

Universidad Central “Marta Abreu” de Las Villas
Facultad de Matemática, Física y Computación



Trabajo para optar por Título de:
Licenciado en Ciencia de la Computación

Implementación de un algoritmo basado en Colonias de Hormigas para la optimización de funciones con datos mezclados

Autor

Sergio Ramón Pérez León

Tutores

Lic. Gonzalo Nápoles Ruiz

Dr. Jorge L. Moya Rodríguez

2013

Hago constar que el presente Trabajo para optar por Título de Licenciado en Ciencia de la Computación ha sido realizado en la facultad de Matemática, Física y Computación de la Universidad Central “Marta Abreu” de Las Villas (UCLV) como parte de la culminación de los estudios de Licenciatura en Ciencia de la Computación, autorizando a que el mismo sea utilizado por la institución para los fines que estime conveniente, tanto de forma total como parcial y que además no podrá ser presentado en eventos ni publicado sin la previa autorización de la UCLV.

Firma del Autor

Los abajo firmantes, certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y que el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del Tutor

Lic. Gonzalo Nápoles Ruiz

Firma del Tutor

Dr. Jorge L. Moya Rodríguez

Dedicatoria

A toda mi familia.

Agradezco a todas las personas que han contribuido al desarrollo de este trabajo, en especial:

- A mi mamá, mi papá y hermano por estar conmigo en todo.
- Al resto de mi familia por su ánimo y confianza.
- A mis tutores: Lic. Gonzalo Nápoles Ruiz por sus invaluables consejos e indicaciones y Dr. Jorge L. Moya Rodríguez por su apoyo y recibimiento.
- A la Dra. María M. García Lorenzo por sus recomendaciones.
- A los amigos de la carrera por soportarme; especialmente a Magdiel, Daril, Leandro, Fernando.
- Al Dr. Ramiro y su esposa Beatriz por su gran favor a inicio de carrera.
- A los profesores que se empeñaron en hacer de mí un buen profesional.

*“Las computadoras no resuelven los problemas,
pero capacitan a las personas para hacerlo ”*

Steve Ballmer

RESUMEN

Desde su introducción la meta-heurística ACO ha sido utilizada para solucionar complejos problemas combinatorios. Debido a su éxito en este tipo de problemas, los investigadores han trabajado en su modificación y generalización. Estas modificaciones incluyen la mejora de las soluciones obtenidas o la reducción del tiempo de cálculo necesario. Sin embargo, esta meta-heurística es mayormente utilizada para la resolución de problemas discretos. Por otra parte, los algoritmos poblacionales para solucionar problemas mixtos son escasos y no ofrecen soluciones muy precisas (por ejemplo, Algoritmos Genéticos). Precisamente el objetivo de este Trabajo de Diploma es desarrollar un método inspirado en la meta-heurística Colonias de Hormigas para la optimización de funciones con datos mezclados (variables discretas y continuas) llamado MACO (Mixed Ant Colony Optimization, por sus siglas en inglés). Además, un nuevo procedimiento de búsqueda local fue diseñado para este algoritmo para garantizar una mejor convergencia. Con el objetivo de validar esta propuesta, son diseñadas 40 funciones mixtas de pruebas que permiten estudiar el comportamiento de MACO respecto a otros enfoques como Algoritmos Genéticos y Recocido Simulado. Además, el nuevo procedimiento se valida mediante la resolución de un problema de optimización real en el campo de la Ingeniería Mecánica, particularmente en el área de la optimización de los intercambiadores de calor de tubo y coraza.

ABSTRACT

Since its inception the Ant Colony Optimization meta-heuristic has been widely used for solving complex combinatorial problems. Due to its success to solve this kind of problems, researches have been working in its modification and generalization. These variations include improvements of the found solutions or the reduction of the execution time. However, this meta-heuristic is mostly used for solving discrete problems. On the other hand, the existing population-based algorithms are less studied, and they do not offer precise solutions (e.g. Genetic Algorithms). Precisely, the goal of the present work is to develop a new method inspired on the Ant Colony Optimization meta-heuristic called Mixed Ant Colony Optimization (MACO), for optimizing functions having mixed data. Besides, a new local search procedure was specially designed for the proposed procedure, allowing better convergence rate. In order to validate the proposal, 40 test functions (benchmark) are designed for studying the behavior of MACO with respect to other approaches such as Genetic Algorithms and Simulated Annealing. To conclude, MACO is also validated through a real-case optimization problem concerning to Mechanical Engineering, particularly in the field of the design of shell-and-tube heat exchangers.

TABLA DE CONTENIDOS

| | |
|--|----|
| INTRODUCCIÓN | 1 |
| CAPÍTULO 1. OPTIMIZACIÓN GLOBAL DE FUNCIONES USANDO METAHEURÍSTICAS: OPTIMIZACIÓN DE COLONIA DE HORMIGAS. | 6 |
| <i>1.1. El problema de optimización</i> | 6 |
| 1.1.1. Clasificaciones..... | 6 |
| 1.1.3. Optimización global | 7 |
| <i>1.2. Métodos de optimización</i> | 8 |
| 1.2.1. Optimización lineal. | 8 |
| 1.2.2. Optimización no lineal. | 8 |
| • <i>Los métodos diferenciales</i> | 9 |
| • <i>Los métodos no diferenciales</i> | 9 |
| <i>1.3. Meta-heurísticas</i> | 10 |
| 1.3.1. Técnicas inspiradas en la evolución | 12 |
| 1.3.2. Técnicas inspiradas en los procesos físicos..... | 13 |
| 1.3.3. Técnicas inspiradas en el aprendizaje | 14 |
| <i>1.4 Algoritmo de colonia de hormigas (ACO)</i> | 14 |
| 1.4.1 Las colonias de hormigas naturales..... | 14 |
| 1.4.2 De las hormigas naturales a la meta-heurística de ACO | 16 |
| 1.4.3 Tipos de problemas resolubles por ACO | 17 |
| 1.4.4 La hormiga artificial..... | 18 |
| 1.4.5 Similitudes y diferencias entre las hormigas naturales y artificiales | 19 |
| 1.4.6 Modo de funcionamiento y estructura de un algoritmo ACO | 20 |
| <i>1.5 Modelos de Optimización Basada en Colonias de Hormigas</i> | 23 |
| 1.5.1 El Sistema de Hormigas | 23 |
| 1.5.2 El Sistema de Colonia de Hormigas..... | 26 |
| 1.5.3 El Sistema de Hormigas Max-Min..... | 28 |
| <i>1.6 Conclusiones parciales</i> | 30 |
| CAPÍTULO 2. PROCEDIMIENTO BASADO EN COLONIAS DE HORMIGAS PARA EL MANEJO DE DATOS MEZCLADOS. | 31 |
| <i>2.1 Problema de optimización</i> | 31 |
| 2.1.1 Método propuesto..... | 31 |
| <i>2.2 Procedimiento de búsqueda local</i> | 38 |
| <i>2.3 Estudio del rendimiento del procedimiento MACO</i> | 42 |
| <i>2.4 Técnicas estadísticas para el análisis de los resultados</i> | 48 |
| 2.4.1 Análisis estadístico..... | 51 |
| <i>2.5 Conclusiones parciales</i> | 54 |

| | |
|---|-----------|
| CAPÍTULO 3. VALIDACIÓN DEL MÉTODO PROPUESTO ANTE UN EJEMPLO REAL MODELADO EN EL CAMPO DE LOS INTERCAMBIADORES DE CALOR DE TUBO Y CORAZA..... | 55 |
| <i>3.1. Intercambiadores de calor de Tubo y Coraza.....</i> | <i>55</i> |
| <i>3.2 Métodos para la optimización de los intercambiadores de calor de Tubo y Coraza.....</i> | <i>57</i> |
| <i>3.3 Modelo matemático para el diseño y cálculo de un intercambiador de Tubo y Coraza (método Kern).</i> | <i>59</i> |
| 3.3.1 Función objetivo: Minimización del costo de inversión y de operación..... | 60 |
| 3.3.2 Lado del tubo..... | 62 |
| 3.3.3 Lado coraza | 64 |
| 3.3.4 Costo total de operación de descuento y caída de presión | 65 |
| <i>3.4. Resultados</i> | <i>67</i> |
| <i>3.5 Conclusiones parciales</i> | <i>72</i> |
| CONCLUSIONES..... | 74 |
| RECOMENDACIONES | 75 |
| REFERENCIAS BIBLIOGRÁFICAS | 76 |

INTRODUCCIÓN

Existen problemas de optimización complejos en diversos campos como la economía, el comercio, la ingeniería, la industria o la medicina. Sin embargo, a menudo estos problemas son muy difíciles de resolver en la práctica. El estudio de esta dificultad inherente para resolver dichos problemas tiene cabida en el campo de la teoría de las Ciencias de la Computación, ya que muchos de ellos pertenecen a la clase de problemas NP-duros, lo que significa que no existe un algoritmo conocido que los resuelva en un tiempo polinomial (Johnson, 1979).

Día tras día, siguen apareciendo nuevos problemas, lo que ha dado lugar a que se hayan realizado muchas propuestas de algoritmos para tratar de solucionarlos. Las técnicas existentes se pueden clasificar básicamente en algoritmos exactos o aproximados. Los algoritmos exactos intentan encontrar una solución óptima y demostrar que la solución obtenida es de hecho la óptima global; estos algoritmos incluyen técnicas como, por ejemplo: procesos de vuelta atrás (*backtracking*), ramificación y poda (*Branch and Bound*), programación dinámica, etc. (Steiglitz, 1982, Bratley, 1996). Debido a que los algoritmos exactos muestran un rendimiento pobre para muchos problemas, se han desarrollado otros tipos de algoritmos aproximados que proporcionan soluciones de alta calidad para estos problemas combinatorios (aunque no necesariamente la óptima) en un tiempo computacional razonable.

En los métodos aproximados, denominados también heurísticos, la rapidez del proceso es tan importante como la calidad de la solución obtenida. En (A. Díaz and F.T., 1996) se define los algoritmos heurísticos como:

“Un método heurístico es un procedimiento para resolver un problema de optimización bien definido mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para obtener una buena solución”

En contraposición a los métodos exactos que proporcionan una solución óptima del problema, los métodos heurísticos se limitan a proporcionar una buena solución no necesariamente óptima. Lógicamente, el tiempo invertido por un método exacto para encontrar la solución óptima de un problema difícil, si es que existe tal método, es de un orden de magnitud muy superior al del heurístico (pudiendo llegar a ser tan grande en muchos casos, que sea inaplicable). Una clasificación usual de los problemas de optimización es en: Estocástica y Determinista. La

diferencia radica en los valores que toman las variables en el proceso de optimización, en la estocástica son valores probabilísticos y en la determinista los valores pueden ser discretos o continuos, de ahí que la optimización determinista se clasifique en dos, Optimización Combinatoria o Discreta y Optimización Continua.

En los últimos años ha habido un crecimiento espectacular en el desarrollo de procedimientos heurísticos para resolver problemas de optimización. Aunque hemos mencionado el caso de la resolución de un problema difícil, existen otras razones para utilizar métodos heurísticos, entre las que podemos destacar:

- El problema es de una naturaleza tal que no se conoce ningún método exacto para su resolución.
- Aunque existe un método exacto para resolver el problema, su uso es computacionalmente muy costoso.
- El método heurístico es más flexible que un método exacto, permitiendo, por ejemplo, la incorporación de condiciones de difícil modelación.
- El método heurístico se utiliza como parte de un procedimiento global que garantiza el óptimo de un problema. Existen dos posibilidades:
 - El método heurístico proporciona una buena solución inicial de partida.
 - El método heurístico participa en un paso intermedio del procedimiento, como por ejemplo las reglas de selección de la variable a entrar en la base en el método Simplex.

Los algoritmos heurísticos dependen en gran medida del problema concreto para el que se han diseñado. Las técnicas e ideas aplicadas a la resolución de un problema son específicas de éste y aunque, en general, pueden ser trasladadas a otros problemas, han de particularizarse en cada caso. Así pues, es necesario referirse a un problema concreto para estudiar con detalle los procedimientos heurísticos.

En los últimos años han aparecido una serie de métodos bajo el nombre de meta-heurísticos con el propósito de obtener mejores resultados que los alcanzados por los heurísticos tradicionales. El término meta-heurístico fue introducido por Fred Glover. En algunos textos podemos encontrar la expresión “heurísticos modernos” refiriéndose a los meta-heurísticos (Reeves, 1995). Los profesores Osman y Kelly en (Kelly, 1996) introducen la siguiente definición:

“Los procedimientos meta-heurísticos son una clase de métodos aproximados que están diseñados para resolver problemas difíciles de optimización combinatoria, en los que los heurísticos clásicos no son efectivos. Los meta-heurísticos proporcionan un marco general para crear nuevos algoritmos híbridos combinando diferentes conceptos derivados de la inteligencia artificial, la evolución biológica y los mecanismos estadísticos”

Los procedimientos meta-heurísticos se sitúan conceptualmente “por encima” de los heurísticos en el sentido que guían el diseño de éstos. Así, al enfrentarnos a un problema de optimización, podemos escoger cualquiera de estos métodos para diseñar un algoritmo específico que lo resuelva aproximadamente.

En estos momentos existe un gran desarrollo y crecimiento de estos métodos, los cuales han probado su eficiencia sobre una colección significativa de problemas; entre los que podemos destacar: Búsqueda Tabú (Glover, 1977), Recocido Simulado (S. Kirkpatrick, 1983), GRASP (Resende, 1989), Algoritmos Genéticos (Goldberg, 1998), Optimización de Enjambre de Partículas (Eberhart, 1995) (*Particle Swarm Optimization*, PSO) y Optimización Basada de Hormigas (*Ant Colony Optimization*, ACO) (Dorigo, 1997, Stutzle, 1999), entre otros. Estas últimas dos meta-heurísticas caen en la categoría de algoritmos bioinspirados (vida artificial) y de inteligencia colectiva, ya que la potencialidad de estos modelos para resolver problemas está dada por la cooperación entre individuos de una forma directa o indirecta. Específicamente en la meta-heurística ACO, la cooperación se realiza entre hormigas de una forma indirecta.

Es notable el creciente el interés científico de los modelos ACO, principalmente por su alta aplicabilidad para resolver diferentes problemas de optimización discreta. Estos enfoques se basan en usar el comportamiento de las colonias de hormigas naturales en el proceso de búsqueda, las cuales minimizan el recorrido entre su nido y cualquier fuente de alimento.

Esta meta-heurística desde su surgimiento ha representado una herramienta adecuada para solucionar problemas de optimización combinatoria, pero el hecho de que sólo pueda operar sobre problemas discreto, constituye una limitación. Ampliar el área de trabajo de esta meta-heurística a problemas mixtos le daría una mayor robustez, además de constituir una nueva herramienta en el área de la optimización de problemas mixtos.

Objetivo general

Desarrollar un procedimiento inspirado en la meta-heurística Colonias de Hormigas para la optimización de funciones con datos mezclados (variables discretas y continuas), con aplicación en la estimación de parámetros que caracterizan el diseño de intercambiadores de calor.

Objetivos específicos

1. Diseñar e implementar un procedimiento para la optimización de funciones con variables mezcladas utilizando la meta-heurística Colonia de Hormigas.
2. Comparar el método propuesto frente a otros algoritmos, utilizando 40 funciones de pruebas.
3. Aplicar el nuevo enfoque a la optimización de parámetros mixtos que caracterizan el diseño de intercambiadores de calor de Tubo y Coraza.

Tareas de Investigación

1. Diseñar un procedimiento, que permita la optimización de funciones con datos mezclados, usando la meta-heurística ACO.
2. Implementar la meta-heurística ACO para la optimización de funciones mixtas.
3. Diseñar un análisis de los datos que permita validar el método.
4. Valorar la factibilidad del método en la solución del problema de aplicación haciendo uso de los resultados del análisis.

Justificación

La optimización se utiliza en diversas áreas: ciencias aplicadas, ingeniería, economía, medicina, procesos industriales, entre otras ramas. De las diversas formas de abordar el problema de optimización, la meta-heurística ha mostrado la estrecha relación existente entre los diversos procesos del entorno y la solución del problema de optimización.

Además, existe un gran desarrollo y crecimiento de estos métodos que han probado su eficiencia sobre una colección significativa de problemas; entre los que podemos destacar, Búsqueda Tabú, Recocido Simulado, GRASP, Algoritmos Genéticos (Goldberg, 1998)(Goldberg, 1998)(Goldberg, 1998)(Genetic Algorithms), Optimización Basada en Enjambre de Partículas (Particle Swarm Optimization) y Optimización Basada en Colonias de Hormigas (Ant Colony Optimization), etc.

En los últimos años los algoritmos poblacionales y de inteligencia colectiva de optimización han ganado popularidad y son utilizados en varias áreas del conocimiento debido a su relativa simplicidad y costo computacional; representando desde su surgimiento una herramienta adecuada para solucionar problemas de optimización combinatoria.

La meta-heurística ACO permite la optimización de problemas combinatorios complejos. Se trata de problemas de optimización donde se minimiza una determinada función, donde las soluciones posibles conforman un conjunto discreto. Recientemente la meta heurística ACO ha sido enriquecida por distintas variantes donde se introducen nuevos elementos a los algoritmos con la finalidad de mejorar las soluciones obtenidas, o con el objetivo de reducir los tiempos de cálculo necesarios para la ejecución del proceso de búsqueda.

A pesar de esto, los problemas mixtos no pueden ser resueltos con el uso de la meta-heurística ACO de una forma natural, pues la meta-heurística original fue concebida para solucionar problemas discretos. Por lo tanto, la extensión de este algoritmo para solucionar problemas mixtos resulta relevante desde el punto de vista teórico y práctico.

Hipótesis de investigación

1. Es posible construir un nuevo procedimiento que usando ACO permita la optimización de funciones con variables mixtas.
2. El nuevo método permite obtener resultados superiores a otras técnicas de Inteligencia Artificial como son los Algoritmos Genéticos y Recocido Simulado.
3. Es factible optimizar los parámetros en el diseño de intercambiadores de calor de tubo y coraza, con el nuevo método basado en colonia de hormigas.

Diseño de la Tesis

Después de esta Introducción, la tesis se estructura en tres capítulos. El primero de ellos habla sobre la optimización global con meta heurísticas y la meta-heurística ACO. El segundo se ocupa sobre la explicación y validación del nuevo método. En el tercer capítulo se resuelve con el nuevo método, un problema real de mecánica en el área de los intercambiadores de calor y se analizan los resultados. Finalmente se formulan las conclusiones y recomendaciones y se relacionan las Referencias Bibliográficas.

CAPÍTULO 1. OPTIMIZACIÓN GLOBAL DE FUNCIONES USANDO METAHEURÍSTICAS: OPTIMIZACIÓN DE COLONIA DE HORMIGAS.

1.1. El problema de optimización

Las primeras ideas de optimización se remontan a los inicios del cálculo diferencial pero, sólo hasta los años 40 se unen en un área específica con el desarrollo de la programación lineal. Esta área ha tenido un rápido desarrollo posterior en lo que se refiere a métodos e implementaciones para la solución de problemas de optimización (Pierre, 1986).

La optimización se aplica al minimizar o maximizar, costos, tiempo, riesgos, ganancias, calidad, eficiencia entre otros. De estas aplicaciones básicas pueden derivarse innumerables clases de problemas diferentes entre sí. Muchos de estos problemas se han abordado con éxito, pero, aun así, hay problemas en el mundo real que no han sido solucionados por los métodos existentes; y otros, que aunque teóricamente pueden ser solucionados por los métodos existentes, el tiempo y costos necesarios para resolverlos los hace inviables desde la práctica (Pierre, 1986, Nash, 1996, Talbi, 2009).

1.1.1. Clasificaciones

Existen diferentes tipos de problemas de optimización, en la Figura 1.1 se muestra la clasificación de cada tipo de problema. A continuación se definen las características que permiten clasificar los diferentes tipos de problema.

- El problema puede o no tener restricciones.
- Si la función o funciones objetivo y las restricciones son lineales se dice que el problema es lineal; en caso contrario es no lineal.
- Las funciones objetivo pueden ser una o varias; cuando hay más de una función objetivo se llama optimización multiobjetivo.
- Las variables pueden ser todas continuas, todas discretas o mixtas; definiendo los problemas como continuos, combinatorios o mixtos.

Cada una de estas variantes generales dan paso a un tipo de problema diferente, que a su vez tienen subtipos de problemas cada vez más específicos; por ejemplo, las ecuaciones cuadráticas son un tipo específico dentro de la clasificación de optimización no lineal que presenta características particulares (Pierre, 1986, Rao, 1996).

La amplia gama de problemas descritos impiden que un único método de solución resuelva de manera eficiente; es decir, que cada método está diseñado para resolver cierto tipos de problemas específicos eficientemente, y en particular, un método diseñado para cierto problema específico podrá solucionarlo mejor que otros métodos (Pierre, 1986, Nash, 1996, Wolpert, 1997).



Figura 1.1 Clasificación general de los problemas de optimización.

1.1.3. Optimización global

En muchas ocasiones, las funciones tienen más de un mínimo en la región de soluciones factibles; los mínimos para una subregión de A se denominan mínimos locales, es decir, pueden encontrarse varios mínimos para una misma región a los que el método puede llegar. El problema de optimización global se define formalmente como (Bäck, 1996):

Dada una función $f : A \subseteq \mathbb{R}^n \rightarrow \mathbb{R}, A \neq \emptyset$, para $\vec{x}^* \in A$ el valor de $f^* := f(\vec{x}^*) > -\infty$

$$\forall \vec{x} \in A : f(\vec{x}^*) \leq f(\vec{x}) \tag{1.3}$$

Entonces, \vec{x}^* es mínimo global, f es la función objetivo, y el conjunto A es la región factible. El problema de determinar un punto de mínimo global es llamado problema de optimización global.

Esta definición puede extenderse al problema de optimización global sin restricciones donde $A \subseteq \mathbb{R}^n$; es importante aclarar que aunque en este caso se define la región factible como los reales, otros autores tienen una visión menos estricta al respecto. Aun así, las regiones acotadas; al ser continuas, siguen siendo espacios de búsqueda infinitos (Vesterstrom, 2004, Brest, 2006).

1.2. Métodos de optimización

El problema de optimización ha sido abordado desde diferentes perspectivas. En las siguientes secciones se describirán los principales enfoques desarrollados de acuerdo al tipo de problemas.

1.2.1. Optimización lineal.

También llamada programación lineal; en este caso, todas la funciones del problema deben ser lineales. El método más ampliamente aplicado a este tipo de problema es el Simplex; este método permite abordar problemas con varias funciones objetivo, restricciones lineales de igualdad e inecuaciones y varias variables. El método como tal, ha evolucionado desde sus inicios incluyendo varias técnicas, lo que lo hace más eficiente (Pierre, 1986, Nash, 1996).

1.2.2. Optimización no lineal.

La mayoría de los problemas reales tienen características de no linealidad. A su vez, la no linealidad puede ser descrita de muchas maneras de acuerdo a su forma (polinomial, no polinomial); de acuerdo a su grado de no linealidad (cuadrática, cúbica, etc.); de acuerdo al número de mínimos: unimodal o multimodal; de acuerdo a su geometría convexa o no convexa. Cada característica de la función enmarca un problema diferente que requiere diferentes métodos de solución y se pueden clasificar en dos grandes grupos: métodos diferenciales y no diferenciales (Pierre, 1986, Jang, 1996, Nash, 1996).

