Colonias de Hormigas¹⁴ han demostrado una gran eficiencia y menor complejidad que los métodos exactos, por lo que pueden ser implementados con requerimientos tecnológicos mínimos.

Considerando la importancia del estudio de nuevos métodos y sus aplicaciones a la industria colombiana, este proyecto busca mostrar una fundamentación teórica acerca de los sistemas de colonias de hormigas en forma general y su aplicación al problema particular planteado.

⁶ En el numeral 8.3 *Definición del objetivo* puede encontrarse una definición detallada de cada objetivo.

⁷ Léase definición en el numeral 2.1. *Definición de ruteo y scheduling*.

⁸ Véase el numeral 2.3. *Tipos de Problemas de Ruteo de Vehículos*.

⁹ Una descripción se encuentra en 2.4. *Descripción de Problemas de Ruteo con Mismo Origen y Destino*

¹⁰ En el numeral 4.5. *Optimización multiobjetivo* se presenta una esquematización detallada.

¹¹ Que se dificultan a medida que más variables se integran al problema. (no pueden manejar un gran número de variables)

¹² Véase el numeral 4.6. *Esfuerzo computacional asociado a los problemas de ruteo*

¹³ Véase definición en 4.2.1. *Clasificación de los problemas de decisión*.

¹⁴ Véase definición en el numeral 5. *Los sistemas de colonia de hormigas*

El modelo¹⁵ que se presenta no representa una creación novedosa, sino una ejemplificación de la utilización de algoritmos de Sistemas de Colonia de Hormigas en la construcción de algoritmos particulares.

¹⁵ En el desarrollo del modelo presentado, se utiliza el axioma de que siempre encontrar una solución aproximada a un modelo exacto será mejor que encontrar una solución exacta a un modelo aproximado.

2. SISTEMAS LOGÍSTICOS

2.1. SISTEMAS LOGÍSTICOS

2.1.1. Sistema. Se utiliza el término sistema para explicar en general el conjunto de elementos interrelacionados (Objetos, Personas, Información, etc.) utilizados según un proceso dinámico con el fin de alcanzar los objetivos señalados¹⁶.

Las interacciones de los elementos del sistema se realizan tanto hacia adentro como hacia fuera del conjunto, en una búsqueda de la integración, así lo expresa Tejada:

Si consideramos a un conjunto, formado por partes, estas partes son afectadas por factores internos y externos a ese conjunto. Además, cada parte debe interactuar entre sí y con otras provenientes del medio externo. También, éste conjunto debe, a su vez, interactuar como su conjunto lógico para dar respuestas a sus propios requerimientos y a los que el medio externo espera de él. Un sistema integrado se logra cuando cada parte de este todo, actúa e interrelaciona interna y externamente mediante un plan, método y orden¹⁷.

2.1.2. Sistema logístico. Se han dado múltiples definiciones al término “logística” algunos se encuentran a continuación¹⁸:

- El movimiento de los materiales desde una fuente u origen hasta un destino o usuario.
- Actividad compuesta por dos funciones básicas: La gestión de los materiales: encargado de los flujos materiales en el aprovisionamiento de las materias

¹⁶ TEJADA, Carlos. Ideas orientadoras sobre el sistema logístico del componente ejercito en el teatro de operaciones. En revista Military Review, p. 44. Buenos Aires:Escuela superior de guerra, Diciembre 2001

¹⁷ Ibid., p 45

¹⁸ ANAYA, Julio. ARBONES, Eduardo. BALLOW, Ronald. BITLE, L y RAMSEY, J. Logística Integral(1991), Logística Empresarial (1994). Enciclopedia del management. En PÉREZ, Marisol. De la Gestión de la Producción a la Gestión de la Cadena de Suministro (Online) Holguín: Editorial Uho, 2003. www.monografias.com (consultado: 6 de abril de 2005).

primas y componentes y en las operaciones de fabricación hasta el envase del producto terminado y la gestión de distribución: encargada del embalaje, el control de los inventarios de los productos terminados, pasando por los procesos de manipulación, almacenamiento y transporte hasta la entrega del producto o del servicio al cliente.

- Es un enfoque que permite la gestión de una organización a partir del estudio de flujo de materiales y el flujo informativo que a él se asocia, desde los suministradores hasta los clientes, partiendo de cinco funciones básicas que se desarrollan en las organizaciones: la gestión de aprovisionamiento, la gestión de procesos, la distribución física, la planificación integrada y el aseguramiento de la calidad.
- El sistema que garantiza el movimiento óptimo de las cargas y la información de la fuente hasta un cliente.
- El conjunto de todas las actividades relacionadas con el flujo de material desde el punto proveedor hasta el punto consumidor, contempla además de las actividades materiales aquellas mediante las que se planifica, organiza, regula y controla dicho flujo material (dirección) de forma eficiente entendiéndose por eficiente llegar al punto consumidor con la cantidad y calidad requerida en el momento y lugar demandado con el menor costo posible.
- Comprende la planificación, organización y el control de todas las actividades relacionadas con la obtención, traslado y almacenamiento de materiales y productos, desde la adquisición hasta el consumo, a través de la organización y como un sistema integrado.

En la actualidad para hablar de la Logística hay que hacerlo refiriéndonos a ella como un sistema formado por procesos que se integran alrededor del flujo bienes, fondos y/o información, donde se deben considerar todos los aspectos relacionados a su flujo con una fuerte orientación al cliente final, con el objetivo de lograr altos niveles de satisfacción de éste que garanticen la sostenibilidad de la Organización y del ecosistema.

2.2. LA LOGÍSTICA COMO UNA VENTAJA COMPETITIVA

La mejora en la eficiencia de la logística debe incidir positivamente en dos aspectos fundamentales:

La competitividad de las empresas.

La reducción del impacto ambiental.

Según Esparcia y Sharman “la mejora en la competitividad viene dada por las siguientes causas”¹⁹:

- Facilidad de distribución de sus productos a nuevos mercados.
- Reducción de tiempos desde la concepción del producto hasta la puesta a disposición del consumidor, lo que permite adecuarse rápidamente a sus gustos y anticipar ventas respecto a competidores menos preparados.
- Reducción de las necesidades de almacenamiento, con la consiguiente reducción de costes y mejora en la rotación de productos.
- Automatización y disminución de errores en pedidos/envíos, obteniéndose mejoras en costes de gestión y satisfacción de clientes.
- Optimización del transporte de productos utilizando las rutas y medios o combinación de medios (carretera, marítimo, ferrocarril...) más adecuados teniendo en cuenta coste y tiempo.
- Organización de la producción y distribución teniendo en cuenta la globalización y las posibilidades de la localización de parte de los procesos productivos que las hagan más competitivas.
- En el sector de alimentos, la posibilidad de ofrecer productos más frescos, de mayor calidad.

En cuanto a la reducción del impacto ambiental se obtiene principalmente por:

- Optimización del transporte teniendo en cuenta las variables que afectan al medio ambiente y reduciendo su impacto.

2.2.1. Principales criterios de la calidad logística. Los criterios logísticos más importantes que se deben tener en cuenta en el servicio al cliente según Angulo²⁰ son:

Grado de certeza No es tan necesario llegar rápido con el transporte, como llegar a tiempo, con el mínimo rango de variación.

Grado de confiabilidad Una cadena se conforma de diferentes eslabones. Eso es una cadena logística. Si se agregan algunos eslabones que no están

¹⁹ ESPARCIA, Anna y SHARMAN, Ken. Algoritmos Bio-inspirados En Revista Actualidad Tic (On line), 2004 ; p. 2, <http://www.iti.upv.es/cas/> (consultado: 8 de mayo de 2005)

²⁰ANGULO, Cesar, La Logística (Online), 2005; p. 2, www.monografias.com (consultado: 11 de abril de 2005)

relacionados, se segmentan las responsabilidades; el cliente final pierde la confianza, al aparecer mayores errores de interpretación y responsables difusamente identificables. El cliente debe poder manifestar cuál es su criterio de confiabilidad, cómo entiende que deberían ser atendidos.

Grado de flexibilidad Implica que el prestador pueda adaptarse eficientemente a los picos de demanda. Un operador logístico que considera excesivo la solicitud de eficiencia cuando se da un salto por estacionalidad, desconoce qué es valor para su cliente.

Aspectos cualitativos Se trata aquí del servicio, del cual debe buscarse su homogeneidad en toda la cadena logística. En muchos casos, se cuida minuciosamente el proceso productivo, se diseña con cuidado el packaging (empaquetado), se llega hasta decir cómo debe transportarse y almacenar en el depósito. Pero también se debe controlar cómo llegarán hasta el cliente esos productos.

La mejora continua Día a día se deben replantear los parámetros que presenten deficiencias de acuerdo a los objetivos pensados, pero también se deben revisar aquellos que están bien y pueden ser mejorados. Es mucho más saludable cuestionar internamente lo que aparentemente resulta bien, a que lo haga el mercado. La mejora de las variables logísticas se debe entender como una exigencia.

2.3. DISTRIBUCIÓN FÍSICA

“La distribución física implica la planeación, la instrumentación y el control del flujo físico de los materiales y los bienes terminados desde su punto de creación/extracción hasta los lugares de su utilización, con el fin de satisfacer las necesidades de los clientes a cambio de una ganancia. El mayor costo de la distribución física corresponde al transporte, seguido por el control de inventario, el almacenaje y la entrega de pedidos con servicios al cliente”²¹.

²¹ CHAMOSO, Raul. Sist. de Control y Distribución de Ventas (Online), 2004; p 3. www.monografias.com (consultado: 10 de abril de 2005)

Las decisiones erróneas sobre la distribución física pueden provocar altos costos. Aún las grandes compañías a veces utilizan poco las modernas herramientas de decisión para coordinar los niveles de inventario, las formas de transporte y la localización de la planta, la bodega y las tiendas.

Además, la distribución física no es sólo un costo, sino una poderosa herramienta de creación de demanda. Las compañías pueden atraer más clientes otorgándoles mejor servicio o precios más bajos por medio de una mejor distribución física. En cambio, pierden clientes cuando no logran suministrarles los bienes a tiempo²².

2.3.1. Medios de transporte. Es un elemento de mucha importancia dentro de la distribución física. Para transportar productos dentro de una ciudad y a otras se utiliza las diferentes vías de comunicación. Los medios mas comunes son:

Camión Los camiones han ido aumentando constantemente su participación en el transporte y representan el mayor porcentaje de movilización del total de carga movilizada. Se encargan de la mayor parte del transporte dentro de las ciudades. Pueden llevar productos de puerta a puerta, lo cual ahorra a los compradores la necesidad de transferir sus productos de un lugar a otro²³.

Vías Fluviales “El medio más económico para transportar grandes cargas”²⁴. El inconveniente de este método son las limitaciones naturales de estas vías y la necesidad de puertos de cargue.

Ferrocarril: es uno de los medios más eficientes con relación a su costo para enviar grandes volúmenes de productos como carbón, arena, minerales o productos agrícolas. Su principal limitación en Colombia es la limitada extensión de su infraestructura.

Otros Tipos de Transporte

Vehículos automóviles

Tuberías

²² Ibid., p. 6

²³ CHAMOSO, Raul. Sist. de Control y Distribución de Ventas (Online), 2004; p 3. www.monografias.com (consultado: 10 de abril de 2005)

²⁴ Ibid., p 3.

Líneas aéreas

2.3.2. Criterios para la selección de transporte²⁵.

Costos Se evalúa si los beneficios de uso de un medio de transporte en realidad equivalen a lo que cuesta utilizarlo; en comparación con los demás medios.

Tiempos en tránsito Es el tiempo total en el cual las mercancías se encuentran en poder del transportista.

Confiabilidad Se refiere a la integridad tanto en la seriedad como en la consistencia del servicio que ofrezca el medio de transporte. El tiempo y la confiabilidad afectan los costos de exigencia del vendedor además de las posibles ventas que, por no tener disponible la mercancía no se puede realizar.

Capacidad Es el espacio que tiene el transporte para dar un servicio adecuado a cada tipo de mercancía, ya que existen productos diferentes y por tanto existirá un trato diferente.

Asequibilidad Es la disponibilidad de la ruta o red específica para circular por ella.

Seguridad La mercancía debe llegar en optimas condiciones hasta el consumidor, generalmente el transporte se hace responsable de todas las perdidas y daños en la mercancía.

2.3.3. Canales de distribución²⁶. El canal de distribución lo constituye un grupo de intermediarios relacionados entre sí que hacen llegar los productos y servicios de los fabricantes a los consumidores y usuarios finales.

²⁵ Ibid., p 8.

²⁶ Se toma textualmente de CHAMOSO, Raúl. Sist. de Control y Distribución de Ventas (Online), 2004; p 3. www.monografias.com (consultado: 10 de abril de 2005)

Canales de distribución para productos industriales. Los productos industriales tienen una distribución diferente a las de los productos de consumo y emplean cuatro canales que son:

Productores – usuarios industriales: este es el canal más usual para los productos de uso industrial ya que es el más corto y más directo; utiliza representantes de ventas de la propia fábrica.

Productores – distribuidores industriales – consumidores industriales: en este caso los distribuidores industriales realizan las mismas funciones de los mayoristas, algunas ocasiones desempeñan las funciones de los mayoristas y en otras ocasiones desempeñan las funciones de fuerza de ventas de los fabricantes.

Productores – agentes – distribuidores industriales – usuarios industriales: en este canal la función del agente es facilitar las ventas de los productos y la función del distribuidor es almacenar los productos hasta que son requeridos por el usuario industrial.

Productores – agentes – usuarios industriales: en este caso los distribuidores industriales no son necesarios y, por lo tanto, se eliminan. Ejemplo: productos agrícolas.

Canales de distribución para productos de consumo Los canales para productos de consumo se dividen a su vez en cinco tipos que se consideran los más usuales:

Productores Consumidores: esta es la vía más corta y rápida que se utiliza en este tipo de productos. La forma que más se utiliza es la venta de puerta a puerta, la venta por correo, el telemarketing y la venta por teléfono. Los intermediarios quedan fuera de este sistema.

Productores – minoristas – consumidores: este es el canal más visible para el consumidor final y gran número de las compras que efectúa el público en general se realiza a través de este sistema. Ejemplos de este canal de distribución son los concesionarios automotrices, las gasolineras y las tiendas de ropa. En estos casos el productor cuenta generalmente con una fuerza de ventas que se encarga de

hacer contacto con los minoristas que venden los productos al público y hacen los pedidos después de lo cual los venden al consumidor final.

Productores – mayoristas – minoristas o detallistas: este tipo de canal se utiliza para distribuir productos tales como medicina, ferretería y alimentos. Se usa con productos de gran demanda ya que los fabricantes no tienen la capacidad de hacer llegar sus productos a todo el mercado consumidor.

Productores – intermediarios – mayoristas – consumidores: este es el canal más largo, se utiliza para distribuir los productos y proporciona una amplia red de contactos; por esa razón, los fabricantes utilizan a los intermediarios o agentes. Esto es muy frecuente en los alimentos perecederos.

2.4. INTEGRACIÓN DE LOS SISTEMAS LOGÍSTICOS

El factor humano y la evolución de teorías y tendencias han sido condicionantes de primer orden en el desarrollo de la logística. La evolución de teorías, técnicas y herramientas han emanado de los principales centros de Investigación Universitarios Americanos y Europeos principalmente. Este conocimiento ha experimentado en los últimos 10 años un elevado nivel de difusión en términos de nueva literatura, revistas especializadas, cursos, carreras y diplomados, entre otros. Uno de los principales aportes que ha hecho la logística a la gerencia de empresas hace referencia a su tendencia integradora y colaborativa desde el punto de vista de la planificación, ya sea a nivel intra-empresas (integración de funciones de la empresa como desarrollo de productos, abastecimiento, producción, distribución y ventas), como también a nivel inter-empresas (integración de procesos con proveedores y clientes). Esta nueva visión presiona la necesidad de adaptar los procesos actuales a la nueva realidad estratégica, táctica y operacional, desde la perspectiva de la planificación²⁷.

²⁷ DOMÍNGUEZ, José A. y otros. Dirección de Operaciones. Aspectos tácticos y operativos en la producción y los servicios, p. 303 España: Editorial MC Graw Hill; 1999.

2.4.1. Colaboración en operaciones. Se realiza con el objetivo de reducir o eliminar todas las actividades que no añadan valor al producto y pretende la integración y colaboración estrecha entre cliente y proveedor.

En este ámbito se han desarrollado diferentes técnicas, entre las más utilizadas se encuentran²⁸:

- **EDI- *Electronic Data Interchange*.** Comunicación electrónica de los datos de pedidos, facturas, etc.
- **EPOS- *Electronic Point of Sales*.** Conexión del proveedor con los escáners de los puntos de venta para conocer la demanda en tiempo real, básico para desencadenar el sistema de suministro más eficaz y como base de la demanda de los productos y el comportamiento de los clientes.
- **ER- *Efficient Replenishment*.** Es el proveedor el que se encarga de determinar los pedidos en cantidad, tiempo y frecuencia a partir de unos criterios acordados con el cliente.
- **ASN- *Advance Shipping Notes*.** Comunicación de forma avanzada sobre cantidades que se tienen que suministrar y fechas de entrega para facilitar las tareas posteriores a la entrega, por parte de los clientes.
- **EOS- *Efficient Operations Standard*.** Acuerdos sobre procedimientos que permitan mejorar las operaciones de distribución:
 - Tipos y altura de palets.
 - Códigos de barra de identificación de los productos, cajas y unidades de expedición, y de los palets.
 - Sinergia en el transporte con el fin de llenar al máximo los camiones en el trayecto de ida y vuelta (evitar desplazamientos sin mercancías).
 - Coordinar la demanda y la fiabilidad del transporte para facilitar la gestión de las nuevas plataformas logísticas convertidas en almacenes de tránsito sin inventarios permanentes.
- **DSD- *Direct Store Delivery*.** Gestión integrada por parte del proveedor para la entrega de la mercancía directamente en el punto de venta.
- **CAO- *Computer Assisted Ordering*.** Diseño del pedido del cliente final con soporte informático para conformar la demanda hecha a la medida para ese cliente, escogiendo entre las diferentes alternativas y opciones que presenta el producto y comunicación directa del pedido al proveedor desde la misma terminal del punto de venta.
- **EA- *Efficient Administration*.** Gestión integrada de la base de datos de los productos-opciones, alternativas, precios, etc-, los pedidos, las facturas, los

²⁸ Tomadas de: PÉREZ, Marisol. De la Gestión de la Producción a la Gestión de la Cadena de Suministro (Online) Holguín: Editorial Uho, 2003. www.monografias.com (consultado: 6 de abril de 2005).

pagos, las reclamaciones, etc.

- **DPP- *Direct Product Profitability***. Forma estándar de determinar la rentabilidad del producto para el cliente, considerando no solo el margen comercial, sino también la rotación del producto.
- **SPACEMAN-**. Extensión de la anterior en la que la rentabilidad del producto se mide en función de la superficie y de la inversión económica dedicada a dicho producto.

2.4.2. Colaboración en Marketing. Los beneficios derivados de esta colaboración se centran en un aumento de la facilidad para captar oportunidades de crecimiento en áreas de actividad, una racionalización de las inversiones en promoción, merchandising y desarrollo de productos. Hay tres áreas básicas de colaboración²⁹:

- **ESA- *Efficient Store Assortment***. Las decisiones tomadas por el detallista para maximizar las ventas (como la distribución del espacio de exposición, localización del producto, gama de productos, política de precio, marcas, etc) son compartidas con el proveedor. De esta manera se obtienen beneficios más importantes, ya que este aporta su conocimiento especializado sobre el producto y el tipo de consumidor.
- **EP- *Efficient Promotion***. Suministrador y detallista colaboran en campañas de promoción que pueden abarcar tres ámbitos diferentes:
 - Desarrollo de las ventas a través de campañas orientadas al consumidor.
 - Reducción de costos utilizando técnicas promocionales más racionales (*couponing* y eliminación de *packs* especiales)
 - Reducción de las compras especulativas que dificultan la programación logística eficiente.
- **EPD- *Efficient Product Launch and Development***. La introducción de nuevos productos es imprescindible en la actividad comercial puesto que mejora la posición del suministrador frente a la competencia y permite la detallista ofrecer productos más adaptados a las necesidades del cliente. La colaboración entre ambos en este campo se centra en los frentes:
 - Lanzamiento y desarrollo de nuevos productos.
 - Desarrollo de marcas blancas (marcas con el nombre del distribuidor y no del fabricante)

²⁹ Tomadas de PÉREZ, Marisol. De la Gestión de la Producción a la Gestión de la Cadena de Suministro (Online) Holguín: Editorial Uho, 2003. www.monografias.com (consultado: 6 de abril de 2005).

2.4.3. Colaboración con proveedores. A mediados de los años 90 se desarrolla un gran auge por los sistemas de Planificación de Recursos de la Empresa ERP (Enterprise Resource Planning), una arquitectura de software que facilita el flujo de información entre las funciones de manufactura, logística, finanzas y recursos humanos de una empresa y/o entre varias, con un enfoque hacia la organización.

En años más recientes el auge se trasladó a los sistemas de gestión de la cadena de abastecimiento SCM, (Supply Chain Management) una arquitectura de software que comprende la coordinación, programación, adquisición, producción, inventarios y entrega de los productos y servicios a los clientes, con un enfoque hacia fuera de la organización.

En primera instancia se podría pensar que los ERP son la base de los SCM, no obstante, los orígenes de ambos tipos de sistemas son muy distintos. Ambos sistemas manejan en alguna medida la misma información, sin embargo hay importantes diferencias en la manera en que abordan el problema de interacción con otras empresas (proveedores, clientes, etc...).

Una de las claves del éxito de los sistemas SCM, es la manera en que promueve la integración de las actividades, la cooperación, la coordinación y el intercambio de información a lo largo de toda la cadena, desde los proveedores hasta los clientes finales.

La práctica moderna va conduciendo a estructuras más ajustadas con menores costos donde la logística es un órgano de dirección al más alto nivel que organiza y regula todo el flujo material desde el aprovisionamiento hasta la distribución física, los servicios de post-venta y el flujo informativo asociado a tales actividades. El Sistema SCM es el encargado de mantener la integración que se requiere para que el sistema tenga éxito³⁰.

³⁰ PÉREZ, Marisol. De la Gestión de la Producción a la Gestión de la Cadena de Suministro (Online) Holguín: Editorial Uho, 2003. p. 13; www.monografias.com (consultado: 6 de abril de 2005).

2.4.4. Enfoques de Integración de las cadenas de Abastecimiento.

Existen dos orientaciones que se destacan en la integración de las cadenas de abastecimiento³¹:

Integración vertical: Se combinan dos o más etapas de la cadena bajo una dirección. Esto trae como resultado la compra de las operaciones de un eslabón de la cadena o la realización de las operaciones de este eslabón. Por ejemplo, un gran comerciante de ventas masivas, como las tiendas de descuento, pueden almacenar y transportar los productos que le compra el fabricante, con lo cual se elimina la necesidad de utilizar al mayorista.

Integración horizontal: consiste en combinar instituciones al mismo nivel de operaciones bajo una administración única. Un ejemplo serán las tiendas departamentales. Esta integración proporciona ahorros importantes en especialistas de publicidad, investigación de mercados, compras, etc. Y la puede llevar a cabo una organización al fusionarse con otras organizaciones o incrementando el número de unidades.

2.4.5. Uso de intervalos de recepción (ventanas de tiempo). El uso de intervalos de recepción se hace más útil conforme las empresas crecen y su número de proveedores y de productos aumenta. El proceso de recepción de mercancías comienza a ser controlado por sistemas especializados los cuales gestionan la recepción de materiales a un horario específico para cada proveedor. Este horario en el cual un proveedor específico será atendido se conoce como ventana de tiempo, debido a que si en ese intervalo de tiempo el proveedor ingresa a la bodega para entrega su mercancía no podrá hacerlo hasta se le sea asignada una nueva ventana. Mediante este requisito los clientes se aseguran de que sus bodegas de recepción tengan la capacidad suficiente de atender a determinado número de clientes y evita que se produzcan congestiones o esperas por mercancías.

Existen otras razones que han llevado a que algunos clientes establezcan un horario para la recepción de mercancías de sus proveedores, tal es el caso de métodos JIT de producción, hora de atención al público, manejo delicado del producto (alimentos), entre otros.

³¹ de CHAMOSO, Raúl. Sist. de Control y Distribución de Ventas (Online), 2004; p 11. www.monografias.com (consultado: 10 de abril de 2005)

Las ventanas de tiempo exigen una adecuada planeación para el ruteo de los proveedores ya que el arribo de la mercancía muy temprano supone costos de oportunidad debidos a la espera y el arribo tardío supone la pérdida de la entrega.

2.5. SISTEMAS DE INFORMACIÓN UTILIZADOS EN DISTRIBUCIÓN

Para hacer frente al reto de la integración, se necesitan herramientas sofisticadas de soporte a la decisión (DSS, Decision Support Systems) basados en métodos cuantitativos y la utilización de los avances en las tecnologías de la información y de la comunicación. Estos métodos cuantitativos consisten en modelos matemáticos de optimización que generalmente requieren de una cooperación con eslabones siguientes de la cadena de abastecimiento³².

Sistemas DRP Dentro de la cadena logística, el abastecimiento puede ser descentralizado, si un proveedor entrega la mercadería en distintas plantas, o centralizado, si la entrega es en un sólo lugar. En el primer caso se programa directamente a los proveedores para entregar en cada sitio. En el segundo, se utilizan dos herramientas: la programación de proveedores y las técnicas de DRP. Éstas permiten unir las demandas entre múltiples sitios, incluso en distintos países. Para el empleo del DRP, en primer lugar, se define una red en la que hay nodos o sitios, que son los centros de distribución, plantas, etc. Luego es establecen cuáles son las relaciones entre los nodos. Hasta este momento, el modelo es de definición paramétrica. Lo que hace mover a este modelo son unas variables definidas, partiendo de la demanda. La demanda se calcula sobre cada uno de los nodos que proveen los productos independientes al mercado. El objetivo de los pronósticos es partir de los niveles de existencia en cada nodo y las políticas de planeamiento por producto, en cada lugar, para determinar la demanda en cascada al eslabón anterior de la cadena de abastecimiento interna. Para cada producto se define el nivel de servicio, el stock que se desea mantener y el pronóstico de venta en cada lugar. El DRP también maneja el balanceo de cargas, trabajando sobre pesos y volúmenes, y también la demanda estacional³³.

³² SERRA, Daniel. Investigación operativa y logística empresarial (Online), Fabra:Organización de Empresas GREL-IET. 2004, p.2. www.almacenar.com (Consultado: 8 de mayo de 2005)

³³ DOMÍNGUEZ, José A. y otros. Dirección de Operaciones. Aspectos tácticos y operativos en la producción y los servicios, España: Editorial MC Graw Hill; 1999. 303p

Sistemas de planeamiento Los sistemas de planeamiento utilizan herramientas tales como pronósticos de venta, DRP, MRP, CRP y MPS (Programa Maestro de Producción); en distintos niveles. Pueden emplear pronósticos de venta y MPS para los productos terminados, y herramientas de MRP para los semielaborados, materias primas y materiales de empaque. Los sistemas de planeamiento calculan en qué momento se quebrará el stock de seguridad para saber cuándo debe reponerse la mercadería y, por lo tanto, el lead time (tiempo de aprovisionamiento) en función del cual se determina la fecha de liberación de los productos³⁴.

Sistemas de Estimación de la demanda Las posibilidades van desde pronosticar las ventas, conocer el inventario del siguiente eslabón de la cadena y coordinar con él, hasta llegar a acuerdos comerciales y logísticos donde se amplíe la base de conocimiento. En el balanceo entre la demanda y el abastecimiento, las ventas y los pronósticos de venta pueden tener picos y valles. Al planear la demanda se debe tratar de moderar esas fluctuaciones por medio de una mayor flexibilidad, más rapidez en los cambios³⁵.

2.6. TENDENCIAS DE LOS SISTEMAS LOGÍSTICOS

2.6.1. Respecto a los sistemas de operación. Las tareas futuras en el avance de los sistemas logísticos incluyen el desarrollo y la incorporación de mejores algoritmos heurísticos de solución y “clustering”, la generación y ejecución automática de modelos matemáticos de manera transparente al usuario, la mejora de las facilidades de ayuda existentes, la generación automática de información con el fin de mejorar la implementación actual de los operadores de mejora intra e inter-ruta, y la definición de interfaces con Sistemas de Información Geográfica³⁶.

2.6.2. Respecto a la integración. En la actualidad se puede percibir un cambio evidente en el estilo de vida de los consumidores ligado al avance de las comunicaciones (Internet, telefonía móvil, etc), cosa que favorece que se puedan hacer pedidos de productos desde cualquier lugar. Para afrontar estos cambios, la industria, debe evolucionar modificando la estructura de la cadena de suministro,

³⁴ Ibid., p. 305

³⁵ DOMÍNGUEZ, José A. y otros. Dirección de Operaciones. Aspectos tácticos y operativos en la producción y los servicios, España: Editorial MC Graw Hill; 1999. 305p

³⁶ BARROS, Oscar. VARGAS, Samuel. WEBER, Richard. Evaluación de Prácticas de Gestión en la Cadena de Valor de Empresas Chilenas. (Online) En Revista Serie Gestion N°44, p. 40 Enero, 2004; www.obarros.cl (consultado: 9 de Mayo 2005)

las relaciones entre sus componentes y los roles que desempeñan. Estas relaciones se basaran en compromisos a largo plazo edificados sobre la confianza entre la totalidad de los componentes de la cadena³⁷.

Las principales ventajas que pueden derivarse de la integración son las siguientes:

- Generación automática de las órdenes de pedido a lo largo de la cadena sobre la base de la información directa del punto de venta.
- Sistema *pull* de extremo a extremo de la cadena.
- Ausencia de inventarios a causa del reaprovisionamiento continuo y exacto.

³⁷ Ibid., p 42

3. EL PROBLEMA DEL RUTEO

3.1. DEFINICIÓN DE RUTEO Y SCHEDULING

El problema de Ruteo y “Scheduling” (R&S) de vehículos según Paez³⁸ “aborda la definición de las mejores rutas y/o programas para coleccionar/entregar bienes o pasajeros en un sistema de distribución/recolección”. El interés por métodos de R&S satisfactorios ha sido estimulado por varios factores, quizás, el más importante de estos es la globalización y su impacto directo en las actividades logísticas”.

