

Metaheurísticas: una visión global*

Belén Melián Batista, José A. Moreno Pérez, José M. Moreno Vega
DEIOC.

Universidad de La Laguna
38271 La Laguna
{mbmelian,jamoreno,jmmoreno}@ull.es

Resumen

Las metaheurísticas pueden concebirse como estrategias generales de diseño de procedimientos heurísticos para la resolución de problemas con un alto rendimiento. En este trabajo se tratan, en primer lugar, los fundamentos para establecer el concepto de metaheurística. Los estrategias metaheurísticas se refieren al diseño de alguno de los tipos fundamentales de procedimientos heurísticos de solución de un problema de optimización. Se realiza una descripción de las principales metaheurísticas para métodos de relajación, procesos constructivos, búsquedas por entornos y procedimientos evolutivos. Se presta atención especial a las metaheurísticas de búsqueda que constituyen el paradigma central de estas técnicas en la resolución de problemas de optimización. Se proponen y analizan las características deseables de las metaheurísticas, desde el punto de vista de su estudio teórico y de su aplicación práctica. Finalizamos con las conclusiones derivadas de nuestra perspectiva.

1. Introducción

En Inteligencia Artificial (IA) se emplea el calificativo *heurístico*, en un sentido muy genérico, para aplicarlo a todos aquellos aspectos que tienen que ver con el empleo de conocimiento en la realización dinámica de tareas. Se habla de heurística para referirse a una técnica, método o procedimiento inteligente de realizar una tarea que no es producto de un riguroso análisis formal, sino de conocimiento experto sobre la tarea. En especial, se usa el término heurístico para referirse a un procedimiento que trata de aportar soluciones a un problema con un buen rendimiento, en lo referente a la calidad de las soluciones y a los recursos empleados.

En la resolución de problemas específicos han surgido procedimientos heurísticos exitosos, de los que se ha tratado de extraer lo que es esencial en su éxito para aplicarlo a otros problemas o en contextos más extensos. Como ha ocurrido claramente en diversos campos de la IA, en especial con los sistemas expertos, esta

línea de investigación ha contribuido al desarrollo científico del campo de las heurísticas y a extender la aplicación de sus resultados. De esta forma se han obtenido, tanto técnicas y recursos computacionales específicos, como estrategias de diseño generales para procedimientos heurísticos de resolución de problemas. Estas estrategias generales para construir algoritmos, que quedan por encima de las heurísticas, y van algo más allá, se denominan metaheurísticas. Las metaheurísticas pueden integrarse como un sistema experto para facilitar su uso genérico a la vez que mejorar su rendimiento.

En este trabajo se presenta una visión global actualizada del campo de las metaheurísticas, centrada en torno a la noción de metaheurística, la clasificación de las más relevantes y el análisis de las cualidades deseables de éstas. Sin embargo, una discusión rigurosa del concepto de metaheurística, una clasificación estructurada y exhaustiva de las diferentes estrategias, o el estudio completo de las características apropiadas de una metaheurística es una empresa imposible de contemplar y a la que han contribuido diversos autores con reflexiones intercaladas en libros o artículos sobre metaheurísticas específi-

*Este trabajo ha sido parcialmente financiado con el proyecto TIC2002-04242-C03-01 con fondos FEDER en un 70 %

cas (ver, por ejemplo, [26], [37], [36] y [77]).

En la siguiente sección se describen los fundamentos que permiten establecer, partiendo de la noción de heurística, el concepto de metaheurística y se establece una primera clasificación de las metaheurísticas a partir de los diferentes tipos de procedimientos heurísticos para los que establecen pautas de diseño. En la tercera sección se describen las metaheurísticas de búsqueda, considerando tanto búsqueda local como global, y las estrategias evolutivas. En la cuarta sección se analiza el papel de las metaheurísticas y se enumeran las principales características deseables de las mismas. El trabajo finaliza con unas breves conclusiones.

2. Las Metaheurísticas

2.1. Concepto de metaheurística

La idea más genérica del término *heurístico* está relacionada con la tarea de resolver inteligentemente problemas reales usando el conocimiento disponible. El término heurística proviene de una palabra griega con un significado relacionado con el concepto de encontrar y se vincula a la supuesta exclamación *eureka* de Arquímedes al descubrir su famoso principio.

La concepción más común en IA es interpretar que heurístico es el *calificativo* apropiado para los procedimientos que, empleando conocimiento acerca de un problema y de las técnicas aplicables, tratan de aportar soluciones (o acercarse a ellas) usando una cantidad de recursos (generalmente tiempo) razonable. En un problema de optimización, aparte de las condiciones que deben cumplir las soluciones factibles del problema, se busca la que es óptima según algún criterio de comparación entre ellas. En Investigación Operativa, el término heurístico se aplica a un procedimiento de resolución de problemas de optimización con una concepción diferente. Se califica de *heurístico* a un procedimiento para el que se tiene un alto grado de confianza en que encuentra soluciones de alta calidad con un coste computacional razonable, aunque no se garantice su optimalidad o su factibilidad, e incluso, en algunos casos, no se llegue a establecer lo cerca que se está de dicha situación. Se usa el calificativo heurístico en contraposi-

ción a *exacto*, que se aplica los procedimientos a los que se les exige que la solución aportada sea óptima o factible. Una solución *heurística* de un problema es la proporcionada por un método heurístico, es decir, aquella solución sobre la que se tiene cierta confianza de que es factible y óptima, o de que alcanza un alto grado de optimalidad y/o factibilidad. También es usual aplicar el término heurística cuando, utilizando el conocimiento que se tiene del problema, se realizan modificaciones en el procedimiento de solución del problema que, aunque no afectan a la complejidad del mismo, mejoran el rendimiento en su comportamiento práctico.