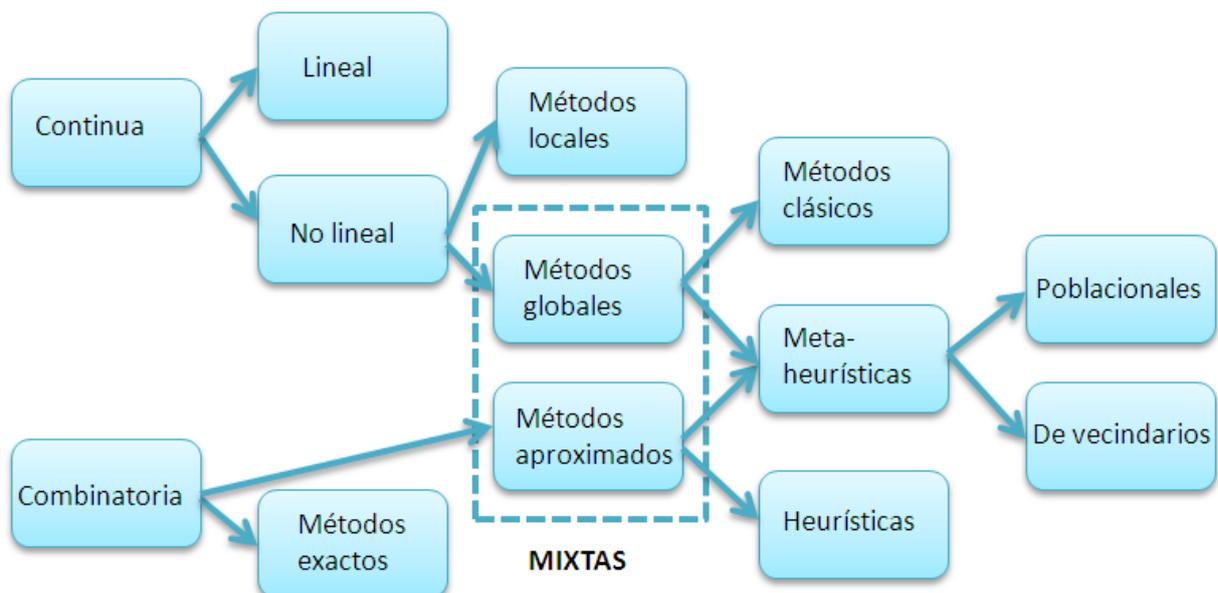


Figura 1.2 Métodos de optimización de acuerdo al tipo de problema.

En (Feoktistov, 2006) se presenta una clasificación de los métodos según el tipo de variable que describe muy bien la problemática y como se ha solucionado dando una visión más global de la relación entre los tipos de problemas de optimización y la forma de resolverlos. Esta clasificación se puede observar en la Figura 1.2: los métodos diferenciales estarían incluidos en la categoría de métodos clásicos en la rama de la optimización continua, no lineal y global; los otros métodos observados no utilizan las derivadas para resolver el problema.

- ***Los métodos diferenciales***

Históricamente el problema de optimización no lineal se solucionó con la teoría clásica de máximos y mínimos que utiliza los criterios de las derivadas para hallar los puntos estacionarios que muestran las inflexiones de una función y pueden ser: puntos máximos, mínimos o de cambio de concavidad. Este se denomina un método indirecto o analítico, pues la solución se determina por las propiedades diferenciales de la función o funciones involucradas, y no por la evaluación directa de la misma. Es un método que es de uso muy restringido a funciones no lineales derivables (Pierre, 1986).

Otros métodos utilizan la información de los puntos estacionarios encontrados por las derivadas y los mezcla con algoritmos numéricos de búsqueda directa, lo que permite mayor velocidad y abarcan mayor cantidad de tipos de problema. Estos métodos han sido ampliamente aplicados, son rápidos y eficaces siempre y cuando las condiciones iniciales necesarias estén dadas: deben ser funciones donde la labor analítica inicial sea posible (funciones derivables), lo que los aleja de muchos problemas del mundo real. Entre estos métodos se pueden mencionar: el método del gradiente y sus variaciones, método de Newton y sus variaciones, punto interior (Jang, 1996, Nash, 1996).

- ***Los métodos no diferenciales***

Estos métodos ignoran la información de las derivadas, lo que les permite abordar una gran gama de problemas; a pesar de no ser tan eficientes en términos de velocidad para problemas de fácil solución analítica (es decir, fácilmente derivables, baja no-linealidad y continuos) pueden solucionar problemas de diversa índole, con varias dimensiones, alta complejidad, para variables continuas, discretas y mixtas, mostrando gran potencial para problemas reales (Jang, 1996, Vesterstrom, 2004). Dentro de estos métodos se pueden destacar los métodos directos determinísticos y aleatorios entre los cuales se encuentran los métodos de Rosenbrock, Neldermead, Powell, búsqueda aleatoria repetitiva, entre otros (Himmelblau, 1972, Nash, 1996).

En la Figura 1.3, este tipo de métodos correspondería a las heurísticas y meta-heurísticas utilizadas en la solución de problemas no lineales y con cualquier tipo de variables; estos métodos son directos, heurísticos y pueden ser determinísticos o estocásticos. Dentro de estos métodos hay un subconjunto importante: las meta-heurísticas (de Castro, 2007).

Las meta-heurísticas no tienen un fundamento matemático robusto, son intuitivas y llegan a una solución mediante ciertas reglas planteadas empíricamente; se inspiran en el entorno, y generalmente llegan a una buena solución, en términos de cercanía al óptimo global del problema; pero, no tienen la precisión de los métodos diferenciales. Las primeras ideas al respecto se encuentran en la literatura en los años 60's cuando Holland propone los Algoritmos Genéticos (Goldberg, 1989) y Fogel la Programación Evolutiva (Bäck, 1997). Aunque inicialmente no pudieron ser muy desarrollados por la tecnología computacional de la época, a partir de los 80's se han desarrollado una serie de métodos que se presentarán más adelante.

1.3. Meta-heurísticas

Las meta-heurísticas son técnicas de inspiración biológica y física, que por medio de heurísticas emulan los procesos de optimización que se pueden observar en la naturaleza, como por ejemplo: la evolución. Los algoritmos meta-heurísticos no tienen una base matemática estricta y s las funciones buscando directamente en el universo de soluciones con diversas estrategias. Por sus características, generalmente ellos proveen una respuesta en menor tiempo y de alta calidad a problemas de gran complejidad, mostrando así su innegable potencial en aplicaciones del mundo real (Michalewicz, 2002, Dorigo, 2003).

Los algoritmos meta-heurísticos abordan problemas que por su condición no pueden ser resueltos satisfactoriamente por los métodos derivativos y por su estructura no necesitan la información diferencial. Sin embargo, aunque resuelven problemas más variados, su naturaleza iterativa y la ausencia de información sobre las derivadas hacen que sean susceptibles al punto (o puntos) inicial de optimización y hacen que los parámetros del algoritmo ganen gran importancia en el funcionamiento y resultado del mismo (Rao, 1996, Feoktistov, 2006).

En la Tabla 1.1 se presenta un resumen de las publicaciones sobre las técnicas más populares en aplicaciones, híbridos, mejoras metodológicas, comparaciones y pruebas. Se tienen en cuenta artículos, conferencias, libros, capítulos de libros de las principales publicaciones del área. Se aprecia el aumento del uso de los métodos meta-heurísticos y la mayor aceptación de algunos

métodos, como Algoritmos Genéticos (AG), Recocido Simulado (SA) u Optimización de Enjambre de Partículas (PSO), mientras que otros algoritmos como Programación Evolutiva y Estrategias Evolutivas, a pesar de ser métodos más antiguos, no son tan frecuentemente aplicados. De igual manera, las meta-heurística consideradas tanto en la Tabla 1.1 como en las secciones siguientes se limitan a las que tienen más acogida según la literatura.

Las meta-heurísticas toman su inspiración de procesos que mejoran sus condiciones o cuyo objeto o comportamiento emergente muestre una notable mejoría en alguna característica, ya sea en la calidad final o en el refinamiento de ciertos parámetros del proceso. Por ejemplo, el temple de un material, es un proceso que mejora la calidad del material por el control de la temperatura (Jang, 1996, de Castro, 2007). Teniendo en mente los procesos, se intenta simularlos por medio de heurísticas emulando el comportamiento de mejoría (optimización) que presenta el proceso de donde nace la idea; así, estos métodos han logrado soluciones aceptables a problemas de gran dificultad en un tiempo y a unos costos razonables (Bäck, 1997).

Tabla 1.1: Número de artículos por técnicas bio-inspiradas (Delgado, 2010).

| Meta-heurística | Números de artículos (metodologías y aplicaciones) | | | |
|--|--|-------------|-------------|-------|
| | antes del 2001 | 2001 - 2005 | 2006 - 2010 | Total |
| Algoritmos Genéticos | 7285 | 14723 | 13850 | 35858 |
| Recocido Simulado | 3490 | 3287 | 3388 | 10165 |
| PSO | 7 | 742 | 5262 | 6011 |
| Colonias de Hormigas | 47 | 699 | 2223 | 2969 |
| Búsqueda Tabú | 465 | 692 | 793 | 2969 |
| Evolución diferencial | 93 | 382 | 1362 | 1837 |
| Algoritmos Inmunológicos | 65 | 420 | 1180 | 1665 |
| Programación evolutiva | 252 | 382 | 321 | 955 |
| Estrategias evolutivas | 168 | 342 | 620 | 930 |
| GRASP | 235 | 298 | 392 | 925 |
| Algoritmos Meméticos | 5 | 141 | 396 | 542 |
| Búsqueda dispersa | 21 | 93 | 170 | 284 |
| Hooke and Jeeves Pattern search | 113 | 34 | 55 | 202 |
| Quimiotaxis bacteriana | N.A. | 1 | 22 | 23 |

Las meta-heurísticas tienen diversas fuentes de inspiración, pero se pueden identificar básicamente tres grandes grupos: evolución, procesos físicos y aprendizaje, dentro de los cuales se pueden enmarcar los diferentes métodos que han sido desarrollados, ya sea dentro de una de las líneas estrictamente o en una combinación de ellas. En la Figura 1.3 se observan las técnicas más representativas de las diferentes categorías.

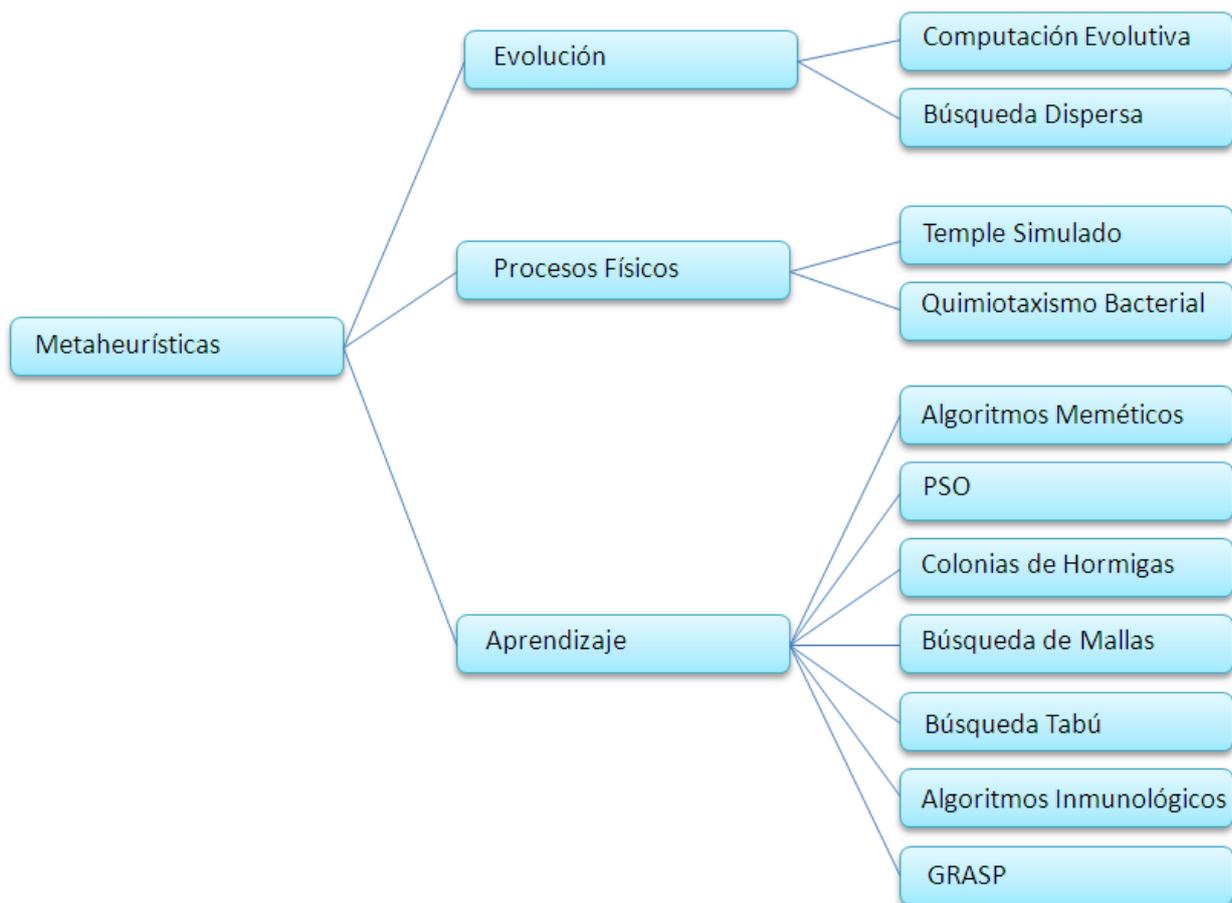


Figura 1.3 Meta heurísticas de acuerdo a su inspiración

1.3.1. Técnicas inspiradas en la evolución

Basado en los trabajos de Darwin, la evolución se describe como el mejoramiento de una población por medio de la reproducción de los seres más aptos. Más tarde, Lamarck en su trabajo sobre evolución, le da un peso mayor a las mutaciones de los individuos y describe cómo pequeños cambios en los individuos logran que una población se adapte mejor a su medio o a un cambio en él. En la Tabla 1.2 se observa un paralelo entre la teoría de evolución y el problema de optimización, el mismo paralelo se presenta en cada uno de los métodos de optimización bio-inspirados; la población representa un conjunto de soluciones diferentes y el medio sería la función de adaptación que describe cuales son los mejores individuos de acuerdo al criterio del medio; en algunos casos se toma la misma función objetivo o algún escalamiento de la misma; en este enfoque, los métodos presentan una búsqueda elitista, tienen una alta tasa de convergencia y exploran especialmente a nivel local. Se han desarrollado estrategias para evitar la convergencia prematura y la injerencia de los parámetros creando parámetros adaptativos para algunos tipos de problemas (Coello, 2005).

Tabla 1.2: Equivalente entre la teoría de evolución y el problema de optimización

| Evolución | | Modelo de Optimización |
|-----------------------|---|--|
| Individuo (Población) | → | Solución |
| Adaptación | → | Valor de la función objetivo |
| Generación | → | Iteración del ciclo |
| Recombinación | → | Combinación de individuos para obtener mejores soluciones |
| Mutación | → | Cambios aleatorios que dan nuevas características al individuo |
| Selección | → | Toma de los mejores individuos |

1.3.2. Técnicas inspiradas en los procesos físicos

Estas técnicas se inspiran en fenómenos físicos en los que algún parámetro es mejorado, por ejemplo: en la quimiotaxis bacteriana, que es el proceso por el cual una bacteria direcciona sus movimiento de acuerdo la concentración química del medio y de esta forma encuentra su alimento (Muller, 2000, de Castro, 2007). Al igual que en el caso anterior, se tiene el fenómeno y su aproximación al modelo de optimización.

Aunque este es el grupo que presenta más diferencias entre los procesos de donde toman su inspiración, tienen como común denominador que el proceso físico descrito mejora sus condiciones por el control o la búsqueda de una variable específica. En el temple simulado, por ejemplo, se tiene como variable de control de mejoramiento a la temperatura. Las técnicas enmarcadas en este enfoque tienen una forma de búsqueda metódica de más grande a más pequeño que se va moviendo de acuerdo a la mejor solución encontrada y su convergencia depende especialmente del paso de la búsqueda, los parámetros del algoritmo y el punto de inicio de la misma. Son heurísticas de búsqueda en el vecindario.

Una de las técnicas más empleadas en la actualidad es el Recocido Simulado. En la Tabla 1.3 se observa un paralelo entre el método de Recocido Simulado y el problema de optimización

Tabla 1.3: Equivalente entre Recocido Simulado y el problema de optimización

| Metalurgia | | Modelo de Optimización |
|-----------------------------|---|-------------------------------|
| Configuración | → | Solución Factible |
| Energía de la Configuración | → | Costo de la solución |
| Mínima Energía | → | Función Objetivo |
| Configuración fundamental | → | Solución Óptima |
| Temperatura | → | Parámetro |

1.3.3. Técnicas inspiradas en el aprendizaje

El aprendizaje, aunque tiene en común con la evolución el principio de transformación para mejorar, tiene su fundamento principal en la memoria, es decir, las decisiones pasadas tienen inferencia en las decisiones futuras del individuo (solución) y por lo tanto aprende como hacer mejor las cosas. Se inspira en diversos fenómenos y comportamientos emergentes, como el comportamiento gregario de las hormigas, donde el aprendizaje individual comunicado logra un mejor comportamiento global; o sistemas de aprendizaje como el sistema inmune que genera anticuerpos luego de vencer a cierta bacteria o virus, mejorando su comportamiento ante el mismo estímulo en el futuro (de Castro, 2007, Talbi, 2009). Esta línea de inspiración mejora algunas falencias de los grupos anteriores por implementación de los mecanismos de aprendizaje, logrando una mayor exploración del espacio de búsqueda en algunos casos. Aun así, su exactitud depende mayormente de la calibración de los parámetros del algoritmo y de la cantidad de dimensiones del problema.

1.4 Algoritmo de colonia de hormigas (ACO)

1.4.1 Las colonias de hormigas naturales

Las hormigas son insectos sociales que viven en colonias y que, debido a su colaboración mutua, son capaces de mostrar comportamientos complejos y realizar tareas difíciles desde el punto de vista de una hormiga individual. Un aspecto interesante del comportamiento de muchas especies de hormigas es su habilidad para encontrar los caminos más cortos entre su hormiguero y las fuentes de alimento. Este hecho es especialmente interesante si se tiene en cuenta que muchas de las especies de hormigas son casi ciegas, lo que evita el uso de pistas visuales.

Mientras que se mueven entre el hormiguero y la fuente de alimento, algunas especies de hormigas depositan una sustancia química denominada feromona (una sustancia que puede “olerse”). 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. De hecho, los experimentos realizados por biólogos han demostrado que las hormigas prefieren de manera probabilística los caminos marcados con una concentración superior de feromona (J. M. Pasteels, 1987, S. Goss, 1989). En la práctica, 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 es la probabilidad de elegirlo. Puesto que las hormigas depositan

feromona en el camino que siguen, este comportamiento lleva a un proceso de auto-refuerzo que concluye con la formación de rastros señalados por una concentración de feromona elevada. Este comportamiento permite además a las hormigas encontrar los caminos más cortos entre su hormiguero y la fuente del alimento (S. Goss, 1989).

En la Figura 2.1 se ilustra cómo este mecanismo permite a las hormigas encontrar el camino más corto. Inicialmente no existe ningún rastro de feromona en el medio y, cuando una hormiga llega a una intersección, elige de manera aleatoria una de las bifurcaciones posibles. Según transcurre el tiempo y mientras que las hormigas están recorriendo los caminos más prometedores, estos van recibiendo una cantidad superior de feromona. Esto ocurre gracias a que al ser los caminos más cortos, las hormigas que los siguen consiguen encontrar la comida más rápidamente, por lo que comienzan su viaje de retorno antes. Entonces, en el camino más corto habrá un rastro de feromona ligeramente superior y, por lo tanto, las decisiones de las siguientes hormigas estarán dirigidas en mayor medida a dicho camino. Además, este camino recibirá una proporción mayor de feromona por las hormigas que vuelven por él, que por el de las que vuelven por el camino más largo. Este proceso finaliza haciendo que la probabilidad de que una hormiga escoja el camino más corto aumente progresivamente y que al final el recorrido de la colonia converja al más corto de todos los caminos posibles.

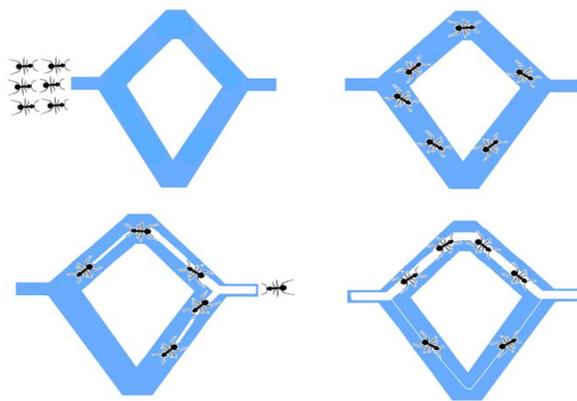


Figura 2.1 Comportamiento de la colonia: obtiene el camino más corto entre dos puntos.

Esta convergencia se complementa con la acción del entorno natural que provoca que la feromona se evapore después de cierto tiempo. Así, los caminos menos prometedores pierden progresivamente feromona porque son visitados cada vez por menos hormigas. Sin embargo, algunos estudios biológicos han demostrado que los rastros de feromona son muy persistentes - 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. (Bonabeau, 1999) - lo que provoca una menor influencia del efecto de la evaporación del rastro de feromona en el proceso de búsqueda del camino más corto entre dos puntos.

En (Bonabeau, 1999), numerosos experimentos muestran que, debido a la gran persistencia de feromona, es difícil que las hormigas “olviden” un camino que tiene un alto nivel de feromona aunque hayan encontrado un camino aún más corto. Hay que tener en cuenta que si se traslada este comportamiento directamente al ordenador para diseñar un algoritmo de búsqueda podemos encontrarnos con que se quede rápidamente estancado en un óptimo local.

1.4.2 De las hormigas naturales a la meta-heurística de ACO

Los algoritmos de ACO se inspiran directamente en el comportamiento de las colonias reales de hormigas para solucionar problemas de optimización combinatoria. Se basan en una colonia de hormigas artificiales, esto es, unos agentes computacionales simples que trabajan de manera cooperativa y se comunican mediante rastros de feromona artificiales.

Los algoritmos basados en ACO son esencialmente algoritmos constructivos: en cada iteración, cada hormiga construye una solución al problema recorriendo un grafo de construcción. Cada arista del grafo, que representa los posibles pasos 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 r hasta el nodo s , o sea, de recorrer la arista a_{rs} . Se nota por η_{rs} . 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 r a s . 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 hormigas. Se nota por τ_{rs} .
- En esta sección se presentan los pasos que llevan desde las hormigas reales a la ACO. A partir de ahora debe tenerse en cuenta que los algoritmos ACO presentan una doble perspectiva:
 - Por un lado, son una abstracción de algunos patrones de comportamiento naturales 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).

1.4.3 Tipos de problemas resolubles por ACO

El tipo de problema que se puede resolver por medio de hormigas artificiales pertenece al grupo (restringido) de problemas de camino mínimo que se pueden caracterizar por los siguientes aspectos (se sigue principalmente la presentación expuesta en (Caro, 1999, Stützle, 2003)):

- Existe un conjunto de restricciones Ω definido por el problema a solucionar.
- Existe un conjunto finito de componentes $N = \{n_1, n_2, \dots, n_l\}$.
- El problema presenta diversos estados que se definen según secuencias ordenadas de componentes $\delta = \langle n_r, n_s, \dots, \mu_n, \dots \rangle$ ($\langle r, s, \dots, n, \dots \rangle$ para simplificar) sobre los elementos de N . Si Δ es el conjunto de todas las secuencias posibles, se llamará $\tilde{\Delta}$ al conjunto de posibles (sub)secuencias que respetan las restricciones Ω . Los elementos definen los estados posibles. $|\delta|$ es la longitud de una secuencia δ , esto es, el número de componentes en la secuencia.
- Existe una estructura de vecindario definida como sigue: δ_2 es un vecino de δ_1 si: (i) tanto δ_1 como δ_2 pertenecen a Δ , (ii) el estado δ_2 puede alcanzarse desde δ_1 en un paso lógico, es decir, si r es la última componente de la secuencia δ_1 , debe existir una componente $s \in Y$ tal que $\delta_2 = \langle \delta_1, s \rangle$, o sea, debe existir una transición válida entre r y s . El vecindario alcanzable de δ_1 es el conjunto que contiene todas las secuencias $\delta_2 \in \tilde{\Delta}$. Si $\delta_2 \notin \tilde{\Delta}$, δ_2 está en el vecindario no alcanzable de δ_1 .
- Una solución S es un elemento de $\tilde{\Delta}$ que verifica todos los requisitos del problema.
- Existe un costo $C(S)$ asociado a cada solución S .
- En algunos casos, se puede asociar un costo o una estimación del mismo puede estar asociado a los estados.