Un sistema de ruteo consiste en varios vehículos de capacidad conocida que operan desde una o múltiples bases/depositos con el objeto de visitar a un conjunto de puntos de servicio de localización conocida. Los problemas reales de ruteo de vehículos comprenden una amplia variedad de restricciones que se vinculan a vehículos, a clientes o a ambos, además, existen diferentes características que deben ser contempladas, así como las posibles opciones de solución para cada una. Como resultado, de considerar estas distintas características y restricciones, y las diferentes medidas de desempeño, surge una amplia variedad de problemas de Ruteo, que a su vez han sido abordados con una multiplicidad de enfoques (modelos de programación matemática, algoritmos basados en grafos y genéticos, “simulated annealing”, otros métodos heurísticos, etc.)³⁹.

³⁸ PAEZ, Bernardo. GONZALES, Pedro. Router: Ambiente Amigable Para Ruteo Y “Scheduling” De Vehiculos (online) En Congreso Encuentro de jóvenes Investigadores (Sexta edición: 2005:Rosario), p 2 , www.universia.com.ar/contenidos/investigacion/unl/TECNOLOGIA/ingenieria/63.htm (consultado: 8 de mayo de 2005)

³⁹ Ibid., p. 4

3.2. FACTORES A CONSIDERAR EN SISTEMAS DE RUTEO

Varios problemas importantes se presentan dentro del área de la Distribución Física. A nivel estratégico se debe decidir sobre la capacidad y localización de instalaciones (plantas, almacenes), a nivel táctico, la definición del tamaño y mezcla de los medios de transporte, y a nivel operativo, la determinación de la ruta de los vehículos, y su programación para satisfacer limitaciones de tiempo y capacidad⁴⁰.

La particularidad de cada empresa hace incluso que el objetivo de estos sistemas varíe de una a otra, siendo para algunos casos prioritaria la disminución en el costo, para otras el tiempo, para otras la distancia total, etc..

Los aspectos principales que deben ser considerados en los problemas de ruteo son^{41 42}:

Tipos de Vehículos:

- Pasajeros
- Buses
- Taxis.

Sistemas de viajes:

- Recoger y entregar
- Circuito
- Por Prioridades
- Ida y vuelta entre orígenes y destinos

Flota:

- Un vehículo
- Múltiples vehículos:
 - Vehículos iguales
 - Vehículos de distinto tipo (costo, capacidad, etc.)

Relaciones Contractuales:

- Vehículos propios: Permiten mayor control y seguridad de servicio.

⁴⁰ COMAS, Jose. Ruteo de vheiculos. (online). Valencia:Universidad politecnica Diapositiva 4 www.mat.up.es/%7EEeqvm/opp.html (Consultado: 8 de mayo de 2005)

⁴¹ Ibid., Diapositivas 8-11

⁴² AUTOR DESCONOCIDO. Gestión de Operaciones. Logística del transporte(Online); www.monografias.com (consultado: 7 de Mayo de 2005).

- Vehículos subcontratados: Permiten descentralización administrativa
- Sistema mixto: Vehículos propios y subcontratados

Número de depósitos (orígenes):

- Único
- Múltiples

Número de destinos Finales:

- Único
- Múltiples

Demanda:

- Determinística
- Estocástica:
 - En volumen
 - En localización
 - En tiempo de viaje

Componentes de Costo:

- Costos fijos:
 - De inversión
 - De operación (sueldos, patente, etc.)
- Costo variables de operación

Frecuencia de viajes:

- Múltiples por día
- De duración de más de un día

Tiempos de trabajo:

- Con sobretiempo
- Sin sobretiempo

Tiempos y distancias de viaje:

- Conocidos
- Estimados
- Aleatorios

Objetivos:

- Minimizar costos reales
- Minimizar costos de contrato
- Minimizar tiempo de servicio
- Maximizar la calidad del servicio (clientes atendidos a tiempo)
- Minimizar el riesgo de no cumplir la demanda
- Maximizar beneficio neto

Algunos otros Factores adicionales:

- Ventanas de tiempo
- Problemas de congestión
- Camiones con compartimentos para distintos productos

- Enlace cliente-vehículo
- Transbordos: camión-tren
- Prohibición de transitar ciertos vehículos en ciertas áreas a determinadas horas
- Además de entregar se requiere recoger artículos.
- Limitación en el tiempo total del viaje.
- Limitaciones de precedencia entre clientes.
- Entrega a ciertos clientes es opcional incurriéndose en castigo.

Enfoques de Solución:

- Métodos manuales: Usando mapas, afiches, cartas Gantt, pizarras...
- Solución por lógica o experiencia
- Modelos heurísticos
- Modelos matemáticos exactos

3.3. TIPOS DE PROBLEMAS DE RUTEO DE VEHÍCULOS

En la literatura actual se identifican 3 tipos básicos de problemas de ruteo:

- Encontrar una ruta en una red donde el origen es diferente al punto destino.
- Definir rutas de transporte entre múltiples orígenes y destinos.
- Rutear vehículos cuando el origen y el punto destino son los mismos.

Dentro del tercer tipo de problema se incluyen diversos modelos como:

- El agente viajero (TSP)
- Ruteo de vehículos (VRP)
- El problema de recoger y entregar (PDP)
- El problema de ruteo e inventario (IRP)
- El VRP con ventanas de tiempo (VRPTW)

3.4. DESCRIPCIÓN DE PROBLEMAS DE RUTEO CON MISMO ORIGEN Y DESTINO

Este tipo de problemas es enfrentado mediante la creación de modelos según las características del problema. Algunos problemas se han destacado por su cotidianidad y han recibido gran atención en la industria y en la academia teniendo un nombre establecido. Los más destacados son⁴³:

Problema del Agente Viajero: Consiste en encontrar la secuencia de puntos que el agente debe visitar iniciando y regresando al mismo punto, minimizando la distancia o tiempo total.

Problema de Ruteo de Vehículos: Un número de vehículos con capacidad limitada debe servir un conjunto determinado de consumidores con cierta demanda, satisfaciendo sus limitaciones. El objetivo es el de asignar vehículos a consumidores y definir la secuencia de visitas para minimizar costos, distancias o tiempos.

El problema de Recoger y Entregar: Al problema anterior se le incluye la posibilidad de recoger artículos.

Problema de Ruteo e Inventarios: Un almacén central con capacidad ilimitada sirve un conjunto de detallistas con cierta demanda fija. El objetivo es el de asignar vehículos y determinar sus cargas para reponer inventarios de los detallistas al mínimo costo de transporte e inventarios.

Problema de ruteo de vehículos con ventanas de tiempo: al VRP se le agrega la obligación de entregar los pedidos dentro de un intervalo de tiempo limitado, específico para cada cliente.

⁴³ Tomados de: COMAS, Jose. Ruteo de vheiculos. (online). Valencia:Universidad politecnica Diapositiva 16-19 www.mat.up.es/%7EEqvm/opp.html (Consultado: 8 de mayo de 2005)

4. PROBLEMAS DE DECISIÓN PRESENTES EN RUTEO

Se debe entender el concepto y genealogía de los problemas para comprender de una forma mas adecuada el problema que se afronta en esta investigación.

4.1. PROBLEMAS DE DECISIÓN

De acuerdo al diccionario de la Real Academia Española, un problema matemático es: "Proposición dirigida a averiguar el modo de obtener un resultado cuando ciertos datos son conocidos.⁴⁴". Como tal, un problema incluye una interrogante referida a un conjunto (posiblemente desconocido) de parámetros, y un enunciado acerca de las propiedades que la solución debe tener.

Al especificar los valores de los parámetros, se especifica una instancia del problema. El tiempo necesario para descubrir la solución suele ser una función de la instancia, sin embargo, expuesto de esta manera, el concepto de problema es aún muy ambiguo para ser de utilidad.

La primera clasificación que se puede hacer es de si se trata de un problema de decisión o no. Un problema de decisión es un problema cuyas únicas respuestas pueden ser "si" o "no". "¿Existe vida en Marte?" es un problema de decisión con instancia única; pero no es del tipo de problemas cuya solución pueda hallarse a través de algoritmos computacionales sino empíricos, ya que requiere de la experimentación para decidirse⁴⁵.

Por el contrario, "¿Existe un número positivo A que sumado a 2 resulte en 3?" es un problema de decisión matemático que se puede verificar aplicando un algoritmo.

⁴⁴ REAL ACADEMIA ESPAÑOLA. Diccionario de la Lengua Española; edición 21, Madrid: Editorial Espasa, 1992.

⁴⁵ CÓRDOVA, Juan. Optimización a través de colonias de hormigas. San Carlos:Universidad de San Carlos, p 2. 2004.

4.2. LA NATURALEZA DE LOS PROBLEMAS DE DECISIÓN

Casi cualquier problema⁴⁶ puede convertirse o en un problema de decisión, o en una colección de ellos, incluso problemas que a simple vista no lo son. Por ejemplo: Hallar un x entero que satisfaga la ecuación $x^2 = 25$, puede convertirse primero en “¿Existe algún x entero cuyo cuadrado sea igual a 25?” y si la respuesta es “sí” se procede a preguntar: “¿Es $0^2 = 25$?”, “¿Es $1^2 = 25$?”, hasta encontrar el primero de tales problemas de decisión que responda “sí” y con él validamos la solución propuesta.

Claramente tal forma de proceder no tiene garantía de encontrar una solución. Y, aún cuando la solución exista no resulta ser la forma más eficiente de hallarla. Para poder resolver problemas y comparar los métodos mediante los cuales se obtienen las soluciones (algoritmos) es necesario expresar el problema y el método en algún tipo de idioma universal. El instrumento habitual que ejecuta al algoritmo (método) es una computadora universal y el problema debe codificarse razonablemente para que los tiempos de ejecución del algoritmo propuesto sean significativos⁴⁷.

Hay que agregar que no todos los algoritmos pueden resolver (responder “sí” o “no”) un problema de decisión. De hecho existen problemas de decisión que no pueden resolverse en un sistema formal. Cabe anotar que los problemas de ruteo con aquí está caracterizado por tener tiempos de ejecución finitos⁴⁸ y por lo tanto, en principio, puede ser resuelto.

4.2.1. Clasificación de los problemas de decisión. Primero, se tiene que definir lo que se llama tiempo de ejecución. Éste es el tiempo que transcurre entre el momento en que el algoritmo comienza su ejecución y el momento en que entrega la una solución definitiva, este tiempo es proporcional al número de instrucciones básicas que la computadora ejecuta, bajo la dirección del algoritmo, al resolver el problema. Midiendo este tiempo de ejecución se puede calcular la eficiencia de un algoritmo sobre otro. Tal eficiencia esta dada en términos del orden del algoritmo. Un algoritmo es de orden polinomial si su tiempo de ejecución

⁴⁶ Para problemas que puedan resolverse a través de algoritmos matemáticos.

⁴⁷ CORDOVA, Juan. Optimización a través de colonias de hormigas. San Carlos:Universidad de San Carlos, p 2. 2004.

⁴⁸ GAREY, M. JOHNSON ,D. Computers and Intractability A Guide to the Theory of NP-Completeness; New York :W. H. Freeman and Company, 1997.

esta limitado por una función polinomial en el tamaño de la entrada. Un problema es polinomial determinístico si existe algún algoritmo polinomial que lo resuelva sin adivinar. Tales problemas pertenecen a la clase **P** de problemas⁴⁹.

Segundo, se debe verificar si las conclusiones que se derivan respecto al problema y algoritmo en la computadora universal son generalizables o no. Afortunadamente en el medio computacional actual sí lo son, y por ello no es tan importante la elección específica de la plataforma computacional sobre la cual se trabaja⁵⁰.

Estos últimos dos puntos permiten subclasificar los problemas de decisión en aquellos que se pueden resolver en tiempo polinomial usando un algoritmo determinístico y aquellos que se pueden resolver en tiempo polinomial utilizando un algoritmo que adivina (o no-determinístico). Los primeros pertenecen a la clase **P** de problemas y los últimos a la clase **NP**⁵¹.

Como lo expresa Cordova “si un problema pertenece a **NP** debe existir un algoritmo determinístico que lo resuelva en un tiempo de ejecución exponencial. Aquí es donde reside la dificultad de los problemas que pertenecen a **NP**: de resolverse con algoritmos determinísticos sólo puede garantizarse un tiempo de ejecución exponencial y esto implica que para instancias suficientemente grandes no hay mucha esperanza de resolverlos rápidamente”⁵².

Varios problemas de optimización caen en la categoría de problemas pertenecientes a **NP**. Un subconjunto de éstos pertenece a una clase denominada “**NP-complete**” o “**NP-Duros**” cuyos miembros representan algo así como problemas “típicos” en el sentido de que si un problema pertenece a **NP-complete** debe existir una transformación (de tiempo polinomial) que convierte cualquier problema de **NP** en este particular. “Los problemas de **NP-complete** son, los problemas más difíciles de la clase **NP** y resolverlos eficientemente implicaría que todos los problemas de **NP** podrían ser resueltos eficientemente”⁵³.

⁴⁹ CORDOVA, Juan. Optimización a través de colonias de hormigas. San Carlos:Universidad de San Carlos, p 2. 2004.

⁵⁰ GAREY, M. JOHNSON ,D. Computers and Intractability A Guide to the Theory of NP-Completeness; New York :W. H. Freeman and Company, 1997

⁵¹ CORDOVA, Juan. Op. cit., P 4.

⁵² Ibid., p 4

⁵³ GAREY, M. JOHNSON ,D. Op. cit., p 23

4.3. CLASES DE PROBLEMAS DE RUTEO.

Los problemas que se enfrentan en ruteo son fundamentalmente de dos tipos: de optimización y de diseño. En el primer caso, dado un sistema ya existente se trata de encontrar los parámetros de funcionamiento que permitan aproximarse lo más posible a un objetivo o conjunto de objetivos. La solución que se busca en estos problemas es de carácter operativo (rutas a tomar, cargas a transportar..etc) y táctico (tamaño del flota), y no permite tomar decisiones respecto a cambios de infraestructura.

En el segundo caso, se trata de diseñar el sistema desde el principio para que se cumplan los objetivos prefijados. La solución que se busca en estos problemas es de carácter estratégico (ubicación de la bodega, capacidad de almacenaje).

4.3.1. Los Problemas de optimización. Si un problema está formulado de una manera uniforme, y el espacio de todas las soluciones posibles es conocido, un método de búsqueda trata de encontrar las soluciones que satisfagan el enunciado del problema. Dicho así parece una redundancia, pero no todos los problemas son problemas de búsqueda, ni todos los espacios de búsqueda se pueden definir también de forma uniforme⁵⁴.

Si un problema se puede efectuar como problema de búsqueda, en muchos casos se puede reducir a un problema de hallar un máximo o mínimo, o sea, un óptimo. Pero esto sólo va a ser posible en el caso de que “se pueda calcular la bondad de una solución: la solución del problema será aquella, o aquellas, que optimicen una función de bondad, ajuste, evaluación o fitness; en muchos casos, por lo tanto, un problema de búsqueda se puede reducir a un problema de optimización (maximización o minimización)”⁵⁵.

⁵⁴ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

⁵⁵ Ibid., p. 3

4.4. LOS PROBLEMAS CON RESTRICCIONES

Los problemas con restricciones tienen una larga tradición. Una restricción es simplemente una condición que tiene que cumplir la solución, habitualmente formulada como una función sobre una combinación de las variables del espacio de búsqueda. En el caso del ruteo, una restricción puede consistir en que no se pueda pasar por un cliente dos veces. En otros casos, se expresan como una desigualdad; por ejemplo, la suma de las variables debe ser 0. En un problema de asignación de trabajos a revisores en un congreso, por ejemplo, ningún revisor deberá revisar más de 6 trabajos, y preferentemente, no menos de 5. En este caso se trata de una restricción suave, que se puede incumplir si se amerita; en el caso de las restricciones fuertes, la solución será inválida en el caso de que se incumplan⁵⁶.

Las restricciones fuertes se pueden manejar de múltiples maneras válidas. En algunos casos, el problema serio puede ser manejar las restricciones suaves. Se puede optar simplemente por pasar de ellas; pero si no se pasa, hay que asignarles un peso, o una prioridad, con lo cual, el manejo de las restricciones suaves se convierte, en sí, en un problema.

4.5. OPTIMIZACIÓN MULTIOBJETIVO

Al planear resolver un Problema de Ruteo donde no sólo se desee minimizar distancia y se consideran otros criterios para la determinación de la ruta, es necesario introducir en el análisis herramientas teóricas adicionales que permitan un análisis del problema con varios criterios de decisión. La nueva estructura teórica es conocida como optimización multiobjetivo.

En la definición del problema objeto de este proyecto, se incluyen ya, cuatro criterios a tener en cuenta, por un lado minimizar el número de vehículos, por el otro la distancia a recorrer cumpliendo con ventanas de tiempo, por otro lado el tiempo de recorrido total y por ultimo el tiempo de atención máximo. El “problema” va a estar en que los criterios están en conflicto, lo cual implica que cuando

⁵⁶ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

estamos mejorando uno de los criterios, al menos vamos a estar empeorando otro (u otros), haciendo imposible que exista dicha alternativa⁵⁷, por tanto, es necesaria una metodología que permita desarrollar el problema como un Problema de Ruteo Multiobjetivo.

La optimización multiobjetivo (u optimización multicriterio, multiprestaciones o optimización vectorial) se define como el problema de encontrar un vector de variables de decisión que satisfagan unas restricciones y optimicen una función vectorial cuyos elementos representan a la función objetivo. Estas funciones forman una descripción matemática de criterios de evaluación que generalmente están en conflicto unos con otros. Por tanto, el término "optimizar" implica encontrar una solución que daría los valores de todas las funciones objetivo aceptables para el diseñador⁵⁸.

En general no existirá una combinación que sea a la vez solución de todos los objetivos. De esta forma el primer concepto que se pierde es el de óptimo tal y como se entiende en la programación monoobjetivo tradicional, y lo que buscaremos para solucionar nuestro problema serán las denominadas soluciones eficientes o Pareto (soluciones que presentan un equilibrio entre los objetivos) .

El concepto de óptimo fue propuesto originalmente por Edgeworth, y más adelante replanteado por Pareto por lo que se le suele denominar óptimo Pareto u óptimo tipo Pareto. Un vector es óptimo Pareto si no existe ninguno cuyos valores sean mejores para cada uno de los componentes (estrictamente, si son menores o iguales para todos, y al menos menor estricto para uno de ellos) o descrito de otra manera, es un óptimo que responde a la representación de las preferencias de un decisor por un orden parcial de tipo paretiano. Lo más habitual es que no exista un solo óptimo Pareto, sino un conjunto de ellos; a este conjunto de soluciones se le suele denominar conjunto no dominado, y a su gráfico se le suele llamar frente Pareto⁵⁹ . .

⁵⁷ SAAVEDRA, Caridad. FOSADO, Osvaldo. FERNANDEZ, Pedro. Tratamiento Multicriterio de Problemas de Ruta solucionados a través de Técnicas de Simulación Monte Carlo (Online) Cuba: Universidad de Pinar del Río. www.monografias.com (consultado: 8 de Mayo de 2005).

⁵⁸ NESMACHNOW, Sergio. Una Versión Paralela del Algoritmo Evolutivo para Optimización Multiobjetivo NSGA-II y su Aplicación al Diseño de Redes de Comunicaciones Confiables. En Actas del Tercer Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados. Córdoba: España. 2004.

⁵⁹ CRICHIGNO, Jorge. TALAVERA, Francisco. PRIETO, Joel. BARÁN, Benjamín. Enrutamiento Multicast Utilizando Optimización Multiobjetivo p. 6. Asunción: Centro Nacional de Computación. 2004

La mayoría de las técnicas de resolución de problemas de Toma de Decisiones Multiobjetivo tratan de, en algunos casos, encontrar una buena representación del conjunto de soluciones Pareto, y en otros, a partir de unos niveles de aspiración conseguir, en el caso que le sea posible, una alternativa Pareto que satisfaga dichos niveles (solución satisfactoria), y en el caso que no le sea posible, una alternativa Pareto que más 'se acerque' en alguna medida a dichos niveles (solución no satisfactoria).

4.6. ESFUERZO COMPUTACIONAL ASOCIADO A LOS PROBLEMAS DE RUTEO

Una consideración importante tanto en la formulación como en la solución de los problemas de Ruteo es el esfuerzo computacional asociado. Dado que la mayoría de los problemas de ruteo son NP-Complete, los métodos empleados para hallar soluciones óptimas sufren un crecimiento exponencial de la carga computacional con el tamaño del problema. Las limitaciones vienen dadas en función de la capacidad de memoria del computador para trabajar con un conjunto extremadamente alto de datos como los generados por los problemas de ruteo grandes, y por la velocidad de procesamiento requerida para evaluar toda la superficie de respuestas del problema en tiempos razonablemente útiles.

4.7. ENFOQUES UTILIZADOS PARA RESOLVER EL PROBLEMA DE RUTEO

En la actualidad, los enfoques utilizados para la solución del problema de ruteo se dividen en aquellos que utilizan métodos matemáticos exactos y los que utilizan algoritmos heurísticos⁶⁰ y operadores de mejora para determinar una solución al problema. Sin embargo, los diferentes tipos de problemas presentan diferentes comportamientos al ser abordados por uno u otro método. "En general la tendencia es utilizar métodos matemáticos siempre y cuando sea factible obtener soluciones útiles en un tiempo de computación razonable y recurrir a métodos heurísticos cuando esto no sea posible"⁶¹.

⁶⁰ Véase Siguiendo capítulo

⁶¹ BACILIERI, Nino. GARCIA, William. Algoritmos Heurísticos. En: III Congreso Colombiano y I Conferencia Andina Internacional De Investigación De Operaciones, Cartagena. 2004

5. CONCEPTOS DE HEURÍSTICA Y ALGORITMOS HEURÍSTICOS

Como se definió anteriormente, los problemas de ruteo son del tipo NP-Complete. Lo cual nos garantiza únicamente una solución determinística en un tiempo exponencial. Para comprender la magnitud de este problema, se presenta a continuación un esquema de una solución determinística del problema típico del agente viajero:

El agente debe viajar por 3 ciudades: A, B, C, los recorridos posibles serán:

- A B C
- A C B

Se evalúa la distancia para cada uno y se escoge el menor.

Ahora, el agente debe viajar por 4 ciudades: A, B, C, D, los recorridos posibles serán:

- A B C D
- A B D C
- A C B D
- A C D B
- A D B C
- A D C B
-

Se evalúa la distancia para cada ruta y se escoge el menor.

Si el agente debiera viajar por 5 ciudades tendería 24 rutas posibles, y si fueran 6 ciudades tendrían 120, así, a medida que aumentan las ciudades a visitar, aumenta de forma exponencial el número de posibles rutas a evaluar.

En realidad las empresas no tendrían problemas para encontrar las rutas óptimas si tuvieran 6 o menos clientes, pero para aquellas empresas grandes que deben entregar a n clientes, sabiendo que el número total de recorridos diferentes es de $(n - 1)!$ el problema es otro⁶².

⁶² TORRES, Marcelo. Algoritmos heurísticos para resolución del vendedor viajero. (online) Santiago de Chile: USACH, 2004. <http://optimos2.diinf.usach.cl/VendedorViajero/Teoria.htm> (Consultado: 16 de Mayo de 2005)

Si la empresa tuviera 20 clientes tendría 121645100408832000 rutas asimétricas posibles.

Si tuviera 100 clientes tendría 9.3326×10^{155} rutas asimétricas posibles.

En este caso el tiempo y el costo computacional necesario para hallar la ruta optima seria muy elevado como para que la solución sea útil (la disminución en los costos por usar la ruta optima es menor que el costo de hallar esa ruta).

Para problemas de esta naturaleza se hace admisible encontrar soluciones suficientemente buenas (cercanas al óptimo) siempre que puedan encontrarse en un marco temporal razonable.

5.1. DEFINICIÓN DE HEURÍSTICA

Por definición “la heurística es el estudio del método y reglas de descubrimiento e invención. Esta es la naturaleza de la heurística, la de buscar reglas de búsqueda mas que el dominio del problema, basados en la experiencia y en la intuición, y de usar esas reglas para guiar la selección, eliminación y generación de soluciones”⁶³

Los métodos que aplican heurística para la solución de problemas tienen una justificación intuitiva, por ello no garantizan la producción de una solución óptima, en algunos casos solo están diseñados para vencer la complejidad del problema.

Los métodos heurísticos están asociados con búsquedas heurísticas. El principal objetivo de los métodos heurísticos es el de cortar el espacio de búsqueda de la solución, recobrase de los fallos, y encontrar una solución tan cercana a la optima como sea posible⁶⁴.

⁶³ BACILIERI, Nino. GARCIA, William. Algoritmos Heuristicos. En: II congreso latinoamericano de Investigación de Operaciones. Memorias, Cartagena. 2004

⁶⁴ Ibid., p 5.

5.2. APLICACIÓN DE LOS MÉTODOS HEURÍSTICOS

Los métodos heurísticos pueden ser usados en dos diferentes situaciones:

- En donde un problema no puede tener una solución exacta, con ambigüedades inherentes en la declaración del problema, o en los datos disponibles.
- En donde un problema tiene una solución exacta, pero el tiempo y costo computacional son prohibitivos. El estado de crecimiento de ciertos problemas es explosivo, con un número de posibles estados de crecimiento incrementándose exponencial o factorialmente mientras el algoritmo procede.

5.3. PATRONES DE INFERENCIA PLAUSIBLE

El proceso heurístico de descubrimiento en matemáticas está guiado por mecanismos de inferencia no deductivos, que tienen mucho trabajo de "adivinanza" y que aunque no son totalmente certeros, son signos de progreso en la solución de un problema⁶⁵. Los principios que gobiernan este tipo de razonamiento se conocen como patrones de inferencia plausible.

Para algunos autores el trabajo de adivinanza sigue ciertos patrones que pueden caracterizarse. Entre los patrones de razonamiento plausible más discutidos se encuentran definidos por Polya⁶⁶:

5.3.1. Patrones Inductivos. "La verificación de la consecuencia hace que la conjetura sea más creíble". Por ejemplo, la conjetura "llovió anoche" se hace más creíble cuando se verifica la consecuencia "el patio está mojado".

⁶⁵ POLYA. Mathematics and Plausible Reasoning. Volume II Patterns of Plausible Inference. p 54 Princeton :University Press, 1968.

⁶⁶ Tomados de: ATOCHA Aliseda Heurística, hipótesis y demostración en matemáticas (online) Mexico D.F.:Instituto de Investigaciones Filosóficas. <http://minerva.filosoficas.unam.mx/> (consultado: 12 de mayo de 2005)

5.3.2. Verificación sucesiva de varias consecuencias "La verificación de una nueva consecuencia cuenta más (o menos) si la nueva consecuencia difiere más (o menos) de la primera de las consecuencias verificadas". Por ejemplo, si se trata de corroborar la conjetura: "todos los cuervos son negros" y observamos n cuervos americanos, todos ellos negros, nuestra credibilidad en esta conjetura se incrementará sustancialmente si el cuervo (n_1) es un cuervo negro de Australia y no de América.

5.3.3. Verificación de consecuencias improbables. "La verificación de una consecuencia cuenta más (o menos) dependiendo de si la consecuencia es más (o menos) probable en si misma" Por ejemplo, la conjetura "llovió anoche" es más creíble con la evidencia "el techo está goteando" que con la observación más común: "el pasto está mojado".

5.3.4. Inferencia por Analogía. "Una conjetura se hace más creíble cuando una conjetura análoga resulta ser verdadera". Por ejemplo, la conjetura "De todos los objetos con el mismo volumen, la esfera es la que tiene menor superficie" se hace más creíble cuando se prueba el teorema relacionado "De todas las curvas que tienen la misma área, el círculo es el de perímetro más corto".

Estos patrones presentan modos de razonamiento inciertos, en donde se muestran condiciones para hacer más o menos creíble una conjetura. La credibilidad depende en gran medida de la verificación de las consecuencias asociadas a la hipótesis (como los primeros tres patrones plausibles) o de probar conjeturas análogas (como en el último patrón). A este tipo de conclusiones se les llama "conclusiones heurísticas". No son certeras, pero guían el camino a aquellas conjeturas que posteriormente se demuestran formalmente⁶⁷.

El razonamiento plausible se rige de reglas, aunque éstas no son como las reglas que caracterizan al razonamiento demostrativo, en el cual las conclusiones a las que se llega son totalmente certeras.

⁶⁷ ATOCHA Aliseda Heurística, hipótesis y demostración en matemáticas (online) Mexico D.F.:Instituto de Investigaciones Filosóficas. <http://minerva.filosoficas.unam.mx/> (consultado: 12 de mayo de 2005)

5.4. SILOGISMO HEURÍSTICO (DE LAS SOLUCIONES HEURÍSTICAS)

El comportamiento inteligente de una computadora se modela en general diseñando estrategias que convierten una búsqueda exhaustiva, como por ejemplo calcular todas las jugadas posibles en una partida de ajedrez, en una búsqueda heurística, es decir aquella que solo calcula las jugadas que tienen posibilidad de éxito, de acuerdo a una función de evaluación predeterminada.

Así, la noción de heurística varía de uno a otro campo. Mientras que en matemáticas se identifica con el descubrimiento de teoremas o de soluciones a problemas, en inteligencia artificial la heurística se ha incorporado con acciones específicas, como el diseño de estrategias de búsqueda inteligente.

Este razonamiento⁶⁸ se reconoce como la base de estudios en filosofía de la ciencia. El enfoque matemático en particular, fue profundizado por Lakatos⁶⁹ en el libro "pruebas y refutaciones" donde se argumenta que "aunque no hay una lógica del descubrimiento que obtenga resultados con certeza, si hay una lógica falible del descubrimiento, la lógica del progreso científico, que no es ni psicología ni lógica, sino una disciplina independiente, la lógica de la heurística".

Sin embargo, la propuesta sobre los patrones de razonamiento plausible no es considerada en la lógica contemporánea como estimable de ser estudiada como inferencia lógica. Un matemático o un lógico dirían que si la conclusión de una forma argumentativa no es certera, no sirve. De hecho, al silogismo heurístico presentado se le conoce más popularmente como la falacia de afirmar el consecuente⁷⁰. La conclusión de este silogismo heurístico no es certera, es solo una pista, una hipótesis (ej. tal vez esté lloviendo) que puede ser confirmada o refutada con información adicional.