Unas heurísticas para resolver un problema de optimización pueden ser más *generales* o *específicas* que otras. Los métodos heurísticos específicos deben ser diseñados a propósito para cada problema, utilizando toda la información disponible y el análisis teórico del modelo. Los procedimientos específicos bien diseñados suelen tener un rendimiento significativamente más alto que las heurísticas generales. Las heurísticas más generales, por el contrario, presentan otro tipo de ventajas, como la sencillez, adaptabilidad y robustez de los procedimientos. Sin embargo, las heurísticas generales emanadas de las metaheurísticas pueden mejorar su rendimiento utilizando recursos computacionales y estrategias inteligentes.

El término metaheurísticas se obtiene de anteponer a *heurística* el sufijo *meta* que significa “más allá” o “a un nivel superior”. Los conceptos actuales de lo que es una metaheurística están basados en las diferentes interpretaciones de lo que es una forma inteligente de resolver un problema. Las *metaheurísticas* son estrategias inteligentes para diseñar o mejorar procedimientos heurísticos muy generales con un alto rendimiento. El término metaheurística apareció por primera vez en el artículo seminal sobre búsqueda *tabú* de Fred Glover en 1986 [20]. A partir de entonces han surgido multitud de propuestas de pautas para diseñar buenos procedimientos para resolver ciertos problemas que, al ampliar su campo de aplicación, han adoptado la denominación de metaheurísticas.

La relevancia de las metaheurísticas se refleja en la publicación de libros sobre este campo en los últimos años, entre los que los más recientes son [57],[61], [45], [67], [73], [46] y [25]. Diversos artículos de revisión, monografías

y volúmenes especiales sobre metaheurísticas han venido apareciendo en diversas colecciones editoriales o revistas periódicas de los campos de Investigación Operativa, Inteligencia Artificial, Ingeniería y Ciencias de la Computación. Además, en estas publicaciones se observa un incremento considerable del número de trabajos que incluyen procedimientos heurísticos en los que se realizan planteamientos estándares de las metaheurísticas. Desde 1985 se viene publicando la revista *Journal of Heuristics* que concentra una parte importante de las publicaciones en este campo.

2.2. Tipos de metaheurísticas

Las metaheurísticas son estrategias para diseñar procedimientos heurísticos. Por tanto, los tipos de metaheurísticas se establecen, en primer lugar, en función del tipo de procedimientos a los que se refiere. Algunos de los tipos fundamentales son las metaheurísticas para los métodos de relajación, las metaheurísticas para los procesos constructivos, las metaheurísticas para las búsquedas por entornos y las metaheurísticas para los procedimientos evolutivos.

- Las metaheurísticas de *relajación* se refieren a procedimientos de resolución de problemas que utilizan relajaciones del modelo original (es decir, modificaciones del modelo que hacen al problema más fácil de resolver), cuya solución facilita la solución del problema original.
- Las metaheurísticas *constructivas* se orientan a los procedimientos que tratan de la obtención de una solución a partir del análisis y selección paulatina de las componentes que la forman.
- Las metaheurísticas *de búsqueda* guían los procedimientos que usan transformaciones o movimientos para recorrer el espacio de soluciones alternativas y explotar las estructuras de entornos asociadas.
- Las metaheurísticas *evolutivas* están enfocadas a los procedimientos basados en conjuntos de soluciones que evolucionan sobre el espacio de soluciones.

Algunas metaheurísticas surgen combinando metaheurísticas de distinto tipo, como la metaheurística GRASP (*Greedy Randomized Adaptive Search Procedure*) [65], [31], que combina una fase constructiva con una fase de búsqueda de mejora. Otras metaheurísticas se centran en el uso de algún tipo de recurso computacional o formal especial como las redes neuronales, los sistemas de hormigas o la programación por restricciones y no se incluyen claramente en ninguno de los cuatro tipos anteriores.

Por otro lado, de una u otra forma, todas las metaheurísticas se pueden *concebir* como estrategias aplicadas a procesos de búsqueda, donde todas las situaciones intermedias en el proceso de resolución del problema se interpretan como elementos de un espacio de búsqueda, que se van modificando a medida que se aplican las distintas operaciones diseñadas para llegar a la resolución definitiva. Por ello, y porque los procesos de búsqueda heurística constituyen el paradigma central de las metaheurísticas, es frecuente interpretar que el término metaheurística es aplicable esencialmente a los procedimientos de búsqueda sobre un espacio de soluciones alternativas. Por este mismo motivo se dedica una parte importante de este trabajo a las metaheurísticas de búsqueda.