Como se ha comentado todas las características anteriores se dan en problemas de optimización combinatoria que pueden representarse en forma de grafo ponderado $G = (N, A)$, donde A es el conjunto de aristas que conectan el conjunto de componentes N . El grafo G se denomina grafo de construcción $G.2$ por tanto, se tiene que:

- las componentes n_r son los nodos del grafo,
- los estados δ (y por tanto las soluciones S) se corresponden con caminos en el grafo, esto es, secuencias de nodos o aristas,
- las aristas del grafo, a_{rs} , son conexiones/transiciones que definen la estructura del vecindario. $\delta_2 = \langle \delta_1, s \rangle$ será vecino de δ_1 si el nodo r es la última componente de δ_1 y la arista a_{rs} existe en el grafo,

- deben existir los costos explícitos c_{rs} asociados con cada arista, y
- 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.

1.4.4 La hormiga artificial

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. Tiene las siguientes propiedades (Caro, 1999, Stützle, 2003):

- Busca soluciones válidas de costo mínimo para el problema a solucionar.
- Tiene una memoria L que almacena información sobre el camino seguido hasta el momento, esto es, L almacena la secuencia generada. Esta memoria puede usarse para (i) construir soluciones válidas, (ii) evaluar la solución generada, y (iii) reconstruir el camino que ha seguido la hormiga.
- Tiene un estado inicial $\delta_{inicial}$, que normalmente se corresponde con una secuencia unitaria y una o más condiciones t de parada asociadas.
- Comienza en el estado inicial y se mueve siguiendo estados válidos, construyendo la solución asociada incrementalmente.
- Cuando está en un estado $\delta_r = \langle \delta_{r-l}, r \rangle$ (es decir, actualmente está localizada en el nodo r y ha seguido previamente la secuencia δ_{r-l} , puede moverse a cualquier nodo s de su vecindario posible $N(r)$, definido como $N(r) = \{S | (a_{rs} \in A) \text{ y } ((\delta_r, s) \notin \tilde{\Delta})\}$.
- 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 r hasta el s , puede actualizar el rastro de feromona τ_{rs} asociado al arco a_{rs} .
- Este proceso se llama actualización en línea de los rastros de feromona paso a paso.
- 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 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).

1.4.5 Similitudes y diferencias entre las hormigas naturales y artificiales

Las colonias de hormigas naturales y artificiales comparten una serie de características. Las más importantes pueden resumirse a continuación (M. Dorigo, 1999):

- 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 estimérgica 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 artificiales viven en un mundo discreto y tienen algunas capacidades adicionales:

- Las hormigas artificiales pueden hacer uso de la información heurística (no sólo 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. Otra gran diferencia se refiere al momento de depositar la feromona. Las hormigas artificiales normalmente sólo depositan feromona después de generar una solución completa.
- La evaporación de feromona en los algoritmos de ACO 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. Esto evita una convergencia prematura del algoritmo hacia óptimos locales.

- Para mejorar la eficiencia y eficacia del sistema, los algoritmos ACO pueden enriquecerse con habilidades adicionales. Ejemplos típicos son la capacidad de mirar más allá de la siguiente transición (“*lookhead*”) (Middendorf., 1998), la optimización local (Gambardella., 1997, Hoos., 1997) y “*backtracking*” (cuyo uso no está muy extendido), que persiguen mejorar la eficacia, o la llamada lista de candidatos que contiene un conjunto de los estados vecinos más prometedores (Gambardella., 1997, Caro, 1999) para mejorar la eficiencia del algoritmo.

1.4.6 Modo de funcionamiento y estructura de un algoritmo ACO

Como se ha visto en las secciones anteriores, el modo de operación básico de un algoritmo de ACO es como sigue: las m hormigas (artificiales) de la colonia se mueven, concurrentemente y de manera asíncrona, 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 (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 a 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 ACO 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 como un mecanismo que evita el estancamiento en la búsqueda y permite que las hormigas busquen y exploren nuevas regiones del espacio. Las acciones del demonio son acciones opcionales -que no tienen un contrapunto natural- para implementar tareas desde una perspectiva global que no pueden llevar a cabo las hormigas por la perspectiva local que ofrecen. Ejemplos 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, o 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. La estructura de un algoritmo de ACO genérico es como sigue (Caro, 1999, M. Dorigo, 1999):

Procedimiento meta-heurística_ACO()

```

1 Inicialización_de_parámetros
2 mientras (criterio_de_terminación_no_satisfecho)
3     Programación_de_actividades
4         Generación_de_Hormigas_y_actividad ()
5         Evaporación_de_Feromona ()
6         Acciones_del_demonio () opcional
7     fin Programación_de_actividades
8 fin mientras

```

Procedimiento Generación_de_Hormigas_y_actividad ()

```

1 repetir en paralelo desde k=1 hasta m (número_hormigas)
2     Nueva_Hormiga (k)
3 fin repetir en paralelo

```

Procedimiento Nueva_Hormiga (id_Hormiga)

```

1 inicializa_hormiga (id_Hormiga)
2 L = actualiza_memoria_hormiga ()
3 mientras (estado_actual ≠ estado_objetivo)
4     P = calcular_probabilidades_de_transición (A, L, Ω)
5     siguiente_estado = aplicar_política_decisión (P, Ω)
6     mover_al_siguiente_estado (siguiente_estado)
7     si (actualización_feromona_en_linea_paso_a_paso)
8         depositar_feromona_en_el_arco_vistado ()
9     fin si
10    L = actualizar_estado_interno ()
11 fin mientras
12 si (actualización_feromona_en_linea_a_posteriori)
13     para cada arco visitado
14         depositar_feromona_en_el_arco_visitado()
15     fin para
16 fin si
17 liberar_recursos_hormiga (id_Hormiga)

```

El primer paso incluye la inicialización de los valores de los parámetros que utiliza el algoritmo. Entre otros, se deben fijar el rastro inicial de feromona asociado a cada transición, τ_0 , que es un valor positivo pequeño, normalmente el mismo para todas las componentes/conexiones, el número de hormigas en la colonia, m , 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 meta-heurística ACO controla, mediante el constructor Programación_de_Actividades, la planificación de las tres componentes mencionadas en esta sección: (i) la generación y puesta en funcionamiento de las hormigas artificiales, (ii) la evaporación de feromona, y (iii) las acciones del demonio. La implementación de este constructor determinará la sincronía existente entre cada una de las tres componentes. Mientras que la aplicación a problemas “clásicos” NP-duros (no distribuidos), normalmente usa una planificación secuencial, en problemas distribuidos como el enrutamiento en redes, el paralelismo puede ser explotado de manera sencilla y eficiente.

Como se ha comentado antes, varias componentes son o bien opcionales, como las acciones del demonio, o bien dependientes estrictamente del algoritmo de ACO 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).

Por otro lado, hay que remarcar 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 L . 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.).

Por último, comentar que 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 Ω para establecer el proceso de transición probabilístico hacia otros estados válidos.

1.5 Modelos de Optimización Basada en Colonias de Hormigas

En la literatura se han propuesto diversos algoritmos que siguen la meta-heurística ACO. Entre los algoritmos de ACO disponibles para problemas de optimización combinatoria NP-duros, se encuentran el Sistema de Hormigas (*Ant System*, AS) (M. Dorigo, 1996), el Sistema de Colonia de Hormigas (*Ant Colony System*, ACS) (Gambardella., 1997), el Sistema de Hormigas Max-Min (*Max-Min Ant System*, MMAS) (Hoos., 2000b) y el Sistema de la Mejor-Peor Hormiga (*Best-Worst Ant System*, BW-ACO) (O. Cerdón, 1999, O. Cerdón, 2002). A continuación, se presenta una breve descripción de algunos de estos algoritmos.

1.5.1 El Sistema de Hormigas

El Sistema de Hormigas (SH, siglas en español) (M. Dorigo, 1996) fue desarrollado por Marco Dorigo y fue el primer algoritmo de ACO. Inicialmente, se presentaron 3 variantes distintas: SH-densidad, SH-cantidad y SH-ciclo, que se diferenciaban en la manera en que se actualizaban los rastros de feromona. En los dos primeros, las hormigas depositaban feromona mientras que construían sus soluciones (esto es, aplicaban una actualización en línea paso a paso de feromona), con la diferencia de que la cantidad de feromona depositada en el SH-densidad es constante, mientras que la depositada en SH-cantidad dependía directamente de la deseabilidad heurística de la transición η_{rs} . Por último, en SH-ciclo, la deposición de feromona se lleva a cabo una vez que la solución está completa (actualización en línea a posteriori de feromona). Esta última variante era la que obtenía mejores resultados y es por tanto la que se conoce como SH en la literatura (y en el resto de este trabajo).

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 como sigue. En cada paso de construcción, una hormiga k escoge ir al siguiente nodo con una probabilidad que se calcula como:

$$p_{rs}^k = \begin{cases} \frac{[\tau_{rs}]^\alpha * [\eta_{rs}]^\beta}{\sum_{u \in N_r^k} [\tau_{ru}]^\alpha * [\eta_{ru}]^\beta}, & \text{si } s \in N_k(r) \\ 0, & \text{en otro caso} \end{cases} \quad (1.1)$$

donde $N_k(r)$ es el vecindario alcanzable por la hormiga k cuando se encuentra en el nodo r , y $\alpha, \beta \in \mathbb{R}$ son dos parámetros que ponderan la importancia relativa de los rastros de feromona y la información heurística. Cada hormiga k almacena la secuencia que ha seguido hasta el momento y su memoria L_k , tal como se explicó antes, se utiliza para determinar $N_k(r)$ en cada paso de construcción.

Volviendo a los parámetros α y 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 que el 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.

Como se ha dicho, 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:

$$\tau_{rs} \leftarrow (1 - \rho) * \tau_{rs} \quad (1.2)$$

donde $\rho \in (0,1)$ 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_k) y deposita una cantidad de feromona $\Delta\tau_{rs}^k$ en cada conexión por la que ha viajado:

$$\tau_{rs} \leftarrow \tau_{rs} + \Delta\tau_{rs}^k, \forall a_{rs} \in S_k \quad (1.3)$$

donde $\Delta\tau_{rs}^k = f(C(S_k))$, es decir, la cantidad de feromona que se deposita depende de la calidad $C(S_k)$ de la solución S_k construida por la hormiga k . Para resumir la descripción del SH, se muestra a continuación el procedimiento Nueva_Hormiga para este algoritmo:

Procedimiento Nueva_Hormiga (id_Hormiga)

```

2   $k = \text{id\_Hormiga}; r = \text{generar\_estado\_inicial}; S_k = r$ 
3   $L_k = r$ 
4  mientras (estado_actual  $\neq$  estado_objetivo)
5      para cada  $s \in N_k(r)$  hacer  $p_{rs}^k = \frac{[\tau_{rs}]^{\alpha} * [\eta_{rs}]^{\beta}}{\sum_{u \in N_r^k} [\tau_{ru}]^{\alpha} * [\eta_{ru}]^{\beta}}$ 
6      siguiente_est = aplicar_política_de_decisión( $P, N_k(r)$ )
7       $r = \text{siguiente\_est}; S_k = \langle S_k, r \rangle$ 
8      ---
9       $L_k = L_k \cup r$ 
10 fin mientras
        {se ejecuta el procedimiento Evaporación_de_feromona() se lanza y evapora la
        feromona en cada arco  $a_{rs}: \tau_{rs} = \{(1 - \rho)\tau_{rs}\}$ 
11 para cada arco  $a_{rs} \in S_k$  hacer
12      $\tau_{rs} \leftarrow \tau_{rs} + e * f(C(S_k))$ 
13 fin para
14 liberar_recursos_hormiga(id_Hormiga)

```

La línea 8 vacía se incluye para hacer notar que no existe una actualización de feromona en-línea paso a paso y que, antes de la línea 12, el demonio debe haber aplicado la evaporación de feromona. De hecho, este es un ejemplo donde la construcción Programación_de_actividades interfiere con el funcionamiento de los procedimientos fundamentales de la meta-heurística ACO, tal como se indicaba anteriormente.

Antes de finalizar esta sección es importante resaltar que los creadores del SH también propusieron una versión extendida del algoritmo que normalmente mejoraba los resultados obtenidos, llamada SH elitista (M. Dorigo, 1996). 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:

$$\tau_{rs} \leftarrow \tau_{rs} + e * f(C(S_{\text{mejor-global}})), \forall a_{rs} \in S_{\text{mejor-global}} \quad (1.4)$$

1.5.2 El Sistema de Colonia de Hormigas

El Sistema de Colonias de Hormigas (SCH, siglas en español) (Gambardella., 1997) es uno de los primeros sucesores del SH que introduce tres modificaciones respecto a dicho algoritmo de ACO:

1. El SCH usa una regla de transición distinta, denominada regla proporcional pseudo-aleatoria. Sea k una hormiga situada en el nodo r , $q_0 \in [0,1]$ un parámetro y q un valor aleatorio en $[0,1]$, el siguiente nodo s se elige aleatoriamente mediante la siguiente distribución de probabilidad:

Si $q \leq q_0$

$$p_{rs}^k = \begin{cases} 1, & \text{si } s = \arg \max_{u \in N_k(r)} \{ \tau_{ru} * \eta_{ru}^\beta \} \\ 0, & \text{en otro caso} \end{cases} \quad (1.5)$$

si no ($q_0 > q$)

$$p_{rs}^k = \begin{cases} \frac{[\tau_{rs}]^\alpha * [\eta_{rs}]^\beta}{\sum_{u \in N_k(r)} [\tau_{ru}]^\alpha * [\eta_{ru}]^\beta}, & \text{si } s \in N_k(r) \\ 0, & \text{en otro caso} \end{cases} \quad (1.6)$$

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 $q_0 > q$ 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.

2. 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, $S_{\text{mejor-global}}$ (aunque en algunos trabajos iniciales se consideraba también una actualización basada en la mejor hormiga de la iteración (Gambardella., 1997), en ACO casi siempre se aplica la actualización por medio de la 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:

$$\tau_{rs} \leftarrow (1 - \rho) * \tau_{rs}, \forall a_{rs} \in S_{\text{mejor-global}} \quad (1.7)$$

A continuación el dominio deposita feromona respondiendo a la regla:

$$\tau_{rs} \leftarrow \tau_{rs} + \rho * f(C(S_{\text{mejor-global}})), \forall a_{rs} \in S_{\text{mejor-global}} \quad (1.8)$$

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

3. 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_{rs} , aplica la regla:

$$\tau_{rs} \leftarrow (1 - \varphi) * \tau_{rs} + \varphi * \tau_0 \quad (1.9)$$

donde $\varphi \in (0,1]$ es un segundo parámetro de decremento 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, se 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.

Procedimiento Nueva_Hormiga (id_Hormiga)

```

1  k= id_Hormiga; r = Generar_Estado_Inicial; Sk = r
2  Lk = r
3  mientras (estado_actual ≠ estado_objetivo)
4      para cada s ∈ Nk(r) hacer calcular brs = τrsηrsβ
5      q = generar_valor_aleatorio_en_[0,1]
6      si (q ≤ q0)
7          siguiente_estado = max(brs, Nk(r))
8      si no
9          para cada s ∈ Nk(r) hacer prsk =  $\frac{b_{rs}}{\sum_{u \in N_r^{(k)}} b_{ru}}$ 
10         sig_est=aplicar_política_de_decisión (p, Nk(r))
11     fin si
12     r = sig_est; Sk = ⟨Sk, r⟩
13     τrs = (1 - φ)τrs + φτ0
14     Lk = Lk ∪ r
15 fin mientras
16 ---
17 ---
18 ---
19 liberar_recursos_hormiga(id_Hormiga)

```

Procedimiento Acciones_del_dominio()

```

1  para cada Sk hacer búsqueda_local (Sk) {opcional}
2  Smejor-actual = mejor_solucion (Sk)
3  si (mejor (Smejor-actual, Smejor-global))
4      Smejor-global = Smejor-actual
5  fin si
6  para cada arista ars ∈ Smejor-global hacer
        {se ejecuta el procedimiento evaporación_de_feromona () se lanza y evapora
        feromona en la arista ars: τrs = {(1 - ρ)τrs}}
7  τrs ← τrs + ρ * f(C(Smejor-global))
8  fin para

```

1.5.3 El Sistema de Hormigas Max-Min

El Sistema de Hormigas Max-Min (SHMM) (Stützle., 1999, Hoos., 2000a, Hoos., Agosto 1996), 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:

1. 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 hayan construido su solución cada rastro de feromona sufre una evaporación:

$$\tau_{rs} \leftarrow (1 - \rho) * \tau_{rs} \quad (1.10)$$

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

$$\tau_{rs} \leftarrow \tau_{rs} + f(C(S_{\text{mejor}})), \forall a_{rs} \in S_{\text{mejor}} \quad (1.11)$$

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 (Stützle., 1999, Hoos., 2000a). Además, en el SHMM las soluciones que ofrecen las hormigas suelen ser mejoradas usando optimizadores locales antes de la actualización de feromona.

2. 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 $\tau_{\max}^* = 1/(\rho * C(S^*))$, donde S^* es la solución óptima. 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} .

Para poder incrementar la exploración de nuevas soluciones, el SHMM utiliza en ocasiones re-inicializaciones de los rastros de feromona (Stützle., 1999, Hoos., 2000a, Hoos., Agosto 1996).

3. 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 acusadas, lo que

no ocurre cuando los rastros de feromona se inicializan a un valor muy pequeño. La estructura del procedimiento Acciones_del_dominio en el SHMM se muestra a continuación:

Procedimiento Acciones_del_dominio()

```

1 para cada  $S_k$  hacer búsqueda_local( $S_k$ ) {opcional}
2  $S_{mejor-actual} = mejor\_solucion(S_k)$ 
3 si (mejor( $S_{mejor-actual}, S_{mejor-global}$ ))
4      $S_{mejor-global} = S_{mejor-actual}$ 
5 fin si
6  $S_{mejor} = decisión(S_{mejor-global}, S_{mejor-actual})$ 
7 para cada arista  $a_{rs} \in S_{mejor}$  hacer
8      $\tau_{rs} \leftarrow \tau_{rs} + f(C(S_{mejor}))$ 
9     si ( $\tau_{rs} < \tau_{min}$ )  $\tau_{rs} = \tau_{min}$ 
10 fin para
11 si (condición_de_estancamiento)
12     para cada arista  $a_{rs}$  hacer  $\tau_{rs} = \tau_{max}$ 
13 fin si

```

1.6 Conclusiones parciales

Resulta satisfactorio el aporte científico, económico y social que ofrece el uso de técnicas computacionales para la optimización global. Proveen una respuesta en un tiempo razonable y de mejor calidad a problemas de alta complejidad mostrando así gran potencial en aplicaciones del mundo real. Los procedimientos meta-heurísticos se sitúan conceptualmente “por encima” de otros, por su capacidad de resolver problemas difíciles de optimización. Así, al enfrentar el problema de optimización, se puede escoger uno de estos métodos para diseñar un algoritmo específico que resuelva aproximadamente el problema. De los Modelos de Optimización basado en colonia de Hormigas, en este trabajo se utiliza *Max-Min Ant System* como optimizador pues no requiere especificar información heurística y evita los estados de estancamiento.

CAPÍTULO 2. PROCEDIMIENTO BASADO EN COLONIAS DE HORMIGAS PARA EL MANEJO DE DATOS MEZCLADOS.

En este capítulo se presenta el proceso de construcción del nuevo algoritmo basado en colonias de hormigas para el manejo de datos mezclados. Además, se prueba con 40 funciones mixtas y mediante pruebas estadísticas, se comparan los resultados con las meta heurísticas: Algoritmo Genético (AG) y Recocido Simulado (SA).

2.1 Problema de optimización

El algoritmo que se desarrolla, utiliza ACO para resolver problemas de optimización de funciones mixtas (continuas y discretas) no lineales. El problema a resolver debe ser de minimización o reducirse a este, debido a que el algoritmo busca devuelve un conjunto de variables que hacen mínimo la función objetivo. Debe ser sin restricciones. Para tratar problemas con restricciones, esta debe ser incluida por la persona en la función objetivo como una penalización, debido a que el procedimiento no trabaja con restricciones.

2.1.1 Método propuesto

Para que ACO pueda resolver problemas con datos mezclados, se transforma el problema mixto (P) en un problema complemente discreto (P*). Es sobre este problema discreto sobre el que trabaja ACO. Para esto se dispone de cuatro pasos:

- **Paso #1:** El primer paso del algoritmo que se propone consiste en tomar toda la información de las variables.
- **Paso #2:** Transforma la codificación de los individuos. El genoma con variables mezcladas se transforma en una codificación totalmente discreta.
- **Paso #3:** Resolver el problema discreto transformado, utilizando la meta-heurística Colonia de Hormigas.
- **Paso #4:** Transformar la solución discreta encontrada a su equivalente pero con las variables originales mixtas.

❖ Lectura de las variables del problema.

Primeramente se lee la cantidad de variables con que cuenta el problema que se quiere resolver. Seguidamente la grado de exactitud (lugares decimales) con que se quiere la respuesta. Por último se lee variable por variable.

De cada variable se lee el nombre, la descripción, donde se argumenta aspectos importantes de las variables, que sean fundamentales para su comprensión, el tipo, el tipo de variable puede ser discreta y continua. Dependiendo del tipo se leen los valores que pueden tomar las variables. En caso de que sean continuas se leen dos números, referentes al rango inferior y superior de valores que puede tomar la variable.

Si la variable es discreta se puede introducir el valor de dos formas diferentes. La primera consiste en introducir todos los valores que forman parte del dominio de la variable. La segunda consiste en introducir el rango inferior y superior de valores que puede tomar la variable (estos valores serían enteros). De esta forma a una variable cuyo rango sea de -100 a 100, no habría que introducirle todos los números enteros entre -100 y 100, sino que con introducir el rango inferior y superior sería suficiente.

Las variables discretas y continuas, están caracterizadas por:

- Nombre
- Descripción
- Tipo (discreta o continua)
- Valor mínimo
- Valor máximo
- Conjunto de valores (para variables discretas)

En el caso de las discretas, puede tener los valores mínimos y máximos cuando son enteras o un conjunto de valores del dominio.

❖ Transformación del problema mixto en discreto (Transformar $P \rightarrow P^*$)

Cada variable continua x_i es transformada en un conjunto de variables discretas, de la forma:

$$x_i^* = \{z_i, d_{i1}, d_{i2}, \dots, d_{ij}, \dots, d_{ik}\} \quad (2.1)$$

donde z_i es la parte entera del número x_i , d_{ij} representa el j-ésimo dígitos de la parte decimal del número x_i y k denota el número de cifras de precisión (en este caso, cantidad de números después de la coma).

Ejemplo: Considere el problema: $\min f(x_1, x_2, x_3, x_4, x_5)$, Tabla 2.1. Se quiere resolver este problema para dos cifras de precisión ($k = 2$) de las variables continuas.