⁶⁸ Refiriéndose específicamente a los pensamientos de Polya.

⁶⁹ LAKANTOS, Proofs and Refutations. The Logic of Mathematical Discovery. Cambridge:University Press. 1976.

⁷⁰ ATOCHA Aliseda Heurística, hipótesis y demostración en matemáticas (online) Mexico D.F.:Instituto de Investigaciones Filosóficas. <http://minerva.filosoficas.unam.mx/> (consultado: 12 de mayo de 2005)

5.5. METAHEURÍSTICAS

Desafortunadamente, los algoritmos de mejora iterativos pueden estancarse en soluciones de baja calidad (óptimos locales muy lejanos al óptimo global). Para permitir una mejora adicional en la calidad de las soluciones, la investigación en este campo en las últimas dos décadas se ha centrado en el diseño de técnicas de propósito general para guiar la construcción de soluciones o la búsqueda local en las distintas heurísticas. Estas técnicas se llaman comúnmente metaheurísticas y consisten en conceptos generales empleados para definir métodos heurísticos⁷¹. 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 si ya existe previamente algún método heurístico específico para el problema. De hecho, las metaheurísticas son ampliamente reconocidas como una de las mejores aproximaciones para atacar los problemas de optimización combinatoria.

Las metaheurísticas incorporan conceptos de muchos y diversos campos como la genética, la biología, la inteligencia artificial, las matemáticas, la física y la neurología, entre otras. Algunos ejemplos de metaheurísticas son: Enfriamiento simulado, búsqueda tabú, búsqueda local iterativa, GRASP (greedy randomized adaptative search procedures) y por supuesto una de las más recientes, la Optimización basada en Colonias de Hormigas (OCH).

A diferencia de las heurísticas clásicas, los algoritmos metaheurísticos hacen énfasis en realizar una exploración profunda de las regiones más prometedoras de la superficie de soluciones y no de toda la superficie como era habitual. La solución obtenida con respecto a las heurísticas normales es de mayor calidad y por lo general encontrada en menor tiempo

5.6. HIPERHEURÍSTICAS

Hay dos tipos básicos de heurísticas de acuerdo a su nivel de conocimiento del problema: las heurísticas de bajo nivel de abstracción que tratan directamente con

⁷¹ TORRES, Marcelo. Algoritmos heurísticos para resolución del vendedor viajero. (online) Santiago de Chile: USACH, 2004. <http://optimos2.diinf.usach.cl/VendedorViajero/Teoria.htm> (Consultado: 16 de Mayo de 2005)

el problema y las hiperheurísticas que consisten en una o varias heurísticas de mayor nivel de abstracción que trabajan administrando heurísticas de menor nivel.

Una hiperheurística se puede definir como una metaheurística de alto nivel que administra otras metaheurísticas más específicas de acuerdo al problema en cuestión. El objetivo principal de una hiperheurística es proporcionar un nivel de generalidad más alto del que por lo general una metaheurística proporciona⁷².

La motivación detrás del desarrollo de hiperheurísticas es el siguiente: Una heurística es aplicable y aceptable en una situación pero en otra se vuelve evidente la necesidad de aplicar una heurística completamente diferente o antagónica. Esto nos lleva a pensar que “dado un conjunto de heurísticas aplicables debe existir un conjunto óptimo (o al menos uno sumamente adecuado) que nos lleve a un estado meta; la manera en la cual se arbitran estas heurísticas (que son de dominio específico) es tarea de una hiperheurística”⁷³.

Ahora bien, dicho conjunto de heurísticas puede crecer mientras el algoritmo itera e incluso dichas heurísticas pueden ser parametrizadas cambiando la transformación de estados a los que se presente esta heurística, también es factible añadir nuevas heurísticas por medio de un ciclo de aprendizaje. Otra opción a explorarse es considerar una hiperheurística como un agente administrador de otros agentes (con heurísticas y restricciones) en un ambiente multiagente como es el caso de los algoritmos de colonias de hormigas.

Una hiperheurística tiene la característica importante de no contener conocimiento del problema que se está resolviendo, solo tiene acceso a un conjunto de heurísticas más específicas que si interactúan con el problema a solucionar. Pero esto de ninguna manera evita que una hiperheurística contenga un estado interno, este estado puede contener información sobre el desempeño de las heurísticas que opera, lo que abre camino a la posibilidad de explorar combinaciones de heurísticas, afinar su desempeño en línea, u observar como operan al ser aisladas.

⁷² TORRES, Marcelo. Algoritmos heurísticos para resolución del vendedor viajero. (online) Santiago de Chile: USACH, 2004. <http://optimos2.diinf.usach.cl/VendedorViajero/Teoria.htm> (Consultado: 16 de Mayo de 2005)

⁷³ Ibid.

5.7. TÉCNICAS HEURÍSTICAS DE BÚSQUEDA DE SOLUCIONES

En general, los métodos de búsqueda heurísticos y los métodos clásicos se dividen, a grandes rasgos, en métodos globales y locales. Los métodos globales tratan de encontrar el máximo global de un problema, mientras que los locales se concentran en la vecindad de la solución generada inicialmente, y por tanto, necesitan alguna técnica adicional, como comienzos múltiples, para acercarse al máximo global⁷⁴.

En general, los métodos locales no tienen ninguna garantía de que el máximo encontrado sea global, ni una medida del error en que se incurre dando como bueno el máximo encontrado; los globales sí suelen tener una de las dos cosas, o las dos. Algunos métodos híbridos combinan algoritmos de los dos tipos como es el caso del modelo presentado en este proyecto.

Los métodos heurísticos se dividen también en heurísticas constructivas y heurísticas de remamamiento: Las heurísticas constructivas poco a poco construyen desde cero la solución observando las restricciones y criterios de optimalidad mientras que las heurísticas de remamamiento empiezan con una solución candidata y esta se va remando hasta llegar a un criterio de terminación. Es posible que estas dos heurísticas se retroalimenten durante los ciclos de solución.

5.7.1. Métodos de búsqueda exhaustiva. En algunos problemas donde el espacio de búsqueda es suficientemente pequeño y enumerable, se pueden usar métodos de búsqueda exhaustiva o enumerativas: se examinan una por una todas las soluciones posibles, y se da por buena aquella que tenga el máximo valor de la función objetivo⁷⁵.

El problema con la búsqueda exhaustiva no sólo es que no siempre sea factible, sino que su mal comportamiento con respecto al aumento de tamaño del espacio de búsqueda, y el hecho de que necesite una cantidad de memoria de ordenador

⁷⁴ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

⁷⁵ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

proporcional al tamaño del problema, hace que no se pueda aplicar en la mayor parte de los casos. Sin embargo, en algunos casos puede ser útil; por ejemplo, en la fase final de la resolución del problema cuando el subespacio de búsqueda esté suficientemente acotado. Algunos algoritmos usan técnicas para descartar partes del espacio de búsqueda que se van a recorrer, haciendo lo suficientemente pequeño el tamaño de búsqueda como para usar métodos exhaustivos.

5.7.2. Métodos de búsqueda local. Los algoritmos de búsqueda local parten de una solución inicial, y, aplicándole operadores de variación, la van alterando; si la solución alterada es mejor que la original, se acepta, si no lo es, se vuelve a la inicial. El procedimiento se repite hasta que no se consigue mejora en la solución.

Los procedimientos de búsqueda local más usados son los basados en el gradiente: en este caso, el operador de variación selecciona una nueva solución teniendo en cuenta la derivada de la función que se quiere optimizar en el punto, tratando de ascender o descender usando el gradiente hasta llegar a un punto de inflexión donde no se puede obtener ninguna mejora adicional.

Estos procedimientos se suelen llamar de hillclimbing, o de escalado de colinas, el truco en ellos consiste en escoger correctamente el tamaño del paso para ascender y el punto de inicio. Pero se trata de algoritmos de búsqueda locales, y, como tales, sólo van a encontrar el máximo local más cercano al punto de inicio. Esto se puede resolver usando una estrategia de multicomienzo, pero, en todo caso, no garantizan que se encuentre, o incluso se acerque al máximo global. En todo caso, tiene la ventaja de que en cada iteración del algoritmo, se tiene una solución válida, aunque no necesariamente sea mejor⁷⁶.

En el caso del Agente viajero, un procedimiento de búsqueda local creado se denomina 2-opt. consiste en escoger una permutación inicial aleatoria, e ir probando con todas las posibles soluciones en la vecindad de esta, entendiendo como vecindad el conjunto de todos los recorridos que pueden ser alcanzados intercambiando dos caminos no adyacentes en el recorrido original. Un recorrido reemplaza al original si es mejor; una vez que no se puede mejorar un recorrido, se considera óptimo y se da como solución. El mejor algoritmo de búsqueda local conocido para el problema del agente viajero es el llamado Lin-Kernighan, que es una variación del anterior, este permite que el número de caminos a intercambiar

⁷⁶ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

en cada iteración varié, además, en vez de tomar cualquier mejora, prueba varias y se queda con la mejora de mayor valor⁷⁷.

5.7.3. Métodos que usan soluciones parciales. Estos métodos van construyendo soluciones variable a variable, no obteniendo una solución completa al problema hasta que todas las variables han sido asignadas.

En el caso del Agente Viajero, un algoritmo voraz comenzaría con una ciudad elegida aleatoriamente, y elegiría la ciudad más cercana a ésta. Para esta segunda ciudad, elegiría la más cercana, y así sucesivamente. En este caso, el problema estaría simplemente parametrizado con la ciudad inicial elegida.

El algoritmo representativo es el de Kart & Thompson, que es el que ha perdurado por su flexibilidad para considerar restricciones de diversa naturaleza. En este algoritmo las ciudades no se incluyen necesariamente al final del recorrido que se esta creando sino en la posición mas conveniente. En el anexo uno puede verse el esquema de funcionamiento de este método mediante un ejemplo de un problema del Agente viajero.

Otros métodos consisten en dividir el problema general en problemas más simples, y hallar la solución a estos problemas más simples; el método se denomina *divide y vencerás*; en algunos casos, la solución a un problema de orden n puede venir resolver a otro problema de orden $n-x$. En el anexo dos se presenta un ejemplo de este tipo de métodos: el método de Barrido(sweep) aplicado al problema de ruteo con múltiples vehículos.

Uno de los métodos más conocidos son los algoritmos greedy, que funcionan de la forma siguiente: para cada variable, se asigna aquél valor que maximice alguna función de utilidad. Los algoritmos greedy sustituyen un criterio de optimalidad global por un criterio en cada paso, no siempre los dos se corresponden, y por tanto, no tienen porqué hallar el máximo global del problema⁷⁸.

⁷⁷ LEVINE, John. DUCATELLE, Frederick. Ant Colony Optimisation and Local Search for Bin Packing and Cutting Stock Problems. Edinburgh: Centre for Intelligent Systems and their Applications. 2003.

⁷⁸BRAYSY, Olli. Efficient Local Search Algorithms for the Vehicle Routing Problem with Time Windows (On line). Finlandia:University of Vaasa. En Cuarta Conferencia Internacional de Metaheurísticas, Porto, Portugal, julio de 2001. (Documento encontrado directamente a través de www.google.com)

Algunos métodos consisten en ir descartando partes del espacio de búsqueda, con el objeto de acotar cada vez más la solución al problema. El *branch and bound*, es un método que imagina el espacio de búsqueda como un árbol de decisión, en cada paso del algoritmo, se toma la decisión de *podar* las ramas del árbol que parezcan menos prometedoras⁷⁹.

5.7.4. Métodos de búsqueda global. Los métodos de búsqueda global tratan de escaparse de los máximos locales, explorando con más eficiencia el espacio de búsqueda. Generalmente, añaden algún componente aleatorio a la búsqueda, de forma que, si se encuentra un mínimo local, se salte a otro punto del espacio de búsqueda, donde pueda haber otro máximo, posiblemente global.

Uno de los más conocidos es el recocido simulado (*simulated annealing*). A grandes rasgos, consiste en un procedimiento de ascenso de gradiente, pero tal que no siempre se escoge la mejor solución; dependiendo de una *temperatura*, la probabilidad de escoger una solución peor que la actual va descendiendo con el tiempo, hasta que al final del algoritmo, cuando la temperatura es 0, se escoge de forma determinista siempre la mejor solución.

Otros métodos, como el *Greedy Randomized Adaptive Search Procedure* o GRASP, se basan en la combinación de fases de explotación del conocimiento, en los cuales se siguen algoritmos de búsqueda local, con fases de exploración, en los cuales se usan procedimientos aleatorios para expandir el espacio de búsqueda recorrido. El algoritmo GRASP mantiene una lista de posibles soluciones a un problema, que se genera aleatoriamente. Se escoge una de las soluciones, y se mejora hasta que no se pueda mejorar más; si es mejor que la mejor almacenada, se actualiza la lista de soluciones almacenadas. GRASP equivale a un algoritmo de ascenso/descenso de gradiente con multicomienzo; mejora los resultados con respecto a una aplicación simple del mismo, pero no es tan eficiente como algoritmos que incluyen componentes estocásticos. Un algoritmo similar, denominado *búsqueda tabú*, incluye una memoria que almacena los últimos movimientos realizados, y que se usa para no volver a caer en los mismos máximos locales, a esta memoria se le denomina *lista tabú*. Pero también tiene un método que permite escapar de los máximos locales, que se denomina *nivel de aspiración*; que consiste en un criterio para aceptar movimientos incluidos en la lista tabú. Aparte de esa memoria, que se podría llamar a *corto plazo*, diversas variantes de la búsqueda tabú incluyen memorias a *medio plazo*,

⁷⁹ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

registrando los atributos más comunes de un conjunto de soluciones, para poder explotar esa zona del espacio de búsqueda, y memorias *a largo plazo*, que diversifica la búsqueda sobre regiones que no se han explorado con anterioridad⁸⁰.

⁸⁰ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

6. LOS SISTEMAS DE COLONIA DE HORMIGAS

En los últimos años los investigadores se han volcado a analizar el comportamiento de algunas especies animales, en busca de explicaciones a comportamientos que durante muchos años pasaron desapercibidos, o simplemente no presentaban otra utilidad aparente.

Los cisnes cuando migran, al volar en V, sin proponérselo calculan una distribución espacial que minimiza el esfuerzo total requerido por la bandada para trasladarse. En realidad lo que se aplica es un mecanismo simple que le permite a un cisne decidir que posición tomar: buscar aquella en la que se canse menos. Con esta regla se ubica justo en los puntos alrededor de otros cisnes donde se fatigue lo menos posible. Eventualmente el cisne "líder" se cansa y busca una posición más cómoda y uno de los cisnes a su lado toma su posición al no haber cerca ninguna otra de menor esfuerzo disponible⁸¹. Este comportamiento imitado por escuadrones aéreos militares para ahorrar combustible, pareciera tener pocos usos en otras áreas, pero si se tiene en cuenta que es la base de nuevas teorías de administración y economía⁸², la percepción hacia estos comportamientos cambia.

6.1. *INTELIGENCIA DE ENJAMBRE Y COMPUTACIÓN EMERGENTE*

Abejas que utilizan las formas geométricas más resistentes, formaciones de bancos de peces, patrones de ataque de depredadores, etc., son ejemplos de comportamientos relativamente complejos que optimizan alguna función (energía, tiempo, etc.) invertida en la satisfacción de una necesidad (locomoción, alimentación, protección, etc.). La función resulta optimizada, sin que los individuos conscientemente lo decidan.

⁸¹ CÓRDOVA, Juan. Optimización a través de colonias de hormigas. San Carlos:Universidad de San Carlos, p 2. 2004.

⁸² BENTLEY, Meter. Digital Biology.

Tales comportamientos, aparentemente inteligentes, corresponden a una propiedad del sistema como un agregado de individuos que no se muestra en los componentes individuales⁸³.

La expresión inteligencia de enjambre o “swarm intelligence” se acuñó en 1988 a partir del trabajo de los investigadores Beni, Hackwood y Wang. El término, se ha defendido, para denominar el comportamiento inteligente que proviene de la aglomeración más allá de una masa crítica de entidades que por sí solas no tienen más que un comportamiento limitado pero que en conjunto son capaces de desplegar un amplio e interesante espectro de comportamiento dinámico⁸⁴.

Las aplicaciones de esta inteligencia son variadas y promisorias pero no son infalibles. Estos sistemas pueden ayudar a resolver ciertos problemas y proporciona las primeras aproximaciones a otros, pero como con otros métodos habrán ocasiones en las que no se desempeñe bien.

La aplicación del concepto de inteligencia de enjambre en la solución de problemas es conocida como Computación Emergente⁸⁵, ya que los individuos crean soluciones que con cada ciclo se van mejorando, conduciendo con el tiempo hacia una solución global aceptable.

La observación de la naturaleza ha sido una de las principales fuentes de inspiración para la propuesta de nuevos paradigmas computacionales. Así nacieron diversas técnicas de Inteligencia Artificial como: los Algoritmos Genéticos, Recocido Simulado, Redes Neuronales, y entre estas técnicas, el sistema de Colonias de Hormigas.

⁸³ CÓRDOVA, Juan. Optimización a través de colonias de hormigas. San Carlos:Universidad de San Carlos, p 3. 2004.

⁸⁴ BONABEAU, E. DORIGO M. THERAULAZ, G.. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York, NY, 1999.

⁸⁵ CÓRDOVA, Juan. Op. cit. p.2.

6.2. LAS HORMIGAS REALES Y SUS COLONIAS

Las hormigas son insectos sociales que viven en colonias y que tienen un comportamiento dirigido al desarrollo de la colonia como un todo más que a un desarrollo individual. Atraídos por este comportamiento los mirmecólogos Pierre-Paul Grassé y Edward O. Wilson, realizaron un estudio donde se define la colonia de hormigas como un solo organismo, cuyos órganos son las hormigas en si. Los componentes de una colonia se comportan de tal forma que la conducta de la misma es mucho mas que lo que cabria deducir de la suma de sus partes, o incluso de la inteligencia de cada una de sus partes. Para explicar este fenómeno, Pierre Paul Grassé introdujo el concepto de Estimergia, que consiste en la sinergia por la comunicación que se realiza entre los miembros de la colonia a través del ambiente en el que se mueven⁸⁶.

6.2.1. El camino entre la colonia y la comida. La mayoría de especies de hormigas transportan el alimento que encuentran hasta la colonia para almacenarlo. Para esta tarea normalmente son utilizados un numero considerado de hormigas, las cuales realizan varios viajes trayendo cada una parte del alimento. Una característica muy interesante que fue observada, es como estos insectos trazan caminos cuyos contornos suelen ser suaves y cuya ruta es más o menos una línea recta, bordeando obstáculos mayores, entre la colonia y la fuente de alimentos. En otras palabras, estos insectos son capaces de encontrar las rutas mas cortas entre las fuentes de alimento y la colonia. Este hecho es especialmente interesante si se tiene en cuenta que muchas de las especies de hormigas son casi ciegas, lo que limita el uso de referencias visuales⁸⁷.

Los estudios de Jean-Louis Deneubourg, Pasteels y Verhaeghe de la Université Libre de Bruxells sobre la hormiga argentina *Linphitema humile*, mostraron que colonias de este tipo de hormigas logran determinar caminos cortos hacia fuentes de alimentos. En un experimento se uso una colonia de hormigas, cuyo hormiguero se separo de una fuente de alimento por medio de un puente de dos ramas de igual longitud, las hormigas se dejan en libertad y se mide en el tiempo el porcentaje de hormigas que escogen cada una de las ramas. El resultado muestra que después de una fase inicial de transición en la que pueden existir

⁸⁶ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005).

⁸⁷ CÓRDOVA, Juan. Optimización a través de colonias de hormigas. San Carlos:Universidad de San Carlos, p 2. 2004.

oscilaciones, las hormigas tienden a converger a la misma ruta normalmente la más corta⁸⁸.

6.2.1. El uso de feromonas en la consecución del camino más corto. Al comenzar la búsqueda, las hormigas se desplazan de una manera aislada esencialmente aleatoria, mientras se mueven depositan una sustancia química denominada feromona (una sustancia que pueden oler). Si no se encuentra ningún rastro de feromona, las hormigas se mueven de manera básicamente aleatoria, pero cuando existe feromona depositada, tienen mayor tendencia a seguir el rastro, dejando a su vez su propia feromona a lo largo del camino recorrido y por tanto lo hacen más atractivo, puesto que se ha reforzado el rastro de feromona. El rastro de feromonas también les permite encontrar el camino de regreso al hormiguero y transmitir esta información a otras hormigas. De hecho, los experimentos realizados demostraron que las hormigas prefieren de manera probabilística los caminos marcados con una concentración superior de feromona. La elección entre distintos caminos toma lugar cuando varios caminos se cruzan, entonces, las hormigas eligen el camino a seguir con una decisión probabilística sesgada por la cantidad de feromona: cuanto más fuerte es el rastro de feromona (mayor concentración de la sustancia), mayor es la probabilidad de elegirlo⁸⁹.

Se ha demostrado experimentalmente que el rastro de feromonas le sirve a la colonia de hormigas para encontrar la ruta mas corta del hormiguero a la fuente de alimento cuando existen rutas alternativas. La hormiga que consigue encontrar la comida más rápidamente necesariamente encontró el camino mas corto hasta el momento (en comparación con las compañeras que salieron a la vez desde la colonia en la búsqueda aleatoria), por lo que comienza su viaje de retorno antes, siguiendo su rastro y dejando nuevamente una huella de feromona. Entonces, en el camino más corto habrá un rastro de feromona ligeramente superior y, por lo tanto, las siguientes hormigas que lleguen a la comida preferirán ligeramente esta trayectoria en comparación con otras creadas que solo cuentan con la mitad de feromona en ellas, y al hacerlo intensifican el rastro de feromona haciendo la ruta aún más atractiva para nuevas hormigas. Como al principio el rastro de feromona no es muy fuerte, las primeras hormigas que siguen el camino hacia la comida pueden desviarse y encontrar atajos dentro del mismo, con lo que se perfecciona la solución propuesta. Eventualmente el proceso hace que la probabilidad de que

⁸⁸ DENEUBOURG, J. GOSS, S. Collective patterns and decision making; *Ethology, Ecology, Evolution*, Vol. 1, 1989, págs. 295-311.

⁸⁹ ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco *La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques*. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

una hormiga escoja el camino más corto aumente progresivamente y que al final el recorrido de la colonia converja hacia el más corto de todos los caminos posibles⁹⁰.

Esta convergencia se complementa con la acción del entorno natural que provoca que la feromona se evapore transcurrido un cierto tiempo. Así, los caminos menos prometedores pierden progresivamente feromona porque son visitados cada vez por menos hormigas.

En la Figura 1, se ilustra el proceso. Las hormigas llegan a un punto donde tienen que decidir por uno de los caminos que se les presenta, Figura(a), lo que resuelven de manera aleatoria. En consecuencia, la mitad de las hormigas se dirigirán hacia un extremo y la otra mitad hacia el otro extremo, como ilustra la Figura (b). Como las hormigas se mueven aproximadamente a una velocidad constante, las que eligieron el camino más corto alcanzarán el otro extremo más rápido que las que tomaron el camino más largo, quedando depositado mayor cantidad de feromona por unidad de longitud, Figura (c). La mayor densidad de feromonas depositadas en el trayecto más corto hace que éste sea más deseable para las siguientes hormigas y por lo tanto la mayoría elige transitar por él. Considerando que la evaporación de la feromona hace que los caminos menos transitados sean cada vez menos deseables y la realimentación positiva en el camino con más feromonas, resulta claro que al cabo de un tiempo casi todas las hormigas transiten por el camino más corto, Figura (d)⁹¹.

⁹⁰ DORIGO, Marco. DI CARO, Gianni. GAMBARDELLA, Luca. Ant Algorithms for Discrete Optimization. En Revista Artificial Life, Bruselas: MIT Press, 1999.

⁹¹ BARÁN, Benjamín. ALMIRÓN, Marta. Colonia de Hormigas en un Ambiente Paralelo Asíncrono. p. 11. San Lorenzo: Centro Nacional de Computación. 2003

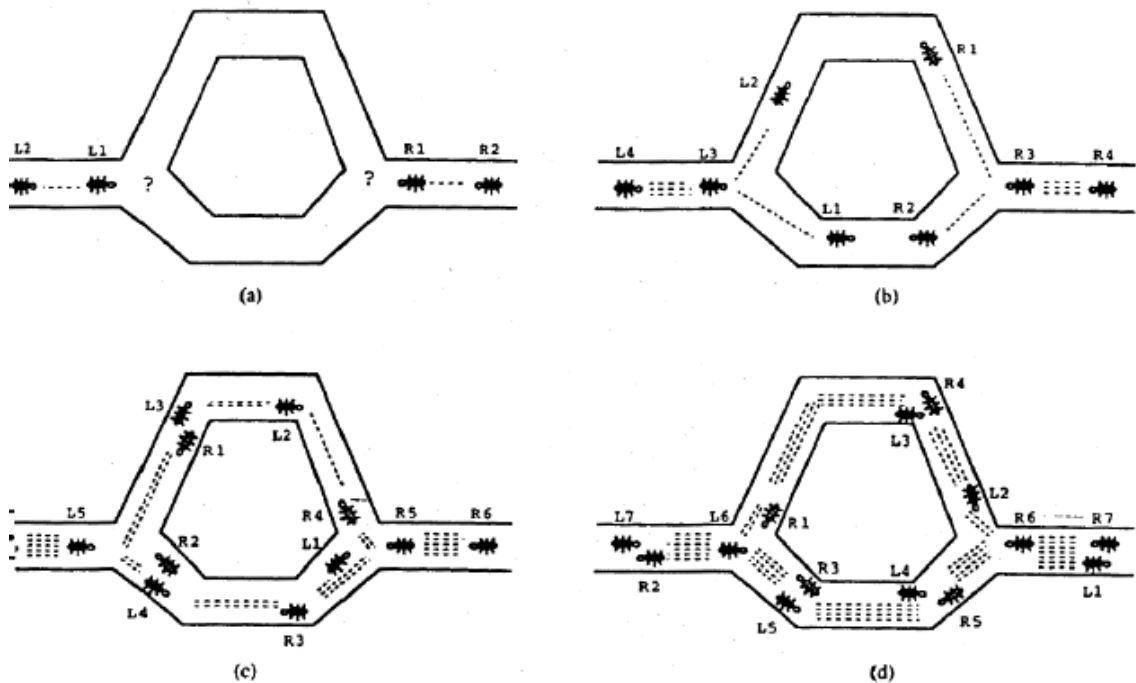


Figura 1. Uso de feromona para la consecución de la ruta mas corta⁹²

De esta manera la feromona depositada constituye la información que permite la coordinación de los esfuerzos del enjambre en la realización de obras colectivas sin necesidad de la intervención de una autoridad central. Es de notar que las hormigas solo se comunican de manera indirecta, la comunicación entre hormigas es bastante escasa, y su entendimiento limitado; la mayor parte de la comunicación se realiza a través de los actos que otras hormigas han realizado, o bien a través de la acumulación de feromonas en el suelo y otras modificaciones del espacio físico que perciben. Claramente, esta es una forma de comunicación Estimergia.

6.2.2. Un método no infalible. puede afirmarse correctamente, que el proceso descrito se caracteriza por una retroalimentación positiva, en la que la probabilidad con la que una hormiga escoge un camino aumenta con el número de hormigas que previamente hayan elegido el mismo camino. Esta afirmación parecería hacer del sistema un método infalible para encontrar las mejores rutas, sin embargo el sistema presenta ciertas debilidades. Algunos estudios biológicos han demostrado

⁹² Tomada de : BARÁN, Benjamín. ALMIRÓN, Marta. Colonia de Hormigas en un Ambiente Paralelo Asíncrono. p. 11. San Lorenzo: Centro Nacional de Computación. 2003

que los rastros de feromona son muy persistentes⁹³, la feromona puede permanecer desde unas pocas horas hasta varios meses dependiendo de varios aspectos distintos, como la especie de las hormigas, el tipo de suelo, etc., lo que provoca una menor influencia del efecto de la evaporación en el proceso de búsqueda del camino más corto. En algunos casos esto representa un inconveniente, ya que, debido a la gran persistencia de feromona, difícilmente las hormigas cambiarán un camino que tiene un alto nivel de feromona aunque hayan encontrado un camino aún más corto. Si se traslada este comportamiento directamente al computador para diseñar un algoritmo de búsqueda, podría presentarse que se quede rápidamente estancado en un óptimo local.

Por otro lado, si por algún azar, una hormiga encuentra un camino largo antes de que otra encuentre el corto (lo cual es probabilísticamente posible), entonces la colonia adoptará el camino largo y convergerá a una solución que no es óptima, por lo tanto el método no es infalible pero, la mayor parte de las veces el camino corto es el favorecido porque también es el que tiene mayor probabilidad de ser encontrado en un principio, mediante una búsqueda aleatoria.

Aunque se trata de una técnica interesante y prometedora, es evidente que la optimización por colonia de hormigas, al igual que otras metaheurísticas, en muchos casos no pueda competir con métodos de búsqueda local especializados.

6.3. PRINCIPIOS DE LAS HEURÍSTICAS CON MÚLTIPLES AGENTES BIOINSPIRADAS

Tanto las colonias de hormigas y otros insectos, como los algoritmos que las imitan, se basan en tres principios:

Retroalimentación positiva: mediante la feromona que se deposita

Retroalimentación negativa: mediante la evaporación de la feromona

Fluctuaciones múltiples: debido a la toma de decisión probabilística de las hormigas.

⁹³ BONABEAU, E. DORIGO M. THERAULAZ, G.. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York, NY, 1999.

La retroalimentación positiva, junto con la retroalimentación negativa conlleva a una amplificación de las fluctuaciones, en otras palabras produce fluctuaciones múltiples debido a que las hormigas se nutren de la información vigente producida en la iteración anterior a la hora de tomar decisiones (que camino escoger). Debe haber un equilibrio entre los dos tipos de retroalimentación, si no, se llegaría a una situación de caos o estancamiento. Por otro lado, estos comportamientos sólo pueden tener lugar si hay múltiples individuos e interacciones, en el caso de que haya solo unos pocos, los comportamientos serán predecibles a corto plazo, y poco innovadores⁹⁴.