2.2.1. Metaheurísticas de Relajación

Una cuestión relevante al abordar un problema real es la obtención de un modelo que permita emplear una técnica de resolución apropiada. Si con este modelo el problema resulta difícil de resolver se acude a modelos modificados en los que es más sencillo encontrar buenas soluciones o en los que los procedimientos son más eficientes. Una relajación de un problema es un modelo simplificado obtenido al eliminar, debilitar o modificar restricciones (u objetivos) del problema real. En cualquier formulación siempre existe algún grado de simplificación, lo que puede afectar en mayor o menor medida al ajuste a la realidad de los procedimientos de resolución y de las soluciones del problema propuestas. Los modelos muy ajustados a la realidad suelen ser muy difíciles de resolver, y sus soluciones difíciles de implementar exactamente, por lo que se acude a modelos relajados. Las *metaheurísticas de relajación* son estrategias para el empleo de relajaciones

del problema en el diseño de heurísticas. Se refieren al diseño, tanto de procedimientos que utilizan formulaciones relajadas del problema para proponer sus soluciones, como soluciones del problema, como de procedimientos que usan dichas relajaciones para guiar las operaciones realizadas para su resolución.

Muchas heurísticas de relajación modifican elementos del problema para proponer la solución de estas modificaciones como solución heurística del problema original. Las buenas relajaciones son las que simplifican el problema y hacen más eficientes los procedimientos de solución, pero cuya resolución proporciona muy buenas soluciones del problema original. Por ejemplo, para un problema de programación lineal entera, su relajación lineal consiste en ignorar la restricción de que las variables sean enteras. Se utiliza frecuentemente para aplicar procedimientos eficientes de programación lineal, como el método del Simplex, a dicha relajación y proponer una solución entera muy próxima a la solución del problema relajado.

Entre las metaheurísticas de relajación se encuentran los métodos de relajación lagrangiana [5], [33] o de restricciones subordinadas. Otras metaheurísticas de relajación alteran las restricciones o los objetivos del problema para usar su solución en la conducción de la búsqueda de la solución del problema original. Esta modificación puede estar encaminada a relajar las restricciones a las que debe estar sometida la solución, permitiendo que el recorrido borde la región factible para acercarse al óptimo global incluso desde la región no factible. Otras estrategias modifican la función objetivo para obtener, de forma más rápida, valoraciones aproximadas (por exceso o por defecto) de la calidad de la solución que orientan la búsqueda, al menos en los estados iniciales. Es frecuente encontrar problemas en los que evaluar la función objetivo puede significar resolver otro problema de gran dificultad, realizar un proceso de simulación o realizar algún tipo de inversión o consumo de recursos. Para estos problemas es muy útil encontrar funciones sencillas de calcular que den una idea aproximada de la calidad de las soluciones sin necesidad de una evaluación ajustada de la función objetivo.

2.2.2. Metaheurísticas Constructivas

Las heurísticas constructivas aportan soluciones del problema por medio de un procedimiento que incorpora iterativamente elementos a una estructura, inicialmente vacía, que representa a la solución. Las *metaheurísticas constructivas* establecen estrategias para seleccionar las componentes con las que se construye una buena solución del problema. Entre las metaheurísticas primitivas en este contexto se encuentra la popular estrategia voraz o *greedy*, que implica la elección que da mejores resultados inmediatos, sin tener en cuenta una perspectiva más amplia. Dentro de este tipo de metaheurística, destaca la aportación de la metaheurística GRASP [65], [31] que, en la primera de sus dos fases, incorpora a la estrategia *greedy* pasos aleatorios con criterios adaptativos para la selección de los elementos a incluir en la solución.

2.2.3. Metaheurísticas de búsqueda

El tipo de metaheurística más importante es el de las *metaheurísticas de búsqueda*, que establecen estrategias para recorrer el espacio de soluciones del problema transformando de forma iterativa soluciones de partida. Las búsquedas evolutivas se distinguen de éstas en que es un conjunto de soluciones, generalmente llamado *población* de búsqueda, el que evoluciona sobre el espacio de búsqueda.

La concepción primaria de heurística más frecuente era la de alguna regla inteligente para mejorar la solución de un problema que se aplicaba iterativamente mientras fuera posible obtener nuevas mejoras. Tales procesos se conocen como búsquedas monótonas (descendentes o ascendentes), algoritmos escaladores (*hill-climbing*) o búsquedas locales. Esta última denominación obedece a que la mejora se obtiene en base al análisis de soluciones similares a la que realiza la búsqueda; denominadas *soluciones vecinas*. Estrictamente hablando, una *búsqueda local* es la que basa su estrategia en el estudio de soluciones del vecindario o entorno de la solución que realiza el recorrido. Las metaheurísticas de búsqueda local son las estrategias o pautas generales para diseñar métodos de búsqueda local, como la estrategia voraz o *greedy*. Esta metaheurística establece como

pauta, una vez consideradas cuales son las soluciones que intervienen en el análisis local, elegir iterativamente la mejor de tales soluciones mientras exista alguna mejora posible.

Sin embargo, se suele asumir que las búsquedas locales sólo modifican la solución que realiza el recorrido mediante una mejora en su propio entorno. El principal inconveniente de estas búsquedas locales es que se quedan atrapadas en un óptimo local, una solución que no puede ser mejorada por un análisis local. Por ello, el propósito fundamental de las primeras metaheurísticas era extender una búsqueda local para continuarla más allá de los óptimos locales, denominándose *Búsqueda Global*.