Tabla 2.1 Clasificación de las variables

| | | | | | |
|----------------------|----------------|----------------|----------------|----------------|----------------|
| Variabes | X ₁ | X ₂ | X ₃ | X ₄ | X ₅ |
| Clasificación | Discreta | Discreta | Continua | Continua | Continua |

El algoritmo de Colonia de Hormigas tiene que resolver el problema discreto X^* asociado al genoma modificado, tabla 2.2.

Tabla 2.2 Conversión de las variables discretas a continuas

| | | | | | |
|-----------------------------|----------------|----------------|--|--|--|
| Genoma original X | X ₁ | X ₂ | X ₃ | X ₄ | X ₅ |
| Genoma modificado X* | X ₁ | X ₂ | Z ₃ d ₃₁ d ₃₂ | Z ₄ d ₄₁ d ₄₂ | Z ₅ d ₅₁ d ₅₂ |

El problema mixto se convierte en un problema de variables discretas, a partir de la idea anterior. Este procedimiento se realiza a la hora de construir la matriz de feromona. La matriz de feromona contendría por las filas las variables discretas y por las columnas la unión de los dominios de todas las variables discretas. De esta forma un valor a_{ij} representa la “relevancia” de ir de la variable i al valor j , donde siempre se cumple que $j \in \text{dominio}(\text{variable } i)$.

Para esto se construye la matriz de feromona donde la cantidad de filas sería igual al total de variables discretas que requiere el problema mixto una vez que se reduzca a un problema discreto. Esto se obtendría mediante la fórmula:

$$\|M\| = \|x^d\| + \|x^c\|*(ge + 1) \tag{2.2}$$

Donde la cantidad de filas ($\|M\|$) recibiría la suma de la cantidad de variables discretas ($\|x^d\|$), más la cantidad de variables continuas que representarían a la parte entera de las variables continuas ($\|x^c\| * 1$), más la cantidad de variables continuas por el grado de exactitud deseado ($\|x^c\|*ge$), una por cada lugar decimal que se quiera de exactitud.

La cantidad de columnas de la matriz de feromona para cada fila, sería igual a la cantidad de valores del dominio que tiene la variable discreta que corresponda a esa fila. Como ilustra la Tabla 2.3. De esta forma para acceder a los valores de la variable, se indexa el índice de la fila, con el valor del dominio de la variable correspondiente al índice de la tabla. Por ejemplo en la tabla 2.3 para $V_{6,3}$ se tomaría el valor del dominio de la variable X_4 , correspondiente al índice 3.

En caso de las partes decimales d_{ij} se toma el propio índice como valor.

Tabla 2.3 Matriz de feromona

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|-----------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| X ₁ | V _{1,0} | V _{1,1} | V _{1,2} | V _{1,3} | V _{1,4} | V _{1,5} | V _{1,6} | | | | | | |
| X ₂ | V _{2,0} | V _{2,1} | V _{2,2} | V _{2,3} | V _{2,4} | V _{2,5} | V _{2,6} | V _{2,7} | V _{2,8} | V _{2,9} | V _{2,10} | V _{2,11} | V _{2,12} |
| Z ₃ | V _{3,0} | V _{3,1} | V _{3,2} | V _{3,3} | | | | | | | | | |
| d ₃₁ | V _{4,0} | V _{4,1} | V _{4,2} | V _{4,3} | V _{4,4} | V _{4,5} | V _{4,6} | V _{4,7} | V _{4,8} | V _{4,9} | | | |
| d ₃₂ | V _{5,0} | V _{5,1} | V _{5,2} | V _{5,3} | V _{5,4} | V _{5,5} | V _{5,6} | V _{5,7} | V _{5,8} | V _{5,9} | | | |
| Z ₄ | V _{6,0} | V _{6,1} | V _{6,2} | V _{6,3} | V _{6,4} | V _{6,5} | V _{6,6} | V _{6,7} | V _{6,8} | V _{6,9} | V _{6,10} | | |
| d ₄₁ | V _{7,0} | V _{7,1} | V _{7,2} | V _{7,3} | V _{7,4} | V _{7,5} | V _{7,6} | V _{7,7} | V _{7,8} | V _{7,9} | | | |
| d ₄₂ | V _{8,0} | V _{8,1} | V _{8,2} | V _{8,3} | V _{8,4} | V _{8,5} | V _{8,6} | V _{8,7} | V _{8,8} | V _{8,9} | | | |
| Z ₅ | V _{9,0} | V _{9,1} | V _{9,2} | | | | | | | | | | |
| d ₅₁ | V _{10,0} | V _{10,1} | V _{10,2} | V _{10,3} | V _{10,4} | V _{10,5} | V _{10,6} | V _{10,7} | V _{10,8} | V _{10,9} | | | |
| d ₅₂ | V _{11,0} | V _{11,1} | V _{11,2} | V _{11,3} | V _{11,4} | V _{11,5} | V _{11,6} | V _{11,7} | V _{11,8} | V _{11,9} | | | |

donde V_{ij} , corresponde al valor del rastro de feromona. El siguiente pseudocódigo muestra cómo se crea la matriz de feromona:

Procedimiento Construir_matriz ()

- 1 $\|M\| = \|x^d\| + \|x^c\| * (ge + 1); j = 1$
 - 2 para cada variable $x_i \in P$ hacer
 - 3 $\|M_j\| = \|x_i\|$
 - 4 si (continua (x_i))
 - 5 para $k \in (1, \dots, ge)$
 - 6 $j = j + 1$
 - 7 $\|M_j\| = \{1, \dots, 10\}$
 - 8 fin para
 - 9 fin si
 - 10 $j = j + 1$
 - 11 fin para
-

donde M constituye la matriz de feromona y $\|M\|$ la dimensión (número de filas), x^d , las variables discretas y $\|x^d\|$ la dimensión de x^d (cantidad de variables discretas), x^c , las variables continua y $\|x^c\|$ la dimensión de x^c (cantidad de variables continuas), ge, el grado de precisión(cantidad de lugares decimales), j recorre P^* , i recorre P, x_i representa las variables mixtas. P, reprenta al problema mixto (el problema con variables discretas y continuas). P^* , representa el problema discreto (el problema mixto ya reducido a un problema discreto).

❖ **Transformación de la solución discreta a su equivalente mixto.**

Una vez resuelto el problema, se construyen las variables continuas a partir de los valores devueltos por ACO. La mejor solución construida por las hormigas durante el proceso de búsqueda es transformada, teniendo en cuenta el genoma original con las variables mezcladas. La siguiente ecuación muestra esta idea:

$$x_i = z_i + \sum_{j=1}^n d_{ij} * 10^{-j} \tag{2.3}$$

donde z_i es la parte entera de x_i , d_{ij} representa las cifras de precisión de la variable optimizada x_i , mientras que k es el número de cifras de precisión.

Partiendo de la idea anteriormente explicada se realiza la programación. Primero se realizan tantas iteraciones como cantidad de variables tenía el problema mixto (P). En cada iteración se asigna a la variable discreta del problema original (P), la variable discreta en el problema transformado (P*). En caso de las variables continuas, se recorren las próximas k variables discretas del problema transformado (P*) y se construye según la ecuación 2.3 la variable continua que es asignada a la variable discreta correspondiente al problema mixto (P).

Este procedimiento se ejecuta cada vez que se evalúa la función y al final cuando se devuelve la solución, el siguiente pseudocódigo muestra el procedimiento donde ge , el grado de precisión (cantidad de lugares decimales), j recorre P*, i recorre P, x_i representa las variables mixtas donde se devolverá la solución, s_j representa la solución (discreta), devuelta por ACO. P, representa al problema mixto (el problema con variables discretas y continuas). P*, representa el problema discreto (el problema mixto ya reducido a un problema discreto)

Procedimiento Construir_variables ()

```

1  j = 1; i = 1
2  para cada variable  $x_i \in P$  hacer
3       $x_i = s_j$ 
4      si (continua ( $x_i$ ))
5          para  $k \in (1, \dots, ge)$ 
6               $j = j + 1$ 
7               $x_i = x_i + s_j * 10^{-j+i}$ 
8          fin para
9      fin si
10      $j = j + 1; i = i + 1$ 
11 fin para

```

❖ **El vector solución**

El vector solución guarda cada valor obtenido durante la formación de la solución. Al final contendría los valores de cada variable discreta del problema transformado.

Tabla 2.4 Vector solución

| | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|----------|----------|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| l_1 | l_2 | l_3 | l_4 | l_5 | l_6 | l_7 | l_8 | l_9 | l_{10} | l_{11} |

En el vector solución los índices corresponderían a las variables discretas del problema transformado, mientras el valor l , no sería el valor que tomaría la variable, sino un índice que referenciaría ese valor en el dominio de la variable.

El diagrama de la figura 2.1 ilustra donde se realizan los pasos anteriormente explicados en el nuevo procedimiento. La transformación del problema mixto en discreto, se corresponde a la opción **crear matriz**, donde por las filas de esta se tendría las variables discretas, que representan a las variables mixtas del problema. La opción **crear variable** se correspondería con transformación de la solución discreta a su equivalente mixto. Mientras realizar paso, sería equivalente a mover en el pseudocódigo del método ACO Max Min.

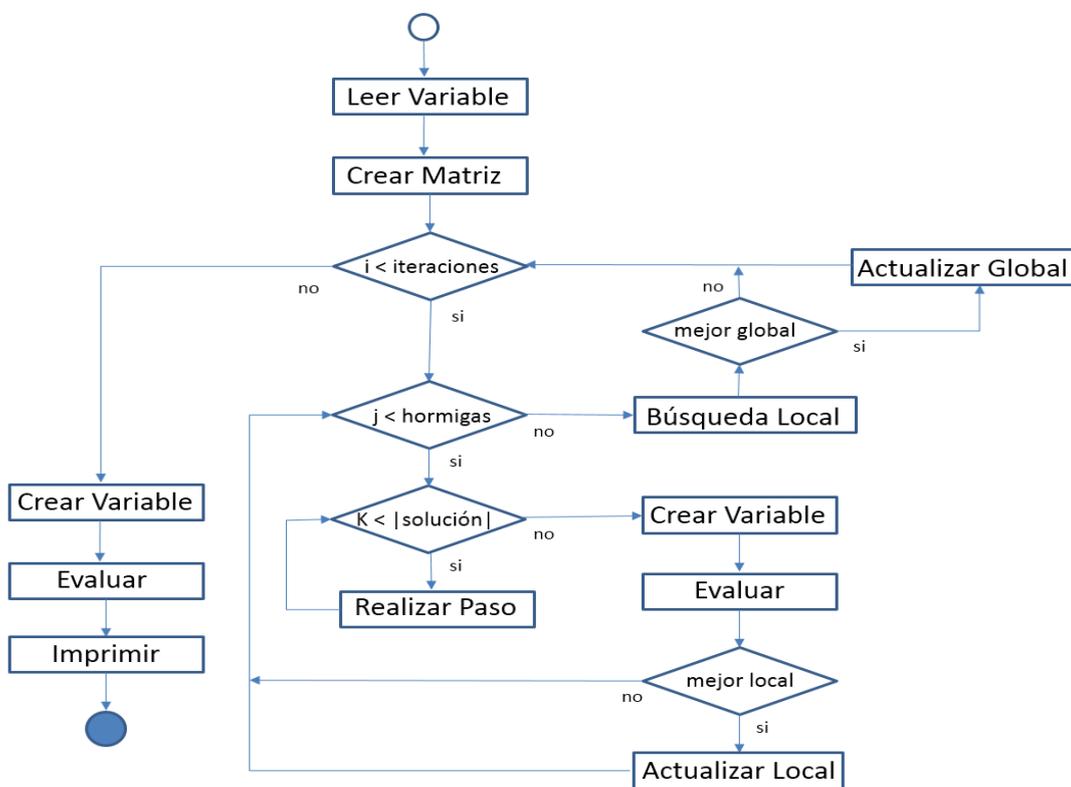


Figura 2.1 Diagrama general del nuevo procedimiento MACO.

❖ **Método ACO_MAX_MIN**

Una vez que el problema es llevado a un problema netamente discreto, se ejecuta el método ACO_MAX_MIN, como muestra el siguiente pseudocódigo. El siguiente pseudocódigo muestra el Método ACO con el que se resuelve el problema:

Procedimiento ejecutar ()

```

1  desde i = 1, hasta cantidad_de_ciclo hacer
2      desde j = 1, hasta cantidad_de_hormigas
3          Inicializar (hormiga_j)
4      fin desde
5      desde k = 1, hasta cantidad_de_estados
6          desde j = 1, hasta cantidad_de_hormigas
7              Mover (hormiga_j, estado_k)
8          fin desde
9      fin desde
10     Guardar_Mejor_Solucion
11     Hacer_Busqueda_Local
12     Actualizar_Matriz_feromona
13     Calcular_Feromona_Max_y_Min
14     Regular_Matriz_Feromona
15     fin desde

```

❖ **Realizar movimiento de Hormiga**

En el problema Inicializar (hormiga_j) coincide con Mover (hormiga_1, estado_k), por ser un caso especial de Mover, solo se explica Mover. En el diagrama 2.1 sería Realizar Paso. El método Mover, calcula para cada valor permisible (un número), mediante la fórmula (2.4). Basado en la formula anterior, luego escoge un valor de forma probabilística, apoyándose en la fórmula anterior. Este valor incide en la selección del próximo nodo.

$$p_{rs}^k = \begin{cases} \frac{\tau_{rs}}{\sum_{u \in N_k^r} \tau_{ru}}, & \text{si } s \in N_k(r) \\ 0, & \text{en otro caso} \end{cases} \quad (2.4)$$

❖ **Guardar Mejor Solución**

Cada vez que se ejecuta el método Guardar_Mejor_Solucion, se recorren las soluciones encontradas por cada hormiga y se transforma la solución discreta a su equivalente mixto. Luego se evalúa y si mejora a la solución global, entonces se actualiza la solución global, con la nueva solución encontrada.

❖ **Actualización de la Matriz de Feromona**

El método Actualizar_Matriz_Feromona actualiza la matriz de feromona, para esto primeramente se recorre la matriz, actualizando el rastro de feromona mediante la fórmula:

$$\tau_{rs} \leftarrow (1 - \rho) * \tau_{rs} \tag{2.5}$$

Esto correspondería al proceso de evaporización. Posteriormente se recorre el mejor camino y actualiza la feromona mediante la fórmula:

$$\tau_{rs} \leftarrow \tau_{rs} + \rho * 1/C(S_{\text{mejor-global}}), \forall a_{rs} \in S_{\text{mejor-global}} \tag{2.6}$$

Este paso incrementa el rastro, en los valores que conforman la mejor solución.

❖ **Cálculo de la Feromona Máxima y Mínima**

El método Cálculo_Feromona_Max_Min recalcula la feromona máxima y mínima mediante las fórmulas:

$$\tau_{\text{max}} \leftarrow (1/(1 - \rho)) * 1/C(S_{\text{mejor-global}}) \tag{2.7}$$

$$\tau_{\text{min}} \leftarrow \tau_{\text{max}}/(2 * \text{tamaño_del_problema}) \tag{2.8}$$

❖ **Regular Matriz Feromona**

El método Regular_Matriz_Feromona, recorre la matriz y si $\tau_{rs} < \tau_{\text{min}}$, $\tau_{rs} \leftarrow \tau_{\text{min}}$, sino si $\tau_{rs} > \tau_{\text{max}}$, $\tau_{rs} \leftarrow \tau_{\text{max}}$, manteniendo en el rango el valor de las feromonas. De esta forma se logra que no sea descartado ningún camino y se evita que el algoritmo quede estancado en un camino determinado.

2.2 Procedimiento de búsqueda local

Los procedimientos de búsqueda local, también llamados de mejora, se basan en explorar el entorno o vecindad (*Variable Neighborhood Search*, VNS) (Matthijs, 2001, Nenad, 2003) de una solución. Utilizan una operación básica llamada movimiento que, aplicada sobre los diferentes elementos de una solución, proporciona las soluciones de su entorno. Formalmente:

Definición: Sea X el conjunto de soluciones del problema combinatorio. Cada solución x tiene un conjunto de soluciones asociadas $N(x) \subseteq X$, que denominaremos entorno de x .

Definición: Dada una solución x , cada solución de su entorno, $x' \in N(x)$, se puede obtener directamente a partir de x mediante una operación llamada movimiento.

Un procedimiento de búsqueda local parte de una solución inicial x_0 , calcula su entorno $N(x_0)$ y escoge una nueva solución x_1 en él. Dicho de otro modo, realiza el movimiento m_1 que aplicado a x_0 da como resultado x_1 . Este proceso se aplica reiteradamente, describiendo una trayectoria en el espacio de soluciones.

Los métodos de búsqueda local suelen ser muy rápidos y proporcionan soluciones que, en promedio, están relativamente cerca del óptimo global del problema. Además, dichos métodos suelen ser el punto de partida en el diseño de algoritmos meta-heurísticos más complejos.

Existen distintos algoritmos de búsqueda local basados en el intercambio de vecindad entre los que se encuentran: el Algoritmo de Lin y Kernighan (Keld, 2000, Kernighan, 1973, McPeak, 2000, Chris, 2001), el Algoritmo de 2-intercambio, de 3-intercambio (Keuthen, 2003) y más general el Algoritmo de k -intercambio (Freisleben, 2000). Estos algoritmos son muy dependientes del problema que resuelven, son muy utilizados en problemas como el TSP. De acuerdo a las peculiaridades de problema de optimización reformulado, estos métodos locales resultan inaplicables de ahí que se desarrolló un nuevo método de búsqueda local.

La idea del nuevo método es muy simple y consiste en tomar la mejor solución encontrada por las hormigas (solución local) e intercambiar valores con el resto de las soluciones encontradas por el resto de las hormigas. Esta operación tiene como objetivo ver si el intercambio mejora la solución; en caso positivo se toma la nueva solución como la mejor solución global.

Para esto el nuevo método recorre todas las soluciones encontradas por todas las hormigas. Para cada hormiga recorre cada valor de la solución e intercambia la posición con la mejor solución encontrada, si al evaluar en la función objetivo el valor es menor que el de la mejor solución encontrada, entonces se toma como la mejor solución. Este proceso se repite hasta recorrer todas las soluciones de todas las hormigas y dentro de las soluciones todas las posiciones.

El siguiente pseudocódigo detalla el procedimiento:

Procedimiento Hacer_Busqueda_Local ()

```

1 desde i = 1, hasta cantidad_de_hormigas
2     tmp = s
3     desde j = 1, hasta cantidad_de_estados
4         tmpj = tij           //INTERCAMBIO
5         si (Evaluar (tmp) < Evaluar (s))
6             s = tmp
7         sino
8             tmpj = sj       //DESINTERCAMBIO
9         fin si
10    fin desde
11 fin desde

```

donde s , es la mejor solución global encontrada, y t_i es la solución encontrada por la hormiga i .

También se mejoró el método de búsqueda local antes explicado, para esto además de realizar una búsqueda local a partir de la mejor solución local encontrada por las hormigas en una iteración, se realizó de nuevo la búsqueda local, pero a partir de la mejor solución global encontrada hasta el momento. Esta variante como se verá en el análisis estadístico 2.5.1, demostró dar mejores soluciones, que la explicada inicialmente. Es importante resaltar que el algoritmo de búsqueda local mejora las soluciones alcanzadas por las hormigas de forma considerable, como se puede apreciar en la tabla 2.6. Sin embargo, este proceso trae consigo que el costo computacional asociado para solucionar el problema se eleve, pero usualmente es el precio a pagar por obtener mejores aproximaciones (soluciones).

Para la implementación de este procedimiento se crearon 3 paquetes, `aco_max_min`, `ecuacion`, `function`. En el primero se tiene todo lo referente al método ACO. Este cuenta con tres clases:

- La clase **ACO_MAX_MIN** además tener el método que se encarga de la ejecución del algoritmo, tienen los métodos de búsqueda local, el método que chequea que los valores estén en el rango.
- La clase **Ant** se encarga de elegir el próximo estado a visitar por cada hormiga, utilizando el método ruleta que ayuda probabilísticamente a escoger el próximo estado.

- La clase **Matriz** presenta lo referente a las operaciones sobre la matriz como calcular feromona máxima y mínima, inicializar la matriz de feromona, actualizarla y también un método que la regula.

El segundo tiene las clases:

- **Ecuacion**, donde en la clase ecuación se guarda la información leída de las variables, también permite adicionar nuevas variables y tiene dos métodos, `construir_matriz` y `construir_variables`. El primero como se ha explicado anteriormente permiten transformar las variables mixtas (continua y discreta) en un conjunto de variables discretas y el segundo permite una vez transformadas las variables predictoras en discretas, retornar los valores como mixto.
- **Funcion**, en esta clase están implementadas las cinco funciones discretas que se emplean en el problema de optimización.

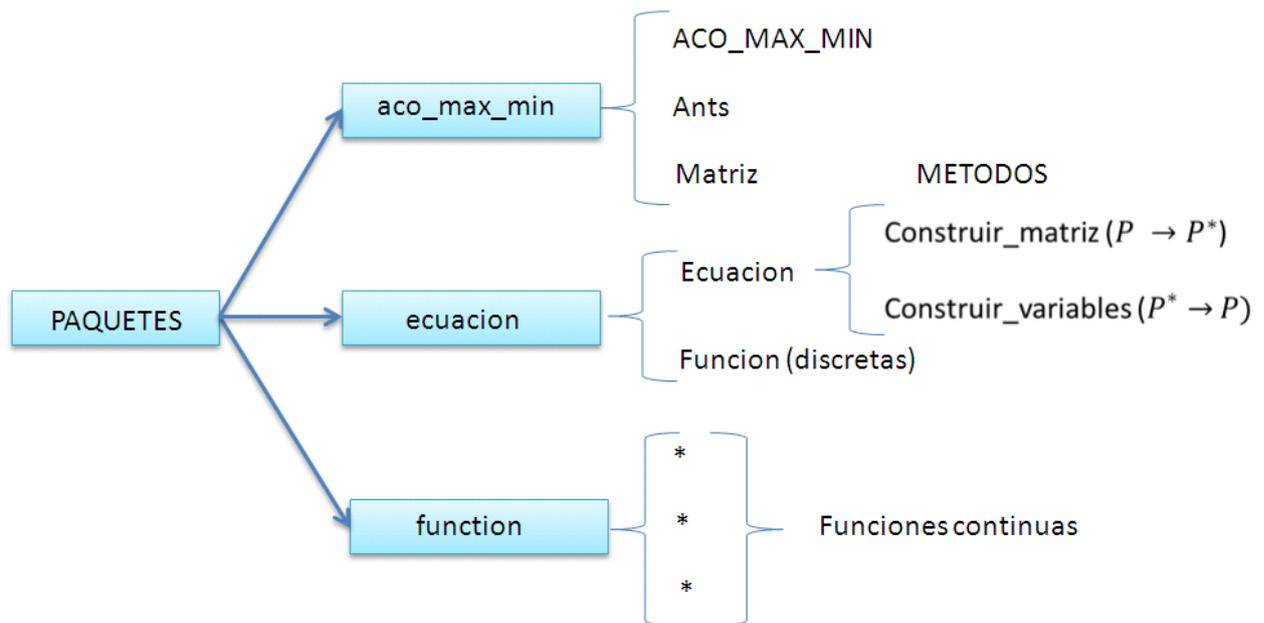


Figura 2.2 Paquetes creados

Mientras que las funciones discretas están incluidas en el paquete `ecuacion`, las funciones continuas cuentan con su propio paquete. Este tiene 25 clases, una para cada función continua, dentro de estas se encuentran las 8 empleadas en la validación, además tiene una clase *benchmark*, en donde se llaman a las funciones para su evaluación.