6.4. METAHEURISTICA DE SISTEMA DE COLONIA DE HORMIGAS

6.4.1. Origen de la metaheurística de colonia de hormigas. El algoritmo de colonia de hormigas fue propuesto inicialmente por Marco Dorigo de la Université Libre de Bruxells en 1992 para la solución de problemas combinatorios y de asignación cuadrática. El algoritmo original Ant system(AS)⁹⁵, nació de aprovechar el comportamiento de las hormigas naturales en la creación de algoritmos de búsqueda y simulación, desde entonces, casi todos los comportamientos de una colonia de hormigas, desde la construcción de estructuras, hasta la localización de comida, se han utilizado para enfrentar algún tipo de problema de búsqueda. Este tipo de algoritmos inspirados en las hormigas, ha evolucionado en una familia de algoritmos denominada optimización por colonia de hormigas (OCH)⁹⁶. En general, tienen en común las hormigas, las feromonas, y un entorno en el cual se desarrolla la labor de la colonia.

6.4.2. Aspectos generales de los OCH. La optimización por colonia de hormigas consiste en procedimientos de construcción estocástica de soluciones que simulan el comportamiento de las hormigas reales. Las hormigas artificiales se implementan como procesadores paralelos cuya misión es construir soluciones al problema dado, utilizando un procedimiento en el que iterativamente se añaden componentes a soluciones parciales movidas por una combinación de⁹⁷:

⁹⁴ MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005. <http://geneura.ugr.es> (consultado: 7 de Mayo de 2005).

⁹⁵ DORIGO, M., MANIEZZO, COLORNÍ A. An investigation of some properties of an Ant algorithm. proceedings of the parallel problem solving from nature conference. Bruselas - Bélgica, 1992.

⁹⁶ MERELO, J.J Op., cit.

⁹⁷ OTERO, Juan. GUERRERO, Danilo. Clasificación Automática Mediante Colonia De Hormigas. La Habana: Universidad de la habana. P 12.

- Información y datos del problema a resolver
- Una memoria común que corresponde a los Trazos de feromona artificial que cambian dinámicamente durante la ejecución del algoritmo para reflejar la experiencia adquirida por las hormigas en la búsqueda de una solución.
- Una función heurística utilizada para evaluar los sucesivos pasos de la construcción.

En general en los algoritmos de colonia de hormigas se tiene un conjunto finito de agentes que cooperativamente buscan una solución de calidad al problema en consideración. Cada hormiga construye una solución partiendo de un estado al que ha sido asignada, va recolectando información con la cual mide su desempeño, modifica los valores de las variables de representación del problema y los transmite a las demás hormigas. Las hormigas son individuos relativamente simples que pueden construir por si solos soluciones pobres al problema, las soluciones de alta calidad se obtienen por la interacción y cooperación entre las hormigas. Los movimientos de cada hormiga son locales y se realizan de acuerdo a la definición de vecinos dada para el problema, a la información privada de cada hormiga, a las reglas de decisión estocásticas y al rastro de feromonas que es accesible para todos los miembros de la colonia. Un aspecto importante es que los componentes estocásticos de las reglas de decisión sobre el movimiento siguiente que realizara una hormiga, así como la evaporación de las feromonas, permiten que no se presente convergencia prematura hacia óptimos locales⁹⁸.

La metaheurística está compuesta de tres pasos fundamentales:

- La actividad de las hormigas
- Un mecanismo de evaporación de feromona
- Decisiones globales (Demonio)

El comportamiento de las hormigas puede resumirse entonces del siguiente modo: Una colonia de hormigas se mueve de forma concurrente y asincrónica a través de estados adyacentes del problema construyendo caminos.

Se mueven aplicando una regla de decisión estocástica local que hace uso de los trazos de feromona e información heurística. En su movimiento las hormigas van construyendo soluciones del problema de optimización. Una vez construida una solución o durante el proceso de construcción de una solución las hormigas

⁹⁸ DORIGO, Marco. DI CARO, Gianni. GAMBARDELLA, Luca. Ant Algorithms for Discrete Optimization. En Revista Artificial Life, Bruselas: MIT Press, 1999.

evalúan la solución (o solución parcial) y depositan feromona en las conexiones entre puntos utilizadas. Esta información será utilizada por las próximas hormigas.

La evaporación de la feromona es el procedimiento mediante el cual la intensidad de la traza de feromona sobre los arcos decrece en el tiempo. Este procedimiento es una cierta forma de olvido, favoreciendo la exploración de nuevas áreas en el espacio de búsqueda y evitar convergencia muy rápida del algoritmo a una región subóptima.

Las decisiones globales pueden ser utilizadas para implementar acciones centralizadas, que no pueden ser realizadas por una sola hormiga. Por ejemplo, se puede evaluar la solución obtenida por cada hormiga y depositar feromona adicional en las conexiones utilizadas por la hormiga que construyó la mejor solución. A este proceso que no tiene ningún contrapunto natural se le conoce como **demonio**⁹⁹.

6.5. LAS HORMIGAS ARTIFICIALES

Si se desea transplantar el comportamiento de hormigas reales a un entorno cibernético, se debe proveer a estas hormigas de vida, espacio, objetivos y reglas. De hecho las hormigas (o agentes) son programas que corren sobre un sustrato adecuado (una computadora o red de computadoras), cobrando vida en los momentos en que su código está siendo ejecutado. El espacio es la memoria de la computadora o computadoras involucradas.

Los objetivos dependen del problema a resolver y las reglas de interacción local son las decisiones embebidas en el programa de los agentes que les indican cómo obtener información del universo (la memoria) y reaccionar a esta información, tanto ejecutando alguna acción como marcando al mismo tiempo el universo para transmitir información a futuros agentes que visiten la misma región.

⁹⁹ ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco. La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

Las hormigas artificiales son programas más o menos independientes que tienen un objetivo e interactúan con el universo que es alguna estructura de datos adecuada que provee información al mismo tiempo que es susceptible de ser modificada y por lo tanto también es depositaria de información. Las hormigas, como programas, pueden residir en una sola máquina o trasladarse en una red según sea el tipo de problema para el que están diseñadas.

La hormiga artificial es un agente computacional simple que intenta construir soluciones posibles al problema explotando los rastros de feromona disponibles y la información heurística. Sin embargo, en algunos problemas, puede también construir soluciones no válidas que podrán ser penalizadas dependiendo de lo inadecuado de la solución.

Características de las hormigas Artificiales¹⁰⁰:

- Busca soluciones válidas de costo mínimo para el problema a solucionar.
- Tiene una memoria que almacena información sobre el camino seguido hasta el momento, esto es, la memoria almacena la secuencia generada. Esta memoria puede usarse para:
 - Construir soluciones válidas
 - Evaluar la solución generada
 - Reconstruir el camino que ha seguido la hormiga.
- Tiene un estado inicial, que normalmente corresponde con una secuencia unitaria y una o más condiciones de parada asociadas.
- Comienza en el estado inicial y se mueve siguiendo estados válidos, construyendo la solución asociada incrementalmente.
- Cuando la hormiga está en un nodo y ha seguido previamente una secuencia, puede moverse a cualquier nodo no visitado de su vecindario posible. El movimiento se lleva a cabo aplicando una regla de transición, que es función de los rastros de feromona que están disponibles localmente, de los valores heurísticos de la memoria privada de la hormiga y de las restricciones del problema.
- Cuando durante el procedimiento de construcción una hormiga se mueve desde el nodo i hasta el j , puede actualizar el rastro de feromona asociado al arco a_{rs} . Este proceso se llama actualización en línea de los rastros de feromona paso a paso.

¹⁰⁰ ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco. La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

- El procedimiento de construcción acaba cuando se satisface alguna condición de parada, normalmente cuando se alcanza un estado objetivo.
- Una vez que la hormiga ha construido la solución puede reconstruir el camino recorrido y actualizar los rastros de feromona de los arcos/componentes visitados(as) utilizando un proceso llamado actualización en línea a posteriori. Este es el único mecanismo de comunicación entre las hormigas, utilizando la estructura de datos que almacena los niveles de cada arco/componente (memoria compartida).

6.6. SIMILITUDES Y DIFERENCIAS ENTRE LAS HORMIGAS NATURALES Y ARTIFICIALES

Las colonias de hormigas naturales y artificiales comparten una serie de características. Las más importantes se resumen a continuación¹⁰¹:

- Uso de una colonia de individuos que interaccionan y colaboran para solucionar una tarea dada.
- Tanto las hormigas naturales como las artificiales modifican su entorno a través de una comunicación estímulo-respuesta basada en la feromona. En el caso de las hormigas artificiales, los rastros de feromona artificiales son valores numéricos que están disponibles únicamente de manera local.
- Ambas, las hormigas naturales y las artificiales, comparten una tarea común: la búsqueda del camino más corto (construcción iterativa de una solución de costo mínimo) desde un origen, el hormiguero (decisión inicial), hasta un estado final, la comida (última decisión).
- Las hormigas artificiales construyen las soluciones iterativamente aplicando una estrategia de transición local estocástica para moverse entre estados adyacentes, tal como hacen las hormigas naturales.

Sin embargo, estas características por sí solas no permiten desarrollar algoritmos eficientes para problemas combinatorios difíciles. De hecho, las hormigas

¹⁰¹ DORIGO, Marco. DI CARO, Gianni. GAMBARDELLA, Luca. Ant Algorithms for Discrete Optimization. En Revista Artificial Life, Bruselas: MIT Press, 1999.

artificiales viven en un mundo discreto y tienen algunas capacidades adicionales¹⁰²:

- Las hormigas artificiales pueden hacer uso de la información heurística (no solo de los rastros locales de feromona) en la política estocástica de transición que apliquen.
- Tienen una memoria que almacena el camino seguido por la hormiga.
- La cantidad de feromona depositada por la hormiga artificial es función de la calidad de la solución encontrada.
- Las hormigas artificiales normalmente sólo depositan feromona después de generar una solución completa.
- Tienen que incluir al sistema un mecanismo de evaporación más fuerte que las naturales para evitar que el algoritmo se estanque en óptimos locales
- Pueden realizar operaciones adicionales para mejorar la eficiencia y eficacia del sistema

Como se comentó, la evaporación de feromona en los algoritmos de OCH es diferente a como se presenta en la naturaleza, ya que la inclusión del mecanismo de evaporación es una cuestión fundamental para evitar que el algoritmo se quede estancado en óptimos locales. La evaporación de feromona permite a la colonia de hormigas artificiales olvidar lentamente su historia pasada para redireccionar su búsqueda hacia nuevas regiones del espacio.

Para mejorar la eficiencia y eficacia del sistema, los algoritmos OCH pueden enriquecerse con habilidades adicionales. Ejemplos típicos son la capacidad de mirar más allá de la siguiente transición (lookhead), la optimización local, y el backtracking, que persiguen mejorar la eficacia, o la llamada lista de candidatos que contiene un conjunto de los estados vecinos mas prometedores para mejorar la eficiencia del algoritmo¹⁰³.

¹⁰² DORIGO, Marco. DI CARO, Gianni. GAMBARDELLA, Luca. Ant Algorithms for Discrete Optimization. En revista Artificial Life, Bruselas: MIT Press, 1999.

¹⁰³ ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco. La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

6.7. LA METAHEURÍSTICA DE OPTIMIZACIÓN BASADA EN COLONIAS DE HORMIGAS

Los algoritmos de OCH son esencialmente algoritmos constructivos: en cada iteración del algoritmo, cada hormiga busca construir una solución al problema recorriendo un grafo de construcción que representa el problema. Cada arista del grafo, que representa los posibles pasos (ciudades, clientes...) que la hormiga puede dar, tiene asociada dos tipos de información que guían el movimiento de la hormiga:

- Información heurística, que mide la preferencia heurística de moverse desde el nodo v_i hasta el nodo v_j . Las hormigas no modifican esta información durante la ejecución del algoritmo.
- Información de los rastros de feromona artificiales, que mide la deseabilidad aprendida del movimiento de i a j . Imita a la feromona real que depositan las hormigas naturales. Esta información se modifica durante la ejecución del algoritmo dependiendo de las soluciones encontradas por las hormigas¹⁰⁴.

De esta manera, las hormigas al encontrarse en una ciudad tiene la siguiente información a su disposición:

- La lista que lleva consigo, que le indica qué clientes aún no visita.
- La información local que le indica a qué clientes (ciudades, etc..) se puede acceder inmediatamente desde donde esta, cuáles son las distancias(costo, tiempo) y cuántas hormigas ya han tomado cada ruta.

Una hormiga estará tentada a seguir una ruta corta hacia alguna ciudad no visitada, pero esto compite con su tendencia a transitar por caminos que ya han sido frecuentados por otras hormigas. Es decir que la probabilidad de tomar una ruta (hacia una ciudad no visitada) es inversamente proporcional a la distancia y directamente proporcional a la cantidad de feromona depositada en la ruta.

¹⁰⁴ ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco. La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

En este punto se debe tener en cuenta que los algoritmos OCH presentan una doble perspectiva:

- Por un lado, son una abstracción de algunos patrones de comportamiento naturales relacionados con el comportamiento de las hormigas que permite encontrar el camino más corto.
- Por otro lado, incluyen algunas características que no tienen una contrapartida natural, pero que permiten que se desarrollen algoritmos para obtener buenas soluciones al problema que se pretende resolver (por ejemplo, el uso de información heurística que guíe el movimiento de las hormigas).

6.8. MODO DE FUNCIONAMIENTO DE UN ALGORITMO DE OCH

Como se ha visto hasta el momento, el modo de operación básico de un algoritmo de OCH es como sigue: las m hormigas (artificiales) de la colonia se mueven, concurrentemente y de manera asíncronica, a través de los estados adyacentes del problema (que puede representarse en forma de grafo con pesos). Este movimiento se realiza siguiendo una regla de transición que está basada en la información local disponible en las componentes (nodos). Esta información local incluye la información heurística y memorística (representada por los rastros de feromona) para guiar la búsqueda. Al moverse por el grafo de construcción, las hormigas construyen incrementalmente soluciones.

Opcionalmente, las hormigas pueden depositar feromona cada vez que crucen un arco (conexión) mientras que construyen la solución (actualización en línea paso a paso de los rastros de feromona). Una vez que cada hormiga ha generado una solución se evalúa ésta y puede depositar una cantidad de feromona que es función de la calidad de su solución (actualización en línea de los rastros de feromona). Esta información guiará la búsqueda de las otras hormigas de la colonia en el futuro.

Además, el modo de operación genérico de un algoritmo de OCH incluye dos procedimientos adicionales, la evaporación de los rastros de feromona y las acciones del **demonio**. La evaporación de feromona la lleva a cabo el entorno y se usa para permitir que las hormigas busquen y exploren nuevas regiones del espacio. Las acciones del demonio como se había especificado, son acciones

opcionales, para implementar tareas desde una perspectiva global que no pueden llevar a cabo las hormigas por la perspectiva local que ofrecen.

Algunos ejemplos de acciones del demonio son:

- Observar la calidad de todas las soluciones generadas y depositar una nueva cantidad de feromona adicional sólo en las transiciones/componentes asociadas a algunas soluciones.
- Aplicar un procedimiento de búsqueda local a las soluciones generadas por las hormigas antes de actualizar los rastros de feromona.

En ambos casos, el demonio reemplaza la actualización en línea a posteriori de feromona y el proceso pasa a llamarse actualización fuera de línea de rastros de feromona.

6.9. ESTRUCTURA GENÉRICA DE UN ALGORITMO DE OCH

El siguiente algoritmo representa el modelo genérico de OCH:

- 1 Procedimiento Metaheurística OCH ()
 - 2 Inicialización de parámetros
 - 3 Mientras (criterio de terminación no satisfecho)
 - 4 Programación de actividades
 - 5 Generación de Hormigas y actividad ()
 - 6 Evaporación de Feromona ()
 - 7 Acciones del demonio () {opcional}
 - 8 Fin Programación de actividades
 - 9 Fin mientras
- 10 Fin Procedimiento

El primer paso incluye la inicialización de los valores de los parámetros que se tienen en consideración en el algoritmo. Entre otros, se debe fijar el rastro inicial de feromona asociado a cada transición, que es un valor positivo pequeño, normalmente el mismo para todas las componentes/conexiones, el número de hormigas en la colonia m (en algunos casos este valor podrá ser una incógnita del problema), y los pesos que definen la proporción en la que afectarán la información heurística y memorística en la regla de transición probabilística.

El procedimiento principal de la metaheurística OCH controla, mediante el constructor *Programación de Actividades*, la planificación de las tres componentes principales:

- La generación y puesta en funcionamiento de las hormigas artificiales
- La evaporación de feromona
- Las acciones del demonio.

La forma de implementación de este constructor determinará la sincronía existente entre cada una de las tres componentes.

La Planificación secuencial es usualmente aplicada a problemas “clásicos” NP-duros no distribuidos (problemas estáticos). El paralelismo es usado de forma más eficiente en problemas distribuidos como los de enrutamiento en redes (problemas dinámicos).

Algunas componentes del algoritmo son opcionales, como las acciones del demonio, o dependientes estrictamente del algoritmo de OCH específico, por ejemplo cuándo y cómo se deposita la feromona. Generalmente, la actualización en línea paso a paso de los rastros de feromona y la actualización en línea a posteriori de los rastros de feromona son mutuamente excluyentes y no suelen estar presentes a la vez ni faltar ambas al mismo tiempo (si las dos faltan, el demonio suele actualizar los rastros de feromona).

El siguiente algoritmo representa el procedimiento *generación y puesta en funcionamiento de las hormigas artificiales*:

- 1 Procedimiento Generación de Hormigas y actividad ()
 - 2 repetir en paralelo desde $k = 1$ hasta m (numero hormigas)
 - 3 Nueva Hormiga (k)
 - 4 fin repetir en paralelo
- 5 Fin Procedimiento

El procedimiento *Nueva hormiga* se ilustra en el siguiente algoritmo:

- 1 Procedimiento Nueva Hormiga (id Hormiga)
 - 2 inicializa hormiga (id Hormiga)
 - 3 L = actualiza memoria hormiga ()
 - 4 mientras (estado actual estado objetivo)
 - 5 P = calcular probabilidades de transición (A,L,W)

```

6 siguiente estado = aplicar política decisión (P,W)
7 mover al siguiente estado (siguiente estado)
Si (actualización feromona en línea paso a paso)
    8 depositar feromona en el arco visitado ()
Fin si
9 L = actualizar estado interno ()
10 Fin mientras
Si (actualización feromona en línea a posteriori)
    11 Para cada arco visitado
        12 Depositar feromona en el arco visitado ()
    13 Fin para
Fin si
14 Liberar recursos hormiga (id Hormiga)
15 Fin Procedimiento

```

Es de destacar que el procedimiento *actualiza memoria hormiga()* se encarga de especificar el estado inicial desde el que la hormiga comienza su camino y, además almacenar la componente correspondiente en la memoria de la hormiga. La decisión sobre cuál será dicho nodo depende del algoritmo específico (puede ser una elección aleatoria o una fija para toda la colonia, o una elección aleatoria o fija para cada hormiga, etc.).

Los procedimientos *calcular probabilidades de transición* y *aplicar política decisión* tienen en consideración el estado actual de la hormiga, los valores actuales de la feromona visibles en dicho nodo y las restricciones del problema para establecer el proceso de transición probabilístico hacia otros estados válidos.

6.10. DIFERENCIAS ENTRE LAS METAHEURÍSTICA OCH, LOS ALGORITMO OCH Y LOS ALGORITMOS DE HORMIGAS

Es importante señalar que mediante el término metaheurística OCH se entiende el modo genérico de operación de la OCH. El nombre algoritmo de OCH se usa para referirse a cada instancia específica del algoritmo genérico. La metaheurística OCH incluye un amplio rango de algoritmos que pueden tener formas muy variadas. Estas variaciones se deben principalmente a que existen distintos tipos de interacciones complejas en el constructor Programación de actividades, entre las actividades *Generación de Hormigas y actividad ()*, *Evaporación de Feromona*

() y *Acciones del demonio* (). En cualquier caso, la línea de acción general de los algoritmos de OCH actuales sigue el flujo de actividades expuesto anteriormente.

Por otra parte, la metaheurística OCH no es lo suficientemente general como para cubrir la familia completa de algoritmos de hormigas, que pueden definirse como métodos aproximados para solucionar problemas combinatorios basados en características del comportamiento genérico de las hormigas naturales. Algunos ejemplos de algoritmos de hormigas que no están incluidos en la metaheurística OCH son el *SH-rápido* (“Fast Ant System”) un algoritmo de construcción que se basa en el modo de operación de una única hormiga y que no utiliza de manera explícita la evaporación de feromona, y el *SH-Híbrido* (“Hybrid Ant System”) un procedimiento de búsqueda local que hace uso de la información de los rastros de feromona para generar soluciones vecinas.

En particular, el presente proyecto utiliza un algoritmo metaheurístico OCH por lo que en lo sucesivo se considerara que tanto la feromona como la información heurística están asociadas únicamente a las conexiones(nodos), el cual es el caso de la gran mayoría de las aplicaciones de la OCH a problemas de ruteo o asignación.

6.11. MODELOS DE ALGORITMOS OCH

El primer algoritmo basado en la optimización mediante colonias de hormigas fue aplicado al Problema del agente viajero, obteniéndose resultados bastante alentadores (1992). A partir de dicho algoritmo se han desarrollado diversos heurísticos que incluyen varias mejoras¹⁰⁵. Entre los algoritmos de OCH disponibles para problemas de optimización combinatoria NP-duros, se encuentran:

El Sistema de Hormigas (SH o Ant System(AS))

El Sistema de Colonia de Hormigas (SCH o Ant Colony System (ACS))

El Sistema de Hormigas Max-Min (SHMM, Max-Min Ant System)

¹⁰⁵ BARCOS, Lucía. RODRÍGUEZ, Victoria . ÁLVAREZ, Jesús. ROBUSTÉ, Francesc . Routing Optimization for many-to-many freight, Transportation problems using Ant Colonies Techniques. España: Universidad Carlos III En Second world conference on POM, Cancun, Mexico, april 2004.

El SH con ordenación (Rank-Based Ant System)
El Sistema de la Mejor-Peor Hormiga (SMPH o Best-Worst Ant System)
El AntNet (Sistema Especifico para redes computacionales)

El primer algoritmo creado fue el SH, los siguientes cuatro algoritmos mejoraron progresivamente los resultados en los sucesivos 10 años, mientras que el AntNet, ha conseguido bastante éxito únicamente en el enrutamiento de paquetes en redes de comunicaciones.

El algoritmo Sistema de hormigas propuesto por Dorigo proponía tres variantes que básicamente difieren en el momento y la modalidad en que actualizan la matriz de feromonas que sirve para guiar el movimiento de las hormigas, estos algoritmos son:

- *Ant-density*: con actualización constante de las feromonas por donde pasa una hormiga.
- *Ant-quantity*: con actualización de feromonas inversamente proporcional a la distancia entre 2 ciudades recorridas.
- *Ant-cycle*: con actualización de feromonas inversamente proporcional al trayecto completo, al terminar un recorrido. Esta última variante era la que obtenía unos mejores resultados y es por tanto la que se conoce como SH.

Trabajos posteriores de Dorigo y Gambardella extendieron los alcances del *SH* al *Ant-Q* que es un híbrido entre *SH* y *Q-learning*, un conocido algoritmo de aprendizaje con realimentación positiva. Como una extensión del *Ant-Q* se creó el *Sistema de Colonia de Hormigas* (*SCH*) en el que se presentaron mejoras sustanciales a su antecesor *Ant System* en tres aspectos principales:

- i) El procedimiento que realizan las hormigas para la elección de la siguiente ciudad a visitar, pues se ofrece un balance entre la exploración de nuevos caminos y la explotación del conocimiento acumulado acerca del problema
- ii) Actualización global, permitiendo modificar la matriz de feromonas solo con la mejor ruta encontrado hasta el momento;
- iii) La regla de actualización local, que permite a todas las hormigas actualizar la matriz de feromonas al terminar su tour.

Posteriormente se incluyó el *Sistema de Hormigas Max-Min*, donde el registro de feromona es actualizado en diferentes concentraciones según los resultados de cada hormiga al final de cada ciclo.

El siguiente algoritmo con gran éxito fue el *Sistema de la Mejor-Peor Hormiga* en el cual no solo el registro de feromona varía en concentración, sino que la intensidad de la evaporación también depende de los datos recolectados por las hormigas al final de cada ciclo.

En el siguiente capítulo se mostrará en detalle el funcionamiento de estos algoritmos, con excepción del AntNet, el cual es una aplicación específica.

6.12. IMPLEMENTACIÓN DE UN ALGORITMO OCH EN UN HABIENTE PARALELO DE CÓMPUTO

La gran proliferación de redes de computadoras con procesadores heterogéneos, sincrónicos o asíncronicos, y la búsqueda de mejores soluciones obtenidas en menor tiempo de procesamiento, justifican la paralelización de nuevos algoritmos. En este sentido, los algoritmos OCH tienen características que lo hacen especialmente apropiados para su paralelización y distribución entre los diversos procesadores de una red de computadoras o una computadora paralela. En efecto, el paralelismo está implícito en el mismo algoritmo.

7. FUNDAMENTOS MATEMÁTICOS DE LOS SISTEMAS DE COLONIAS DE HORMIGAS¹⁰⁶

7.1. TIPOS DE PROBLEMAS RESOLUBLES POR LA OCH

El una vez entendido el concepto de grafo se puede definir el tipo de problemas que se pueden resolver por medio de OCH. Este tipo de problemas pertenece al grupo (restringido) de problemas de camino mínimo.

Como factor común, estos problemas de optimización combinatoria pueden representarse en forma de grafo ponderado $G = (V, A)$ (Grafo de construcción), donde A es el conjunto de arcos que conectan el conjunto de componentes V . Se destacan los siguientes aspectos:

- Las componentes v_i son los nodos del grafo.
- Los estados δ (*Caminos Hamiltonianos*) corresponden a caminos en el grafo, esto es, secuencias de nodos o aristas.
- Las aristas (arcos) del grafo, a_{ij} , son conexiones/transiciones que definen la estructura del vecindario. $\delta_2 = \langle \delta_1, j \rangle$ será vecino de δ_1 , si el nodo v_j es la última componente de δ_1 , y la arista a_{ij} existe en el grafo.
- Deben existir los costos explícitos c_{ij} asociados con cada arista (Matriz de Costos/Pesos).
- Las componentes o las conexiones deben tener asociados rastros de feromona τ , que representan un tipo de memoria indirecta y a largo plazo del proceso de búsqueda como valores heurísticos η , que representan la información heurística disponible en el problema a resolver.

¹⁰⁶ El contenido de este capítulo fue obtenido principalmente de una sola fuente, a pesar de que gran parte la notación fue cambiada para unificarla a la utilizada en este proyecto, la mayor parte del capítulo es tomado textualmente debido a las características de la información que contiene. La fuente es:
ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco. La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

7.2. MODELOS DE ALGORITMOS OCH

A continuación se presenta una explicación matemática de los algoritmos OCH mas conocidos:

7.2.1. El Sistema de Hormigas. El SH se caracteriza por el hecho de que la actualización de feromona se realiza una vez que todas las hormigas han completado sus soluciones, y se lleva a cabo como sigue: primero, todos los rastros de feromona se reducen en un factor constante, implementándose de esta manera la evaporación de feromona. A continuación cada hormiga de la colonia deposita una cantidad de feromona que es función de la calidad de su solución. Inicialmente, el SH no usaba ninguna acción del demonio, pero es relativamente fácil, por ejemplo, añadir un procedimiento de búsqueda local para refinar las soluciones generadas por las hormigas.

Las soluciones en el SH se construyen de la siguiente. En cada paso de construcción, una hormiga m_i escoge ir al siguiente nodo con una probabilidad que se calcula como:

$$P_{ij}^{m_i} = \begin{cases} \frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{\mu \in N_i^n} [\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}, & si \rightarrow j \in N_{m_i}(i) \\ 0, & si \rightarrow otro_caso \end{cases}$$

donde $N_{m_i}(i)$ es el vecindario alcanzable por la hormiga m_i cuando se encuentra en el nodo i , y $\alpha, \beta \in \mathfrak{R}$, son dos parámetros que ponderan la importancia relativa de los rastros de feromona y la información heurística.

Cada hormiga m_i almacena la secuencia que ha seguido hasta el momento y su memoria L_{m_i} , tal como se explicó antes, se utiliza para determinar $N_{m_i}(i)$ en cada paso de construcción.

Volviendo a los parámetros α, β , su función es la que sigue: si $\alpha = 0$, aquellos nodos con una preferencia heurística mejor tienen una mayor probabilidad de ser escogidos, haciendo el algoritmo muy similar a un algoritmo voraz probabilístico clásico (con múltiples puntos de partida en caso de que las hormigas estén situadas en nodos distintos al comienzo de cada iteración). Sin embargo, si $\beta = 0$, sólo se tienen en cuenta los rastros de feromona para guiar el proceso constructivo, lo que puede causar un rápido estancamiento, esto es, una situación en la que los rastros de feromona asociados a una solución son ligeramente superiores al resto, provocando por tanto que las hormigas siempre construyan las mismas soluciones, normalmente óptimos locales. Por tanto es preciso establecer una adecuada proporción entre la información heurística y la información de los rastros de feromona.

La deposición de feromona se realiza una vez que todas las hormigas han acabado de construir sus soluciones. Primero, los rastros de feromona asociados a cada arco se evaporan reduciendo todos los rastros de feromona en un factor constante:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij}$$

Donde $\rho \in (1, 0]$ es la tasa de evaporación. El siguiente paso de cada hormiga es recorrer de nuevo el camino que ha seguido (el camino está almacenado en su memoria local L_{m_i}) y deposita una cantidad de feromona $\Delta\tau_{ij}$ en cada conexión por la que ha viajado:

$$\tau_{ij} \leftarrow \tau_{ij} + \Delta\tau_{ij}^{m_i}, \forall a_{ij} \in S_{m_i}$$

Donde $\Delta\tau_{ij}^{m_i} = f(C(S_{m_i}))$, es decir, la cantidad de feromona que se deposita depende de la calidad $C(S_{m_i})$ de la solución S_{m_i} construida por la hormiga m_i .