Las metaheurísticas de búsqueda global incorporan pautas para tres formas básicas de escapar de los óptimos locales de baja calidad: volver a iniciar la búsqueda desde otra solución de arranque, modificar la estructura de entornos que se está aplicando y permitir movimientos o transformaciones de la solución de búsqueda que no sean de mejora. Surgen así, respectivamente, las metaheurísticas de arranque múltiple, las metaheurísticas de entorno variable y las metaheurísticas de búsqueda no monótona. Las metaheurísticas de *arranque múltiple* [53], [64] establecen pautas para reiniciar de forma inteligente las búsquedas descendentes. Las metaheurísticas de *entorno variable* modifican de forma sistemática el tipo de movimiento con el objeto de evitar que la búsqueda se quede atrapada por una estructura de entornos rígida. Las búsquedas que también aplican movimientos de no mejora durante el recorrido de búsqueda se denominan *búsquedas no monótonas*.

Las metaheurísticas para búsquedas no monótonas controlan los posibles movimientos de empeoramiento de la solución mediante criterios de aceptación estocásticos o utilizando la memoria del proceso de búsqueda. Las metaheurísticas de *búsqueda estocásticas* establecen pautas para regular la probabilidad de aceptar transformaciones que no mejoren la solución. El *Recocido Simulado* [42], [14] es el exponente más importante de este tipo de metaheurísticas donde la probabilidad de aceptación es una función exponencial del empeoramiento producido. Las metaheurísticas de *búsqueda con memoria* utilizan información sobre el recorrido realizado para evitar que la búsqueda

se concentre en una misma zona del espacio. Fundamentalmente se trata de la *Búsqueda Tabú* [26], [55] cuya propuesta original prohíbe temporalmente soluciones muy parecidas a las últimas soluciones del recorrido.

2.2.4. Metaheurísticas evolutivas

Las *metaheurísticas evolutivas* establecen estrategias para conducir la evolución en el espacio de búsqueda de conjuntos de soluciones (usualmente llamados *poblaciones*) con la intención de acercarse a la solución óptima con sus elementos. El aspecto fundamental de las heurísticas evolutivas consiste en la interacción entre los miembros de la población frente a las búsqueda que se guían por la información de soluciones individuales.

Las diferentes metaheurísticas evolutivas se distinguen por la forma en que combinan la información proporcionada por los elementos de la población para hacerla evolucionar mediante la obtención de nuevas soluciones. Los algoritmos genéticos [30], [66] y meméticos [11], [60] y los de estimación de distribuciones [51] [47] emplean fundamentalmente procedimientos aleatorios, mientras que las metaheurísticas de búsqueda dispersa o de re-encadenamiento de caminos (*Path Relinking*) [44], [54] emplean procedimientos sistemáticos.

2.2.5. Otros tipos de metaheurísticas

Otras metaheurísticas que aparecen en varias clasificaciones corresponden a tipos intermedios entre los anteriores [70], [79]. Entre ellas destacan las metaheurísticas de descomposición y las de memoria a largo plazo.

Las *metaheurísticas de descomposición* establecen pautas para resolver un problema determinando subproblemas a partir de los que se construye una solución del problema original. Se trata de metaheurísticas intermedias entre las de relajación y las constructivas, ya que se refieren básicamente a las características que se pretenden obtener en los subproblemas y a cómo integrar las soluciones de estos subproblemas en una solución del problema original. El objetivo fundamental es obtener subproblemas significativamente más fáciles de resolver

que los originales, y cuyas soluciones puedan ser utilizadas efectivamente. Este es el tipo de metaheurística más apropiada para la aplicación de estrategias de paralelización, donde es muy importante el equilibrio entre los subproblemas obtenidos.

Las *metaheurísticas de memoria a largo plazo* constituyen el caso más relevante de las metaheurísticas de aprendizaje y se sitúan entre las de arranque múltiple y las derivadas de la búsqueda tabú. Por ejemplo, diversas metaheurísticas se refieren al uso de información sobre las características y propiedades comunes a soluciones de alta calidad o sobre las decisiones de mejora adoptadas durante el proceso de solución. Esta información permite mejorar el rendimiento de la búsqueda de arranque múltiple ajustando los parámetros que modulan la exploración y la explotación del proceso. Se incluyen en las metaheurísticas de *aprendizaje* ya que son capaces de emplear información obtenida en la aplicación del propio procedimiento, tanto a un problema específico como a un tipo o clase específica de problemas.

3. Metaheurísticas de búsqueda

Las *metaheurísticas de búsqueda* aportan estrategias para afrontar la resolución de un problema realizando una búsqueda sobre un espacio cuyos elementos representan las soluciones candidatas alternativas. La representación de las soluciones se realiza a través de una codificación que incluya toda la información necesaria para su identificación y evaluación. Una búsqueda sobre un espacio consiste en generar una sucesión de puntos del espacio pasando de uno a otro por medio de una serie de transformaciones o movimientos. Un procedimiento de búsqueda para resolver un problema de optimización realiza recorridos sobre el espacio de las soluciones alternativas y selecciona la mejor solución encontrada en el recorrido. Las metaheurísticas de búsqueda proporcionan pautas para obtener recorridos que, con alto rendimiento, proporcionen soluciones de alta calidad.