2.3 Estudio del rendimiento del procedimiento MACO

Para evaluar el rendimiento de MACO, se utilizaron 40 funciones obtenidas a partir de la combinación de 5 funciones discretas tomadas de (Omran, 2007) y 8 continuas tomadas de (Suganthan, 2005). Las funciones discretas tienen 2 y 4 variables, en el rango de -10 y 10. Las funciones continuas tienen dimensión D , $x = [x_1, x_2, \dots, x_D]$, pero se probaron con 10 variables y presentan características como:

- Unimodales
- Ruidosas y rotadas
- Desplazadas y no separables
- Unimodales con el óptimo en los límites
- Multimodales con muchos óptimos locales
- Existencia de valles entre el óptimo local y el óptimo global
- Continua pero diferenciable sólo en un conjunto de puntos

Funciones Discretas

• F1

La primera función discreta tiene dimensión 2, ($F_1 \in \mathbb{R}^2$). Su valor óptimo es $F_1(x^*) = 0$ y se alcanza para $x^* = (1,1)^T$.

$$F_1(x) = (9x_1^2 + 2x_2^2 - 11)^2 + (3x_1 + 4x_2 - 7)^2 \quad (2.9)$$

• F2

La segunda función discreta tiene dimensión 2, ($F_2 \in \mathbb{R}^2$). Su valor óptimo es $F_2(x^*) = 0$ y se alcanza para $x^* = 0$.

$$F_2(x) = -3803.84 - 138.08x_1 - 232.92x_2 + 123.08x_1^2 + 203.64x_2^2 + 182.25x_1x_2 \quad (2.10)$$

• F3

La tercera función discreta tiene dimensión 2, ($F_3 \in \mathbb{R}^2$). Su valor óptimo es $F_3(x^*) = 0$ y se alcanza para $x^* = 0$.

$$F_3(x) = 2x_1^2 + 3x_2^2 + 4x_1x_2 - 6x_1 - 3x_2 \quad (2.11)$$

• F4

La cuarta función discreta tiene dimensión 4, ($F_4 \in \mathbb{R}^4$). Su valor óptimo es $x^* = 0$ y se alcanza para $F_4(x^*) = 0$.

$$F_4(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4 \quad (2.12)$$

• **F5**

La quinta función discreta tiene dimensión D , ($F_5 \in \mathbb{R}^n$). Su valor óptimo es $x^* = 0$ y se alcanza para $F_5(x^*) = 0$.

$$F_5(x) = \sum_{i=1}^D |x_i| \tag{2.13}$$

Funciones Continuas

• **F1**

$$F_1(x) = \left(\sum_{i=1}^D \left(\sum_{j=1}^i z_j \right)^2 \right) * (1 + 0.4|N(0,1)|) - 450 \tag{2.14}$$

donde $z = x - o$, $x = [x_1, x_2, \dots, x_D]$, D : Dimensión, o : Óptimo global

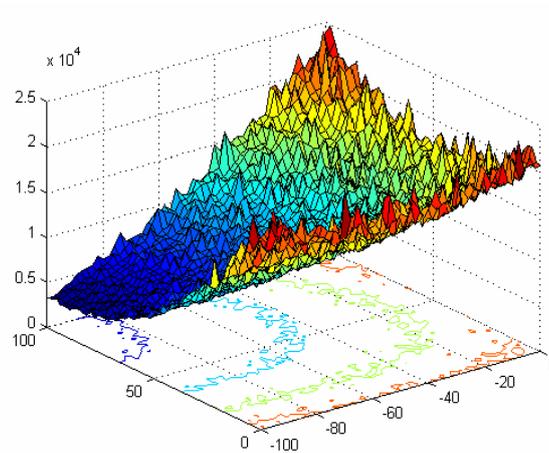


Figura 2.7 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-100,100]^D$ y el óptimo es $F_1(x^*) = -450$

Propiedades:

- Unimodal
- Desplazada
- No separable
- Escalable y ruidosa
- Óptima global sobre la cota

• **F2**

$$F_2(x) = \max\{|A_i x - B_i|\} - 310 \tag{2.15}$$

donde $i = 1, \dots, D$, $x = [x_1, x_2, \dots, x_D]$, D : Dimensión, A es una matriz $D * D$, a_{ij} es un número entero aleatorio en el rango $[-500, 500]$, $\det(A) \neq 0$, A_i es la i -ésima fila de A . $B_i = A_i * o$, o

es un vector $D * 1$, son números aleatorios en el rango $[-100,100]$, $o_i = -100$, para $i = 1, 2, \dots, [D/4]$, $o_i = 100$, para $i = [3D/4], \dots, D$

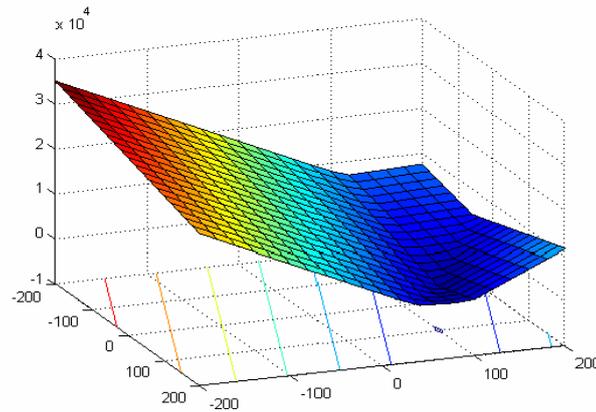


Figura 2.8 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-100,100]^D$ y el óptimo es $F_2(x^*) = -310$.

Propiedades:

- Unimodal
 - Non-separable
 - Escalable
 - Óptima global sobre la cota
- **F3**

$$F_3(x) = \sum_{i=1}^{D-1} (100(z_i^2 - z_{i+1}^2)^2 + (z_i - 1)^2) + 390 \quad (2.16)$$

donde $z = x - o + 1$, $x = [x_1, x_2, \dots, x_D]$, D : Dimensión, o : Óptimo global.

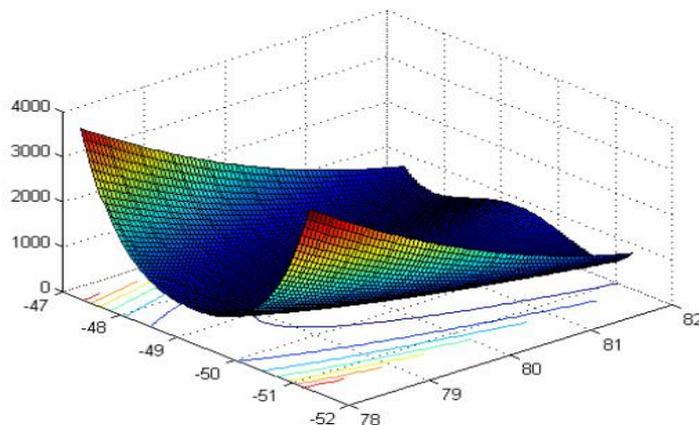


Figura 2.9 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-100,100]^D$ y el óptimo de esta función es $F_3(x^*) = 390F_3(x^*) = 390$.

Propiedades:

- Multi-modal
- Desplazada
- No separable
- Escalable

• **F4**

$$F_4(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D z_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi z_i)\right) + 20 + e - 140 \quad (2.17)$$

donde $z = (x - o) * M$, $x = [x_1, x_2, \dots, x_D]$, D : Dimensión, o : Óptimo global, $o_{2j-1} = -32o_{2j}$ están aleatoriamente distribuido en el rango, para, $j = 1, 2, \dots, \lfloor D/2 \rfloor$, M : Matriz de transformación lineal.

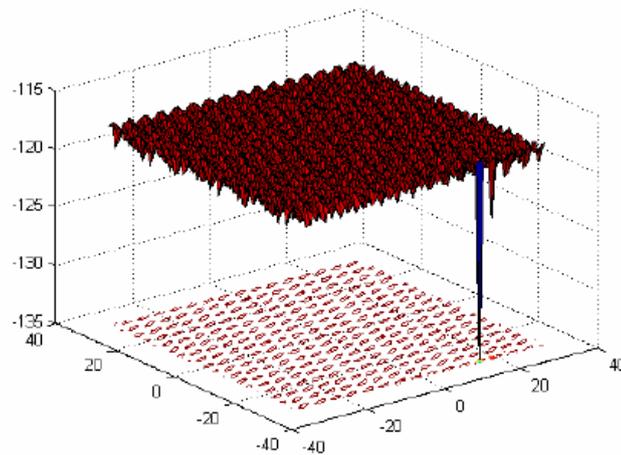


Figura 2.10 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-32, 32]^D$ y el óptimo es $F_4(x^*) = -140$.

Propiedades:

- Multi-modal
- Rotada
- Desplazada
- No separable
- Escalable
- Óptima global sobre la cota

• **F5**

$$F_5(x) = \sum_{i=1}^D (z_i^2 - 10 \cos(2\pi z_i) + 10) - 330 \quad (2.18)$$

donde $z = x - o$, $x = [x_1, x_2, \dots, x_D]$, D : Dimensión, o : Óptimo global.

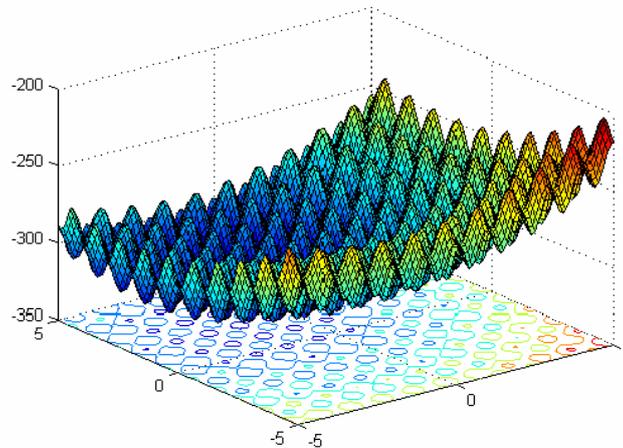


Figura 2.11 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-5,5]^D$ y el óptimo es $F_5(x^*) = -330$.

Propiedades:

- Multi-modal
- Desplazada
- Separable
- Escalable
- Número enorme de óptimo local

• **F6**

$$F_6(x) = \sum_{i=1}^D (z_i^2 - 10 \cos(2\pi z_i) + 10) - 330 \quad (2.19)$$

donde $z = (x - o) * M$, $x = [x_1, x_2, \dots, x_D]$, D : Dimensión, o : Óptimo global, M : Matriz de transformación lineal.

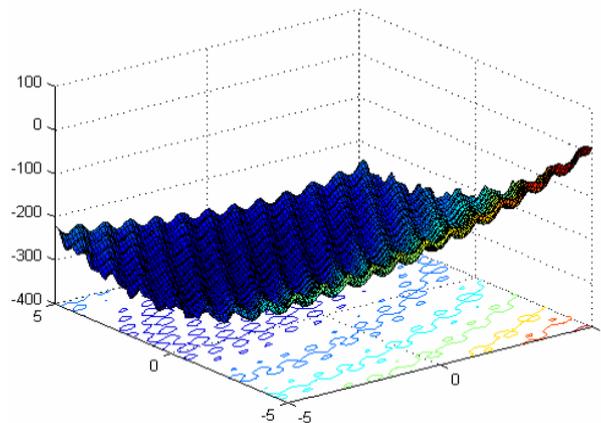


Figura 2.12 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-5,5]^D$ y el óptimo es $F_6(x^*) = -330$.

Propiedades:

- Multi-modal
- Desplazada
- Rotada
- No separable
- Escalable
- Número enorme de óptimo local

• **F7:**

$$F_7(x) = \sum_{i=1}^D \left(\sum_{k=0}^{kmax} \left[a^k \cos \left(2\pi b^k (z_i + 0.5) \right) \right] \right) - D \sum_{k=0}^{kmax} \left[a^k \cos(2\pi b^k * 0.5) \right] + 90 \quad (2.20)$$

donde $a = 0.5$, $b = 3$, $kmax = 20$, $z = (x - o) * M$, $x = [x_1, x_2, \dots, x_D]$, D : Dimensión, o : Óptimo global, M : Matriz de transformación lineal.

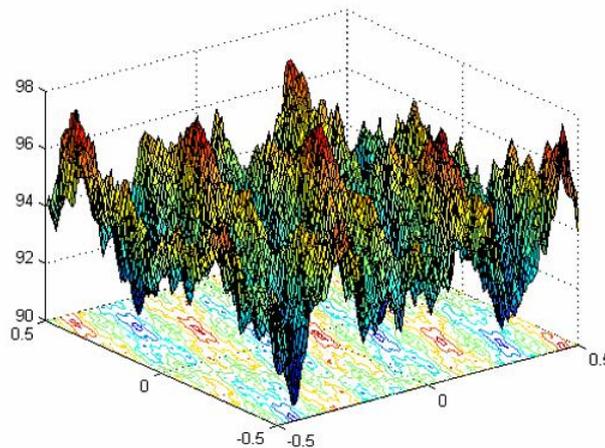


Figura 2.13 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-0.5,0.5]^D$ y el óptimo es $F_7(x^*) = 90$.

Propiedades:

- Multi-modal
- Desplazada
- Rotada
- No separable
- Escalable
- Continua pero diferenciable sólo en un punto

• **F8**

$$F_8(x) = \sum_{i=1}^D (A_i - B_i(x))^2 - 460, \tag{2.21}$$

donde $A_i = \sum_{j=1}^D (a_{ij} \sin \alpha_j + b_{ij} \cos \alpha_j)$, $B_i(x) = \sum_{j=1}^D (a_{ij} \sin x_j + b_{ij} \cos x_j)$, para $i = 1, \dots, D$, D : Dimensión, A, B : son dos matriz $D * D$, a_{ij}, b_{ij} son números enteros aleatorios en el rango $[-100,100]$, $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_D]$, α_{ij} son números enteros aleatorios en el rango $[-\pi, \pi]$.

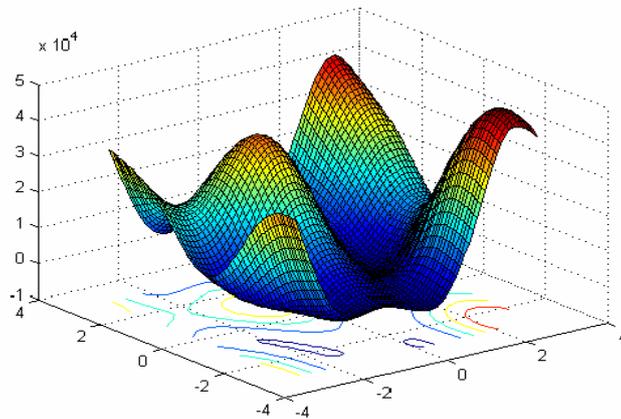


Figura 2.14 Gráfico 3-D para la función en 2-D

El rango de valor de las variables es $x \in [-\pi, \pi]^D$ y el óptimo es $F_8(x^*) = -460$

Propiedades:

- Multi-modal, Rotada, Escalable
- Desplazada, No separable
- Continua pero diferenciable sólo en un punto

2.4 Técnicas estadísticas para el análisis de los resultados

El algoritmo MACO se compara con Recocido Simulado y Algoritmo Genéticos, por ser un enfoque clásico para tratar problemas mixtos. Para esto se tomaron como parámetros los ilustrados por la tabla 2.5. Se hicieron para cada función y cada algoritmo 10 corridas y se promediaron los resultados obteniéndose los valores de la Tabla 2.6.

Tabla 2.5 Parámetros de los diferentes algoritmos

| | Agentes | Generaciones | Adicionales |
|------------|---------------------------------|-----------------------------|-------------|
| AG | 20 | 100 | defecto |
| ACO | 20 | 100 | defecto |
| RS | Cada 20 iteraciones cambio de T | 2000 (100 para una misma T) | defecto |

Tabla 2.6 Resultados de las corridas de los diferentes algoritmos.

| | Mínimo | MACO CON BLOCAL (1) | | AG | | SA | |
|-----|----------|---------------------|---------|-----------|-----------|-----------|-----------|
| | | Resultado | Error | Resultado | Error | Resultado | Error |
| F11 | -450,00 | 3407,31 | 3911,06 | 6868,89 | 14637,79 | 23923,81 | 24472,95 |
| F12 | -310,00 | 1545,65 | 1855,65 | 2687,00 | 5994,18 | 15419,21 | 15729,21 |
| F13 | 390,00 | 4161,99 | 3771,99 | 137140,65 | 273501,30 | 1,28E+10 | 1,28E+10 |
| F14 | -140,00 | -119,71 | 20,28 | -119,28 | 41,43 | -118,46 | 21,53 |
| F15 | -330,00 | -328,99 | 1,00 | -322,62 | 14,75 | -160,04 | 169,95 |
| F16 | -330,00 | -296,36 | 33,63 | -265,20 | 129,59 | -120,86 | 209,13 |
| F17 | 90,00 | 97,16 | 7,16 | 100,36 | 20,72 | 107,75 | 17,75 |
| F18 | -460,00 | 38,84 | 498,84 | 3711,69 | 8343,39 | 101194,49 | 101654,49 |
| F21 | -4283,12 | -546,13 | 3841,36 | 1915,70 | 12397,65 | 11575,43 | 18420,01 |
| F22 | -4143,12 | -1879,78 | 2263,33 | 276,53 | 8839,31 | 10995,90 | 15139,02 |
| F23 | -3443,12 | -687,84 | 2755,27 | 191896,27 | 390678,79 | 1,02E+09 | 1,02E+09 |
| F24 | -3973,12 | -3952,79 | 20,32 | -3949,47 | 47,28 | -3951,67 | 21,44 |
| F25 | -4163,12 | -4162,01 | 1,10 | -4149,14 | 27,94 | -4015,45 | 147,66 |
| F26 | -4163,12 | -4131,08 | 32,03 | -4086,07 | 154,08 | -3947,52 | 215,59 |
| F27 | -3743,12 | -3734,74 | 8,37 | -3723,03 | 40,17 | -3725,24 | 17,87 |
| F28 | -4293,12 | -2333,30 | 1959,81 | 1697,74 | 11981,72 | 86444,26 | 90737,38 |
| F31 | -450,00 | 1862,15 | 2479,48 | 5818,00 | 12520,01 | 13905,66 | 15601,09 |
| F32 | -310,00 | 1494,46 | 1804,46 | 2451,80 | 5523,61 | 11746,49 | 12056,49 |
| F33 | 390,00 | 4212,82 | 3822,82 | 129870,45 | 258960,90 | 1,39E+09 | 1,39E+09 |
| F34 | -140,00 | -125,72 | 14,27 | -124,64 | 30,71 | -124,66 | 15,33 |
| F35 | -330,00 | -335,44 | 5,44 | -326,72 | 6,94 | -220,82 | 109,17 |
| F36 | -330,00 | -304,60 | 25,39 | -264,32 | 131,34 | -175,86 | 154,13 |
| F37 | 90,00 | 90,96 | 1,03 | 94,17 | 8,34 | 99,27 | 9,27 |
| F38 | -460,00 | 157,91 | 617,91 | 4443,85 | 9807,70 | 83778,08 | 84238,08 |
| F41 | -450,00 | 3495,77 | 4064,94 | 4988,68 | 10877,37 | 18858,04 | 20738,37 |
| F42 | -310,00 | 1443,45 | 1753,45 | 4573,74 | 9767,29 | 16409,65 | 16719,65 |
| F43 | 390,00 | 2674,88 | 2284,88 | 118635,95 | 236491,91 | 7,51E+08 | 7,51E+08 |
| F44 | -140,00 | -118,53 | 21,46 | -79,97 | 120,05 | -109,34 | 30,65 |
| F45 | -330,00 | -324,76 | 5,23 | -286,80 | 86,38 | -105,55 | 224,44 |
| F46 | -330,00 | -276,84 | 53,15 | -201,35 | 257,29 | -23,78 | 306,21 |
| F47 | 90,00 | 100,02 | 10,02 | 135,00 | 90,00 | 136,34 | 46,34 |
| F48 | -460,00 | 254,13 | 714,13 | 7168,63 | 15257,26 | 88464,31 | 88924,31 |
| F51 | -450,00 | 3389,18 | 3346,53 | 3195,26 | 7290,52 | 12628,83 | 12960,05 |
| F52 | -310,00 | 1459,22 | 1769,22 | 3359,41 | 7338,82 | 12267,20 | 12577,20 |
| F53 | 390,00 | 2744,32 | 2354,32 | 204091,86 | 407403,72 | 1,44E+09 | 1,44E+09 |
| F54 | -140,00 | -119,66 | 20,33 | -119,20 | 41,59 | -116,50 | 23,49 |
| F55 | -330,00 | -329,26 | 0,73 | -320,85 | 18,28 | -220,65 | 109,34 |
| F56 | -330,00 | -299,82 | 30,17 | -267,83 | 124,33 | -194,18 | 135,81 |
| F57 | 90,00 | 97,46 | 7,46 | 100,00 | 20,01 | 109,42 | 19,42 |
| F58 | -460,00 | 595,24 | 1055,24 | 6295,22 | 13510,44 | 71673,38 | 72133,38 |

La Tabla 2.6 muestra los resultados obtenidos por los métodos ACO MIXTO (MACO), Algoritmo Genético (AG) y Recocido Simulado (SA).

Tabla 2.7 Resultados de las corridas de MACO con diversas variantes de búsqueda local.

| | Mínimo | MACO CON BLOCAL (2) | | MACO CON BLOCAL (1) | | MACO SIN BLOCAL | |
|-----|----------|---------------------|---------|---------------------|---------|-----------------|----------|
| | | Resultado | Error | Resultado | Error | Resultado | Error |
| F11 | -450,00 | -136,98 | 310,00 | 3407,31 | 3911,06 | 20503,31 | 22200,19 |
| F12 | -310,00 | 367,40 | 677,40 | 1545,65 | 1855,65 | 14084,61 | 14394,61 |
| F13 | 390,00 | 542,79 | 152,79 | 4161,99 | 3771,99 | 7,89E+08 | 7,89E+08 |
| F14 | -140,00 | -119,74 | 20,25 | -119,71 | 20,28 | -118,62 | 21,37 |
| F15 | -330,00 | -329,42 | 0,57 | -328,99 | 1,00 | -162,16 | 167,83 |
| F16 | -330,00 | -299,54 | 30,45 | -296,36 | 33,63 | -125,48 | 204,51 |
| F17 | 90,00 | 96,48 | 6,48 | 97,16 | 7,16 | 102,22 | 12,22 |
| F18 | -460,00 | 238,68 | 698,68 | 38,84 | 498,84 | 30882,02 | 31342,02 |
| F21 | -4283,12 | -3888,18 | 370,84 | -546,13 | 3841,36 | 15617,32 | 17931,98 |
| F22 | -4143,12 | -2510,96 | 1632,15 | -1879,78 | 2263,33 | 11529,61 | 15672,73 |
| F23 | -3443,12 | -3030,09 | 413,02 | -687,84 | 2755,27 | 8,69E+08 | 8,69E+08 |
| F24 | -3973,12 | -3952,86 | 20,25 | -3952,79 | 20,32 | -3951,72 | 21,39 |
| F25 | -4163,12 | -4162,54 | 0,57 | -4162,01 | 1,10 | -3986,24 | 176,87 |
| F26 | -4163,12 | -4124,65 | 38,46 | -4131,08 | 32,03 | -3815,52 | 247,59 |
| F27 | -3743,12 | -3735,98 | 7,13 | -3734,74 | 8,37 | -3726,29 | 16,82 |
| F28 | -4293,12 | -3092,04 | 1201,07 | -2333,30 | 1959,81 | 16408,29 | 20701,41 |
| F31 | -450,00 | -196,41 | 283,92 | 1862,15 | 2479,48 | 11912,79 | 11279,15 |
| F32 | -310,00 | 409,37 | 719,32 | 1494,46 | 1804,46 | 11901,16 | 12211,16 |
| F33 | 390,00 | 565,56 | 175,56 | 4212,82 | 3822,82 | 7,32E+08 | 7,32E+08 |
| F34 | -140,00 | -125,75 | 14,24 | -125,72 | 14,27 | -124,78 | 15,21 |
| F35 | -330,00 | -335,71 | 5,71 | -335,44 | 5,44 | 212,84 | 117,15 |
| F36 | -330,00 | -309,11 | 20,88 | -304,60 | 25,39 | -168,45 | 161,54 |
| F37 | 90,00 | 91,19 | 1,50 | 90,96 | 1,03 | 95,89 | 5,89 |
| F38 | -460,00 | 812,48 | 1272,48 | 157,91 | 617,91 | 18906,87 | 19366,87 |
| F41 | -450,00 | 86,02 | 455,77 | 3495,77 | 4064,94 | 19686,83 | 19916,96 |
| F42 | -310,00 | 199,42 | 509,52 | 1443,45 | 1753,45 | 15664,48 | 15974,48 |
| F43 | 390,00 | 654,88 | 264,88 | 2674,88 | 2284,88 | 8,42E+08 | 8,42E+08 |
| F44 | -140,00 | -119,55 | 20,44 | -118,53 | 21,46 | -97,74 | 42,25 |
| F45 | -330,00 | -327,56 | 2,43 | -324,76 | 5,23 | -114,19 | 215,80 |
| F46 | -330,00 | -309,04 | 20,95 | -276,84 | 53,15 | -86,50 | 243,49 |
| F47 | 90,00 | 99,21 | 9,21 | 100,02 | 10,02 | 108,13 | 18,13 |
| F48 | -460,00 | 234,64 | 694,64 | 254,13 | 714,13 | 23463,67 | 29363,67 |
| F51 | -450,00 | -253,79 | 174,68 | 3389,18 | 3346,53 | 14003,45 | 13225,70 |
| F52 | -310,00 | 396,06 | 706,06 | 1459,22 | 1769,22 | 12181,33 | 12491,33 |
| F53 | 390,00 | 534,12 | 144,12 | 2744,32 | 2354,32 | 8,77E+08 | 8,77E+08 |
| F54 | -140,00 | -119,64 | 20,35 | -119,66 | 20,33 | -115,65 | 24,34 |
| F55 | -330,00 | -329,63 | 0,36 | -329,26 | 0,73 | -220,12 | 109,87 |
| F56 | -330,00 | -301,61 | 28,38 | -299,82 | 30,17 | -190,19 | 139,80 |
| F57 | 90,00 | 96,08 | 6,08 | 97,46 | 7,46 | 105,10 | 15,10 |
| F58 | -460,00 | 33,65 | 493,65 | 595,24 | 1055,24 | 12828,30 | 13288,30 |

La Tabla 2.7 muestra los resultados obtenidos por los métodos MACO con búsqueda local (2), MACO con búsqueda local (1) y MACO sin búsqueda local.