Para resumir la descripción del SH, mostramos a continuación el procedimiento Nueva_Hormiga para este algoritmo de OCH en particular:

- 1 Procedimiento Nueva_Hormiga (id_Hormiga)
- 2 $m_i = \text{id_Hormiga}$; $r = \text{generar_estado_inicial}$; $S_{m_i} = r$
- 3 $L_{m_i} = r$
- 4 Mientras (estado_actual \neq estado_objetivo)
- 5 Para cada $S \in N_{m_i}$ (i) hacer

$$P_{ij}^{m_i} = \left[\frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{\mu \in N_i^n} [\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta} \right], si \rightarrow j \in N_{m_i}(i)$$

- 6 Siguiete_est=aplicar_política_de_decisión ($P, N_{m_i}(i)$)
- 7 $r =$ siguiete_est; $S_{m_i} = \langle S_{m_i}, i \rangle$
- 8 $L_{m_i} = L_{m_i} \cup i$
- 9 Fin mientras
 {se ejecuta el procedimiento Evaporación_de_feromona() se lanza y evapora la feromona en cada arco a_{ij} : $\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij}$
- 10 Para cada arco $a_{ij} \in S_{m_i}$ hacer
- 11 $\tau_{ij} = \tau_{ij} + f(C(S_{m_i}))$
- 12 Fin para
- 13 liberar_recursos_hormiga(id_Hormiga)
- 14 Fin Procedimiento

Los creadores del SH también propusieron una versión extendida del algoritmo que normalmente mejoraba los resultados obtenidos, llamada SH elitista. En el SH elitista, una vez que las hormigas han depositado feromona en las conexiones asociadas a sus respectivas soluciones, el demonio realiza una deposición adicional de feromona en las aristas que pertenecen a la mejor solución encontrada hasta el momento en el proceso de búsqueda (esta solución se denomina la mejor global de aquí en adelante). La cantidad de feromona depositada, que depende de la calidad de la mejor solución global, se incrementa en un factor e , que se corresponde con el número de hormigas elitistas que se consideran, tal como sigue:

$$\tau_{ij} = \tau_{ij} + e \cdot f(C(S_{Mejor-Global})), \forall a_{ij} \in S_{Mejor-Global}$$

7.2.2. El Sistema de Colonias de Hormigas (SCH). El SCH es uno de los primeros sucesores del SH que introduce tres modificaciones importantes con respecto a dicho algoritmo de OCH:

El SCH usa una regla de transición distinta, denominada regla proporcional pseudo-aleatoria. Sea m_i una hormiga situada en el nodo i , $q_0 \in [0,1]$ un

parámetro y q un valor aleatorio entre $[0,1]$, el siguiente nodo j se elige aleatoriamente mediante la siguiente distribución de probabilidad:

Si $q \leq q_0$:

$$P_{ij}^{m_i} = \begin{cases} 1, & \text{si } \rightarrow j = \arg_{\mu \in N_{m_i}(i)} \text{MaxN}_{m_i} \{ \tau_{iu} \cdot \eta_{iu}^\beta \} \\ 0, & \text{si } \rightarrow \text{otro_caso} \end{cases}$$

Si no ($q > q_0$):

$$P_{ij}^{m_i} = \begin{cases} \frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{\mu \in N_i^n} [\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}, & \text{si } \rightarrow j \in N_{m_i}(i) \\ 0, & \text{si } \rightarrow \text{otro_caso} \end{cases}$$

Como puede observarse, la regla tiene una doble intención: cuando $q \leq q_0$, explota el conocimiento disponible, eligiendo la mejor opción con respecto a la información heurística y los rastros de feromona. Sin embargo, si $q > q_0$ se aplica una exploración controlada, tal como se hacía en el SH. En resumen, la regla establece un compromiso entre la exploración de nuevas conexiones y la explotación de la información disponible en ese momento.

En Segundo lugar, sólo el demonio (y no las hormigas individualmente) actualiza la feromona, es decir, se realiza una actualización de feromona fuera de línea de los rastros. Para llevarla a cabo, el SCH sólo considera una hormiga concreta, la que generó la mejor solución global, $S_{\text{mejor-global}}$.

La actualización de la feromona se hace evaporando primero los rastros de feromona en todas las conexiones utilizadas por la mejor hormiga global (es importante recalcar que, en el SCH, la evaporación de feromona sólo se aplica a las conexiones de la solución, que es también la usada para depositar feromona) tal como sigue:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij}, \forall a_{ij} \in s_{\text{Mejor-Global}}$$

A continuación, el demonio deposita feromona usando la regla:

$$\tau_{ij} = \tau_{ij} + \rho \cdot f(C(S_{Mejor-Global})), \forall a_{ij} \in S_{Mejor-Global}$$

Adicionalmente, el demonio puede aplicar un algoritmo de búsqueda local para mejorar las soluciones de las hormigas antes de actualizar los rastros de feromona.

En tercer lugar, las hormigas aplican una actualización en línea paso a paso de los rastros de feromona que favorece la generación de soluciones distintas a las ya encontradas.

Cada vez que una hormiga viaja por una arista a_{ij} , aplica la regla:

$$\tau_{ij} \leftarrow (1 - \varphi) \cdot \tau_{ij} + \varphi \cdot \tau_0$$

donde $\varphi \in (0,1]$ es un segundo parámetro de disminución de feromona. Como puede verse, la regla de actualización en línea paso a paso incluye tanto la evaporación de feromona como la deposición de la misma. Ya que la cantidad de feromona depositada es muy pequeña (de hecho, τ_0 es el valor del rastro de feromona inicial y se escogiese de tal manera que, en la práctica, corresponda con el límite menor de rastro de feromona, esto es, con la elección de las reglas de actualización de feromona del SCH ningún rastro de feromona puede caer por debajo de τ_0), la aplicación de esta regla hace que los rastros de feromona entre las conexiones recorridas por las hormigas disminuyan. Así, esto lleva a una técnica de exploración adicional del SCH ya que las conexiones atravesadas por un gran número de hormigas son cada vez menos atractivas para el resto de hormigas que las recorren en la iteración actual, lo que ayuda claramente a que no todas las hormigas sigan el mismo camino.

Los procedimientos *Nueva_hormiga* y *Acciones_del_demonio* (que en este caso interactúa con el procedimiento de *evaporación_de_feromona*) para el SCH son como sigue:

- 1 Procedimiento *Nueva_Hormiga* (*id_Hormiga*)
- 2 $m_i = \text{id_Hormiga}$; $r = \text{Generar_Estado_Inicial}$; $S m_i = r$
- 3 $L m_i = r$
- 4 Mientras (*estado_actual estado_objetivo*)

5 para cada $s \in Nm_i (i)$ hacer calcular $b_{ij} = \tau_{ij} \cdot \eta_{ij}^\beta$
6 $q = \text{generar_valor_aleatorio_en_}[0,1]$
si ($q \leq q_0$)
siguiente_estado = max($b_{ij}, Nm_i^{(i)}$)
si no
para cada $s \in Nm_i (i)$ hacer

$$P_{ij}^{m_i} = \frac{b_{ij}}{\sum_{u \in Nm_i(i)} b_{iu}}$$

sig_est = aplicar_política_de_decisión ($P, Nm_i (i)$)
fin si

7 $r = \text{sig_est}; S_{m_i} = \langle S_{m_i}, i \rangle$
8 $\tau_{ij} = (1 - \varphi) \cdot \tau_{ij} + \varphi \cdot \tau_0$
9 $L_{m_i} = L_{m_i} \cup i$
10 Fin mientras
11 liberar_recursos_hormiga (id_Hormiga)
12 Fin Procedimiento

1 Procedimiento *Acciones_del_demonio*
2 para cada S_{m_i} hacer búsqueda_local(S_{m_i}) {opcional}
3 $S_{\text{mejor-actual}} = \text{mejor_solucion}(S_{m_i})$
4 si (mejor ($S_{\text{mejor-actual}}, S_{\text{mejor-global}}$))
5 $S_{\text{mejor-global}} = S_{\text{mejor-actual}}$
6 Fin si
7 para cada arista $a_{ij} \in S_{\text{mejor-global}}$ hacer
{se ejecuta el procedimiento *evaporación_de_feromona()* se lanza y
evapora feromona en la arista $a_{ij} : \tau_{ij} = (1 - \rho) \cdot \tau_{ij}$ }
8 $\tau_{ij} = \tau_{ij} + \rho \cdot f(C(S_{\text{Mejor-Global}}))$
9 Fin para
10 fin Procedimiento

7.2.3. El Sistema de Hormigas Max-Min (SHMM). Desarrollado por Stützle y Hoos en 1996, es una de las extensiones del SH que mejor rendimiento muestran. Extiende el SH en los siguientes aspectos:

- Se aplica una actualización de los rastros de feromona fuera de línea, de manera similar a como se hace en el SCH. Después de que todas las hormigas han construido su solución cada rastro de feromona sufre una evaporación:

$$\tau_{ij} = (1 - \rho) \cdot \tau_{ij}$$

y a continuación la feromona se deposita siguiendo la siguiente fórmula:

$$\tau_{ij} = \tau_{ij} + f(C(S_{Mejorl})), \forall a_{ij} \in S_{Mejor}$$

La mejor hormiga a la que se le permite añadir feromona puede ser la que tiene una solución mejor de la iteración o la solución mejor global. Los resultados experimentales demuestran que el mejor rendimiento se obtiene incrementando gradualmente la frecuencia de escoger la mejor global para la actualización de feromona. Las soluciones que ofrecen las hormigas suelen ser mejoradas usando optimizadores locales antes de la actualización de feromona.

- Los valores posibles para los rastros de feromona están limitados al rango $[\tau_{\min}, \tau_{\max}]$. Por lo tanto, la probabilidad de un estancamiento del algoritmo disminuye al darle a cada conexión existente una probabilidad, aunque bastante pequeña, de ser escogida. En la práctica, existen heurísticas para fijar los valores de τ_{\min} y τ_{\max} . Se puede ver que, a causa de la evaporación de la feromona, el nivel máximo de feromona en los rastros está limitado a $\tau_{\max}^* = 1/(\rho \cdot C(S^*))$, donde S^* es la solución óptima. Basándonos en este resultado, la mejor solución global puede usarse para estimar τ_{\max} sustituyendo S^* por $S_{\text{mejor-global}}$ en la ecuación de τ_{\max}^* . Para τ_{\min} , normalmente sólo es necesario escoger su valor de tal manera que sea un factor constante menor que τ_{\max} .
- En vez de inicializar los rastros de feromona a una cantidad pequeña, el SHMM los inicializa a una estimación del máximo permitido para un rastro (la estimación puede obtenerse generando una solución S' con una heurística voraz y reemplazando dicha solución S' en la ecuación de τ_{\max}^*). Esto lleva a una componente adicional de diversificación en el algoritmo, ya que al comienzo las diferencias relativas entre los rastros de feromona no serán muy

significativas, lo que no ocurre cuando los rastros de feromona se inicializan a un valor muy pequeño.

La estructura del procedimiento *Acciones_del_demonio* en el SHMM se muestra a continuación:

```

1 Procedimiento Acciones_del_demonio
2 Para cada  $S_{m_i}$  hacer búsqueda_local( $S_{m_i}$ )
3    $S_{\text{mejor-actual}} = \text{mejor\_solucion}(S_{m_i})$ 
4   si ( $\text{mejor}(S_{\text{mejor-actual}}, S_{\text{mejor-global}})$ )
5      $S_{\text{mejor-global}} = S_{\text{mejor-actual}}$ 
6 Fin si
7  $S_{\text{mejor}} = \text{decisión}(S_{\text{mejor-global}}, S_{\text{mejor-actual}})$ 
8 para cada arista  $a_{ij} \in S_{\text{mejor}}$  hacer
9    $\tau_{ij} = \tau_{ij} + f(C(S_{\text{Mejor}}))$ 
10  si ( $\tau_{ij} < \tau_{\text{min}}$ )  $\tau_{ij} \text{ y } \tau_{\text{min}} = \tau_{\text{min}}$ 
11 Fin para
12 si (condición_de_estancamiento)
13   para cada arista  $a_{ij}$  hacer  $\tau_{ij} = \tau_{\text{max}}$ 
14 Fin si
15 Fin Procedimiento

```

7.2.4. El Sistema de Hormigas con Ordenación (SH ordenación). El Sistema de Hormigas con Ordenación es otra extensión del SH propuesta por Bullnheimer, Hartl y Strauss en 1997. Incorpora la idea de ordenar las hormigas para realizar la actualización de feromona, que el demonio realiza, de nuevo, fuera de línea, tal como se muestra a continuación:

- Las m hormigas se ordenan de mejor a peor según la calidad de sus soluciones: (S'_1, \dots, S'_m) , siendo S'_1 la mejor solución construida en la iteración actual.
- El demonio deposita feromona en las conexiones por las que han pasado las $(\sigma - 1)$ mejores hormigas (hormigas elitistas). La cantidad de feromona depositada depende directamente del orden de la hormiga y de la calidad de su solución.
- Las conexiones por las que ha pasado la mejor hormiga global reciben una cantidad adicional de feromona que depende únicamente de la calidad de dicha solución. Esta deposición de feromona se considera la más importante, de hecho, recibe un peso de σ .

Esta metodología de operación tiene efecto al utilizar la siguiente regla de actualización de feromona, la cual se aplica a cada arista una vez que todos los rastros de feromona han sido evaporados:

$$\tau_{ij} \leftarrow \tau_{ij} + \sigma \cdot \Delta \tau_{ij}^{Mg} + \Delta \tau_{ij}^{Orden}$$

Donde

$$\Delta \tau_{ij}^{Mg} = \begin{cases} f(C(S_{mejor-global}, si \rightarrow a_{ij} \in S_{mejor-Global}(i)) \\ 0, si \rightarrow otro_caso \end{cases}$$

$$\Delta \tau_{ij}^{orden} = \begin{cases} \sum_{\mu=1}^{\sigma-1} (\sigma - \mu) \cdot f(C(S'_{\mu})), si \rightarrow a_{ij} \in S'_{\mu} \\ 0, si \rightarrow otro_caso \end{cases}$$

Finalmente, el procedimiento *Acciones_del_demonio* del SH_{ordenación} presenta la siguiente estructura:

- 1 Procedimiento *Acciones_del_demonio*
- 2 Para cada S_{m_i} hacer búsqueda_local(S_{m_i}) {opcional}
- 3 Ordenar (S_1, \dots, S_m) en orden decreciente según la calidad de la solución:
(S'_1, \dots, S'_m)
- 4 si (mejor($S'_1, S_{mejor-global}$))
- 5 $S_{mejor-global} = S'_1$
- 6 Fin si
- 7 desde $\mu = 1$ hasta ($\sigma - 1$) hacer
- 8 para cada arista $a_{ij} \in S'_{\mu}$ hacer
- 9 $\tau_{ij} = \tau_{ij} + (\sigma - \mu) \cdot f(C(S'_{\mu}))$
- 10 Fin para
- 11 Fin desde
- 12 Para cada arista $a_{ij} \in S_{mejor-global}$ hacer
- 13 $\tau_{ij} = \tau_{ij} + \sigma \cdot f(C(S'_{\mu_{mejor-global}}))$
- 14 Fin para

7.2.5. El Sistema de la Mejor - Peor Hormiga (SMPH). Este algoritmo es otra extensión del SH, propuesto por Cordón et. al. en 1999, es un algoritmo de OCH que incorpora conceptos de computación evolutiva.

El SMPH es el algoritmo de OCH que ha obtenido los mejores resultados en la mayoría de la pruebas realizadas.

El SMPH utiliza la misma regla de transición de estados que el SH, así como la misma regla de evaporación de feromona que, al igual que en el SH_{ordenación} y el SHMM, se aplica a todas las transiciones.

Tal como hace el SHMM, el SMPH siempre considera la explotación sistemática de optimizadores locales para mejorar las soluciones de las hormigas.

Formando el núcleo del SMPH podemos encontrar las siguientes tres acciones del demonio:

- La regla mejor-peor de actualización de rastros de feromona, basada en la regla de actualización del vector de probabilidades de PBIL, refuerza las aristas que se encuentran en la mejor solución global. Además, penaliza cada conexión de la peor solución generada hasta el momento, $S_{peor-actual}$, que no se encuentre en la mejor global realizando una evaporación de feromona adicional de esos rastros. Por tanto, la regla de actualización de feromona en el SMPH se convierte en:

$$\tau_{ij} \leftarrow \tau_{ij} + \rho \cdot f(C(S_{mejor-global})), \forall a_{ij} \in S_{mejor-global}$$

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij}, \forall a_{ij} \in S_{peor-actual} \text{ y } a_{ij} \notin S_{mejor-global}$$

- Se realiza una mutación de los rastros de feromona para introducir diversidad en el proceso de búsqueda. Para llevarla a cabo, el rastro de feromona asociado a cada una de las transiciones desde cada nodo (por ejemplo, cada fila de la matriz de rastros de feromona) se muta con una probabilidad P_m utilizando cualquier operador de mutación con codificación real. La propuesta original del SMPH aplicaba un operador que alteraba los rastros de feromona de cada transición mutada añadiendo o restando la misma cantidad en cada iteración. El rango de mutación $mut(it, \tau_{umbral})$, que depende de la media de los rastros de feromona en las transiciones de la mejor solución global, τ_{umbral} , es

más suave en las primeras etapas del algoritmo -donde no hay riesgo de estancamiento- y más fuerte en las últimas etapas, donde el peligro de estancamiento es más fuerte:

$$\tau'_{ij} \leftarrow \begin{cases} \tau_{ij} + mut(it, \tau_{umbral}), si \rightarrow a = 0 \\ \tau_{ij} - mut(it, \tau_{umbral}), si \rightarrow a = 1 \end{cases}$$

donde a es un valor aleatorio en $\{0, 1\}$ e it es la iteración actual.

- Como en otros modelos de OCH, el SMPH considera la reinicialización de los rastros de feromona cuando se estanca la búsqueda, lo que se lleva a cabo fijando cada rastro de feromona a τ_0 . En las primeras versiones del algoritmo se comprobaba si el porcentaje de arcos distintos entre la mejor y la peor solución de la población actual era menor que un cierto valor umbral. En versiones más recientes se utiliza un concepto diferente para evaluar si el algoritmo se ha estancado o no: se comprueba el porcentaje de iteraciones del algoritmo sin haber conseguido una mejora en la mejor solución.

El procedimiento *Acciones_del_demonio* es como sigue:

```

1 Procedimiento Acciones_del_demonio
2 para cada  $S_{m_i}$  hacer búsqueda_local( $S_{m_i}$ )
3  $S_{mejor-actual} = mejor\_solucion(S_{m_i})$ 
4 si ( $mejor(S_{mejor-actual}, S_{mejor-global})$ )
5      $S_{mejor-global} = S_{mejor-actual}$ 
6 fin si
7 para cada arista  $a_{ij} \in S_{mejor-global}$  hacer
8      $\tau_{ij} = \tau_{ij} + \rho \cdot f(C(S_{mejor-global}))$ 
9     suma = suma +  $\tau_{ij}$ 
10 fin para
11  $\tau_{umbral} = \text{sum} / |S_{mejor-global}|$ 
12  $S_{peor-actual} = peor\_solucion(S_{m_i})$ 
13 para cada arista  $a_{ij} \in S_{peor-global}$  y  $a_{ij} \notin S_{mejor-global}$  hacer
14      $\tau_{ij} = (1 - \rho) \cdot \tau_{ij}$ 
15     fin para
16 mut = mut ( $it, \tau_{umbral}$ )
17 para cada nodo  $v \in \{1, \dots, k\}$  hacer
18      $z = generar\_valor\_aleatorio\_en\_ [0,1]$ 

```

```

19  si ( $z \leq P_m$ )
20       $j = \text{generar\_valor\_aleatorio\_en\_}[1, \dots, k]$ 
21       $a = \text{generar\_valor\_aleatorio\_en\_}\{0, 1\}$ 
22      si ( $a = 0$ )  $\tau_{ij} = \tau_{ij} + \text{mut}$ 
23      si no  $\tau_{ij} = \tau_{ij} - \text{mut}$ 
24  fin si
25 Fin para
26 si (condición_de_estancamiento)
27     para cada arista  $a_{ij}$  hacer  $\tau_{ij} = \tau_0$ 
28 Fin si
29 Fin Procedimiento

```

8. METODOLOGÍA DE SOLUCIÓN DE RUTEO UTILIZANDO OCH

Es importante establecer una metodología particular para cada tipo de algoritmo, sin embargo, en general las aplicaciones actuales de OCH tienen unas directivas comunes. Estas directivas se resumen a continuación como una propuesta metodológica:

- (i) Formalizar el problema a resolver generando una representación abstracta de este como un conjunto de componentes y transiciones o a través de un grafo ponderado que será recorrido por las hormigas. Por otro lado, se deben definir claramente los objetivos que el modelo busca cumplir y las relaciones entre la representación del problema y los objetivos.
- (ii) Analizar la literatura existente sobre métodos utilizados para resolver el problema, identificando los problemas encontrados y las ventajas de cada método.
- (iii) Definir de manera apropiada el tipo de decisión que los rastros de feromona inducen. Esto es un paso crucial en la implementación de un algoritmo OCH por lo que se requiere un gran conocimiento de modelo a resolver.
- (iv) Definir de manera apropiada la información heurística asociada a cada componente o transición.
- (v) Escoger un algoritmo de OCH específico que se amolde al problema, para ajustarlo al problema que hay que solucionar, teniendo en cuenta todos los aspectos nuevos o diferentes encontrados.
- (vi) Generar un modelo básico a resolver y evaluar los diseños para las alternativas de solución
- (vii) Reunir algunas instancias específicas del problema que sean representativas y pertinentes al problema en cuestión.
- (viii) Experimentar con el modelo básico.
- (ix) Extender el modelo básico al modelo completo diseñado.

- (x) Diseñar experimentos para efectos de comparación y afinación de los parámetros utilizados.
- (xi) Experimentar con el modelo completo.
- (xii) Afinar los parámetros del algoritmo según los resultados de las experimentaciones o modificar modelo candidato de ser necesario.
- (xiii) Presentar los resultados.

Debe quedar claro que los pasos mostrados sólo dan una idea a *grosso modo* sobre la implementación del algoritmo de OCH. Además, en muchas ocasiones la implementación es un proceso iterativo, en el que a medida que se obtiene una visión más profunda del problema y del comportamiento del algoritmo, se deben revisar algunas de las decisiones iniciales. Por último, es importante insistir en el hecho de que los pasos más importantes son los primeros cinco, ya que una elección poco acertada en esos puntos no suele poder arreglarse con un buen refinamiento del algoritmo.

9. CONSTRUCCIÓN DEL MODELO MATEMÁTICO PARA RESOLVER EL PROBLEMA PLANTEADO

9.1. FORMULACIÓN MATEMÁTICA DEL PROBLEMA

El problema que se enfrenta en este trabajo es un problema combinatorio que puede ser formulado matemáticamente como un grafo dirigido $G = (V, A)$. La notación¹⁰⁷ es la siguiente:

Existe un conjunto de vértices (nodos) $V = \{v_0, v_1, \dots, v_n\}$, donde v_0 representa el depósito (bodega de origen), mientras que v_i ($i = 1, \dots, n$) representa a cada uno de los n clientes a ser visitados.

El conjunto de arcos de G está dado por: $A = \{(v_i, v_j) / v_i, v_j \in V; i \neq j\}$.

C es la matriz de distancias o costos c_{ij} positivos entre cada par de vértices v_i y v_j (incluyendo depósito) definida como $C = \{c_{ij}\} \in \mathfrak{R}^{(n+1) \times (n+1)}$.

$q \in \mathfrak{R}^n$ es el vector de las demandas de los clientes donde q_i representa la cantidad de bienes requeridos por el cliente v_i .

m representa el número de vehículos. Se asumen todos idénticos, con una capacidad Q por vehículo.

R_i es la ruta del vehículo i .

¹⁰⁷ La notación utilizada es la misma utilizada en: BARÁN B. y HERMOSILLA, A. Comparación de un Sistema de Colonias de Hormigas y una Estrategia Evolutiva para el Problema del Ruteo de Vehículos con Ventanas de Tiempo en un Contexto Multiobjetivo. IX Jornadas Iberoamericanas de Informática, Cartagena de Indias, Colombia. 2003.

$[e_i, l_i]$ representa la ventana de tiempo del cliente v_i ; con e_i como la hora más temprana y l_i como la hora más tardía en que se puede iniciar el servicio a dicho cliente v_i .

δ_i representa el tiempo requerido en descargar la cantidad de mercancía q_i en el cliente v_i (tiempo de servicio). Por cuestiones prácticas se utiliza un tiempo de descarga fijo en cada cliente δ_i

Como se había definido anteriormente este trabajo, considera sólo el problema simétrico (misma distancia/costo entre dos ciudades tanto de ida como de vuelta): $c_{ij} = c_{ji}$ para cada $(v_i, v_j) \in A$. En este caso, por simplicidad es común reemplazar A por el conjunto $E = \{(v_i, v_j) / v_i, v_j \in V; i < j\}$

Se asume una velocidad constante y unitaria para los vehículos, con lo que puede asumirse para facilitar el entendimiento, que el tiempo t_{ij} necesario para ir de v_i a v_j es igual a la distancia c_{ij} .

El intervalo $[e_0, l_0]$ en el depósito, es conocido como horizonte de planeación y hace referencia al tiempo en el que la bodega estará en operación.

9.2. DEFINICIÓN DE COSTOS

En los problemas de ruteo, el costo esta asociado normalmente al tamaño de la flota, la distancia recorrida y el tiempo empleado. Como anteriormente definimos la distancia en función del tiempo, el costo estará asociado únicamente al tamaño de la flota y el tiempo requerido en cada ruta R_i .

El costo de m camiones es un costo de carácter fijo según el numero de camiones, por lo que en general los algoritmos de optimización de ruteo deben minimizar en lo posible el tamaño de la flota.

Se define una ruta i como $R_i = \{v_0^i, v_1^i, v_2^i, \dots, v_{k_i}^i, v_{k_i+1}^i\}$, donde $v_j^i \in V$ y $v_0^i = v_{k_i+1}^i = 0$. Donde v_j^i denota al j -ésimo cliente atendido en la ruta i , la cual tiene a su vez un total de k clientes atendidos. Los clientes denotados iguales a cero(0), son la bodega al inicio de la ruta y al final de esta.

Las rutas a establecer estarán regidas por algunas condiciones, propias del problema:

- Las capacidades de los vehículos están limitadas a q por lo tanto:

$$\sum_{j=1}^{k_j} q_j^i \leq Q, \rightarrow i = 1, 2, \dots, m.$$

- Las ventanas de tiempo establecen limitaciones en orden de visitas, si se define a b_v como el instante en que se inicia el servicio al cliente v , entonces ese instante debe estar limitado a :

$$e_{v_j^i} \leq b_{v_j^i} \leq l_{v_j^i}, \rightarrow i = 1, 2, \dots, m, \quad j = 1, 2, \dots, k + 1.$$

Para definir el valor de $b_{v_j^i}$, se establece que todo vehículo viajara al cliente v_j^i después de haber terminado de atender al cliente v_{j-1}^i , entonces:

$$b_{v_j^i} = \max\{e_{v_j^i}; b_{v_{j-1}^i} + \delta_{v_{j-1}^i} + c_{v_{j-1}^i, v_j^i}\}$$

Para efectos prácticos se considera $b_0^i = e_0 = 0$, es decir, los vehículos parten del depósito en el mismo instante. También es práctico considerar a $\delta_0^i = 0$ puesto que el vehículo no debe descargar nada en la bodega al iniciar el recorrido.

Los tiempos de espera w estarán entonces definidos por:

$$w_{v_j^i} = \max\{0; b_{v_j^i} - b_{v_{j-1}^i} - \delta_{v_{j-1}^i} - c_{v_{j-1}^i, v_j^i}\}$$

El tiempo de espera que este trabajo hace referencia se define como el tiempo que un vehiculo debe esperar en la ubicación de un cliente antes de poder entregar su carga de acuerdo al intervalo de recepción establecido.

Finalmente podemos definir el costo asociado al tiempo requerido en la ruta R_i como¹⁰⁸:

$$T(R_i) = \sum_{j=1}^{k_i} c_{v_j^i, v_{j+1}^i} + \sum_{j=1}^{k_i} \delta_{v_j^i} + \sum_{j=1}^{k_i} w_{v_j^i}$$

9.3. DEFINICIÓN DEL OBJETIVO

Como se había definido anteriormente este trabajo presenta un modelo matemático multiobjetivo, en el cual además de optimizar el beneficio económico, se pretende incluir un factor estratégico definido como la satisfacción del cliente, mediante el control del tiempo máximo de servicio.

En un contexto multiobjetivo se manejan diferentes dimensiones de una función objetivo general, cada dimensión representa un objetivo independiente. Por lo general esta función general cuenta con varias soluciones optimas diferentes que pueden ser incluso, nada comparables entre si. La función objetivo general se define como:

$$F = [F_1, F_2, F_3, F_4]^T$$

En donde cada F_z es una funcion definida para un objetivo particular:

$F_1 = m$, se desea minimizar el tamaño de la flota (numero de vehículos). Esta función nos da una idea del costo de la inversión en vehículos y del costo de su mantenimiento.

¹⁰⁸ Analisis Tomado de: BARÁN B. y HERMOSILLA, A. Comparación de un Sistema de Colonias de Hormigas y una Estrategia Evolutiva para el Problema del Ruteo de Vehículos con Ventanas de Tiempo en un Contexto Multiobjetivo. *IX Jornadas Iberoamericanas de Informática*, Cartagena de Indias, Colombia. 2003.

$F_2 = \sum_{i=1}^m \sum_{j=0}^{k_i} C_{v_j^i, v_{j+1}^i}$, se desea minimizar la distancia total recorrida(o tiempo de viaje). Esta función busca beneficios en cuanto al consumo de combustibles, aceites, llantas, etc... de la flota de camiones.

$F_3 = F_2 + \sum_{i=1}^m \sum_{j=1}^{k_i} \delta_{v_j^i} + \sum_{i=1}^m \sum_{j=0}^{k_i} W_{v_j^i}$, se desea minimizar el tiempo total destinado para satisfacer todos los clientes. Esta función busca beneficios en cuanto a los gastos en honorarios a conductores y seguros de vehículos.

$F_4 = \max_i (b_{v_{k_i+1}^i}^i)$, se desea minimizar el tiempo de la ruta mas larga, con lo que se asegura que sin importar la asignación de rutas, el ultimo cliente atendido siempre estará satisfecho.