La descripción general de un proceso de resolución de un problema es, partiendo de una

situación inicial, aplicar iterativamente una operación para modificar la situación actual, hasta que se alcance la situación buscada. Un proceso de búsqueda basado en transformaciones o movimientos sobre un espacio de soluciones posibles consiste en la selección iterativa de movimientos para transformar una solución hasta que se cumpla cierto criterio de parada. El criterio de parada determina cuándo se considera resuelto el problema sin que sea necesario disponer, en una situación intermedia, de información de lo cerca que se está de solucionarlo. Sin embargo, las búsquedas inteligentes deben utilizar este y otro tipo de información en el criterio de parada y en la selección de los movimientos.

En los problemas de optimización, la selección de movimientos y el criterio de parada se realizan teniendo en cuenta, al menos, un indicador de la calidad de las soluciones encontradas en el recorrido. La evaluación de la calidad de las soluciones se realiza a través de una o varias funciones objetivo, teniendo en cuenta las restricciones del problema. La estrategia de búsqueda establece los criterios y mecanismos que guiarán el recorrido. La estrategia de búsqueda puede incorporar herramientas de una o varias metaheurísticas junto a heurísticas específicas para el problema. Por su generalidad, la descripción y análisis de las metaheurísticas de búsqueda se realiza sobre problemas de optimización. A continuación se introducen los aspectos más importantes de los problemas de optimización para describir las metaheurísticas de búsqueda.

Un *problema de optimización* es aquel cuya solución implica encontrar en un conjunto de soluciones candidatas alternativas aquella que mejor satisface unos objetivos. Los problemas de optimización surgen en muchísimos campos científicos y su solución es de crucial importancia para el éxito de multitud de tareas de Inteligencia Artificial. Cada problema de optimización se especifica estableciendo cuáles son las soluciones alternativas y los objetivos perseguidos. Los objetivos se formalizan por una o varias funciones que hay que maximizar o minimizar (supondremos, en la descripción de los métodos de solución, que se trata de minimizar). Formalmente, el problema se compone del espacio de soluciones S y la función objetivo f . Resolver el problema de optimización (S, f) consiste en determinar una solución ópti-

ma, es decir, una solución factible $x^* \in S$ tal que $f(x^*) \leq f(x)$, para cualquier $x \in S$.

Las soluciones alternativas se pueden expresar por la asignación de valores a algún conjunto finito de variables $X = \{X_i : i = 1, 2, \dots, n\}$. Si por U_i se denota al dominio o universo (conjunto de los valores posibles) de cada una de estas n variables, el problema consiste en seleccionar el valor x_i asignado a cada variable X_i del dominio U_i que, sometido a ciertas restricciones, optimiza una función objetivo f . El universo de soluciones se identifica con el conjunto $U = \{x = (x_i : i = 1, 2, \dots, n) : x_i \in U_i\}$. Las restricciones del problema reducen el universo de soluciones a un subconjunto de soluciones $S \subseteq U$, denominado espacio factible.

Los procedimientos de *búsqueda por entornos* recorren el espacio de soluciones U mediante un conjunto de transformaciones o movimientos. Las soluciones que se obtienen de otra mediante uno de los movimientos posibles se denominan vecinas de ésta y constituyen su *entorno*. El conjunto de movimientos posibles da lugar a una relación de vecindad y una *estructura de entornos* en el espacio de soluciones cuya elección es un aspecto trascendental en el éxito de los procesos de búsqueda. Además de una implementación y evaluación eficiente de los movimientos, las propiedades de la estructura de entorno resultante intervienen en esta elección. El esquema general de un procedimiento de búsqueda por entornos consiste en generar una solución inicial y, hasta que se cumpla el criterio de parada, seleccionar iterativamente un movimiento para modificar la solución. Las soluciones son evaluadas mientras se recorren y se propone la mejor solución del problema encontrada.

El entorno de una solución está constituido por las soluciones a las que se puede acceder desde ella por uno de los movimientos posibles. Formalmente, una *estructura de entornos* sobre un espacio o universo de búsqueda U es una función $E : U \rightarrow 2^U$ que asocia a cada solución $x \in U$ un entorno $E(x) \subseteq U$ de soluciones vecinas a x . Gran cantidad de métodos heurísticos propuestos en la literatura pertenece a la clase de procedimientos de búsqueda por entornos [57], [62].

La elección de la estructura de entornos es fundamental en el éxito de los procesos de búsqueda

ya que determina la calidad del conjunto de movimientos aplicados. Aparte de la factibilidad y el grado de mejora de los movimientos aplicados es importante la versatilidad de los mismos. Los movimientos combinados aparecen al ejecutar sucesivamente varios movimientos sobre una solución. Una adecuada combinación de movimientos enriquece los entornos, con lo que se pueden realizar pasos más amplios en el acercamiento al óptimo, pero se corre el riesgo de perjudicar la eficiencia del algoritmo al tener que contemplar un número mayor de movimientos posibles en el proceso de selección.

Otra característica importante de los movimientos es la factibilidad de las soluciones aportadas. Los movimientos factibles son aquellos que siempre proporcionan una solución factible. Esto puede estar ligado o no al hecho de que se aplique sólo a soluciones factibles. En muchos casos, aplicar movimientos más simples, pero no necesariamente factibles, y descartar las soluciones producidas que no sean factibles, es menos eficiente que adaptar el diseño de los movimientos para que sean factibles, sobre todo cuando dicha comprobación es costosa o cuando la probabilidad de que resulte factible es baja. Formalmente, los procedimientos que sólo consideran movimientos factibles están asociados al concepto, algo más restrictivo, de estructura de entornos como una función $E : S \rightarrow 2^S$ que asocia a cada solución factible $x \in S$ un entorno $E(x) \subset S$ de soluciones factibles vecinas a x .