MACO con búsqueda local (1), solo aplica la búsqueda local a la mejor solución encontrada por las hormigas en una iteración. Mientras que MACO con búsqueda local (2), aplica la búsqueda local no solo a la mejor solución encontrada por la hormiga en una iteración, sino también a la mejor solución global encontrada hasta el momento. Si se observan los valores que se muestran en la Tabla 2.6, se puede apreciar a simple vista que los resultados que aporta MACO con búsqueda local, tanto 1 como 2, mejoran los resultados obtenidos por los algoritmos AG y Recocido Simulado en cuanto a calidad de soluciones.

Un análisis a simple vista no sería suficiente para probar lo antes dicho, por lo que se utilizaron técnicas estadísticas para validar los resultados y no dejarlo solamente a la apreciación, este análisis se realizó utilizando el paquete estadístico SPSS versión 18.0, a partir de los datos de la tabla 2.6 y la tabla 2.7.

2.4.1 Análisis estadístico

En un fichero compatible con el SPSS se analiza la existencia o no de diferencias significativas entre ACO-Mixto (MACO), AG y Recocido Simulado, además de varias variantes del método de búsqueda local; ACO-Mixto sin búsqueda local y ACO-Mixto con búsqueda local sobre mejor local (1) y ACO-Mixto con búsqueda local sobre mejor local y mejor global (2), y se determina el algoritmo que ofrece mejores resultados.

Las comparaciones por pares son el tipo más simple de pruebas estadísticas que un investigador puede aplicar en el marco de un estudio experimental. Tales pruebas se dirigen para comparar el rendimiento de dos algoritmos cuando se aplica a un conjunto común de problemas.

La primera prueba consiste en medir el posible ajuste de la muestra a la distribución normal mediante el test de Kolmogorov-Smirnov. Como es posible observar a través de la Tabla 2.8, la significación menor que 0.05 sugiere rechazar la hipótesis de normalidad, por lo que no se deben utilizar pruebas paramétricas.

Precisamente el test de Friedman es una prueba de varianza no paramétrica. Se trata de una prueba de comparaciones múltiples que tiene como objetivo detectar diferencias significativas entre el comportamiento de dos o más algoritmos.

Tabla 2.8 Prueba de Normalidad usando el test de Kolmogorov-Smirnov.

| | | Prueba de Kolmogorov-Smirnov para una muestra | | | | |
|---------------------------|-------------------|---|--------------------------------|-------------|-------------|-------------------------|
| | | ACO MIXTO CON BLOCAL (2) | ACO MIXTO CON BLOCAL (1) | AG | SA | ACO MIXTO SIN BLOCAL |
| N | | 40 | 40 | 40 | 40 | 40 |
| Parámetros | Media | 290,631000 | 1181,081500 | 4,306337E4 | 4,340651E8 | 1,027318E8 |
| normales ^{a,b} | Desviación típica | 397,7558421 | 1,4273221E3 | 1,0669109E5 | 2,0311109E9 | 2,7593414E8 |
| Diferencias | Absoluta | ,237 | ,285 | ,478 | ,460 | ,520 |
| más extremas | Positiva | ,237 | ,285 | ,478 | ,460 | ,520 |
| | Negativa | -,233 | -,204 | -,343 | -,415 | -,355 |
| Z de Kolmogorov-Smirnov | | 1,499 | 1,804 | 3,022 | 2,907 | 3,290 |
| Sig. asintót. (bilateral) | | ,022 | ,003 | ,000 | ,000 | ,000 |

a. La distribución de contraste es la Normal.

b. Se han calculado a partir de los datos.

La hipótesis nula para el test de Friedman afirma la igualdad de medianas entre las poblaciones. La hipótesis alternativa se define como la negación de la hipótesis nula, por lo que no es direccional (Joaquín Derrac 2011).

Tabla 2.9 Rangos medios.

| Rangos | | Rango promedio |
|--------------------------|--|-------------------|
| ACO MIXTO CON BLOCAL (2) | | 1,15 |
| ACO MIXTO CON BLOCAL (1) | | 1,85 |
| AG | | 3,55 |
| SA | | 4,47 |
| ACO MIXTO SIN BLOCAL | | 3,98 |

Tabla 2.10 Significación según test de Friedman.

| Estadísticos de contraste ^a | |
|--|---------|
| N | 40 |
| Chi-cuadrado | 130,780 |
| gl | 4 |
| Sig. asintót. | ,000 |

a. Prueba de Friedman

Los resultados de la Tabla 2.10 basados en la significación del test (menor que 0.05) rechazan la hipótesis principal, por lo que se puede concluir la existencia de diferencias significativas entre los grupos. Además, como muestra la Tabla 2.9, esta prueba ofrece los rangos medios (Mean Rank) coincidiendo el mayor valor con los dos procedimientos MACO con búsqueda local, seguido de Algoritmo Genéticos, Recocido Simulado, y finalmente MACO sin búsqueda local.

La existencia de diferencias significativas de manera grupal que indica Friedman no es suficiente para determinar que MACO con búsqueda local es el más eficiente, también es necesario que exista diferencias entre MACO y los demás clasificadores en comparaciones dos a dos.

El test de Wilcoxon es un procedimiento no paramétrico empleado en situaciones de prueba de hipótesis, que implica un diseño con dos muestras. Se trata de una prueba de pares que tiene como objetivo detectar diferencias significativas entre las dos medias, es decir, el comportamiento de dos algoritmos (Joaquín Derrac 2011).

Tabla 2.11 Significación Test de Wilcoxon.

| Estadísticos de contraste ^c | | | | | | | | | | |
|--|--|----------------------------------|----------------------------------|--|----------------------------------|----------------------------------|--|---------------------|---------------------------|---------------------------|
| | ACO MIXTO CON BLOCAL (1) - ACO MIXTO CON BLOCAL (2) | AG - ACO MIXTO CON BLOCAL (2) | SA - ACO MIXTO CON BLOCAL (2) | ACO MIXTO SIN BLOCAL - ACO MIXTO CON BLOCAL (2) | AG - ACO MIXTO CON BLOCAL (1) | SA - ACO MIXTO CON BLOCAL (1) | ACO MIXTO SIN BLOCAL - ACO MIXTO CON BLOCAL (1) | SA - AG | ACO MIXTO SIN BLOCAL - AG | ACO MIXTO SIN BLOCAL - SA |
| Z | -4,436 ^a | -5,511 ^a | -5,511 ^a | -5,511 ^a | -5,511 ^a | -5,511 ^a | -5,511 ^a | -4,624 ^a | -3,911 ^a | -2,769 ^b |
| Sig. asintót. (bilateral) | ,000 | ,000 | ,000 | ,000 | ,000 | ,000 | ,000 | ,000 | ,000 | ,006 |

a. Basado en los rangos negativos.

b. Basado en los rangos positivos.

c. Prueba de los rangos con signo de Wilcoxon

A través del test de Wilcoxon para muestras pareadas que se muestra en Tabla 2.11 es posible analizar si MACO es el más eficiente. En este caso, la significación calculada para los pares ACO Mixto con búsqueda local-AG y ACO Mixto con búsqueda local-Recocido Simulado es menor que 0.05 por lo que se rechaza la hipótesis principal fundamental conservativa y se confirma la existencia de diferencias significativas entre estos pares siempre a favor de MACO.

La búsqueda local tiende a mejorar significativamente los resultados obtenidos por el nuevo procedimiento basado en colonias de hormigas. La variante 2 de MACO con búsqueda local, que aplica la búsqueda local no solo a la mejor solución encontrada por la hormiga en una iteración, sino también a la mejor solución global encontrada hasta el momento, mejora a MACO con búsqueda local (1), que solo aplica la búsqueda local a la mejor solución encontrada por las hormigas en una iteración. Estas pruebas comparativas en su conjunto, permiten afirmar confiabilmente que ACO Mixto resulta ser el mejor de los métodos. Además de evidenciar que los método de búsqueda local, proporcionan una mejora al nuevo procedimiento.

2.5 Conclusiones parciales

El nuevo procedimiento permite la resolución de problemas de optimización mixtos utilizando ACO. Se basa fundamentalmente en transformar el problema mixto a uno totalmente discreto, resolver este a través de ACO y transformar los valores encontrados a su equivalente en el problema mixto, para su evaluación en la función objetivo y a la hora de devolver la solución.

El método obtiene buenos resultados para funciones complejas (multimodales, ruido, óptimos globales en la frontera, pobremente diferenciables, etc. Además, mediante la realización de procedimientos estadísticos no paramétricos, se demostró que MACO obtiene mejores resultados que métodos muy usados como Algoritmos Genéticos y Recocido Simulado.

Las soluciones encontradas con la aplicación del nuevo método de búsqueda local obtienen mejores resultados que los presentados por el modelo MACO sin búsqueda local. La introducción del método búsqueda local, aunque eleva el costo computacional, mejora ampliamente las soluciones alcanzadas por las hormigas, como se pudo apreciar en la tabla 2.6. Los resultados obtenidos por la nueva técnica MACO superan a métodos clásicos de resolución de problemas mixtos como AG y RS, convirtiendo al procedimiento en una variante de fuerza a la hora de resolver problemas de optimización con variables mixtas.

CAPÍTULO 3. VALIDACIÓN DEL MÉTODO PROPUESTO ANTE UN EJEMPLO REAL MODELADO EN EL CAMPO DE LOS INTERCAMBIADORES DE CALOR DE TUBO Y CORAZA.

Los procesos de transferencia de calor son, sin duda alguna, uno de los problemas más importantes a resolver en el campo de la Ingeniería Química. A continuación se presentan los conceptos que describen el funcionamiento básico de los intercambiadores de calor, sus principales características, cálculo y diseño.

Además se optimizarán los parámetros influyentes de los intercambiadores de calor de tubo y coraza (STHEs, por sus siglas en inglés) desde el punto de vista económico, y se demostrará la eficacia del nuevo procedimiento (MACO), en la optimización del diseño de STHEs desde el punto de vista económico. La capacidad de la técnica de MACO se demuestra utilizando diferentes casos de estudios.

3.1. Intercambiadores de calor de Tubo y Coraza

En los grandes procesos industriales, no es raro que sea necesario transferir cantidades relativamente grandes de energía, entre el sistema y el medio ambiente o entre distintas partes de un sistema dado (WELTY, 1996). Para lograr estos alcances se utilizan principalmente los intercambiadores de calor, los cuales son aparatos que transfieren energía desde un fluido de alta temperatura hacia un fluido de baja temperatura, con ambos fluidos en movimiento a través del mismo.

Los intercambiadores de calor de tubos y coraza están compuestos por tubos cilíndricos, montados dentro de una coraza también cilíndrica, con el eje de los tubos paralelos al eje de la coraza. Un fluido circula por dentro de los tubos, y el otro por el exterior (fluido del lado de la coraza). Son el tipo de intercambiadores de calor más usado en la industria.

Los componentes principales del intercambiador de calor se muestran en la figura 3.1, los cuales son: Coraza (1), es cilíndrica, horizontal, la cual contiene al arreglo de tubos (2), los tubos son lisos y están unidos a los espejos (3), los espejos son placas redondas barrenadas, las cuales soportan a los tubos y a las mamparas (4), las mamparas son también placas delgadas barrenadas, las cuales sirven para soportar a los tubos y dirigir al flujo en la coraza. El ensamble del arreglo de tubos y mamparas se realiza por medio de barras de soporte y espaciadoras, las cuales dan

soporte a las mamparas y mantienen la distancia entre ellas evitando un mal desempeño térmico del intercambiador de calor o daño de los tubos por pandeo o vibración. El fluido que fluye a través de los tubos es dirigido por los cabezales (5), el de cabezal está constituido por canal, tapa y brida, la tapa generalmente es una pieza fundida, cuya función es cerrar al cabezal de entrada y salida, las bridas son placas circulares, las cuales se unen a los cabezales y a los espejos fijos. Las boquillas (6) permiten la entrada y salida del fluido en los tubos y en la coraza, estas son el enlace con los instrumentos de medición.

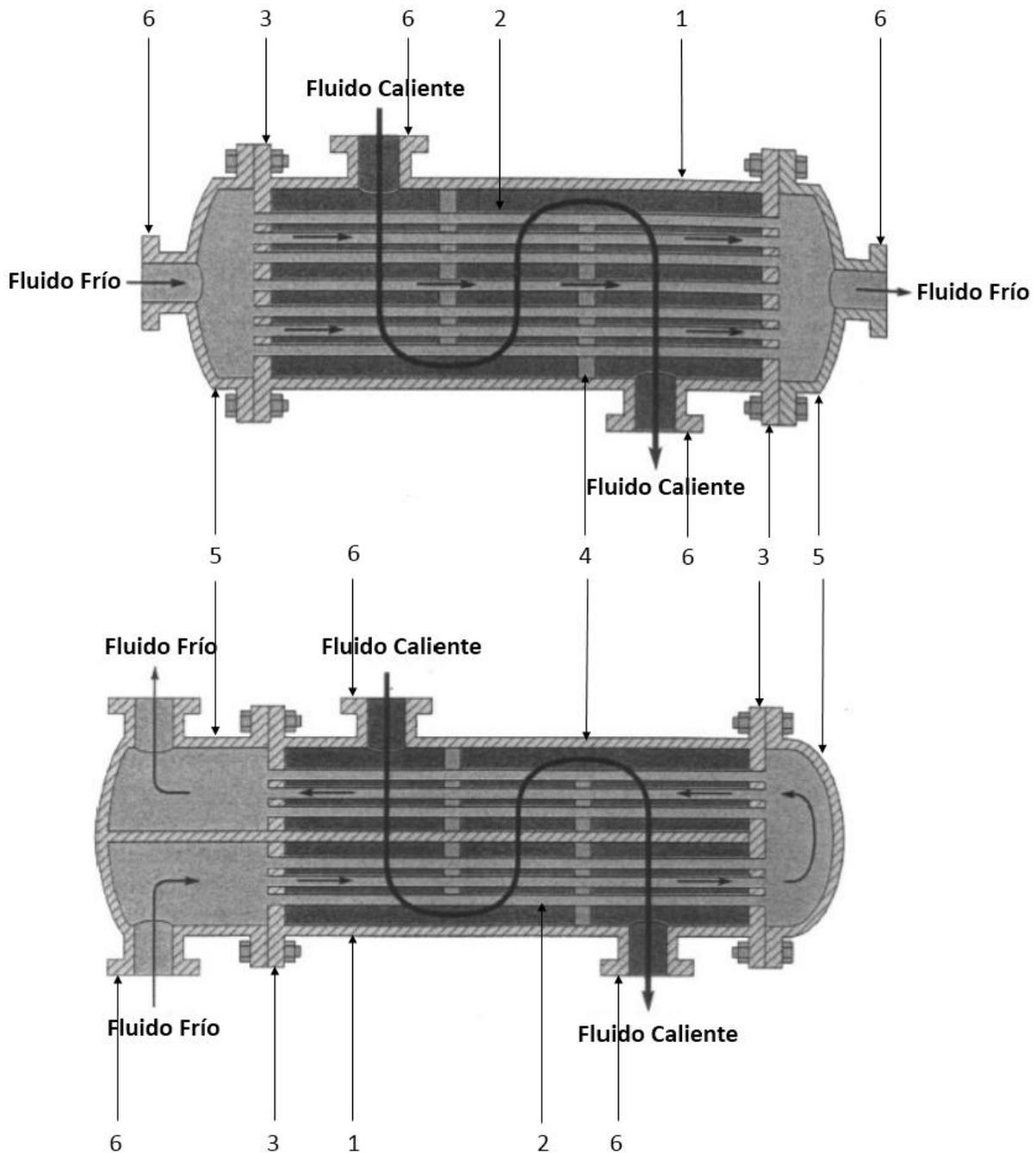


Figura 3.1 Componentes principales del intercambiador de calor de tubo y coraza

Como se puede observar, el fluido que ha de circular en el interior de los tubos ingresa por el cabezal derecho y se distribuye por los orificios de la placa en el haz de tubos. El fluido de la coraza, en cambio, circula por el exterior del haz de tubos, siguiendo una trayectoria tortuosa por el efecto de las pantallas (baffles o mamparas).

3.2 Métodos para la optimización de los intercambiadores de calor de Tubo y Coraza.

La dificultad para integrar en las correlaciones obtenidas en los bancos de tubos y el flujo generado en la carcasa de un intercambiador con deflectores impulsó el desarrollo de "métodos integrales" para el cálculo de la transferencia de calor y pérdida de carga en el lado carcasa.

- Kern (ancestral)
- Método de Tinker (años 50)
- Bell-Delaware (años 60)
- Palen- Taborek (años 60-70)
- Método de Wills and Jhonston.

Las técnicas convencionales de optimización se basan en teorías bien establecidas y funcionan perfectamente bien siempre que sean aplicables. Sin embargo, hay ciertas limitaciones en los métodos antes mencionados.

Este enfoque lleva mucho tiempo, y no garantiza una solución óptima. Las técnicas convencionales de optimización son el mejor medio cuando están disponibles suficientes datos, modelos matemáticos y la información del gradiente y si el atrapamiento en óptimos locales se puede evitar. Todas estas características de estos métodos tradicionales los hacen inaplicable a muchos problemas de la vida real, donde no hay información suficiente sobre modelo matemático del sistema, la dependencia de los parámetros y otra información similar.

Sin embargo, cuando la anterior no se puede asegurar, que es el caso de muchos problemas verdaderos de optimización de la vida, se tienen que buscar técnicas no convencionales de optimización.

En los últimos años los algoritmos poblacionales y de inteligencia colectiva de optimización han ganado popularidad y son utilizados en varias áreas del conocimiento, debido a su relativa simplicidad y costo computacional. Estos algoritmos se están empleando en la optimización de

intercambiadores de calor de Tubo y Coraza, debido a que son capaces de encontrar buenas soluciones con un mínimo de esfuerzo; convergiendo hacia un único óptimo global.

Hay muchos estudios previos sobre la optimización de intercambiadores de calor. Varios investigadores han utilizado diferentes técnicas de optimización considerando diferentes funciones objetivo para optimizar el diseño del intercambiador de calor. En (P.D. Chauduri, 1997) usaron el enfoque del recocido simulado para el diseño óptimo de intercambiadores de calor y desarrollaron un procedimiento de comandos para enlazar el programa HTRI (Heat Transfer Research Inc.) con el algoritmo del recocido. Los autores habían analizado el problema, considerando dos funciones objetivos diferentes llamadas, el área total de transferencia de calor y un coste índice de costo lineal obtenido.

En (R. Selbas, 2006) usaron el algoritmo genético (GA) para el diseño óptimo de STHEs (intercambiadores de calor de tubo y coraza), en el que la caída de presión se aplicó como una restricción para lograr los parámetros óptimos de diseño. Los autores habían considerado la minimización del costo total del intercambiador de calor como una función objetivo. Un estudio de caso fue considerado también para demostrar la aplicación del algoritmo.

Ozcelik (Ozcelik., 2007) consideró un problema mixto de programación no lineal de STHE teniendo en cuenta el tamaño y el costo de la energía del STHE. Los autores habían considerado la minimización de la suma del costo anual del capital y el coste energético de STHEs como una función objetivo y aplicaron un algoritmo genético para la optimización.

En (A.C. Caputo, 2008) llevaron a cabo el diseño del intercambiador de calor basado en optimización económica usando GA (algoritmos genéticos). Los autores habían considerado la minimización del costo total anual como una función objetivo para encontrar la óptima arquitectura del intercambiador de calor.

En (M. Fesanghary, 2009) usaron el análisis de sensibilidad global para identificar los parámetros geométricos más influyentes que afectan el costo total de los STHE con el fin de reducir el tamaño del problema de optimización y llevar a cabo la optimización de los parámetros influyentes de los SYHE desde el punto de vista económico mediante la aplicación del algoritmo de búsqueda armónico. Los autores habían considerado la minimización de los costos de capital y los costos de operación como una función objetivo.

(V.K. Patel, 2010) explora el uso de una técnica de optimización no tradicional, llamada Optimización por Enjambre de Partículas (PSO), para la optimización del diseño de los intercambiadores de calor de tubo y coraza, desde el punto de vista económico. La minimización del costo total anual se considera como función objetivo. Tres variables de diseño tales como el diámetro interior de la coraza, el diámetro exterior del tubo y el espaciamiento entre deflectores se consideran para la optimización. También se consideraron para la optimización dos disposiciones de tubos en forma de triángulo y en forma de cuadrado.

Varios otros investigadores también usaron estrategias basadas en GA (algoritmos genéticos) y algoritmos tradicionales de optimización matemática (I. Ozkol, 2005, R. Hilbert, 2006, G.N. Xie, 2008, J.M. Ponce-Ortega, 2009, J.M. Ponce-Ortega, 2008, M. Saffar-Avval, 1995, M. Reppich, 1994, S. Sun, 1993, P.P.P.M. Lerou, 2005, A.L.H. Costa, 2008, H. Li, 1998, P. Wildi-Tremblay, 2007, B. Allen, 2008, B.V. Babu, 2007, L. Valdevit, 2006, F. Pettersson, 2007, B.K. Soltan, 2004) para diversos objetivos como la generación de entropía mínima (S. Sun, 1993, P.P.P.M. Lerou, 2005) y el costo mínimo de STHes (G.N. Xie, 2008, J.M. Ponce-Ortega, 2009, J.M. Ponce-Ortega, 2008, M. Saffar-Avval, 1995, M. Reppich, 1994, P. Wildi-Tremblay, 2007, B. Allen, 2008, B.V. Babu, 2007, L. Valdevit, 2006, F. Pettersson, 2007, B.K. Soltan, 2004) para optimizar el diseño de intercambiadores de calor. Algunos de estos estudios se focalizan en un solo parámetro geométrico como el espaciamiento óptimo del deflector (M. Saffar-Avval, 1995, H. Li, 1998, B.K. Soltan, 2004) y algunos otros tratan de optimizar una gran variedad de parámetros geométricos y operativos de los STHes. Sin embargo, hay una necesidad de investigar el potencial de la aplicación de técnicas de optimización no tradicionales.