Formalmente el problema queda formulado como:

$$\text{Minimizar} [f_1(x), f_2(x), f_3(x), f_4(x)]^T \rightarrow \text{Sujeto a } x \in S$$

Donde S es la región de soluciones factibles y $x = (x_1, x_2, \dots, x_l)^T$ es un vector de l variables de decisión. Se denota como ψ^x a la solución multiobjetivo dada por la matriz de rutas $M = \{R_i\}$

9.4. METODOLOGÍA DE SOLUCIÓN AL PROBLEMA MULTI OBJETIVO

El problema multiobjetivo crea en si un reto, en razón a que ningún objetivo se considera mas importante que otro y las soluciones obtenidas son difícilmente

comparables, por lo que para escoger un óptimo (u óptimos) se crea la necesidad de establecer un mecanismo para comparar las respuestas obtenidas en principio.

Al establecer los mecanismos de integración entre los procesos de búsqueda y creación de las respuestas y los procesos de toma de decisiones (búsqueda de la mejor respuesta), se debe evitar entrar en nuevos conflictos provocados por la naturaleza no cuantificable de estos algoritmos. La calidad de las respuestas difícilmente podrá ser evaluada si no se pondera de alguna manera las preferencias en el cumplimiento de los objetivos, pero esto a su vez representa un problema si no se conoce (como es común) la sensibilidad en el beneficio general.

El enfoque mas utilizado en la solución de problemas multiobjetivo es el de utilizar el concepto de dominancia. Este concepto estipula que un solución domina a otra(es mejor que otra) si es por lo menos es igual a otra en cada objetivo y por lo menos mejor en un objetivo¹⁰⁹. La definición matemática esta dada por:

Para que una respuesta A domine a otra B , se debe cumplir que:

- (i) A es por lo menos igual que B en cada objetivo:
 $F_k(A) \leq F_k(B), \forall k \in \{1, \dots, k\}$
- (ii) A es mejor que B en por lo menos en un objetivo:
 $\exists k \in \{1, \dots, k\}, F_k(A) < F_k(B)$

Dado que diferentes valores de las variables de decisión representan diferentes compromisos, la resolución del problema multiobjetivo no se concentra en hallar un único valor solución, sino que se plantea hallar un conjunto de soluciones no dominadas, de acuerdo a la definición anterior.

El conjunto de soluciones óptimas al problema de optimización multiobjetivo se compone de los vectores factibles *no dominados*. Este conjunto se denomina conjunto *óptimo de Pareto* y está definido por¹¹⁰:

$$P^* = \{x \in \Omega / \neg \exists x' \in \Omega f(x') \leq f(x)\}$$

¹⁰⁹ CRICHIGNO Jorge, TALAVERA Francisco, PRIETO Joel, BARÁN Benjamín, Enrutamiento Multicast Utilizando Optimización Multiobjetivo, Asunción: Centro Nacional de Computación. 2002

¹¹⁰ TORO, Andrés. GIRÓN, JM. BESADA, E., FERNÁNDEZ, P. LÓPEZ, JA. Aplicación del diseño de control multiobjetivo y multivariable, con algoritmos genéticos, al proceso de la fermentación de la cerveza, Departamento de Arquitectura de Computadores y Automática, Universidad Complutense de Madrid.

La región de puntos definida por el conjunto óptimo de Pareto en el espacio de valores de las funciones objetivo se conoce como frente de Pareto. Formalmente, el frente de Pareto está definido por¹¹¹:

$$FP^* = \{u = (f_1(x), f_2(x), \dots, f_k(x)) / x \in P^*\}$$

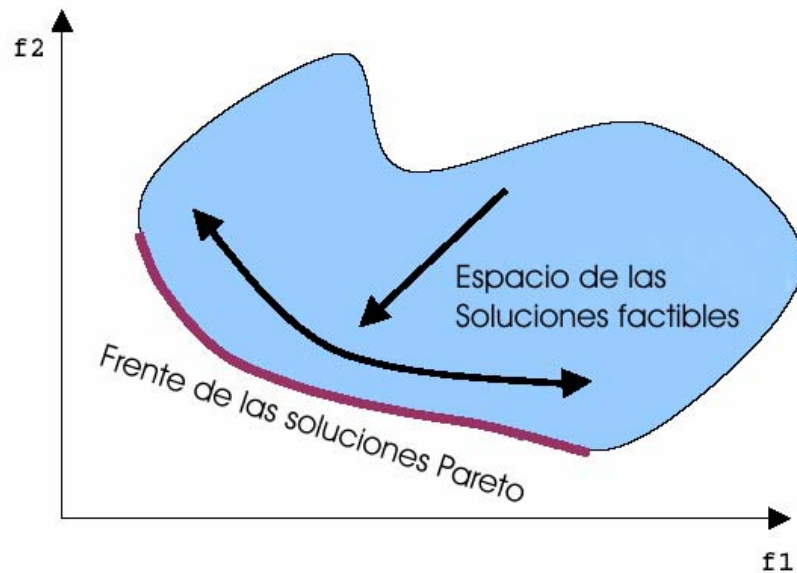


Figura 2. Espacio de las soluciones factibles¹¹²

Las ventajas principales del uso de este enfoque son la facilidad con que se puede lidiar con un número progresivo de objetivos y que disminuye el riesgo de caer en óptimos locales. Además, por tratarse en realidad de un conjunto múltiple de soluciones equivalentes entre sí, el usuario tiene libertad de escoger la que más se ajuste a un determinado compromiso.

¹¹¹ Ibid. P.16

¹¹² Tomada de: NESMACHNOW, Sergio. Una Versión Paralela del Algoritmo Evolutivo para Optimización Multiobjetivo NSGA-II y su Aplicación al Diseño de Redes de Comunicaciones Confiables. En Actas del Tercer Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados. Córdoba: España.2004.

10. CONSTRUCCIÓN DEL ALGORITMO PROPUESTO

Varias técnicas metaheurísticas de optimizaciones con colonias de hormigas son utilizadas en la creación del algoritmo propuesto. En particular se utiliza como base el algoritmo Mejor/Peor Hormiga¹¹³ propuesta por Cordón, puesto que ha demostrado obtener los mejores resultados en la solución del problema del agente viajero¹¹⁴, el MACS-VRPTW¹¹⁵ propuesto por Luca Gambardella et. al, y el MOACS-VRPTW^{116, 117} propuesto por Barán B. et. al.

El algoritmo propuesto tiene como algoritmo base el algoritmo Mejor/Peor hormiga, sin embargo, debido a la naturaleza del problema enfrentado en este proyecto y al aporte investigativo del autor, el modelo propuesto presenta varias diferencias con el algoritmo base incluyendo algunas que son utilizadas por primera vez (según documentación consultada), por lo que difícilmente puede catalogarse como un algoritmo Mejor/Peor hormiga.

El modelo propuesto plantea no solamente la aplicación de conceptos de los algoritmos OCH y en particular el algoritmo Mejor/Peor hormiga sino que además integra los conceptos de búsqueda multiobjetivo en una sola colonia de hormigas que construye soluciones en un tiempo virtual lineal (todas las hormigas parte en el tiempo cero). A diferencia de otros algoritmos que siempre encuentran una solución en cada iteración puesto que construyen soluciones exhaustivamente en tiempos virtuales no lineales (las hormigas parte una a la vez, cada una

¹¹³ CORDÓN, O. HERRERA, F. MORENO, L. Integración de Conceptos de Computación Evolutiva en un Nuevo Modelo de Colonias de Hormigas. *En VIII Conferencia de la Asociación Española para la Inteligencia Artificial*, Murcia (España), 1999, Vol. II, páginas 98-105.

¹¹⁴ ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco. *La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques*. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

¹¹⁵ GAMBARDELLA, Luca. TAILLARD, Éric. AGAZZI, Giovanni. *MACS-VRPTW: A Multiple Ant Colony System For Vehicle Routing Problems With Time Windows (On Line)*. Lugano, Suiza. <http://www.idsia.ch> (Consultado: Mayo 5 de 2005).

¹¹⁶ BARÁN, B. y SCHAEERER, M. A multiobjective Ant Colony System for Vehicle Routing Problem with Time Windows. *Proceedings of the 21st IASTED International Conference APPLIED INFORMATICS*. Innsbruck, Austria. 2003.

¹¹⁷ BARÁN B. y HERMOSILLA, A. Comparación de un Sistema de Colonias de Hormigas y una Estrategia Evolutiva para el Problema del Ruteo de Vehículos con Ventanas de Tiempo en un Contexto Multiobjetivo. *IX Jornadas Iberoamericanas de Informática*, Cartagena de Indias, Colombia. 2003.

empezando en un tiempo cero virtual) el algoritmo propuesto no garantiza que una solución sea encontrada en cada iteración, en contra partida, esta metodología permite que la interacción entre hormigas se de según se construye la ruta sin importar el orden de salida.

El modelo propuesto utiliza una sola colonia de hormigas para minimizar los cuatro objetivos planteados. El algoritmo intenta encontrar una solución factible al problema guiado por los valores heurísticos en los nodos y por los rastros memorísticos de feromona. Los valores memorísticos son utilizados por las cuatro funciones objetivo por lo que el conocimiento intrínseco en los valores de la feromona son de igual importancia para cada objetivo en el frente pareto.

La idea del algoritmo es buscar construir soluciones factibles con un número de vehículos fijo establecido en el problema. Al iniciar el algoritmo el grupo m de vehículos parte a la vez desde la bodega y sucesivamente escogen el siguiente cliente factible (determinado por las restricciones del problema: capacidad del vehiculo y ventana de tiempo) de los que no han sido visitados. Cuando un vehiculo no tiene clientes factibles para escoger, regresa a la bodega. Si al regresar todos los vehículos a la bodega ningún cliente falto por visitar, una solución factible se ha encontrado.

Cuando una solución factible es encontrada se cataloga como la mejor solución global y de ahí en adelante todas las soluciones encontradas son comparadas con la solución global del momento. Cada vez que una mejor solución global es encontrada la solución anterior es olvidada.

10.1. LA VISIÓN MULTIOBJETIVO

Los vehículos determinan el moverse desde el cliente v_i hasta el cliente v_j , utilizando información heurística así como información de la feromona. En cada cliente v_i se asigna una probabilidad de escogencia para cada cliente v_j utilizando la siguiente información:

10.1.1. Estrategia Heurística. La información heurística esta dada por los factores de visibilidad¹¹⁸. Expresión utilizada para denotar la deseabilidad de escoger una ruta para satisfacer un objetivo. El modelo propuesto busca optimizar cuatro objetivos pero el primero es totalmente ajeno en la escogencia del siguiente cliente por lo que se plantean tres factores de visibilidad, uno para cada uno de los objetivos restantes.

La visibilidad para el objetivo F_2 (tiempo de viaje total) esta dada por:

$$\eta_L(i, j) = 1/c_{ij}$$

La preferencia de caminos cortos hace que el tiempo de viaje disminuya.

La visibilidad para el objetivo F_3 (Tiempo total de todas las entregas) esta dada por¹¹⁹:

$$Tiempo_actual(j) = \max(Tiempo_actual(i) + c_{ij}; e_j)$$

$$\Delta c_{ij} = Tiempo_actual(j) - Tiempo_actual(i)$$

$$g_{ij} = \Delta c_{ij} * (l_j - Tiempo_actual(i))$$

$$g_{ij} = \text{Max}(1, d_{ij})$$

$$\eta_o = 1/g_{ij}$$

La preferencia de clientes donde la llegada este después de la apertura de la ventana y lo mas cerca posible del cierre de la ventana hace que el tiempo de espera en general disminuya para todos los clientes.

La visibilidad para el objetivo F_4 (Tiempo de ruta más larga) esta dado por:

$$\eta_h = 1/d_j$$

La preferencia a escoger clientes con menos demanda hace que las rutas se nivelen haciendo mas corto el tiempo de la ruta mas larga.

¹¹⁸ CASAL Ricardo, TONCOVICH Adrián, LÓPEZ Nancy, CORRAL Rafael, Estudio comparativo de estrategias heurísticas para un problema de secuenciamiento multiobjetivo. P. 43 Bahía Blanca:Universidad Nacional del Sur. Argentina 1998

¹¹⁹ Tomada de: GAMBARDELLA, Luca. TAILLARD, Éric. AGAZZI, Giovanni. MACS-VRPTW: A Multiple Ant Colony System For Vehicle Routing Problems With Time Windows (On Line). Lugano, Suiza. <http://www.idsia.ch> (Consultado: Mayo 5 de 2005).

De esta manera se crean tantos factores de visibilidad como objetivos a cumplir se tengan. Si el contexto del problema definiera que un objetivo es mas importante que otro (en el problema planteado todos son iguales) la importancia se vera reflejada en las potencias de cada factor de visibilidad en la formula de probabilidad de escogencia que es asignada a cada cliente.

10.1.2. Estrategias memorísticas (Feromona). La feromona esta inicialmente distribuida por todo el grafo, según el algoritmo se ejecuta los rastros se modifican con el fin de premiar las rutas que han logrado los mejores resultados y desfavorecer los de peores resultados. La matriz de feromona es usada de igual manera por cada uno de los cuatro objetivos del problema por lo que la información guardada en ella es util al grupo de objetivos y no a uno en particular.

10.2. REGLAS DE TRANSICIÓN

El algoritmo propuesto utiliza la misma regla de transición del algoritmo Mejor/Peor Hormiga según la cual:

Para un Q_0 ingresado, y un q aleatorio en cada nodo se decide entre explotación del conocimiento o la exploración aleatoria.

Si $q \leq Q_0$ (exploración) se escoge el siguiente cliente a atender aleatoriamente entre los clientes viables de visitar.

Si $q \geq Q_0$ (explotación del conocimiento) se procede a escoger el cliente con mayor probabilidad P_{ij}^k .

$$P_{ij}^k = \begin{cases} \frac{[\tau_{ij}]^\alpha \cdot [\eta_L^* \eta_o^* \eta_h]^{\beta}}{\sum_{\mu \in N_i^n} [\tau_{ij}]^\alpha \cdot [\eta_L^* \eta_o^* \eta_h]^{\beta}}, & \text{si } i \rightarrow j \in N_k(i) \\ 0, & \text{si } i \rightarrow \text{otro_caso} \end{cases}$$

Donde $N_k(i)$ es el conjunto de los clientes factibles a recorrer en la ruta k desde i, α es la importancia de la feromona y β es la importancia de los valores heurísticos.

Se sabe que del conjunto de $N_k(i)$ clientes, los mas lejanos no serán escogidos pues su P_{ij}^k será siempre alta. Para optimizar tiempo de ejecución y memoria se introduce el concepto de vecinos factibles, los vecinos factibles son los G (parámetro) por ciento clientes factibles más cercanos. De esta manera el conjunto $N_k(i)$ es ahora menor, y se requiere menos tiempo de cálculo con una probabilidad muy escasa de perder una buena solución. Para evitar que el conjunto $N_k(i)$ sea muy pequeño, la opción de limitar a los vecinos aplica únicamente cuando se tengan más de 20 clientes factibles. El concepto de Vecino es también utilizado en la búsqueda local.

10.3. ITERACIONES, CICLOS Y SUBDIVISIONES

El algoritmo esta dividido en R ciclos independientes cuya duración de ejecución debe ser menor o igual a T, cada ciclo contiene un numero S de iteraciones en el cual se intentan crear soluciones, dentro de cada ciclo existen subdivisiones de H iteraciones antes de ejecutar acciones del demonio como la actualización de feromona o la mutación de esta. Según las especificaciones del usuario el algoritmo puede iterar muchas o pocas veces antes de terminar, por defecto se usa 10 ciclos de 100 iteraciones. Las subdivisiones condicionan la frecuencia con que se actualizara la feromona, por defecto es 25 iteraciones.

10.4. ACTUALIZACIÓN DE LA FEROMONA

Para conocer cual es la mejor y la peor ruta que produjo resultados el algoritmo utiliza un indicador para llevar un registro de la calidad de las soluciones encontradas, este indicador esta dado por:

$$Solucion_indicador = 1/[f_2(x), f_3(x), f_4(x)]$$

Donde x representa el conjunto de rutas de la solución encontrada S_{actual} y cada F_y la función de cada objetivo (sin incluir el numero de vehículos pues siempre será constante en la ejecución del algoritmo).

Es de aclarar que esta función no determina si una solución pertenece al frente pareto, únicamente mide la calidad de la solución encontrada.

El proceso de actualización de la feromona se compone de dos etapas:

- El algoritmo itera tantas veces como sea especificado con la feromona inicial, si después de este numero de iteraciones no se encuentra una solución mejor que la global, la matriz de feromona se modifica premiando la ruta con la mejor solución hasta el momento (Global actual) según:

$$\tau_{ij} = \tau_{ij} + \rho * Solucion_indicador \quad \forall a_{ij} \in S_{mejor-global}$$

Si en esas primeras iteraciones no se encuentra una solución se aborta la actualización hasta encontrar una solución.

- De la misma forma, la matriz de feromona se modifica castigando la ruta que produjo el peor resultado, evitando modificar nuevamente los arcos que pertenecen a la mejor solución global:

$$\tau_{ij} = \tau_{ij} * (1 - \rho) \quad \forall a_{ij} \in S_{peor-actual} \text{ y } a_{ij} \notin S_{mejor-global}$$

10.5. INICIACIÓN DE FEROMONA

La iniciación de la feromona en este algoritmo puede ser definida de dos maneras según la escogencia del usuario. El objetivo es crear diversidad en el algoritmo mediante varias opciones que mas adelante serán evaluadas en función de las respuestas y servirán para identificar los caminos a seguir para perfeccionar el algoritmo.

10.5.1. Matriz de feromona Constante. La forma clásica de iniciación de la feromona, establece un valor constante para todos los arcos de la matriz de feromona. Según este método este valor será establecido por la calidad de alguna solución conocida e igual a su *Solucion_indicador*. Sin embargo, no siempre se tiene una solución conocida, por lo que se asigna un valor de feromona promedio de los valores conocidos para instancias similares. Para el caso del algoritmo propuesto utilizaremos un valor igual a $2.37035 * 10^{-12}$, obtenido al calcular la *Solucion_indicador* de una solución encontrada utilizando valores aleatorios de prueba en {0.1,0.95} para el problema C101¹²⁰.

10.5.2. Escalamiento de la matriz de feromonas. En un trabajo previo realizado por Almirón¹²¹ se propone una mejora en los algoritmos OCH mediante el scalamiento lineal en iniciación de la matriz de feromonas. Con esto, se privilegian los caminos más cortos en la probabilidad de que una hormiga elija cierta ciudad, dando a las hormigas un conocimiento previo del problema, al riesgo de poder perder alguna solución globalmente óptima, que utiliza caminos largos entre algunos pares de ciudades.

El método consiste en modificar los rastros de feromona con que inicia el algoritmo, de tal forma que a una mayor distancia le corresponde una cantidad menor de feromonas en $t=0$ conforme:

$$\tau_{ij}(0) = R_{\min} + \left[\frac{R_{\max} - R_{\min}}{c_{\max} - c_{\min}} \right] * (c_{\max} - c_{ij})$$

¹²⁰ Véase 9.11.2. *Instancias*.

¹²¹ ALMIRÓN, Marta. VIVEROS, Enrique. BARÁN, Benjamín, Sistema distribuido de hormigas para el problema del cajero viajante. p. 12 Asuncion: Universidad Nacional de Asunción. 2004

R_{max} = es una constante definida como valor máximo de feromona correspondiente a la menor distancia.(0.1)

R_{min} = es una constante definida como valor mínimo de feromona correspondiente a la menor distancia. (0.95)

c_{max} = es la mayor distancia en la matriz de feromonas d_{ij}

c_{min} = es la menor distancia en la matriz de feromonas d_{ij}

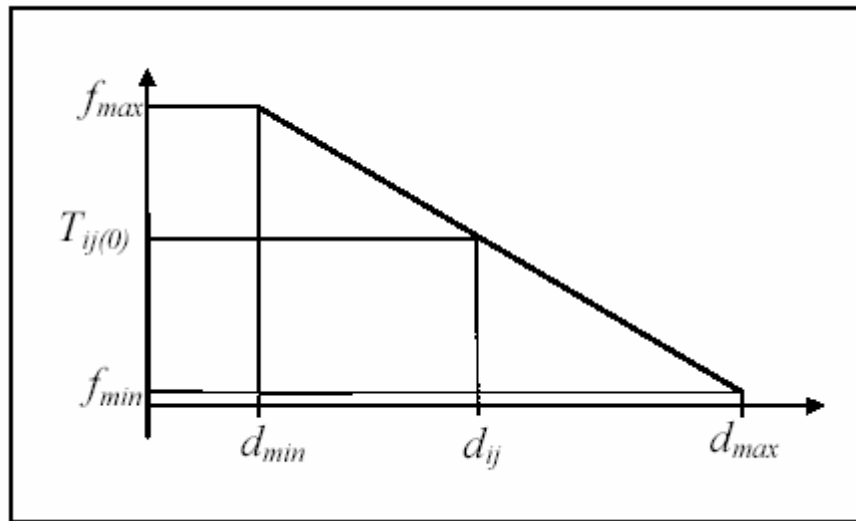


Figura 3. Esquema de valores escalado para la matriz de feromona¹²²

Según el método escogido por el usuario, el algoritmo creará la matriz de feromona inicial τ_0 a ser utilizada durante la ejecución del algoritmo.

10.6. REINICIACIÓN DE FEROMONA

El algoritmo propuesto contempla la reiniciación de los valores de feromona a su nivel inicial τ_0 cuando una solución mejor global ha sido encontrada, de esta manera se desecha la información que está basada en una solución antigua de menor calidad y se comienza a explotar el conocimiento dado por la solución global encontrada.

¹²² Figura tomada de: ALMIRÓN, Marta. VIVEROS, Enrique. BARÁN, Benjamín, Sistema distribuido de hormigas para el problema del cajero viajante. p. 12 Asunción: Universidad Nacional de Asunción. 2004

Cuando $S_{Mejor-Global}$ *concontrada* $\rightarrow \tau_{ij} = \tau_0$

10.7. OPERADOR DE MUTACIÓN

Para lograr una mayor diversidad en el proceso de búsqueda se incluye un operador de mutación que modifica la matriz de feromona. El operador de mutación se ejecuta justo antes de cada iteración y una vez finalizada esta restablece la matriz original para ser mutada en la siguiente iteración.

A diferencia del algoritmo Mejor/Peor hormiga el algoritmo propuesto modifica la matriz de feromona únicamente en una fila y una columna, este método utilizado en los algoritmos genéticos realiza en cada iteración modificaciones a una fila y una columna de la matriz de feromona escogidas aleatoriamente mediante su sobreposición. El método explota la diversidad que puede generar el cambio en todos los rastros de salida de un cliente y en los de llegada de otro¹²³.

$$\tau'_{ij} \begin{cases} \tau_{ia} = \tau_{bi} \rightarrow (i = 0,1,2,\dots,n-1) \\ \tau_{bj} = \tau_{jb} \rightarrow (j = 0,1,2,\dots,n-1) \text{ , donde } a \text{ y } b \text{ son valores aleatorios} \\ \tau_{ij}, \text{ para los demas valores} \end{cases}$$

El algoritmo también presenta la opción de ejecutar otro tipo de mutación. En este caso la modificación se realiza a dos arcos aleatorios de la solución global, intercambiando sus valores. El método explora de una manera más conservadora, la diversidad generada mediante la modificación del camino más deseable memorísticamente por los vehículos.

$$\tau_{ab} = \tau_{ba}, \quad \tau_{ba} = \tau_{ab}, \quad \forall a_{ij} \in S_{mejor-global}$$

a,b son valores aleatorios

¹²³ BRÄYSY, Olli. BERGER, Jean. BARKAOUI, Mohamed. A New Hybrid Evolutionary Algorithm For The Vehicle Routing Problem With Time Windows. p. 28 Vaasa: University of Vaasa.2002

10.8. BÚSQUEDA LOCAL

El algoritmo propuesto utiliza una búsqueda local tipo 2-Opt. para mejorar las soluciones encontradas. Cada vez que una solución es encontrada, se ejecuta una búsqueda local que trata de mejorar los resultados de las funciones objetivo, si la solución creada es mejorada se toma esta nueva solución como la solución a mejorar hasta que no se encuentren mejoras en esta solución.

El método 2-Opt. consiste en mover al menos dos clientes a una posición diferente dentro de la misma u otra ruta a la vez sin violar ninguna restricción.

Para llevar a cabo este cambio, se selecciona aleatoriamente un cliente h_{x+1} y se procede a buscar los clientes vecinos factibles que podrían tomar su posición, si alguno h_y es encontrado se toma como el nuevo h_{x+1} , y h_{y-1} se conecta con el viejo h_{x+1} .

Este cambio puede traer dos casos diferentes si el punto h_y hace parte o no de la ruta de h_{x+1} . Si hace parte de la misma ruta, un intervalo de la ruta debe cambiar de sentido para que la ruta pueda ser cerrada. Si no hace parte de la misma ruta las dos rutas serán cruzadas. En ambos casos las restricciones deben ser comprobadas para todos los puntos siguientes a los modificados en las rutas involucradas¹²⁴.

Si ninguna restricción es violada se evalúan las funciones objetivo para conocer si se mejoro la solución original.

La optimización local puede ser exhaustiva para cada cliente de toda la red o puede ser limitada solo para algunos clientes. El algoritmo propuesto ejecuta la 2-opt. a todos los clientes, lo cual incrementa considerablemente el tiempo de ejecución si muchas soluciones son encontradas.

¹²⁴ GARDEL, Pedro. GÓMEZ, Osvaldo. BARÁN, Benjamín . Estudio del Espacio de Soluciones del Problema del Cajero Viajante. P. 12. San lorenzo: Centro Nacional de Computación. 2001.

10.9. ALGORITMO PROPUESTO

Como fue señalado en capítulos anteriores, los algoritmos OCH pueden ser divididos en bloques. Para facilitar el entendimiento en este trabajo dividiremos el algoritmo en seis bloques:

- (i) Bloque de Iniciación
- (ii) Bloque de actividades de creación de soluciones
- (iii) Bloque de actualización de feromona
- (iv) Bloque de mutación
- (v) Bloque de búsqueda local
- (vi) Bloque de evaluación de soluciones al problema multiobjetivo

10.9.1. Bloque de iniciación. En esta fase se realizan los ajustes para que el algoritmo pueda iniciar sus funciones. El algoritmo lee los parámetros definidos por el usuario que se consideraran mas adelante en el algoritmo como los contadores, el coeficiente de evaporación de la feromona...etc, se lee la matriz de distancias C , el vector de demandas d , valores $[e_i, l_i]$ de los intervalos de recepción de los clientes, los tiempos de descarga δ para cada cliente, ...etc.

En este mismo bloque se realizan algunos cálculos como las distancias de la matriz de costos, se calcula los valores iniciales de la matriz de feromona,

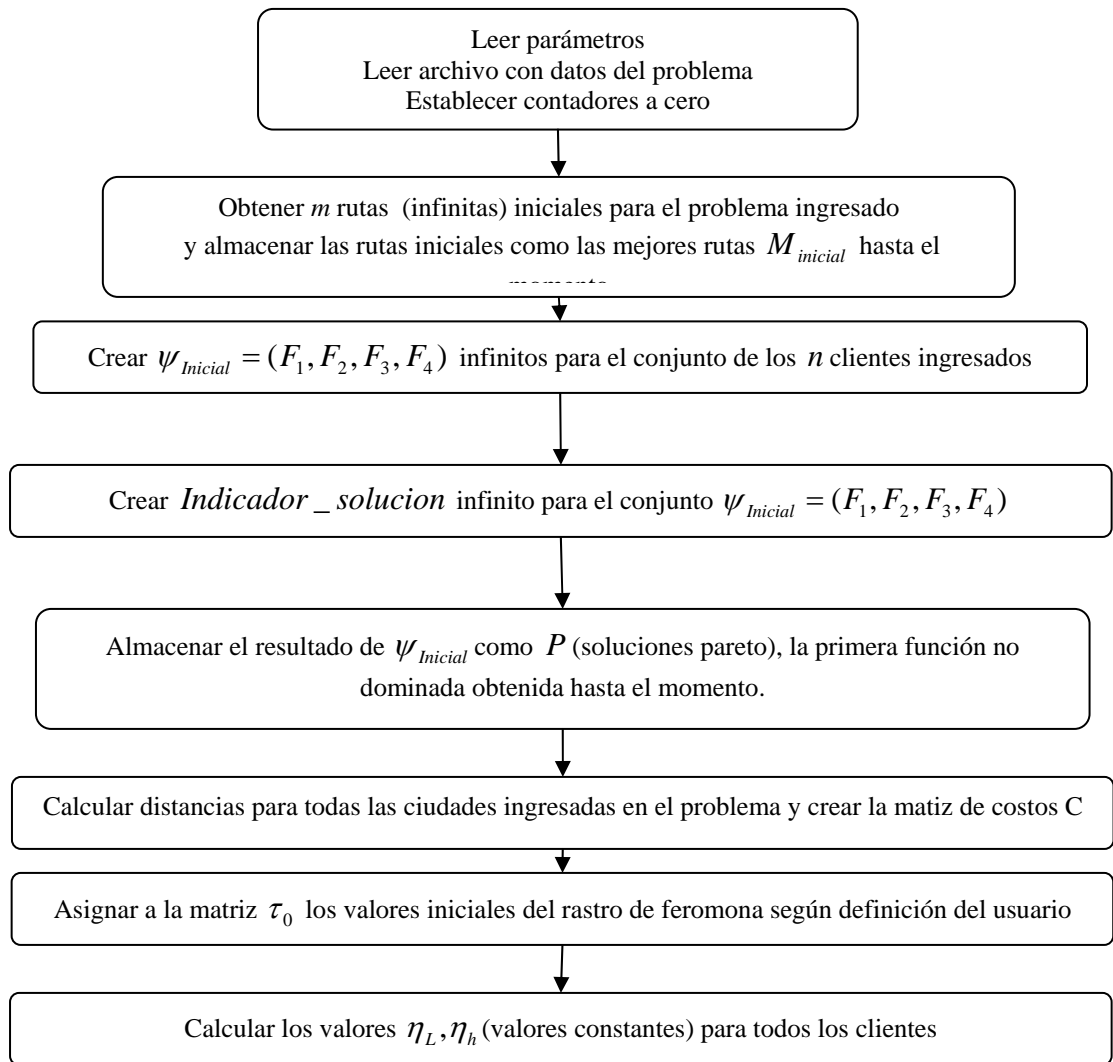


Figura 4. Esquema Bloque de iniciación

10.9.2. Bloque de actividades de creación de soluciones. En esta etapa se agregan los camiones y comienza su actividad de búsqueda de soluciones al problema.