Las principales metaheurísticas de búsqueda por entornos que se describen más adelante se centran sólo en el procedimiento de selección del movimiento. Sin embargo, existen otras cuestiones relevantes en el éxito del procedimiento de búsqueda por entornos. Aparte de la selección de la propia estructura de entornos sobre la que articular la búsqueda, cuestiones importantes son: la evaluación de la función objetivo, el procedimiento de generación de la solución inicial y el criterio de parada.

La posibilidad de realizar una evaluación eficiente de la solución obtenida tras el movimiento es especialmente importante en aquellos problemas en los que la evaluación de la función objetivo sea costosa. Son aplicables las pautas de las metaheurísticas de relajación para evitar cómputos excesivos en la obtención de valoraciones exactas que no son imprescindibles en

la conducción de la búsqueda. Además, se puede contar con procedimientos que evalúan la calidad de los movimientos sin tener que realizar una evaluación completa de la nueva solución desde cero. Para ello se utilizan procedimientos que actualizan rápidamente el valor de la función objetivo tras el movimiento, utilizando el valor anterior y los cambios producidos por el movimiento.

Las pautas de las metaheurísticas constructivas se utilizan para el diseño del procedimiento de generación de la solución inicial. En este sentido, las características fundamentales son la calidad y dispersión de las soluciones iniciales desde la que iniciar la búsqueda. La metaheurística GRASP propone un procedimiento para conseguir un conjunto de diferentes soluciones de alta calidad.

Por último, otra cuestión importante que afecta a cualquier procedimiento de solución de un problema emanado de una metaheurística de búsqueda por entornos es la condición de parada. Los criterios más corrientes se refieren a un límite al número de iteraciones, movimientos, operaciones elementales o tiempo de cómputo total o sin que se produzca alguna mejora.

Dos características fundamentales en el procedimiento de búsqueda por entorno resultante de aplicar metaheurísticas son las capacidades de exploración y de explotación. La *exploración* se refiere a la capacidad del método para explorar las diferentes regiones del espacio de búsqueda para alcanzar la zona en la que se encuentra la solución del problema. La *explotación* de la búsqueda se refleja en el esfuerzo y capacidad por mejorar las soluciones con las que trabaja el procedimiento. Existe un amplio consenso en que estas dos características deben modularse adecuadamente para conseguir el éxito práctico de las aplicaciones de las metaheurísticas.

3.1. Búsquedas Locales

El término *local* se emplea con bastante frecuencia en los estudios teóricos y prácticos del campo de las metaheurísticas de búsqueda. Las estructuras de entorno suelen reflejar algún concepto de proximidad o vecindad entre las soluciones alternativas del problema. Por tanto, el análisis del entorno de la solución actual en el

recorrido de búsqueda para decidir cómo continuarla representa un estudio local del espacio de búsqueda. Por tanto, una búsqueda local es un proceso que, dada la solución actual en la que se encuentra el recorrido, selecciona iterativamente una solución de su entorno. Las *metaheurísticas de búsqueda local* establecen pautas de selección de esta solución del entorno de la solución actual dando lugar a búsquedas locales heurísticas con alto rendimiento. Las búsquedas locales no informadas sólo tienen en cuenta la estructura de entornos para guiar la búsqueda. Las búsquedas monótonas utilizan la evaluación de la función objetivo para admitir sólo cambios en la solución actual que supongan una mejora. Por tanto, las búsquedas locales monótonas quedan atrapadas al llegar a una solución que no admite mejora dentro de su entorno. Las búsquedas globales emplean diversos métodos para escapar de esta situación. A continuación analizamos los aspectos más relevantes de las metaheurísticas para estos procedimientos.

3.1.1. Búsquedas no informadas

Las estrategias de búsqueda por entornos no informadas son aquellas búsquedas locales que sólo prestan atención a la estructura de entornos en el espacio de búsqueda y no utilizan información acerca del valor de la función objetivo en las soluciones encontradas. Las metaheurísticas de búsqueda no informadas aportan estrategias para organizar la exploración eficiente del espacio de búsqueda. Cuando estas pautas se aplican a la exploración del entorno en las búsquedas locales se traducen en metaheurísticas de búsqueda por entornos no informadas. Las metaheurísticas de búsqueda por entornos exhaustiva, parcial y aleatoria son las metaheurísticas de búsqueda no informadas más usuales.