La optimización mediante enjambre de partículas (*Particle Swarm Optimization*) es una de estas técnicas y la misma es investigada en el presente trabajo para evaluar su eficacia.

Sin embargo, hay una necesidad de investigar el potencial de la aplicación de técnicas de optimización no tradicionales. La optimización mediante Colonia de Hormigas es una de estas técnicas y la misma es investigada en el presente trabajo para evaluar su eficacia.

3.3 Modelo matemático para el diseño y cálculo de un intercambiador de Tubo y Coraza (método Kern).

A continuación se presentarán una serie de ecuaciones que se combinarán para bosquejar la solución de un intercambiador de calor de carcasa y tubo.

Kern (KERN, 1997) en 1950 desarrolló uno de los primeros métodos para el estudio del comportamiento térmico de un intercambiador de calor de tubo y coraza. Kern a partir de un intercambiador estándar restringido para un corte de las mamparas del 25% (el mejor en la mayoría de los casos) obtiene las correlaciones para el cálculo de transferencia de calor y la pérdida de cargas. El cálculo se realizará por el método Kern, que todavía se sigue utilizando en la industria para comprobar el funcionamiento térmico de los intercambiadores. Este considera que el flujo a través del arreglo de tubos es únicamente flujo cruzado, y no contempla fugas entre los componentes de la coraza.

Antes de comenzar el cálculo, las condiciones de proceso requeridas para ambos fluidos son:

- Temperaturas de entrada y salida (T_i, T_o).
- Flujos máxicos (m).
- Capacidades caloríficas (C_p).
- Conductividades térmicas (λ).
- Densidades (ρ).
- Viscosidades (μ).
- Factores de obstrucción o ensuciamiento (R_{ensuc}).
- Número de años de vida de la instalación (ny).
- Tasa de descuento anual (i).
- Costo de la energía eléctrica (C_E).
- Horas de trabajo al año (H).

Las variables a optimizar serían.

- Diámetro exterior del tubo (do).
- Diámetro interior de la coraza (D_s).
- Número de pases de los tubos (Np).
- Espaciamiento entre bafles (B).
- Tipo de distribución.

3.3.1 Función objetivo: Minimización del costo de inversión y de operación

El costo total C_{tot} se toma como la función objetivo, que incluye inversión de capital ($C_{inversión}$) y el costo total de descuento de operación de (C_{od}) (A.C. Caputo, 2008).

$$C_{total} = C_{inversión} + C_{od} \quad (3.1)$$

Adoptando la correlación de Hall (M. Taal, 2003), el capital de inversión $C_{\text{inversión}}$ se calcula en función del área superficial del intercambiador.

$$C_{\text{inversión}} = a_1 + a_2 A_T^{a_3} \quad (3.2)$$

Donde para intercambiadores de calor de acero inoxidable, tanto los tubos como la coraza (M. Taal, 2003) $a_1 = 8000$, $a_2 = 259,2$ y $a_3 = 0,91$. Teniendo en cuenta el coeficiente global de transferencia de calor, el área de la superficie del intercambiador de calor (A_T) se calcula:

$$A_T = \frac{Q}{U * F * LMTD} \quad (3.3)$$

Para la transferencia de calor sensible, la tasa de transferencia de calor (Q) está dada por:

$$Q = m_h * C_{ph} (T_{hi} - T_{ho}) = m_c * C_{pc} (T_{co} - T_{ci}) \quad (3.4)$$

Donde T_{ci} es temperatura de entrada del líquido frío en °C, T_{co} , temperatura de salida del líquido frío en °C, T_{hi} , temperatura de entrada del líquido caliente en °C, T_{ho} , temperatura de salida del líquido caliente en °C, C_{ph} , calor específico del líquido caliente en $kJ/kg^\circ C$, C_{pc} , calor específico del líquido frío en $kJ/kg^\circ C$, m_c , flujo másico del fluido frío en kg/seg , m_h , flujo másico del fluido caliente en kg/seg . Teniendo en cuenta el flujo cruzado entre el deflector adyacente, diferencia de temperatura media logarítmica ($LMTD$) se determina por:

$$LMTD = \frac{(T_{hi} - T_{co}) - (T_{ho} - T_{ci})}{\ln\left(\frac{T_{hi} - T_{co}}{T_{ho} - T_{ci}}\right)} \quad (3.5)$$

El factor de corrección (F_t) para la configuración de flujo en cuestión se encuentra en función de la relación de temperatura adimensional para la mayoría de las configuraciones de flujo de interés (Fraas, 1989, M.M. Ohadi, 2000).

$$F_t = \frac{\sqrt{R^2 + 1}}{R - 1} * \frac{\ln\left(\frac{1 - P}{1 - PR}\right)}{\ln\left(\frac{2 - P(R + 1 - \sqrt{R^2 + 1})}{2 - P(R + 1 + \sqrt{R^2 + 1})}\right)} \quad (3.6)$$

donde R es el coeficiente de corrección dado por:

$$R = \frac{T_{hi} - T_{ho}}{T_{co} - T_{ci}} \quad (3.7)$$

P es la eficiencia, dada por:

$$P = \frac{T_{co} - T_{ci}}{T_{hi} - T_{ci}} \quad (3.8)$$

El coeficiente global de transferencia de calor (U) depende de los coeficientes de transferencia de calor y las resistencias al ensuciamiento tanto del lado del tubo como del lado de la coraza dado por (R. Selbas, 2006)

$$U = \frac{1}{\frac{1}{h_s} + R_{ensucs} + \frac{d_o}{d_i} \left(R_{ensuct} + \frac{1}{h_t} \right)} \quad (3.9)$$

Donde d_i es el diámetro interno del tubo y está dado por la expresión:

$$d_i = 0,8 * d_o \quad (3.10)$$

d_o es una variable independiente.

3.3.2 Lado del tubo

De acuerdo con el régimen de flujo, el coeficiente de transferencia de calor en el lado del tubo (h_t) se calcula a partir de la siguiente correlación:

Si ($Re_t < 2\ 300$; (R.K. Sinnott, 1996)),

$$h_t = \frac{\lambda}{d_i} \left[3.657 + \frac{0.0677 \left(Re_t Pr_t \frac{d_i}{L} \right)^{1.33}}{1 + 0.1 Pr_t \left(Re_t \frac{d_i}{L} \right)^{0.3}} \right] \quad (3.11)$$

Si ($2\ 300 < Re_t < 10\ 000$; (R.K. Sinnott, 1996)),

$$h_t = \frac{\lambda}{d_i} \left\{ \frac{\frac{f_t}{8} (Re_t - 1000) Pr_t}{1 + 12.7 \sqrt{\frac{f_t}{8}} (Pr_t^{0.67} - 1)} \left[1 + \left(\frac{d_i}{L} \right)^{0.67} \right] \right\} \quad (3.12)$$

Si ($Re_t > 10\ 000$; (R.K. Sinnott, 1996)),

$$h_t = \frac{\lambda}{d_o} 0.027 Re_t^{0.8} Pr_t^{\frac{1}{3}} \left(\frac{\mu_t}{\mu_w} \right)^{0.14} \quad (3.13)$$

Donde f_t es el factor de fricción de Darcy (Hewitt, 1998) dado por:

$$f_t = (1.92 \log_{10} Re_t - 1.64)^{-2} \quad (3.14)$$

y Re_t es el número de Reynolds en la cara del tubo y está dado como:

$$Re_t = \frac{\rho \cdot v_t \cdot d_i}{\mu} \quad (3.15)$$

Velocidad de flujo para el lado del tubo (v_t) se encuentra por:

$$v_t = \frac{G_t}{\rho} \quad (3.16)$$

Donde la velocidad másica del fluido para el lado del tubo (G_t) está dada por:

$$G_t = \frac{\dot{m}_t}{A_t} \quad (3.17)$$

Donde el área total (A_t) de flujo es:

$$A_t = A_{stt} * \frac{N_t}{N_p} \quad (3.18)$$

Donde N_p es el número de pases por los tubos (N_p se toma como 2) y A_{stt} es el área de la sección transversal del tubo, está dada por:

$$A_{stt} = \frac{\pi}{4} * d_i^2 \quad (3.19)$$

N_t es el número de tubos y que puede ser hallado aproximadamente por la siguiente ecuación.

$$N_t = C \left(\frac{D_s}{d_o} \right)^{n_1} \quad (3.20)$$

C y n_1 son coeficientes que están tomando valores de acuerdo con la disposición de flujo y el número de pasadas. Estos coeficientes se muestran en la Tabla 3.1 para los diferentes arreglos de flujo.

Tabla 3.1 Valores de los coeficientes C y n_1 .

| No. De pases (NP) | Arreglo de tubos triangular | | Arreglo de tubos cuadrado | |
|-------------------|-----------------------------|-------|---------------------------|--------|
| | C | n1 | C | n1 |
| 1 | 0,319 | 2,142 | 0,215 | 2,207 |
| 2 | 0,249 | 2,207 | 0,156 | 2,2291 |
| 4 | 0,175 | 2,285 | 0,158 | 2,263 |
| 6 | 0,0743 | 2,499 | 0,0402 | 2,617 |
| 8 | 0,0365 | 2,675 | 0,0331 | 2,643 |

Sobre la base de la superficie total del intercambiador de calor (A_T), la longitud del tubo necesaria (L) es,

$$L = \frac{A_T}{\pi \cdot d_o \cdot N_t} \quad (3.21)$$

Pr es el número de Prandtl en el lado del tubo y está dado por,

$$Pr = \frac{c_p \cdot \mu}{\lambda} \quad (3.22)$$

Donde λ es la conductividad térmica del fluido (es un dato).

3.3.3 Lado coraza

La Formulación de Kern para intercambiadores de calor de tubo y coraza de separadores segmentados se usa para calcular el coeficiente de transferencia de calor en el lado de la coraza (h_s) (Kern, 1950),

$$h_s = \left(0,36 R_{es}^{0,55} \cdot P_{rs}^{1/3} \cdot \left(\frac{\mu}{\mu_w} \right)^{0,14} \right) \frac{\lambda}{D_e} \quad (3.23)$$

donde, (D_e) es el diámetro hidráulico de la concha y se calcula como (Kern, 1950, R.K. Sinnott, 1996). Para distribución cuadrada:

$$D_e = 4 \cdot \left(\frac{P_t^2 - \frac{\pi \cdot d_o^2}{4}}{\pi \cdot d_o} \right) = \frac{1,27}{d_o} (P_t^2 - 0,785 \cdot d_o^2) \quad (3.24)$$

Para distribución triangular equilátera:

$$D_e = \frac{1,10}{d_o} (P_t^2 - 0,917 \cdot d_o^2) \quad (3.25)$$

Donde el paso entre los tubos (P_t) se calcula como:

$$P_t = (1.25 \div 1.5) \cdot d_o \quad (3.26)$$

El número de Reynolds para el lado de la coraza (Re) es:

$$Re = \frac{\rho \cdot v_s \cdot D_e}{\mu} \quad (3.27)$$

El número de Prandtl para el lado de la coraza (Pr) es:

$$Pr = \frac{c_p \cdot \mu}{\lambda} \quad (3.28)$$

La velocidad de flujo para el lado de la coraza se puede obtener a partir de (R. Selbas, 2006).

$$v_s = \frac{G_s}{\rho} \quad (3.29)$$

Donde la velocidad másica del fluido para el lado de la coraza (G_s) está dada por,

$$G_s = \frac{m_h}{A_s} \quad (3.30)$$

El área de la sección transversal normal a la dirección del flujo se determina por (R. Selbas, 2006),

$$A_s = \frac{D_s \cdot B \cdot (P_t - d_o)}{P_t} = D_s \cdot B \left(1 - \frac{d_o}{P_t}\right) \quad (3.31)$$

Donde D_s es diámetro interior de la coraza y B el espaciamiento entre baffles, ambos variables del problema.

3.3.4 Costo total de operación de descuento y caída de presión

Se verá ahora la segunda parte que interviene en el costo.

El costo total C_{tot} se toma como la función objetivo, que incluye inversión de capital (C_i), el costo de la energía (C_E), el costo de operación anual (C_o) y el costo total de descuento de operación de (C_{od}) (A.C. Caputo, 2008).

El costo de operación relacionado con la potencia consumida en el bombeo (P) para vencer las pérdidas friccionales se determina de las siguientes expresiones:

$$C_{od} = \sum_{k=1}^{ny} \frac{C_o}{(1+i)^k} \quad (3.32)$$

$$C_o = PC_E H \quad (3.33)$$

C_E es el costo de la energía eléctrica en \$/kW h, i la tasa de descuento anual, ny el número de años de vida de la instalación, H las horas de trabajo al año.

$$P = \frac{1}{\eta} \left(\frac{m_t}{\rho_t} \Delta P_t + \frac{m_s}{\rho_s} \Delta P_s \right) \quad (3.34)$$

Donde η es la eficiencia de bombeo (entre 0,5 y 0,7), m_t y m_s son los flujos máscicos en el tubo y la coraza, ΔP_t es la caída de presión en los tubos, ΔP_s es la caída de presión en la coraza, ρ_t es la densidad del fluido que va por el tubo, ρ_s es la densidad del fluido que va por la coraza.

La caída de presión permisible en el intercambiador de calor es la presión estática del fluido que puede ser invertida para conducir el fluido a través del intercambiador. En todos los intercambiadores de calor hay una cerrada afinidad física y económica entre la transferencia de calor y la caída de presión. Para una capacidad calorífica constante en el intercambiador de calor que se va a diseñar, el incremento de la velocidad del flujo provocará un aumento del coeficiente de transferencia de calor lo que se traduce en el diseño compacto del intercambiador y un menor costo de inversión.

Sin embargo, aumentar la velocidad de flujo causará una mayor caída de presión en el intercambiador de calor lo que resulta en costos de funcionamiento adicional. Por esta razón cuando se diseña un intercambiador de calor la caída de presión debe ser considerada con la transferencia de calor y debe ser encontrada la mejor solución para el sistema.

La caída de presión del lado del tubo incluye la caída de presión distribuida a lo largo de la longitud del tubo y las pérdidas de presión concentradas en los codos y en las boquillas de entrada y salida (Kern, 1950).

La caída de presión en los tubos se calcula como la caída de presión propiamente dicha en los tubos más la caída de presión en los codos.

$$\Delta P_t = \Delta P_{lt} + \Delta P_{cod} \quad (3.35)$$

$$\Delta P_t = \frac{\rho_t v_t^2}{2} \left(\frac{L}{d_i} f_t + p \right) N_p \quad (3.36)$$

donde,

$$f_t = 0.00128 + 0.1143(Re_t)^{-0.311} \quad (3.37)$$

Diferentes valores de la constante p son considerados por diferentes autores. Kern (Kern, 1950) supone $p = 4$, mientras que (R.K. Sinnott, 1996) supone $p = 2,5$. Se debe asumir 2,5, N_t es el número de tubos (ya calculada), L la longitud de los tubos (ya calculada), v_t la velocidad del fluido en el lado del tubo (ya calculada).

La caída de presión del lado de la coraza es,

$$\Delta P_s = \frac{\rho_s u_s^2}{2} f_s \frac{L}{B} \frac{D_s}{D_e} \quad (3.38)$$

Donde el factor de fricción en el lado de la coraza se calcula como:

$$f_s = 1,44 Re_s^{-0,15} \quad (3.39)$$

3.4. Resultados

La eficacia del enfoque actual utilizando MACO se evalúa mediante el análisis de tres estudios de caso:

Caso 1: U inicial 600, intercambiador de agua salobre – metanol (R.K. Sinnott, 1996).

Caso 2: U inicial 500, intercambiador de petróleo crudo -kerosene (Kern, 1950).

Caso 3: U inicial 600, intercambiador de agua destilada – agua cruda (R.K. Sinnott, 1996).

Las especificaciones de diseño original, mostradas en las tablas referentes a cada caso de estudio, son suministradas como entrada al algoritmo MACO (Optimización por Colonia de Hormigas Mixtas). Las arquitecturas resultantes de intercambiadores óptimos obtenidas por MACO se comparan con los resultados obtenidos por Algoritmo Genéticos (AG), por Recocido Simulado (SA) y con la solución de diseño original dada por Sinnott (R.K. Sinnott, 1996) y Kern (Kern, 1950) (mostradas en las tablas).

A fin de permitir una comparación coherente, las funciones costo de los cuatro enfoques se calculan como se describe en la sección 3.3. También por la misma razón, todos los valores relacionados con los costos se han tomado de la obra (A.C. Caputo, 2008).

En el enfoque de MACO se impusieron los siguientes límites superior e inferior para las variables de optimización:

- Diámetro interior de la coraza (D_s) en un rango entre 0,1m y 1,5 m,
- Diámetro exterior de los tubos (d_o) en un rango entre los 0.015 m y 0.051 m,
- Espaciamiento entre los deflectores (B), en un rango de 0,05 m a 0,5 m.
- Patrón de diseño de 30 y 45 grados.

Todos los valores de los costos de descuento de operación se calculan con:

- Número de años $ny = 10$ años.

- Tasa de descuento anual (i) = 10%.
- Costo de la energía (C_E) = 0,12 h Euros / kW.
- Cantidad de horas de trabajo anual (H) = 7000 años/horas.

Para obtener la solución óptima usando MACO, se usaron como parámetros:

- Número de hormigas = 50
- Número máximo de iteraciones = 100

Tres estudios de casos se utilizan para mostrar la aplicación de la propuesta algoritmo.

Caso 1: 4,34 Intercambiador de agua salobre – metanol (R.K. Sinnot, 1996).

Este ejemplo implica dos fluidos con las propiedades que se muestran en la Tabla 3.2

Un resumen de los resultados obtenidos con el modelo propuesto (MACO) se dan en la Tabla 3.3, donde una comparación con los resultados reportados por SA, GA y con la solución de diseño original dada por Sinnot (R.K. Sinnot, 1996) y Kern (Kern, 1950).

Tabla 3.1 Propiedades físicas de entrada del caso 1

| | | CASO 1 | |
|------------------------------|-------------|---------|-------------|
| | | Coraza | Tubo |
| | | Metanol | Agua de mar |
| Flujo másico | m | 27,8 | 68,9 |
| Temperatura de entrada | T_i | 95 | 25 |
| Temperatura de salida | T_o | 40 | 40 |
| Densidad del fluido | ρ | 750 | 995 |
| Calor específico | C_p | 2,84 | 4,2 |
| Viscosidad del fluido | μ | 0,00034 | 0,0008 |
| Conductividad térmica | λ | 0,19 | 0,59 |
| Resistencia de ensuciamiento | R_{ensuc} | 0,00033 | 0,0002 |

En los resultados mostrados en la Tabla 3.3 uno puede notar que el diseño obtenido con el algoritmo propuesto tiende a reducir el coste total anual.

La principal diferencia entre los resultados obtenidos por la Literatura, SA y MACO es en la caída de presión en los tubos y la coraza, lo que eleva los costos de bombeo necesarios para el intercambiador, provocando un aumento del costo de operación. La solución encontrada por MACO muestra una reducción en costo anual de operación y el costo total de operación de descuento, determinado por una reducción de la caída de presión. La diferencia obtenida en el

costo total con respecto a AG, es producida por la reducción de la superficie del área del intercambiador de calor, lo que provoca una reducción del capital de inversión. Esto provoca que MACO obtenga un costo total menor, a pesar de que el costo de operación de descuento obtenido por MACO es mayor a la solución estimada por AG.

Tabla 3.3 Óptimas geometrías de intercambiador de calor usando diferentes métodos.

| | Literatura | SA | GA | MACO |
|-------------|------------|----------|--------|--------|
| L | 4,83 | 4,073 | 2,97 | 2,994 |
| do | 0,02 | 0,015 | 0,015 | 0,015 |
| B | 0,356 | 0,499 | 0,497 | 0,5 |
| Ds | 0,894 | 0,72 | 0,86 | 0,84 |
| Pt | 0,025 | 0,0187 | 0,019 | 0,0187 |
| Nt | 918 | 1275 | 1841 | 1829 |
| vt | 0,75 | 0,95 | 0,64 | 0,66 |
| Ret | 14925 | 14308,42 | 9796 | 9997 |
| Prt | 5,7 | 5,69 | 5,69 | 5,69 |
| ht | 3812 | 3969 | 3557 | 3672 |
| ft | 0,028 | 0,0285 | 0,031 | 0,0314 |
| ΔPt | 6251 | 4464 | 1840 | 1985 |
| As | 0,032 | 0,0719 | 0,0856 | 0,084 |
| De | 0,014 | 0,0106 | 0,0107 | 0,0106 |
| vs | 0,58 | 0,51 | 0,43 | 0,43 |
| Res | 18381 | 12127 | 10302 | 10282 |
| Prs | 5,1 | 5,08 | 5,08 | 5,08 |
| hs | 1573 | 1930,23 | 1745 | 1767,3 |
| fs | 0,33 | 0,351 | 0,36 | 0,36 |
| ΔPs | 35789 | 19246 | 12028 | 12322 |
| U | 615 | 707 | 664,7 | 672,8 |
| A | 378,6 | 245,3 | 261,2 | 258 |
| Ci | 51507 | 46760 | 49037 | 48583 |
| Co | 2111 | 1227 | 687 | 713 |
| Cod | 12973 | 7539,5 | 4227 | 4381,3 |
| Ctot | 64480 | 54299 | 53265 | 52964 |

Caso 2: Intercambiador de petróleo crudo -kerosene (Kern, 1950).

Caso 2. Este ejemplo implica dos fluidos con las propiedades que se muestran en la Tabla. 3.4 Un resumen de los resultados obtenidos con el modelo propuesto (MACO) se dan en la Tabla 3.5, donde muestra una comparación con los resultados reportados por SA, GA y con la solución de diseño original dada por Sinnot (R.K. Sinnot, 1996) y Kern (Kern, 1950).