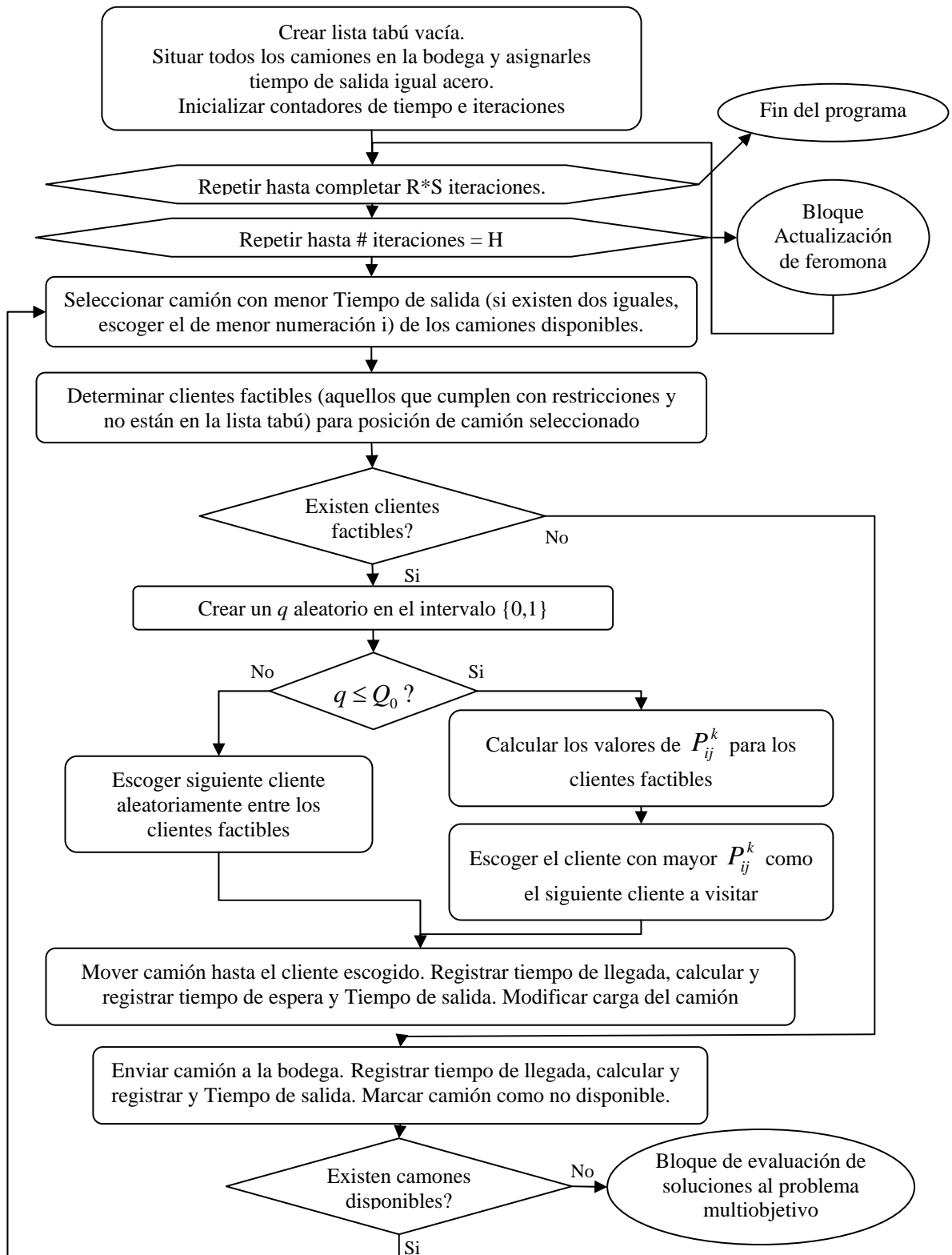


Figura 5. Esquema Bloque de actividades de creación de soluciones

10.9.3. Bloque de actualización de la feromona. En este bloque se registran los cambios que se producen a la matriz de feromona cuando H iteraciones han transcurrido:

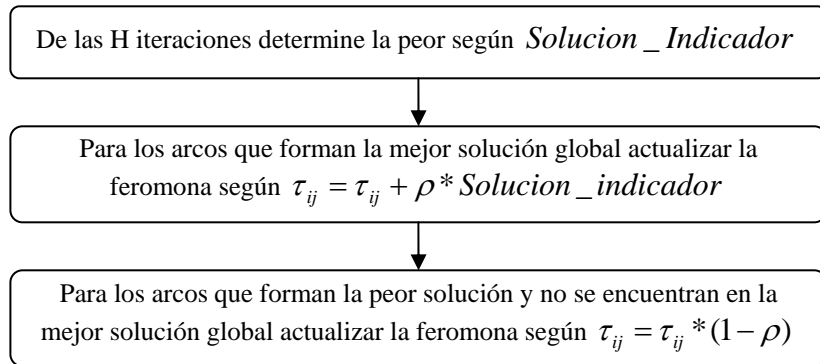


Figura 6. Esquema bloque de actualización de feromona (Actualización)

Si una solución mejor Global es encontrada:

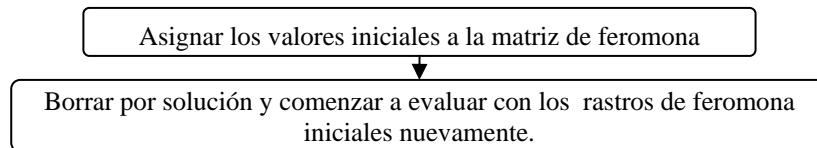


Figura 7. Esquema bloque de actualización de feromona (Reiniciación)

10.9.4. Bloque de Mutación. En este bloque se realizan unos cambios menores a la matriz de feromona

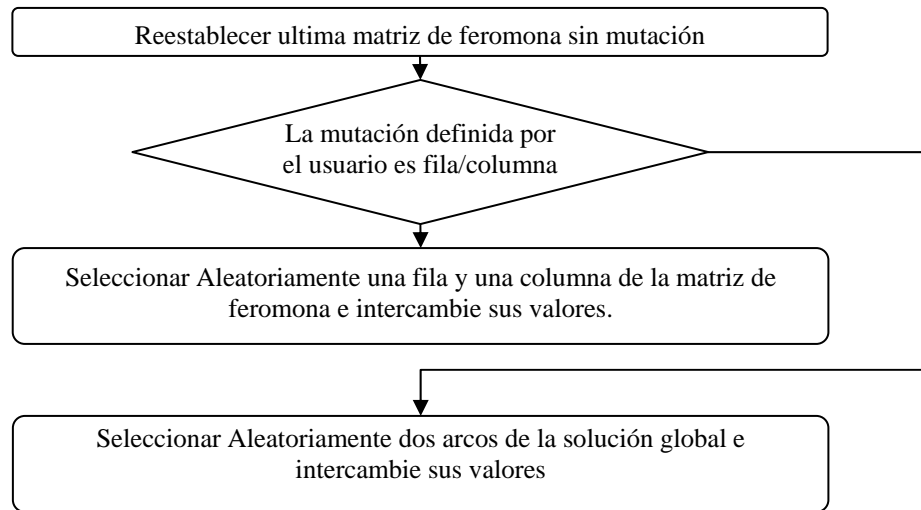


Figura 8. Esquema bloque de mutación

10.9.5. Bloque de Optimización local. En este bloque se intenta mejorar la solución encontrada mediante una optimización 2-opt.

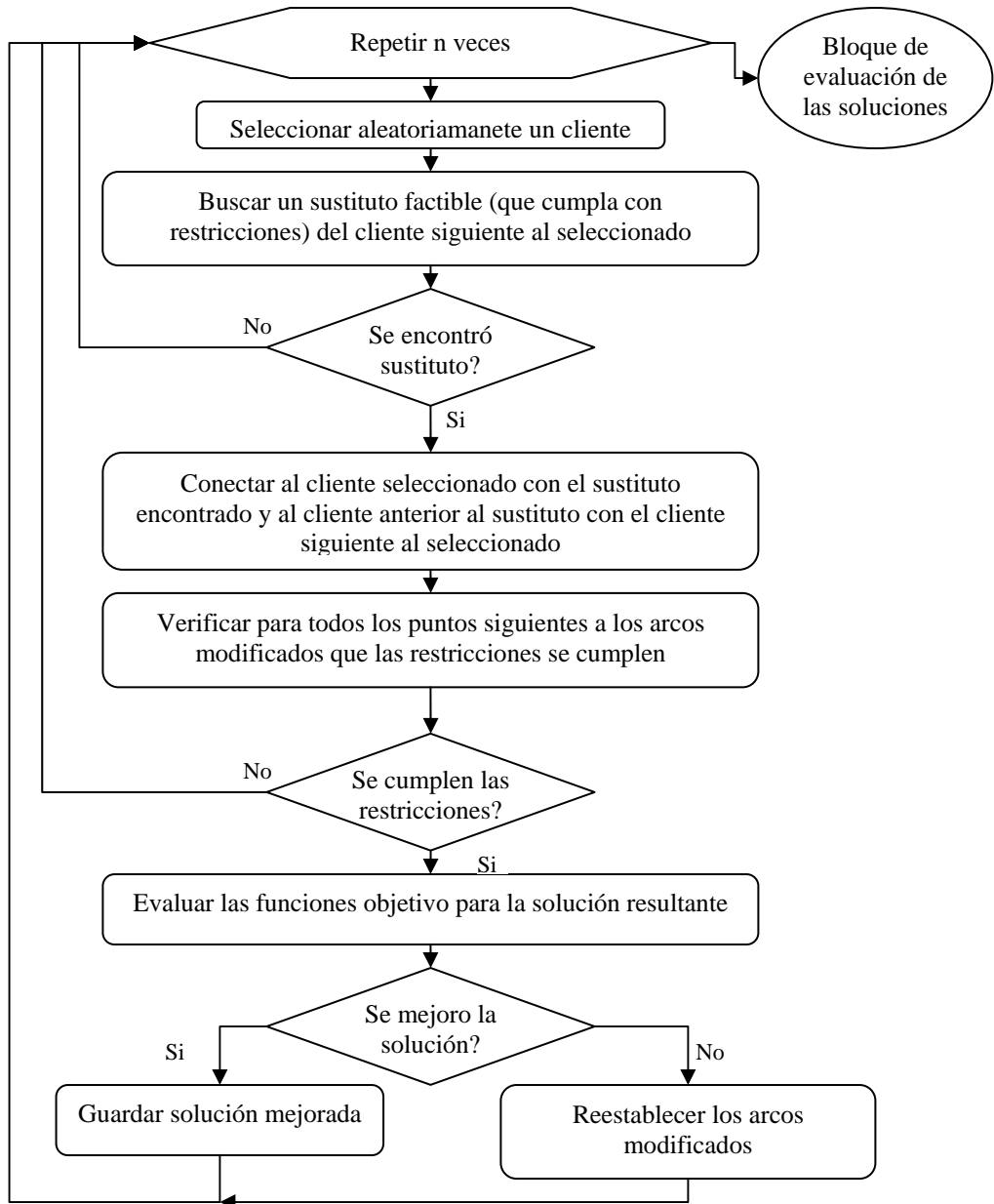


Figura 9. Esquema bloque de optimización local

10.9.6. Bloque de evaluación de las soluciones multiobjetivo. En este bloque se realizan el manejo correspondiente a la evaluación y registro de las soluciones al problema multiobjetivo.

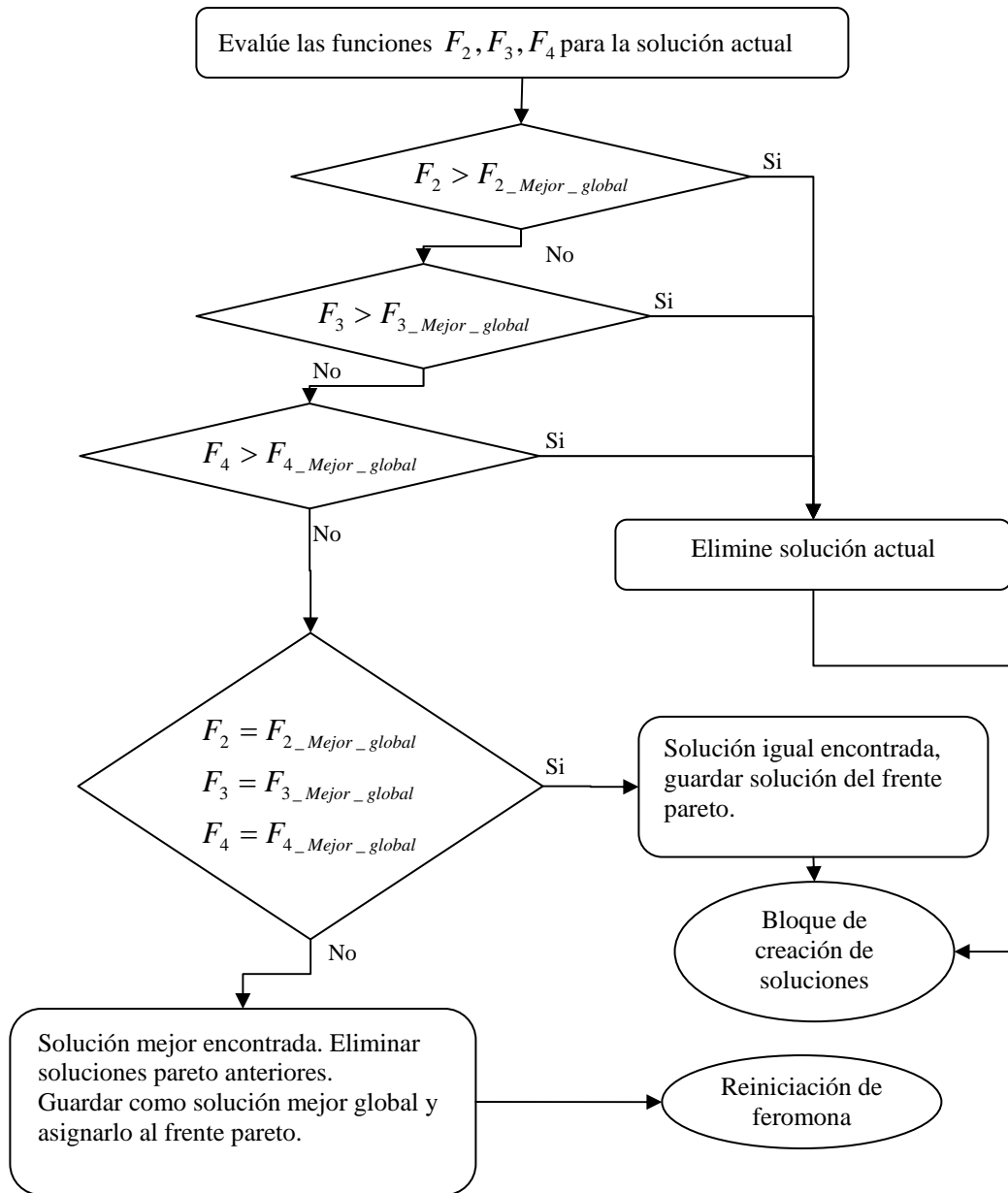


Figura 10. Esquema de evaluación de soluciones

10.10. PSEUDOCÓDIGO DEL ALGORITMO PROPUESTO

Algunos autores utilizan pseudocódigos para facilitar el entendimiento del código que desarrollan haciéndolos independientes del lenguaje utilizado para su programación. A continuación se presenta el pseudocódigo del algoritmo propuesto:

*/*Iniciación*/*

Ingresar datos del problema y Parámetros del algoritmo

Para cada ciudad $v \in \{1, \dots, n\}$ **hacer**

 Calcular distancias a todos los demás clientes según

$$\text{formula } C_{ij} = \sqrt{(i_x - j_x)^2 + (i_y - j_y)^2}$$

Fin para

Calcular valores de la matriz de feromona inicial según $\tau_{ij}(0) = R_{\min} + \left[\frac{R_{\max} - R_{\min}}{c_{\max} - c_{\min}} \right] * (c_{\max} - c_{ij})$

o utilizar valor constante.

Calcular η_L, η_h para cada cliente

Crear Ψ^x Inicial infinito como primera solución con (F_1, F_2, F_3, F_4) infinitos

Crear *Indicador_solucion* inicial infinito

*/*Ciclo principal de creación de soluciones*/*

Repetir*

Repetir

 Asignar los camiones a la bodega

 Lista_tabu → Vacía

 Capacidad de los camiones → Máxima

 Tiempo → cero

Mientras *Camiones_Disponibles* $\neq \{ \}$ **Repetir**

 Seleccionar camión libre k según menor *Tiempo_salida* */* Si hay igualdad seleccionar el de menor numeración i */*

 Calcule $N_k(i)$

Si $N_k(i) = \{ \}$ **hacer** */*no existen nodos factibles*/*

Camion $k \notin \text{Camiones_Disponibles}$

Si no

Seleccionar aleatoriamente $q \rightarrow \{0,1\}$

Si $q \geq Q_0$ **hacer**

 Seleccionar cliente siguiente aleatoriamente entre $N_k(i)$

Si no

Para cada $j \in N_k(i)$ **hacer**

 Calcule η_o

 Compute P_{ij}^k

Fin Para

 Seleccionar cliente siguiente al de mayor P_{ij}^k entre $N_k(i)$

Fin si

$Tiempo_salida_k \leftarrow Tiempo_salida_k + Max(c_{ij} + \delta; e_j + \delta)$

$Capacidad_camion_k \leftarrow Capacidad_camion_k - d_j$

$j \cup lista_tabu$

Fin si

Fin mientras

Si todas las $d_i = 0$ **hacer** /* una solución factible ha sido encontrada*/

Repetir

 Realizar Búsqueda local

Si $S_{optimizada}$ encontrada

$S_{actual} \leftarrow S_{optimizada}$

Si no

 Restablecer S_{actual}

Hasta completar n veces

 Calcular F_1, F_2, F_3, F_4 (actuales)

 Calcular *Indicador_Solucion*

Si $F_{1(actual)} < F_{1(global)}, F_{1(actual)} < F_{1(global)},$
 $F_{1(actual)} < F_{1(global)}, F_{1(actual)} < F_{1(global)}$ **hacer** /*Una solución mejor a la

global ha sido encontrada*/

Asignar $\Psi^x_{Actual} \rightarrow \Psi^x_{Mejor_global}$

$\Psi^x_{Mejor_global} \in P$

Borrar soluciones anteriores de P

$\tau_{ij} = \tau_0$ /*Reiniciación de feromona*/

Si no

Si $F_{1(actual)} = F_{1(global)}, F_{1(actual)} = F_{1(global)},$ **hacer** /*Solución igual*/
 $F_{1(actual)} = F_{1(global)}, F_{1(actual)} = F_{1(global)}$

$\Psi^x_{Actual} \cup P$

Si no /*La solución es peor*/

Si $Indicador_Solucion > Peor_Solucion$ **hacer**

$S_{actual} \rightarrow S_{Peor_atual}$

Si no

Eliminar solución

Fin si

Fin si

Si no

Generar dos valores aleatorios a, b

$$\tau'_{ij} \begin{cases} \tau_{ia} = \tau_{bi} \rightarrow (i = 0,1,2,\dots,n-1) \\ \tau_{bj} = \tau_{jb} \rightarrow (j = 0,1,2,\dots,n-1) \\ \tau_{ij}, \text{ para los demas valores} \end{cases} \quad /*realiza la mutación*/$$

Fin si

Hasta completar H veces

$\tau_{ij} = \tau_{ij} + \rho * Solucion_indicador \quad \forall a_{ij} \in S_{mejor-global}$

$\tau_{ij} = \tau_{ij} * (1 - \rho) \quad \forall a_{ij} \in S_{peor_actual} \text{ y } a_{ij} \notin S_{mejor-global}$

Hasta* completar Ciclo (S iteraciones) /*Fin ciclo Principal*/

10.11. DESARROLLO DEL ALGORITMO COMPUTACIONAL

La aplicación computacional creada en este proyecto fue desarrollada siguiendo el modelo matemático y algorítmico diseñado. Para facilitar su uso, se diseñó una interfase de fácil entendimiento pensando en el usuario como un estudioso del algoritmo más que como un usuario de un paquete de ruteo.

10.11.1. Lenguaje Utilizado. El programa fue desarrollado en java, utilizando para su creación el programa jcreator. Java fue seleccionado como el lenguaje a utilizar básicamente por ser una plataforma independiente, de distribución y uso gratuito, el cual provee un ambiente de trabajo propicio para lograr una codificación fácil de acceder, eficiente y fácil de entender.

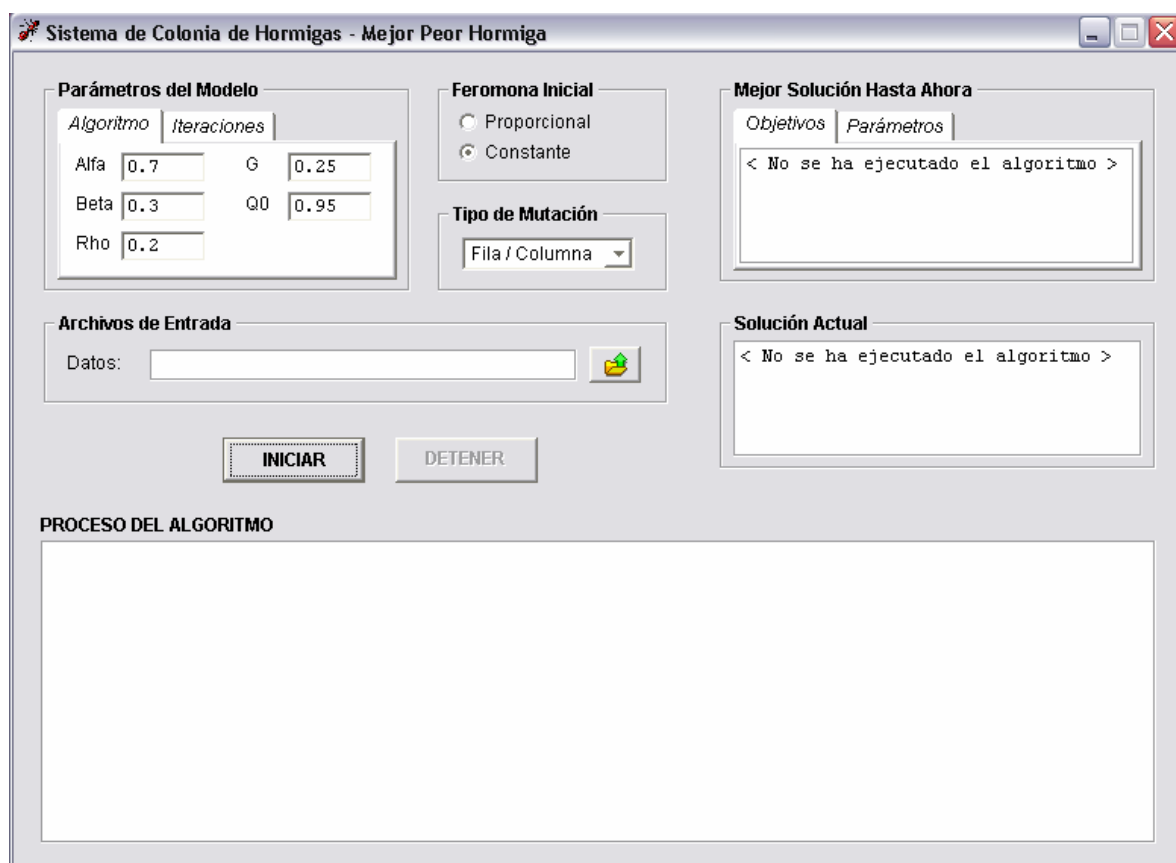


Figura 11. Interfase del algoritmo computacional creado

Java presenta además la característica de ser multiplataforma, brindando al usuario la posibilidad de utilizar el algoritmo desarrollado en plataformas computacionales variadas.

El código contiene comentarios y guías para hacer más fácil su entendimiento y seguimiento, dentro del paquete del programa adjunto a este documento se encuentra un archivo Leame.txt con información más detallada sobre los archivos que contiene el paquete y aspectos técnicos que pueden ser útiles para quien decida estudiar o modificar el código.

10.11.2. Instancias. El algoritmo “lee” los problemas (instancias) de un archivo texto el cual debe tener unas características particulares para que sea reconocido por el programa. El esquema utilizado es el propuesto hace varios años por el profesor Solomon¹²⁵, uno de los más difundidos y claros que existen. El esquema se detalla a continuación:

R111 — Nombre del archivo (requerido)

VEHICLE — Palabra requerida en Ingles o Español

NUMBER 25 CAPACITY 200

CUSTOMER — Palabra requerida en Ingles o Español

CUST NO.	XCOORD.	YCOORD.	DEMAND	READY TIME	DUE DATE	SERVICE TIME
0	35	35	0	0	230	0
1	41	49	10	15	204	10
2	35	17	7	18	202	10
3	55	45	13	54	187	10
4	55	20	19	138	169	10
5	15	30	26	20	199	10
6	25	30	3	76	131	10
7	20	50	5	21	170	10
8	10	43	9	87	112	10
9	55	60	16	88	115	10
10	30	60	16	107	150	10
11	20	65	12	57	86	10
⋮	⋮	⋮	⋮	⋮	⋮	⋮
v_n	x_n	y_n	q_n	e_n	l_n	δ_n

Figura 12. Esquema de instancia requerido

¹²⁵ SOLOMON, M.M. Algorithms for Vehicle Routing and Scheduling Problems with time window constraints. p. 326 Northeastern: Universidad de Boston, 1985.

Los datos deben ser ingresados en el orden en que se encuentran aquí ya que el algoritmo no tendrá en cuenta si el encabezado de las columnas se cambia.

10.12. ANÁLISIS ESTADÍSTICO DE LOS PARÁMETROS

Una vez el algoritmo esta terminado y funcionando, se debe realizar un estudio sobre los valores de las componentes definidas por el usuario (parámetros) a fin de determinar el valor para el cual el conjunto de ellas produce las soluciones de mas alta calidad.

Para obtener conclusiones fiables, se ha aplicado el test estadístico de Análisis de Varianza (ANOVA). Al igual que la regresión, este test se usa para estudiar la relación que existe entre una variable dependiente (variable respuesta) y una o más variables independientes (llamadas factores).

El método ANOVA nos permite comprobar si la diferencia de la respuesta depende de la variación de uno de los factores o si, por el contrario, ha sido fruto del azar.

Antes de poder aplicar este método, debemos comprobar que las observaciones obtenidas en el experimento cumplen las siguientes condiciones:

- Las observaciones son mutuamente independientes.
- Las observaciones se distribuyen siguiendo una distribución normal.
- Todas tienen la misma varianza.
- Las medias se pueden expresar como combinación lineal.

En otras experimentaciones se ha determinado que estas cuatro condiciones se satisfacen.

Con este estudio, se intenta averiguar si los cambios en algunos de los valores de los parámetros de las componentes del algoritmo influyen en la bondad de las soluciones alcanzadas. También se intentara establecer el valor óptimo de dichos parámetros, entendiendo por óptimos aquellos valores con los que el algoritmo consigue las mejores soluciones.

El algoritmo propuesto utiliza principalmente siete parámetros básicos: el tipo de mutación, El tipo de matriz inicial, el peso de la feromona(alfa), el peso del valor heurístico(beta), El coeficiente de evaporización(Rho), el porcentaje de vecinos factibles(G) y el valor de la probabilidad de transición(Q_0). Para poder estudiar adecuadamente la importancia de cada uno de estos parámetros, es necesario trabajar con distintos valores de los mismos y aplicarlos a distintas instancias del problema.

Las características de la prueba son:

- Se trabaja con una instancia representativa del problema para la cual es conocida la respuesta optima:
La instancia a utilizar es la C101 del profesor Salomón de 100 clientes.
- Existe un grupo de parámetros comunes a todas las pruebas a realizar cuyos valores fueron asignados según valores típicos encontrados en la documentación consultada. Estos valores se detallan en la siguiente tabla:

Tabla 1. Parámetros constantes utilizados en la ANOVA

Nombre	Definición	Valor
R	# Ciclos Independientes	10
S	# Iteraciones en cada ciclo	200
H	# iteraciones requeridas para actualizar feromona	50
T	Tiempo máximo por ciclo	ilimitado

- Para cada set de parámetros el algoritmo se corre dos veces, el resultado de ese set de parámetros es el promedio de las respuestas en las dos ejecuciones.
- Los valores con los cuales se experimento para los diferentes parámetros son:

Tabla 2. Parámetros variables utilizados en la ANOVA

Nombre	Definición	Valor
Tipo de mutación	Mutación a aplicar en cada iteración	Fila/Columna, Dos valores
Tipo de Matriz inicial	Matriz a ser utilizada como τ_0	Proporcional escalonada, Valor constante(uniforme)
Alfa	Peso de la feromona	0.5, 1, 2

Beta	Peso del valor heurístico	0.5, 1, 2
Rho	Coefficiente de evaporación de la feromona	0.2, 0.5, 0.9
Q ₀	Valor de la probabilidad de transición	0.9, 0.99
G	porcentaje de vecinos factibles	25, 50

Para realizar la prueba, es necesario realizar un número elevado de corridas del programa con las diferentes combinaciones de parámetros. Para esta prueba de siete factores de los cuales cuatro tienen dos niveles y tres tienen tres niveles se requieren 432 corridas. Para facilitar este trabajo se realizó un programa adicional que varía los datos automáticamente después de cada corrida, este programa se incluye en el paquete adjunto de archivos con el nombre *experimento.java*.

Se realizaron dos corridas para cada combinación generando 864 corridas. Los datos recogidos consistían en la combinación de los siete parámetros y los valores de las funciones objetivo dos, tres y cuatro. Los datos obtenidos fueron ingresados en el programa SPSS:

La hipótesis nula de esta prueba presupone que todas las medias son iguales y apreciables respecto a la bondad de la respuesta obtenida.