Un recorrido exhaustivo de un espacio de búsqueda es el que incluye todos y cada uno de los elementos del espacio. Si el espacio de búsqueda es finito y no excesivamente grande, un procedimiento rudimentario para resolver el problema consiste en implementar un recorrido exhaustivo hasta encontrar la solución. En un problema de optimización, la búsqueda exhaustiva consiste en realizar un recorrido exhaustivo del espacio de soluciones del problema y tomar la mejor de ellas. Un recorrido exhausti-

vo del espacio se consigue empleando una ordenación (implícita o explícita) de todas las soluciones del espacio y utilizando una transformación que obtenga en cada iteración la solución siguiente en dicha ordenación. El procedimiento de generación de la solución inicial debe proporcionar la primera solución de dicha ordenación y el criterio de parada detectar cuándo se ha completado todo el espacio de búsqueda. La ordenación puede comprender sólo las soluciones factibles o un conjunto que las contenga. En este caso sólo habrá que considerar las soluciones factibles para elegir la mejor. A partir de la representación de las soluciones del espacio se determina la ordenación natural consistente en ir modificando sucesivamente los elementos que componen la solución. Dada una estructura de entornos para un problema, la búsqueda por entornos exhaustiva recorrerá sucesivamente y de forma exhaustiva los entornos de las soluciones visitadas. Si la estructura de entornos enlaza todas las soluciones del espacio, la búsqueda será exhaustiva, pero será necesario evitar o controlar las repeticiones para impedir que se cicle indefinidamente.

En algunas circunstancias puede ser suficiente examinar sólo una parte del espacio de búsqueda para obtener una visión global de todo el espacio. Las metaheurísticas de *búsqueda parcial* establecen las pautas para organizar la selección de las soluciones a examinar. Para un problema de optimización, la búsqueda parcial aportará la mejor entre las soluciones examinadas como propuesta de solución. Si las soluciones a examinar se seleccionan de forma completamente al azar se trata de una búsqueda parcial aleatoria pura, conocida como método de *Monte Carlo*. La búsqueda parcial por entornos aleatoria aplica un método parcial para analizar el entorno de las soluciones del recorrido.

La metaheurística de *búsqueda por entornos aleatoria* consiste en seleccionar iterativamente al azar una solución del entorno de la solución actual. Se trata de un recorrido aleatorio puro o uniforme si la distribución de probabilidad en el entorno de la solución actual es uniforme o equiprobable. Para implementar esta metaheurística sólo es necesario disponer de un buen procedimiento que seleccione una solución vecina de la región factible, y una forma rápida de evaluar la nueva solución. La búsqueda se intensifica si la solución del entorno se selecciona

de entre varias soluciones vecinas generadas al azar. La explotación de la búsqueda se ve incluso aumentada si el método de selección de las soluciones a examinar favorece a las de mayor calidad, o a las que se presume que lo van a ser, denominadas soluciones prometedoras. Por otro lado, si la selección parcial de las soluciones a examinar se realiza de forma que se evite la repetición de soluciones examinadas, se obtendrá un mejor aprovechamiento del tiempo de cómputo. La búsqueda parcial sistemática persigue evitar estas repeticiones manteniendo un alto grado de aleatoriedad. Una búsqueda parcial sistemática se obtiene de un recorrido exhaustivo deteniendo la búsqueda sin necesidad de llegar a completar todo el espacio de soluciones. Si la parte del espacio recorrido es pequeña y las soluciones consecutivas, en la ordenación del espacio utilizada, son similares, la visión parcial del espacio de búsqueda sería demasiado sesgada. Para evitar este inconveniente se realiza la búsqueda parcial mediante un recorrido sistemático con arranque aleatorio. Esta estrategia consiste en determinar al azar una solución de arranque y una amplitud de paso no unitario para el recorrido. Además la ordenación es interpretada de forma cíclica (la siguiente de la última solución es la primera) para que el recorrido no se detenga al llegar al final de la ordenación. El recorrido sistemático de m elementos en un conjunto ordenado de n elementos se obtiene fijando una posición de arranque r y una amplitud de paso t . Conviene elegir la amplitud de paso t de forma que $m \cdot t > n$ y tal que t y n sean números primos entre sí o, al menos, con un mínimo común múltiplo suficientemente alto. Dada una estructura de entornos para un problema, la búsqueda por entornos parcial recorrerá sucesivamente y de forma parcial los entornos de las soluciones visitadas. El número de soluciones visitadas en el recorrido de cada entorno determina la intensidad de la búsqueda cuya regulación puede ser estática o dinámica.

3.1.2. Búsquedas Locales Monótonas

Las metaheurísticas de búsqueda anteriores no utilizan la información proporcionada por la evaluación de la función objetivo en la conducción de la búsqueda. Las estrategias de búsqueda pueden incorporar esta información al método de búsqueda para guiar los movimientos

aplicados. Las búsquedas informadas son aquellas que, explícita o implícitamente, utilizan información de la evaluación de la función objetivo. Las búsquedas locales (o por entornos) informadas son las que utilizan información de la función objetivo sólo en el entorno de la solución actual.

Las búsquedas monótonas sólo aceptan mejoras de la solución que realiza el recorrido. Las búsquedas locales monótonas son las búsquedas locales que sólo aplican movimientos que mejoren la solución actual del recorrido [2], [62], [63], [32]. Frecuentemente se interpreta que las búsquedas locales persiguen siempre una mejora en los alrededores de la solución actual, aunque el término *local* hace referencia sólo a que se realiza un análisis en el entorno de la solución actual para guiar la búsqueda. Las búsquedas monótonas no estrictas aceptan también nuevas soluciones que igualan a la solución actual. Estas estrategias presentan la ventaja de que pueden escaparse de las mesetas o zonas llanas del espacio de búsqueda, pero tienen el inconveniente de que podría ciclarse indefinidamente dentro de una de tales mesetas.