Tabla 3.4 Propiedades físicas de entrada del caso2

| | | CASO 2 | |
|------------------------------|-------------|----------|----------------|
| | | Coraza | Tubo |
| | | Kerosene | Petróleo crudo |
| Flujo másico | m | 5,52 | 18,8 |
| Temperatura de entrada | T_i | 199 | 37,8 |
| Temperatura de salida | T_o | 93,3 | 76,7 |
| Densidad del fluido | ρ | 850 | 995 |
| Calor específico | C_p | 2,47 | 2,05 |
| Viscosidad del fluido | μ | 0,0004 | 0,00358 |
| Conductividad térmica | λ | 0,13 | 0,13 |
| Resistencia de ensuciamiento | R_{ensuc} | 0,00061 | 0,00061 |

Tabla 3.5 Óptimas geometrías de intercambiador de calor usando diferentes métodos.

| | Literatura | SA | GA | MACO |
|-------------|------------|--------|--------|--------|
| L | 4,88 | 6,2 | 6,37 | 6,41 |
| do | 0,025 | 0,015 | 0,015 | 0,015 |
| B | 0,127 | 0,273 | 0,28 | 0,29 |
| Ds | 0,539 | 0,34 | 0,33 | 0,33 |
| Pt | 0,031 | 0,0189 | 0,0188 | 0,0187 |
| Nt | 158 | 243 | 233 | 231 |
| vt | 1,44 | 1,34 | 1,43 | 1,45 |
| Ret | 8227 | 4552 | 4786 | 4845 |
| Prt | 55,2 | 56,4 | 56,4 | 56,4 |
| ht | 619 | 811 | 863 | 877 |
| ft | 0,033 | 0,039 | 0,039 | 0,038 |
| ΔPt | 49245 | 13423 | 15300 | 15843 |
| As | 0,0137 | 0,0188 | 0,0193 | 0,0192 |
| De | 0,025 | 0,01 | 0,0106 | 0,01 |
| vs | 0,47 | 0,345 | 0,336 | 0,337 |
| Res | 25281 | 7904 | 7634 | 7636 |
| Prs | 7,5 | 7,6 | 7,6 | 7,6 |
| hs | 920 | 1181 | 1168 | 1172 |
| fs | 0,315 | 0,374 | 0,0376 | 0,376 |
| ΔPs | 24909 | 13737 | 12503 | 12533 |
| U | 317 | 265,94 | 272 | 274 |
| A | 61,5 | 71,8 | 70,2 | 69,7 |
| Ci | 19007 | 20683 | 20426 | 20342 |
| Co | 1304 | 411,4 | 444 | 456,8 |
| Cod | 8012 | 2527 | 2730 | 2807 |
| Ctot | 27020 | 23211 | 23156 | 23150 |

El aumento del coeficiente global de transferencia de calor, es determinado por un aumento de la caída de presión en el lado del tubo, esto provoca que el área de la superficie del intercambiador de calor obtenida por MACO sea menor que las del resto.

Esta reducción del área de transferencia determina una reducción en capital de inversión. Esta reducción del capital de inversión, es el máximo responsable de que el costo total obtenido por MACO, sea ligeramente inferior al obtenido por AG.

Caso 3: Intercambiador de agua destilada – agua cruda (R.K. Sinnot, 1996).

Este ejemplo implica dos fluidos con las propiedades que se muestran en la Tabla 3.6

Un resumen de los resultados obtenidos con el modelo propuesto se dan en la Tabla 3.7, donde muestra una comparación con los resultados reportados por SA, GA y con la solución de diseño original dada por Sinnot (R.K. Sinnot, 1996) y Kern (Kern, 1950).

Tabla 3.6 Propiedades físicas de entrada del caso3

| | | CASO 3 | |
|------------------------------|-------------|----------------|------------|
| | | Coraza | Tubo |
| | | Agua destilada | Agua cruda |
| Flujo másico | m | 22,07 | 35,31 |
| Temperatura de entrada | T_i | 33,9 | 23,9 |
| Temperatura de salida | T_o | 29,4 | 26,7 |
| Densidad del fluido | ρ | 995 | 999 |
| Calor específico | C_p | 4,18 | 4,18 |
| Viscosidad del fluido | μ | 0,0008 | 0,00092 |
| Conductividad térmica | λ | 0,62 | 0,62 |
| Resistencia de ensuciamiento | R_{ensuc} | 0,00017 | 0,00017 |

La solución obtenida por MACO muestra una reducción en la longitud del intercambiador y diámetro de la coraza, haciendo menor el área de la superficie del intercambiador. A pesar de que los costos de operaciones sean mayores, la reducción del área de transferencia trae consigo una disminución en el capital de inversión, que al final es el que determina que el costo total obtenido por MACO sea menor que el del resto.

En estos tres casos, los resultados obtenidos por MACO, mejoran a los de la Literatura y SA, y más estrechamente a los obtenidos por AG. Demostrando la capacidad de la nueva técnica MACO en la optimización de intercambiadores de calor de tubo y coraza.

Tabla 3.7 Óptimas geometrías de intercambiador de calor usando diferentes métodos.

| | Literatura | SA | GA | MACO |
|-------------|------------|--------|--------|--------|
| L | 4,88 | 1,73 | 1,74 | 1,71 |
| do | 0,013 | 0,015 | 0,015 | 0,015 |
| B | 0,305 | 0,476 | 0,499 | 0,5 |
| Ds | 0,387 | 0,61 | 0,59 | 0,58 |
| Pt | 0,023 | 0,0199 | 0,0188 | 0,0187 |
| Nt | 160 | 787 | 795 | 815 |
| vt | 1,76 | 0,7 | 0,74 | 0,76 |
| Ret | 36400 | 9751 | 9977 | 998 |
| Prt | 6,2 | 6,2 | 6,2 | 6,2 |
| ht | 6558 | 3677 | 3878 | 3990 |
| ft | 0,023 | 0,0316 | 0,0314 | 0,0314 |
| ΔPt | 62814 | 1767 | 1995 | 2124 |
| As | 0,0236 | 0,058 | 0,0595 | 0,058 |
| De | 0,013 | 0,011 | 0,0109 | 0,0106 |
| vs | 0,94 | 0,37 | 0,37 | 0,377 |
| Res | 16200 | 5331 | 5071 | 5004 |
| Prs | 5,4 | 5,39 | 5,39 | 5,39 |
| hs | 5735 | 3856,9 | 3878 | 3955,4 |
| fs | 0,337 | 0,397 | 0,4 | 0,401 |
| ΔPs | 67,684 | 5603 | 5252 | 5391 |
| U | 1471 | 1018 | 1038 | 1054 |
| A | 46,6 | 68,34 | 67 | 66,04 |
| Ci | 16549 | 20112 | 19898 | 19740 |
| Co | 4466 | 224 | 224 | 233 |
| Cod | 27440 | 1377 | 1379 | 1435 |
| Ctot | 43989 | 21490 | 21277 | 21175 |

3.5 Conclusiones parciales

Los intercambiadores de calor son un componente integral de todos los sistemas térmicos. Sus diseños deben adaptarse bien a las aplicaciones en que se utilizan, de lo contrario sus comportamientos serán engañosos y sus costos excesivos. El diseño de intercambiadores de calor puede ser una tarea compleja y herramientas avanzadas de optimización son útiles para identificar el intercambiador de calor mejor y más barato para una demanda específica.

El presente estudio ha demostrado la aplicación con éxito de los algoritmos de optimización bioinspirados, particularmente del método MACO para el diseño óptimo de un intercambiador de calor de tubo y coraza desde el punto de vista económico. Además, la técnica MACO presentada es simple en concepto, de pocos parámetros y fácil para su implementación.

Estas características impulsan la aplicación de MACO particularmente en el diseño de sistemas térmicos, donde los problemas suelen ser complejos y tienen están caracterizados por variables de diferente naturaleza, así como discontinuidades en la función objetivo. La capacidad de la técnica MACO discutida se demuestra usando diferentes casos de estudio de la literatura y el comportamiento de los resultados se compara con los obtenidos por otros algoritmos.

De forma general, el algoritmo MACO converge a un valor aceptable de la función objetivo en pocas generaciones. Claramente esta es una característica deseable en problemas de optimización del mundo real, particularmente para la optimización del diseño de intercambiadores de calor. Por último, es válido resaltar que el procedimiento MACO puede ser fácilmente modificado para adaptarse a la optimización de los diferentes sistemas térmicos.

CONCLUSIONES

Desde su introducción, los métodos aproximados para solucionar problemas de optimización han ganado una creciente atención entre los investigadores. De manera general, estos algoritmos son capaces de encontrar soluciones competitivas en un tiempo de ejecución razonable, a diferencia de los métodos directos de búsqueda donde el tiempo de ejecución es impracticable en muchos casos. Por otra parte, los algoritmos heurísticos poblacionales han tenido un desarrollo notable en la solución de problemas continuos y discretos, pero para problemas mixtos el desarrollo ha sido menos evidente. Por ejemplo, los Algoritmos Genéticos han demostrado ser enfoques eficaces pero en ocasiones su desempeño es pobre para problemas con datos mezclados.

Inspirado por este reto, el presente Trabajo de Diploma introdujo un procedimiento que utiliza la Optimización de Colonias de Hormigas para solucionar problemas mixtos. Es válido aclarar que fue seleccionado ACO como optimizador base debido a su éxito para solucionar problemas de optimización combinatorios (preferentemente discretos, aunque existen algunas variantes para problemas continuos). La idea del método resultante (*Mixed Ant Colony Optimization*, MACO) consiste en tres pasos principales: (i) transformar el problema original en su equivalente discreto, (ii) solucionar el problema discreto utilizando ACO y (iii) transformar las soluciones del segundo problema en las soluciones finales, de acuerdo a la codificación original.

Para validar esta implementación, fueron diseñadas 40 funciones mixtas de prueba con diferente complejidad. Resultados experimentales demostraron, con certeza estadística, que el algoritmo MACO es capaz de superar a Algoritmos Genéticos y Recocido Simulado de forma significativa, lo cual confirma la potencialidad de la presente propuesta para solucionar problemas mixtos. Por otra parte, se resalta que la introducción del método búsqueda local mejora considerablemente las soluciones alcanzadas por las hormigas, aunque eleva el costo computacional.

Finalmente, el método MACO es aplicado a un problema real de la Ingeniería Mecánica donde es necesario la optimización en el diseño de intercambiadores de calor. En este marco, resultados experimentales demostraron que MACO es capaz de aproximar el óptimo conocido, superando a los enfoques reportados en la literatura. Por otra parte, desde el punto de vista económico este estudio ha demostrado experimentalmente la factibilidad de la técnica de MACO para el diseño óptimo de un intercambiador de calor de tubo y coraza.

RECOMENDACIONES

- Como fue discutido, el algoritmo MACO es capaz de optimizar funciones caracterizadas por variables mixtas (continuas y discretas), superando a otros enfoques conocidos como Algoritmos Genéticos y Recocido Simulado. Sin embargo, este método conlleva a un aumento en la dimensionalidad del problema (debido a la transformación de la solución) y por tanto podría ser más costoso computacionalmente. Por tal motivo, resulta de interés estudiar la complejidad temporal de esta variante, respecto a variantes clásicas.

REFERENCIAS BIBLIOGRÁFICAS

- A. DÍAZ, F. G., H.M. GHAZIRI, J.L. GONZALEZ, M. LAGUNA, P. MOSCATO AND TSENG, & F.T. 1996. *Optimización Heurística y Redes Neuronales*, Madrid, Paraninfo.
- A.C. CAPUTO, P. M. P., P. SALINI. 2008. Heat exchanger design based on economic optimization. *Applied Thermal Engineering*, 28
- A.L.H. COSTA, E. M. Q. 2008. Design optimization of shell-and-tube heat exchanger. *Applied Thermal Engineering*, 28
- B. ALLEN, L. G. 2008. Optimal geometry and flow arrangement for minimizing the cost of shell-and-tube condensers. *International Journal of Energy Research*, 32
- B.K. SOLTAN, M. S.-A., E. DAMANGIR. 2004. Minimizing capital and operating costs of shell-and-tube condensers using optimum baffle spacing. *Applied Thermal Engineering*, 24
- B.V. BABU, S. A. M. 2007. Differential evolution strategies for optimal design of shell-and-tube heat exchangers. *Chemical Engineering Science*, 62
- BÄCK, T. 1996. *Evolutionary Algorithms in Theory and Practice*. Oxford.
- BÄCK, T., FOGEL, D. Y MICHALEWICZ, Z. 1997. *Evolutionary Computation 1*. IOP Press.
- BONABEAU, E. 1999. *Swarm Intelligence: From natural to artificial systems*, Oxford University Press.
- BRATLEY, G. B. Y. P. 1996. *Fundamentals of Algorithmics*, Englewood Cliffs, NJ, Prentice Hall.
- BREST, J., GREINER, S., BOŠKOVIĆ, B., MERNIK, M. Y ZUMER, V. 2006. *Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems*, IEEE Transactions on Evolutionary Computation 10.
- CARO, M. D. A. G. D. 1999. The Ant Colony Optimization meta-heuristic. In: HILL, M. (ed.) *New Ideas in Optimization*. London UK.
- COELLO, C. 2005. *An introduction to evolutionary algorithms and their applications*, Guadalajara, Springer.
- CHRIS, W. 2001. A Multilevel Lin-Kernighan-Helsgaun Algorithm for the Travelling Salesman Problem.
- DE CASTRO, L. N. 2007. Fundamentals of natural computing: an overview. *Physics of Life Reviews* 4, 1, 1–36.
- DELGADO, E. M. 2010. *Supernova: un algoritmo novedoso de optimización global*. Universidad nacional de Colombia.
- DORIGO, L. G. A. M. 1997. HAS-SOP: An Hybrid Ant System for the Sequential Ordering Problem. *Technical Report IDSIA11-97*. Lugano - Suiza.
- DORIGO, M. Y. S., T. 2003. Handbook of Metaheuristics. In: SPRINGER (ed.).
- EBERHART, J. K. A. R. C. Particle swarm optimization. In: CENTER, I. S., ed. on neural networks, 1995 Piscataway, NJ. 1942-1948.
- F. PETTERSSON, J. S. 2007. Design of robust heat recovery systems in paper machines. *Chemical Engineering and Processing*, 46
- FEOKTISTOV, V. 2006. *Differential Evolution*. Springer.
- FRAAS, A. P. 1989. *Heat Exchanger Design*, New York, John Wiley.
- FREISLEBEN, P. M. A. B. 2000. Greedy and Local Search Heuristics for Unconstrained Binary Quadratic Programming.
- G.N. XIE, B. S., Q.W. WANG. 2008. Optimization of compact heat exchangers by a genetic algorithm. *Applied Thermal Engineering*, 28
- GAMBARDELLA., M. D. Y. L. M. 1997. Ant Colony System: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1: 1, 53-66.

- GLOVER, F. 1977. Heuristics for Integer Programming Using Surrogate Constraints. *Decision Sciences*, 8, 156-166.
- GOLDBERG, D. E. 1989. Genetic Algorithms in Search, Optimization and Machine Learning. *Addison-Wesley*.
- GOLDBERG, D. E. 1998. *Genetic Algorithms in Search*, University of Alabama, Addison-Wesley Publishing Company.
- H. LI, V. K. 1998. Visualization and determination of local heat transfer coefficients in shell-and-tube heat exchangers for staggered tube arrangement by mass transfer measurements. *Experimental Thermal and Fluid Science*, 17.
- HEWITT, G. F. 1998. *Heat Exchanger Design Handbook*, New York, Begell House.
- HIMMELBLAU, D. 1972. Applied nonlinear optimization. *McGraw hill*.
- HOOS., T. S. Y. H. H. 1997. The MAX-MIN Ant System and local search for the traveling salesman problem. In: T. BÄCK, Z. M., Y X. YAO (ed.) *Proceedings of the 1997 IEEE International Conference on Evolutionary Computation (ICEC'97)*. Piscataway, NJ: IEEE Press.
- HOOS., T. S. Y. H. H. 2000a. *MAX-MIN Ant System*, Future Generation Computer Systems.
- HOOS., T. S. Y. H. H. 2000b. *MAX-MIN Ant System*: Future Generation Computer Systems.
- HOOS., T. S. Y. H. H. Agosto 1996. *Improving the Ant system: A detailed report on the MAX-MIN Ant System*, Alemania, Technical Report AIDA-96-12.
- I. OZKOL, G. K. 2005. Determination of the optimum geometry of the heat exchanger body via a genetic algorithm. *International Journal of Heat and Mass Transfer*, 48
- J. M. PASTEELS, J.-L. D., Y S. GOSS. 1987. Self-organization mechanisms in ant societies (I): Trail recruitment to newly discovered food sources. *Experientia Supplementum*, 54, 155-175.
- J.M. PONCE-ORTEGA, M. S.-G., A. JIMENEZ-GUTIERREZ. 2008. Design and optimization of multipass heat exchangers. *Chemical Engineering and Processing*, 47
- J.M. PONCE-ORTEGA, M. S.-G., A. JIMENEZ-GUTIERREZ. 2009. Use of genetic algorithms for the optimal design of shell-and-tube heat exchangers. *Applied Thermal Engineering*, 29
- JANG, J., SUN, C. Y MIZUTAN, E. 1996. Neuro-fuzzy and soft computing. *Prentice Hall*.
- JOAQUÍN DERRAC , S., DANIELMOLINA ,FRANCISCOHERRERA 2011. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, 3–18.
- JOHNSON, M. R. G. A. D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Francisco, H. Freeman and Company.
- KELD, H. A. 2000. An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic.
- KELLY, I. H. O. A. J. P. 1996. *Meta-Heuristics: Theory and Applications*. Ed. *Kluwer Academic, Boston*.
- KERN, D. Q. 1950. *Process Heat Transfer*. New York: McGraw-Hill.
- KERN, D. Q. 1997. “*Procesos de Transferencia de Calor*”, México, Editorial Continental s.a.
- KERNIGHAN, S. L. A. B. W. 1973. An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations research letters*, 21, 498-516.
- KEUTHEN, K. E., BURKE, I. PETER, COWLING AND RALF 2003. Effective Local and Guided Variable.
- L. VALDEVIT, A. P., H.A. STONE, A.G. EVANS. 2006. Optimal active cooling performance of metallic sandwich panels with prismatic cores. *International Journal of Heat and Mass Transfer*, 49
- M. DORIGO, G. D. C., AND L. M. GAMBARDELLA 1999. Ant algorithms for discrete optimization. *Artificial Life*, 5, 137-172.

- M. DORIGO, V. M., Y A. COLORNI. 1996. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems Man, and Cybernetics, Man, and Cybernetics - Part B*, 26: 1, 29-41.
- M. FESANGHARY, E. D., I. SOLEIMANI. 2009. Design optimization of shell-and-tube heat exchangers using global sensitivity analysis and harmony search algorithm. *Applied Thermal Engineering*, 29
- M. REPPICH, J. K. 1994. Optimal design of shell-and-tube heat exchangers. *Computer & Chemical Engineer*, 18
- M. SAFFAR-AVVAL, E. D. 1995. A general correlation for determining optimum baffle spacing for all types of shell-and-tube exchangers. *International Journal of Heat and Mass Transfer*, 38
- M. TAAL, I. B., J. KLEMES, P. STEHLIK 2003. Cost estimation and energy price forecast for economic evaluation of retrofit projects. *Applied Thermal Engineering*, 23
- M.M. OHADI 2000. The Engineering Handbook. *CRC Press LLC*.
- MATTHIJS, B. 2001. Neighborhoods Revisited: An Experimental Investigation into the Effectiveness of Variable Neighborhood Descent for Scheduling.
- MCPEAK, D. B., EUGENE INGERMAN, JOSHUA LEVY AND SCOTT 2000. An Improved Adaptive Multi-Start Approach to Finding Near-Optimal Solutions to the Euclidean {TSP}.
- MICHALEWICZ, Z. Y. S., M. 2002. Evolutionary Optimization. In: PRESS, K. A. (ed.).
- MIDDENDORF., R. M. Y. M. 1998. An Island model based Ant system with lookahead for the shortest supersequence problem. *Fifth International Conference on Parallel Problem Solving from Nature*. A. E. Eiben, T. Bäck, M. Schoenauer, y H.-P. Schwefel ed. Berlín, Alemania: Springer Verlag.
- MULLER, S., MARCHETTO, J., AIRAGHI, S. Y KOURNOUTSAKOS, P. 2000. Optimization algorithms based on a model of bacterial chemotaxis. *proceeding of 6th Int. Conf. Simulation of Adaptive Behavior: From Animals to Animats, SAB 2000*. Paris: Springer-Verlay.
- NASH, S. Y. S., A. 1996. Linear and Nonlinear Programming. *McGraw hill*.
- NENAD, M. A. 2003. A Tutorial on Variable Neighborhood Search.
- O. CORDÓN, F. H., L. MORENO. Integración de Conceptos de Computación Evolutiva en un Nuevo Modelo de Colonias de Hormigas. VIII Conferencia de la Asociación Española para la Inteligencia Artificial, 1999 Murcia (España). 98-105.
- O. CORDÓN, F. H., T. STÜTZLE 2002. A Review on Ant Colony Optimization Metaheuristic: Basis. *Models and New Trends*. Mathware & Soft Computing.
- OMRAN, M. 2007. Global-best harmony search. *Applied Mathematics and Computation*, Vol 198 (2), 643-656.
- OZCELIK., Y. 2007. Exergetic optimization of shell-and-tube heat exchanger using a genetic based algorithm. *Applied Thermal Engineering*, 27
- P. WILDI-TREMBLAY, L. G. 2007. Minimizing shell-and-tube heat exchanger cost with genetic algorithms and considering maintenance. *International Journal of Energy Research*, 31
- P.D. CHAUDURI, U. D., J. LOGSDON. 1997. An automated approach for the optimal design of heat exchangers. *Industrial & Engineering Chemistry Research*, 36.
- P.P.P.M. LEROU, T. T. V., J.F. BURGER, H.J.M. BRAKE, H. ROGALLA. 2005. Optimization of counter flow heat exchanger geometry through minimization of entropy generation. *Cryogenics*, 45
- PIERRE, D. A. 1986. Optimización theory with applications. *Dover*.
- R. HILBERT, G. J., R. BARON, D. THEVENIN. 2006. Multi objective shape optimization of a heat exchanger using parallel genetic algorithm. *International Journal of Heat and Mass Transfer*, 49

- R. SELBAS, O. K., M. REPPICH 2006. A new design approach for shell-and-tube heat exchangers using genetic algorithms from economic point of view. *Chemical Engineering and Processing*, 45
- R.K. SINNOT, J. M. C., J.F. RICHARDSON 1996. *Chemical Engineering Design*. Butterworth-Heinemann, vol. 6.
- RAO, S. 1996. *Engineering Optimization, Theory and practice*. Wiley.
- REEVES, C. R. 1995. *Modern Heuristic Techniques for Combinatorial Problems*, UK, Ed. McGraw-Hill.
- RESENDE, T. F. A. 1989. A probabilistic heuristic for a computational difficult set covering problems. *Operations research letters*, 8, 67–71.
- S. GOSS, S. A., J. L. DENEUBOURG, Y J. M. PASTEELS. 1989. Self-organized shortcuts in the Argentine ant. *Naturwissenschaften*, 76, 579-581.
- S. KIRKPATRICK, C. D. G., M.P. VECCHI 1983. Optimization by simulated. annealing. *Science*, 220, 671-680.
- S. SUN, Y. L., C. YAN. 1993. Optimization in calculation of shell-and-tube heat exchanger. *International Communications in Heat and Mass Transfer*, 20.
- STEIGLITZ, C. H. P. Y. K. 1982. *Combinatorial Optimization - Algorithms and Complexity*, Englewood Cliffs, NJ, Prentice Hall.
- STUTZLE, M. D. A. T. 1999. ACO Algorithms for the Traveling Salesman Problem. *EEUU*, John Wiley & Sons.
- STÜTZLE, M. D. Y. T. 2003. The ant colony optimization metaheuristic: Algorithms, applications and advances. In: F. GLOVER AND G. KOCHENBERGER, E. (ed.) *Handbook of Metaheuristics*. Kluwer Academic Publishers.
- STÜTZLE., T. 1999. Local Search Algorithms for Combinatorial Problems: Analysis, Improvements, and New Applications. 220 of DISKI. Infix, Sankt Augustin, Alemania.
- SUGANTHAN, P. N. 2005. Problem definition and evaluation criteria for the CEC 2005 special session on real-parameter optimization. *Technical Report 2005005*.
- TALBI, E. G. 2009. *Metaheuristics: from design to implementation*. Wiley.
- V.K. PATEL, R. V. R. 2010. Design optimization of shell-and-tube heat exchanger using particle swarm optimization technique. *Applied Thermal Engineering*., 30.
- VESTERSTROM, J. Y. T., R. 2004. *A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems*, NY, Portland, IEEE Press.
- WELTY, J. R. 1996. “*Transferencia de Calor Aplicada a la Ingeniería*”, México, Editorial Limusa.
- WOLPERT, D. H. Y. M., W. G. 1997. No free lunch theorems for optimization. *IEEE Transactions on evolutionary computation* 1, 67-82.