La hipótesis alternativa indica que los parámetros no son iguales y que su importancia es estadísticamente apreciable para un nivel de certeza del 95%

Tabla 3. Resultados de la prueba ANOVA

Tests of Between-Subjects Effects

Dependent Variable: F2

Source	Type II Sum of Squares	df	Mean Square	F	Sig.	Noncent. Parameter	Observed Power ^a
Corrected Model	1.419+202 ^b	10	1.4190+201	251.892	.000	2518.915	1.000
Intercept	3.920+202	1	3.9204+202	6959.059	.000	6959.059	1.000
ALFA	2.924+200	2	1.4618+200	25.948	.000	51.896	1.000
BETA	3.674+200	2	1.8368+200	32.605	.000	65.210	1.000
RHO	6.944+197	2	3.4722+197	.062	.940	.123	.059
G	9.074+199	1	9.0741+199	16.107	.000	16.107	.980
Q0	5.912+201	1	5.9116+201	1049.353	.000	1049.353	1.000
FEROMONA	2.204+201	1	2.2042+201	391.258	.000	391.258	1.000
MUTACION	4.167+198	1	4.1667+198	.740	.390	.740	.138
Error	4.805+201	853	5.6335+198				
Total	5.820+202	864					
Corrected Total	1.900+202	863					

a. Computed using alpha = .05

b. R Squared = .747 (Adjusted R Squared = .744)

De acuerdo con los resultados, se observa que los valores mas significativos son alfa, beta, Qo y la feromona. Los valores menos significativos son Rho y el tipo de mutación.

La prueba también muestra que la feromona con un valor constante y Qo con un valor cercano a 1, generan mejores resultados. Los valores para los cuales se encontraron mejores respuestas son

Tabla 4. Mejores valores para los parámetros según prueba ANOVA

Alfa	1,2
Beta	0.5,1
Rho	0.9,0.5
G	.25
Qo	.99
Feromona	Constante
Mutación	2 valores

Una vez conocidos los parámetros significativos en la calidad de las soluciones y las tendencias de estos, se puede optar por realizar pruebas únicamente con estos valores en las fronteras donde mejores resultados se han obtenido.

Un estudio mas profundo sobre los parámetros de un algoritmo requerirán de un analizar de regresión para hallar las componentes lineales y cuadráticas de cada factor. Una vez hallados estas componentes se debe plantear una ecuación no lineal que arroje los valores óptimos para cada parámetro.

10.13. DESEMPEÑO DEL ALGORITMO

Para realizar un balance sobre el desempeño del algoritmo propuesto utilizaremos los datos obtenidos en la prueba anterior

Tabla 5. Mejores resultados obtenidos para el problema C101

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
F2	864	849.64	1.00+100	6.74E+99	4.6916E+99
F3	864	9859.34	1.00+100	6.74E+99	4.6916E+99
F4	864	1162.03	1.00+100	6.74E+99	4.6916E+99
Valid N (listwise)	864				

Los resultados son comparados con el valor óptimo¹²⁶ y heurístico mejor conocido para la instancia C101:

Tabla 6. Comparación de resultados obtenidos con el algoritmo propuesto, el mejor heurística y el optimo del problema

Algoritmo Propuesto		Optimo		Mejor Heurístico	
Camiones	Distancia total recorrida	Camiones	Distancia total recorrida	camiones	Distancia total recorrida
10	849.64	10	827.3	10	828.94

¹²⁶ Grupo de investigación de los problemas de ruteo.(online). http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem_Instances/solomon100.html (Consultado: 8 de junio de 2005).

A diferencia del algoritmo propuesto los algoritmos utilizados para calcular el óptimo y el mejor heurístico aquí presentados, no optimizan el tiempo requerido para la entrega (F3), ni el tiempo de la ruta mas larga(F4). Sin embargo, el algoritmo propuesto obtiene un valor muy cercano al óptimo (97.37%), mientras optimiza las otras dos funciones.

Durante la realización de pruebas desactivando las funciones objetivo tres y cuatro el algoritmo propuesto encontró una solución igual a la del mejor heurístico conocido. Los parámetros utilizados fueron: Alfa:0.4, beta:0.6, Rho:0.2, G:0.25, Qo:0.98, feromona constante y Mutación de fila columna.

Un estudio profundo del algoritmo requerirá de probar varias instancias, de diferentes tamaños, y autores.

Los resultados experimentales demuestran la convergencia del algoritmo hacia una solución cercana al óptimo, dando un soporte a la metodología utilizada durante el proyecto para la creación del algoritmo.

BIBLIOGRAFÍA

ALMIRÓN, Marta. VIVEROS, Enrique. BARÁN, Benjamín, Sistema distribuido de hormigas para el problema del cajero viajante. p. 12 Asuncion: Universidad Nacional de Asunción. 2004

ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco. Análisis de distintas vertientes para la paralelización de los algoritmos de Optimización basada en Colonias de Hormigas. Actas del Segundo Congreso Español de Metaheurísticas. 2001

ALONSO, Sergio. CORDÓN, Oscar. DE VIANA, Iñaki. HERRERA, Francisco La Metaheurística de Optimización Basada en Colonias de Hormigas: Modelos y Nuevos Enfoques. p. 50. Granada: Departamento de Ciencias de la Computación e Inteligencia Artificial. 2002.

ANGULO, Cesar, La Logistica (Online), 2005; p. 2, www.monografias.com (consultado: 11 de abril de 2005)

ARBONES, Eduardo. BALLOW, Ronald. BITLE, L y RAMSEY, J. Logística Integral (1991), Logística Empresarial (1994). Enciclopedia del management. En PÉREZ, Marisol. De la Gestión de la Producción a la Gestión de la Cadena de Suministro (Online) Holguín: Editorial Uho, 2003. www.monografias.com (consultado: 6 de abril de 2005).

ATOCHA Aliseda Heurística, hipótesis y demostración en matemáticas (online) Mexico D.F.:Instituto de Investigaciones Filosóficas. <http://minerva.filosoficas.unam.mx/> (consultado: 12 de mayo de 2005)

AUTOR DESCONOCIDO. Algoritmos Bio-Inspirados en Logística, En Revista del Instituto Tecnológico de Informática. Marzo 2002

AUTOR DESCONOCIDO. Gestión de Operaciones. Logística del transporte(Online); www.monografias.com (consultado: 7 de Mayo de 2005).

BACILIERI, Nino. GARCIA, William. Algoritmos Heuristicos. En: III Congreso Colombiano y I Conferencia Andina Internacional De Investigación De Operaciones, Cartagena. 2004

BAECK Thomas, Evolutionary Computation

BARÁN, Benjamín. ALMIRÓN, Marta. Colonia de Hormigas en un Ambiente Paralelo Asíncrono. p. 11. San Lorenzo: Centro Nacional de Computación. 2003

BARÁN B. y HERMOSILLA, A. Comparación de un Sistema de Colonias de Hormigas y una Estrategia Evolutiva para el Problema del Ruteo de Vehículos con Ventanas de Tiempo en un Contexto Multiobjetivo. IX Jornadas Iberoamericanas de Informática, Cartagena de Indias, Colombia. 2003

BARÁN, B. y SCHAERER, M. A multiobjective Ant Colony System for Vehicle Routing Problem with Time Windows. Proceedings of the 21st IASTED International Conference APPLIED INFORMATICS. Innsbruck, Austria. 2003.

BARCOS, Lucía. RODRÍGUEZ, Victoria . ÁLVAREZ, Jesús. ROBUSTÉ, Francesc . Routing Optimization for many-to-many freight, Transportation problems using Ant Colonies Techniques (Online). España: Universidad Carlos III En Second world conference on POM, Cancún, Mexico 2004. <http://www.iti.upv.es/cas/> (Consultado 5 de mayo de 2005)

BARROS,Oscar. VARGAS, Samuel. WEBER, Richard. Evaluación de Prácticas de Gestión en la Cadena de Valor de Empresas Chilenas. (Online) En Revista Serie Gestion N°44, p. 40 Enero, 2004; www.obarros.cl (consultado: 9 de Mayo 2005)

BONABEAU, E. DORIGO M. THERAULAZ, G.. Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York, NY, 1999.

BRAYSY, Olli. Efficient Local Search Algorithms for the Vehicle Routing Problem with Time Windows (On line). Finlandia:University of Vaasa. En Cuarta Conferencia Internacional de Metaheurísticas, Porto, Portugal, julio de 2001. (Documento encontrado directamente a través de www.google.com)

BRÄYSY, Olli. BERGER, Jean. BARKAOUI, Mohamed. A New Hybrid Evolutionary Algorithm For The Vehicle Routing Problem With Time Windows. p. 28 Vaasa: University of Vaasa.2002

CASAL Ricardo, TONCOVICH Adrián, LÓPEZ Nancy, CORRAL Rafael, Estudio comparativo de estrategias heurísticas para un problema de secuenciamiento multiobjetivo. P. 43 Bahía Blanca:Universidad Nacional del Sur. Argentina 1998

CHAMOSO, Raul. Sist. de Control y Distribución de Ventas (Online), 2004; p 3. www.monografias.com (consultado: 10 de abril de 2005)

COLORNI A., Dorigo M., and Maniezzo V. An investigation of some properties of an Ant algorithm. Bruselas 1992.

COMAS, Jose. Ruteo de vheiculos. (online). Valencia:Universidad politecnica Diapositiva 4 www.mat.up.es/%7Eeqvm/opp.html (Consultado: 8 de mayo de 2005)

CONSUELO, Ana Díaz. Respuesta estratégica de los operadores logísticos europeos frente a los cambios en el entorno Lic. CC. Empresariales, Tool ULEE.

CORDÓN, O. HERRERA , F. MORENO,L..Integración de Conceptos de Computación Evolutiva en un Nuevo Modelo de Colonias de Hormigas. En VIII Conferencia de la Asociación Española para la Inteligencia Artificial, Murcia (España), 1999, Vol. II, páginas 98-105.

CÓRDOVA, Juan. Optimización a través de colonias de hormigas. San Carlos:Universidad de San Carlos, p 2. 2004.

CRICHIGNO, Jorge. TALAVERA , Francisco. PRIETO, Joel. BARÁN, Benjamín. Enrutamiento Multicast Utilizando Optimización Multiobjetivo p. 6. Asunción: Centro Nacional de Computación. 2004

DENEUBOURG, J. GOSS, S. Collective patterns and decision making; Ethology, Ecology, Evolution, Vol. 1, 1989, págs. 295-311.

DOMÍNGUEZ, José A. y otros. Dirección de Operaciones. Aspectos tácticos y operativos en la producción y los servicios, España: Editorial MC Graw Hill; 1999. 303p

DORIGO M. Optimization, Learning and Natural Algorithms. Ph.D.Thesis, Milan :Politecnico di Milano. 1994

DORIGO, Marco. DI CARO, Gianni. GAMBARDELLA, Luca. Ant Algorithms for Discrete Optimization. En Revista Artificial Life, Bruselas: MIT Press, 1999.

DORIGO M. and GAMBARDELLA L.M. Ant Colony System: A Cooperative Learning Approach to The Traveling Salesman Problem.

DORIGO, M., MANIEZZO, COLORNI A. An investigation of some properties of an Ant algorithm. proceedings of the parallel problem solving from nature conference. Bruselas - Bélgica, 1992.

DORIGO M., MANIEZZO V., and COLORNI A. The Ant System: Optimization by A Colony of Cooperation Agents.

ESPARCIA, Anna y SHARMAN, Ken. Algoritmos Bio-inspirados En Revista Actualidad Tic (On line), 2004 ; p. 2, <http://www.iti.upv.es/cas/> (consultado: 8 de mayo de 2005)

GAMBARDELLA, Luca. TAILLARD, Éric. AGAZZI, Giovanni. MACS-VRPTW: A Multiple Ant Colony System For Vehicle Routing Problems With Time Windows (On Line). Lugano, Suiza. <http://www.idsia.ch> (Consultado: Mayo 5 de 2005).

GARDEL, Pedro. GÓMEZ, Osvaldo. BARÁN, Benjamín . Estudio del Espacio de Soluciones del Problema del Cajero Viajante. P. 12. San lorenzo: Centro Nacional de Computación. 2001.

GAREY, M. JOHNSON ,D. Computers and Intractability A Guide to the Theory of NP-Completeness; New York :W. H. Freeman and Company, 1997

GIPR Grupo de investigación de los problemas de ruteo.(online).
http://neo.lcc.uma.es/radi-aeb/WebVRP/Problem_Instances/solomon100.html
(Consultado: 8 de junio de 2005).

GUASCH, LogiSim, Simulación de procesos logísticos y retos para la optimización de flotas en tiempo real, Cataluña:Universitat Politècnica de Catalunya.1999

LAKANTOS, Proofs and Refutations. The Logic of Mathematical Discovery. Cambridge:University Press. 1976.

LEVINE, John. DUCATELLE, Frederick. Ant Colony Optimisation and Local Search for Bin Packing and Cutting Stock Problems. Edinburgh: Centre for Intelligent Systems and their Applications. 2003.

MELANIE, Mitchell, An Introduction to Genetic Algorithms (Complex Adaptive Systems)

MERELO, J.J. Técnicas heurísticas de resolución de problemas: computación evolutiva y redes neuronales (online) España: Grupo GeNeura, 2005.
<http://geneura.ugr.es> (consultado: 7 de Mayo de 2005)

NESMACHNOW, Sergio. Una Versión Paralela del Algoritmo Evolutivo para Optimización Multiobjetivo NSGA-II y su Aplicación al Diseño de Redes de Comunicaciones Confiables. En Actas del Tercer Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados. Córdoba: España.2004.

OTERO, Juan. GUERRERO, Danilo. Clasificación Automática Mediante Colonia De Hormigas. La Habana: Universidad de la habana. P 12.

PACHECO, J. ARAGÓN, A. DELGADO, C. Diseño de algoritmos para el problema del transporte escolar. Universidad de Burgos. Esapaña. 2001

PAEZ, Bernardo. GONZALES, Pedro. Router: Ambiente Amigable Para Ruteo Y "Scheduling" De Vehiculos (online) En Congreso Encuentro de jóvenes Investigadores (Sexta edición: 2005:Rosario), p 2 , www.universia.com.ar/contenidos/investigacion/unl/TECNOLOGIA/ingenieria/63.htm (consultado: 8 de mayo de 2005)

PÉREZ, Marisol. De la Gestión de la Producción a la Gestión de la Cadena de Suministro (Online) Holguín: Editorial Uho, 2003. www.monografias.com (consultado: 6 de abril de 2005).

POLYA. Mathematics and Plausible Reasoning. Volume II Patterns of Plausible Inference. p 54 Princeton :University Press, 1968.

REAL ACADEMIA ESPAÑOLA. Diccionario de la Lengua Española; edición 21, Madrid: Editorial Espasa, 1992.

SAAVEDRA, Caridad. FOSADO, Osvaldo. FERNANDEZ, Pedro. Tratamiento Multicriterio de Problemas de Ruta solucionados a través de Técnicas de Simulación Monte Carlo (Online) Cuba: Universidad de Pinar del Río. www.monografias.com (consultado: 8 de Mayo de 2005).

SERRA, Daniel. Investigación operativa y logística empresarial (Online), Fabra:Organización de Empresas GREL-IET. 2004, p.2. www.almacenar.com (Consultado: 8 de mayo de 2005)

SOLOMON, M.M. Algorithms for Vehicle Routing and Scheduling Problems with time window constraints. p. 326 Northeastern: Universidad de Boston, 1985.

TEJADA, Carlos. Ideas orientadoras sobre el sistema logístico del componente ejercito en el teatro de operaciones. En revista Military Review, p. 44. Buenos Aires:Escuela superior de guerra, Diciembre 2001

TORO, Andrés. GIRÓN, JM. BESADA, E., FERNÁNDEZ, P. LÓPEZ, JA. Aplicación del diseño de control multiobjetivo y multivariable, con algoritmos genéticos, al proceso de la fermentación de la cerveza, Departamento de Arquitectura de Computadores y Automática, Universidad Complutense de Madrid.

TORRES, Marcelo. Algoritmos heurísticos para resolución del vendedor viajero. (online) Santiago de Chile: USACH, 2004. <http://optimos2.diinf.usach.cl/VendedorViajero/Teoria.htm> (Consultado: 16 de Mayo de 2005)

SITIOS DE INTERNET

<http://duncan.cup.edu/~bro754/sci.htm>

<http://mat.gsia.cmu.edu/orclass/integer/node17.html>

<http://neo.lcc.uma.es/radi-aeb/WebVRP/>

<http://optimos2.diinf.usach.cl/swc/Inicio.htm>

<http://www.ctc.dcu.ie/ctcweb/courses/heuristics/course/vehicle.html>

<http://www.gestiopolis.com>

<http://www.inwap.com/pdp10/hbaker/hakmem/algorithms.html>

<http://www.logistpilot.com>

http://www.nwrel.org/sky/Classroom/Science*Problem_Solving/Problem_Solving.html

<http://www.spcomm.uiuc.edu/projects/vta/PSG/psgl3-ov.html>

ANEXOS

ANEXO A¹²⁷

Método Heurístico de Kart & Thompson¹²⁸. A continuación se presenta un enfoque aplicado al problema del agente viajero, se utiliza este problema clásico por ser parte fundamental de algoritmos de solución del problema de Ruteo de Vehículos. El problema consiste en encontrar la secuencia de puntos que el agente debe visitar iniciando y regresando al mismo punto, minimizando la distancia o tiempo total.

Proceso:

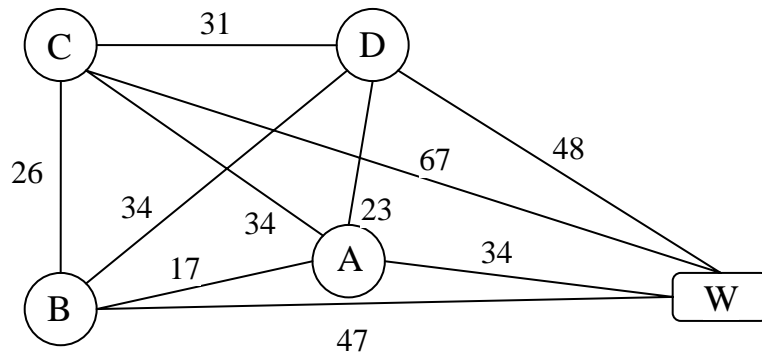
- I. Seleccione 2 clientes, i_1 y i_2 , cualquiera del grupo disponible.
- II. Suponga la existencia de un grupo de k clientes, que forman una ruta definida, con $2 < k < n$, y donde n representa el total de clientes.
- III. Escoja de manera aleatoria un cliente, h , diferente i_1 y i_2 , y calcule d_j para $j = 1$, tal que $d_j = a_{i_j, h} + a_{h, i_{j+1}} - a_{i_j, i_{j+1}}$ donde se define i_{k+1} para ser i_1 cuando $j = k$. Donde las variables a representan la distancia, tiempo o costo entre los pares de clientes, el valor j el orden de visitas desde el punto de origen y el valor k el numero de puntos visitados hasta el momento.
- IV. Defina j^* tal que sea el valor de j asociado al d_j mínimo.
- V. Redefina i_j como i_{j-1} para $j = j^*+1, \dots, k$, y el subíndice h como i_{j^*} . Esto es, incorpore el cliente h en la ruta en la posición de i_{j^*} , teniéndose ahora $(k+1)$ clientes en la misma.
- VI. Si $(k+1) = n$, el algoritmo se termina. De otra manera, establezca k como $k+1$ y continúe.

Ejemplo

Se pide solucionar el problema siguiente:

¹²⁷ Tomados de: COMAS, Jose. Ruteo de vheiculos. (online). Valencia:Universidad politecnica Diapositiva 36-45 www.mat.up.es/%7Eeqvm/opp.html (Consultado: 8 de mayo de 2005)

¹²⁸ Es importante entender la metodología aplicada hasta hace unos años en la solución del problema de ruteo.



Se escogen los puntos $i_1 = A$ y $i_2 = C$, $k = 2$. Se selecciona $B = h$ como un punto que no está en ruta de $k = 2$.

Se calcula d_j para $j = 1, 2$.

Para $j = 1$:

$$\begin{aligned} \mathbf{d}_1 &= a_{i_1, B} + a_{B, i_2} - a_{i_1, i_2} \\ &= a_{A, B} + a_{B, C} - a_{A, C} = 17 + 26 - 34 = 9. \end{aligned}$$

Para $j = 2$

$$\begin{aligned} \mathbf{d}_2 &= a_{i_2, B} + a_{B, i_1} - a_{i_2, i_1} \\ &= a_{C, B} + a_{B, A} - a_{C, A} = 26 + 17 - 34 = 9. \end{aligned}$$

Se selecciona \mathbf{d}_1 , por lo que $\mathbf{j}^* = 1$.

La nueva ruta es ahora: $\mathbf{i}_1 = A$, $\mathbf{i}_2 = B$, $\mathbf{i}_3 = C$.

Dado que $(k+1) = 3 < n = 5$, $k = C$, se continúa con el algoritmo.

Se escoge se escoge aleatoriamente un nuevo punto: W .

Se calcula \mathbf{d}_j para $j = 1, 2, 3$.

Para $j = 1$:

$$\begin{aligned} \mathbf{d}_1 &= a_{i_1, W} + a_{W, i_2} - a_{i_1, i_2} \\ &= a_{A, W} + a_{W, B} - a_{A, B} = 34 + 47 - 17 = 64. \end{aligned}$$

Para $j = 2$:

$$\begin{aligned} \mathbf{d}_2 &= a_{i_2, W} + a_{W, i_3} - a_{i_2, i_3} \\ &= a_{B, W} + a_{W, C} - a_{B, C} = 47 + 67 - 26 = 88. \end{aligned}$$

Para $j = 3 = k$:

$$\begin{aligned} \mathbf{d}_3 &= a_{i_3, W} + a_{W, i_1} - a_{i_3, i_1} \\ &= a_{C, W} + a_{W, A} - a_{C, A} = 67 + 34 - 34 = 67. \end{aligned}$$

El mínimo es \mathbf{d}_1 , por lo que $\mathbf{j}^* = 1$.

El nuevo recorrido es: $\mathbf{i}_1 = A$, $\mathbf{i}_2 = W$, $\mathbf{i}_3 = B$ y $\mathbf{i}_4 = C$.

Como $(k+1) = 4 < n = 5$, se continúa. $k = 4$.

Se escoge D como el nuevo punto h

Se calcula para \mathbf{d}_j para $j = 1, 2, 3, 4$.

$$\begin{aligned} \text{Para } j = 1 & \quad \mathbf{d}_1 = a_{A,D} + a_{D,W} - a_{A,D} = 23 + 48 - 34 = 37. \\ \text{Para } j = 2 & \quad \mathbf{d}_2 = a_{W,D} + a_{D,B} - a_{W,B} = 48 + 34 - 47 = 35. \\ \text{Para } j = 3 & \quad \mathbf{d}_3 = a_{B,D} + a_{D,C} - a_{B,C} = 34 + 31 - 26 = 39. \\ \text{Para } j = 4 = k & \quad \mathbf{d}_4 = a_{C,D} + a_{D,A} - a_{C,A} = 31 + 23 - 34 = 20. \end{aligned}$$

El mínimo \mathbf{d}_j es \mathbf{d}_4 por lo que $\mathbf{j}^* = 4$. Por lo tanto D se incorpora como \mathbf{i}_5 .

El recorrido final sera; $\mathbf{i}_1 = A$, $\mathbf{i}_2 = W$, $\mathbf{i}_3 = B$, $\mathbf{i}_4 = C$ y $\mathbf{i}_5 = D$.

Se puede apreciar que en este método heurístico la solución depende de los puntos iniciales escogidos aleatoriamente, por lo que no se puede asegurar que la ruta obtenida por el algoritmo sea la óptima.

Una buena estrategia es aplicar el algoritmo considerando varios puntos diferentes iniciales y escoger la ruta con la mínima distancia. Se debe reflexionar siempre que el tiempo de ejecución inicial será multiplicado por el número de pares que se evalué.

ANEXO B¹²⁹

Método Heurístico de Barrido (sweep) para los problemas de ruteo con múltiples vehículos. El objetivo es el de asignar m vehículos a n consumidores y definir la secuencia de visitas de cada uno.

Esta metodología contempla 2 fases:

- Asignación de clientes a vehículos
- La definición de la secuencia para visitar cada cliente de cada ruta.

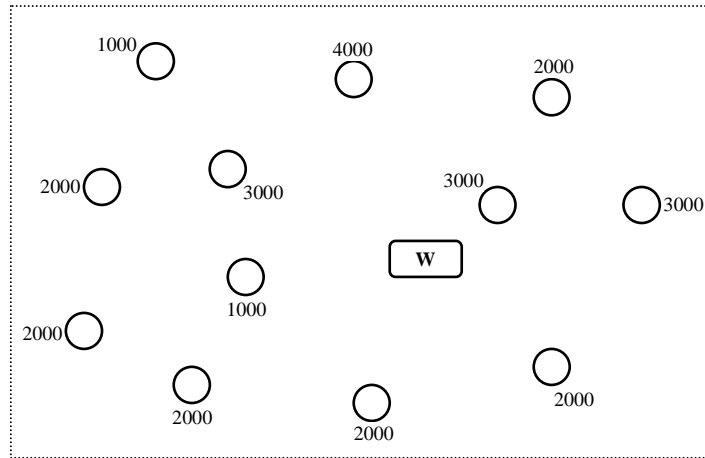
Proceso:

- I. Ubicar los clientes y fuente (origen) en mapa.
- II. Ubicar una línea recta desde la fuente hacia cualquier dirección.
- III. Rotar la línea en una dirección intersectando clientes.
- IV. Cada vez que un cliente se intersecta, se cuestiona la factibilidad de asignarlo a un vehículo (restricciones de capacidad, tiempo de entrega, etc.).
- V. Si es factible se incluye en ruta.
- VI. Se continúa hasta terminar con los clientes o los vehículos.
- VII. Determinar la secuencia de visita de cada cliente para cada ruta aplicando un método de solución del problema del Agente Viajero.

Ejemplo:

Se pide solucionar el siguiente problema con camiones de capacidad de 10.000 unidades.

¹²⁹ Tomados de: COMAS, Jose. Ruteo de vehículos. (online). Valencia: Universidad politécnica Diapositiva 29-35 www.mat.up.es/%7Eeqvm/opp.html (Consultado: 8 de mayo de 2005)

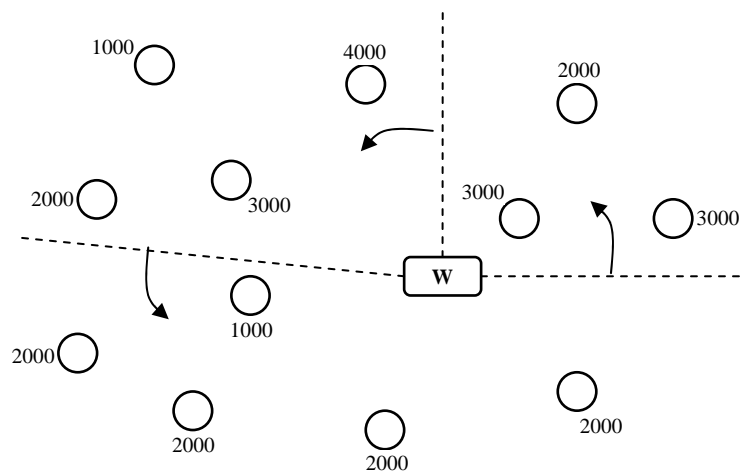


Los círculos representan los clientes, el numero junto a ellos la demanda de unidades y el punto W el almacén de mercancía.

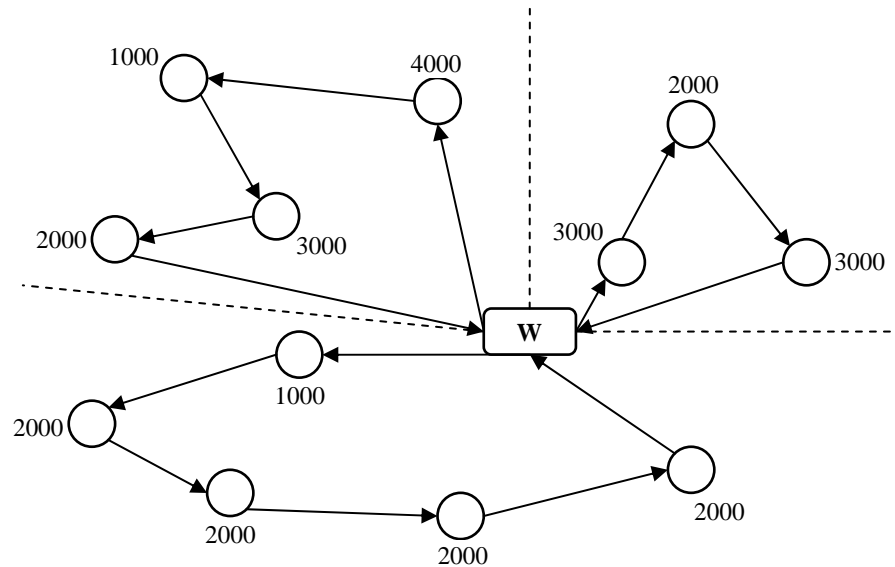
Se traza una línea horizontal hacia la derecha desde W, y se rota en sentido antihorario.

Cada punto que se encuentra se asigna a un camión hasta que este alcanza su capacidad.

Camión 1: $C_1 = 3000 + 2000 + 3000 = 8000$
 Camión 2: $C_2 = 4000 + 1000 + 3000 + 2000 = 10000$
 Camión 3: $C_3 = 1000 + 2000 + 2000 + 2000 + 2000 = 9000$



Se utiliza un método de solución del problema del agente viajero para cada camión tomando a W como punto de inicio.



En este problema se evidencia la importancia de la solución empleada en el problema del agente viajero. Utilizando la metodología de barrido, se hace exhaustivo el uso de algoritmos para el PAV (uno para cada camión), lo cual reduce las posibilidades de evaluar varios pares de inicio y buscar una solución mejor.

Esta metodología como muchas más presenta soluciones locales, indicándonos que incluso si se alcanzaran todos los óptimos locales no necesariamente se alcanzaría el óptimo global.