La metaheurística básica de búsqueda por entorno monótona aleatoria consiste en seleccionar iterativamente una solución al azar del entorno de la solución actual que es sustituida por ésta si se produce una mejora. La solución de partida se puede obtener por cualquier procedimiento arbitrario y el criterio de parada reflejará el estancamiento de la búsqueda en un mínimo local presumible cuando en un cierto número de intentos no se pueda mejorar la solución actual. Las metaheurísticas intensifican la búsqueda en torno a cada solución actual seleccionando el mejor entre una serie de soluciones del entorno obtenidas por un procedimiento del mismo tipo. La intensidad de la búsqueda viene dada por el número o la proporción de soluciones vecinas de la solución actual entre las que se toma la mejor. La metaheurística de intensificación oscilante consiste en hacer oscilar sistemáticamente entre dos valores extremos la intensidad de la búsqueda.

La metaheurística de intensificación oscilante dinámica regula dinámicamente la intensidad de la búsqueda para intensificarla, hasta hacerla exhaustiva al acercarse al óptimo local, pero sin necesidad de encontrar la mejor solución vecina al comenzar los descensos. Una estrategia

autónoma para esta regulación dinámica es, por ejemplo, aumentarla cada vez que no se mejore la solución, hasta alcanzar el tamaño del entorno, y disminuirla mientras se produzcan esas mejoras, sin llegar a anularla.

Las metaheurísticas de búsqueda sistemática mejoran el poder de exploración en el entorno de la solución actual haciendo que las soluciones vecinas entre las que se selecciona la mejor sean distintas. Los procedimientos de búsqueda obtienen una ventaja con esta estrategia si las modificaciones necesarias para garantizar que las soluciones vecinas evaluadas sean distintas no hacen computacionalmente más costoso el procedimiento. El procedimiento se puede implementar, por ejemplo, asumiendo la ordenación implícita del entorno de cada solución y aplicando un procedimiento de muestreo sistemático con arranque aleatorio. Esta ordenación puede venir dada de forma natural o se puede derivar del procedimiento exhaustivo.

Las metaheurísticas de búsqueda local exhaustiva maximizan el poder de explotación de la búsqueda local al examinar, si es necesario, todo el entorno de la solución actual. Las metaheurísticas *voraz* y *ansiosa* aparecen al aplicar las dos reglas fundamentales de selección de esta solución. La metaheurística voraz o (*Greedy*) con la regla de selección de *el mejor primero* y la metaheurística ansiosa (*Anxious*) con la regla de selección de *el primero mejor*. En la primera de ellas se selecciona siempre la mejor solución del entorno de la solución actual y en la segunda se selecciona la primera solución del entorno que mejore la solución actual. En la metaheurística por entornos voraz se recorren siempre todas las soluciones del entorno para seleccionar la mejor, mientras que en la metaheurística por entornos ansiosa se detiene el recorrido cuando se encuentre una solución del entorno mejor que la actual, pero el recorrido se continua de forma exhaustiva si no se encuentra tal mejora.

El punto desde el que comenzar el recorrido del entorno en la estrategia ansiosa es de gran importancia para aumentar la capacidad de exploración del procedimiento. Frente a la elección al azar de este punto, una mejora del poder de explotación de la búsqueda se obtiene si las primeras soluciones vecinas examinadas son las más prometedoras. Las metaheurísticas *golosas* procuran que las primeras soluciones vecinas

evaluadas tengan la mayor probabilidad posible de producir una mejora o que ésta sea de la mayor magnitud posible. El procedimiento se puede implementar usando alguna ordenación del entorno de la solución atendiendo a un análisis de la posible mejora producida por los movimientos mediante una estimación de la calidad de las nuevas soluciones.

3.2. Búsquedas Globales

El principal inconveniente de las búsquedas locales es que si se aproximan a una solución localmente óptima u óptimo local (una solución que es mejor que cualquiera de las de su entorno) la solución actual queda atrapada en su entorno [78], [2]. La regla de parada en las búsquedas monótonas implica detectar los mínimos locales analizando cuando no se mejora la solución actual. Una búsqueda con una perspectiva global del espacio de soluciones debe buscar herramientas para escapar de estas situaciones. Las principales metaheurísticas de búsqueda global surgen de las tres formas principales de escapar de esta situación: a) volver a comenzar la búsqueda desde otra solución inicial, b) modificar la estructura de entornos, y c) permitir movimientos de empeoramiento de la solución actual.

Estas tres opciones dan lugar, respectivamente, a la metaheurística con arranque múltiple, a la metaheurística de entorno variable y a las metaheurísticas de búsqueda no monótonas. La tercera de las opciones incluye diversas metaheurísticas relevantes entre las que destacan la búsqueda probabilística, representada fundamentalmente por el Recocido Simulado (*Simulated Annealing*), y la búsqueda con memoria o Búsqueda Tabú (*Tabu Search*).

Los procedimientos de búsqueda con *arranque múltiple* (*Multi-Start*) realizan varias búsquedas monótonas partiendo de diferentes soluciones iniciales [7], [24], [53], [64]. La búsqueda monótona implicada puede ser cualquiera de las anteriormente descritas. Una de las formas más simples de llevar esto a cabo consiste en generar una muestra de soluciones iniciales o de arranque. Esto es equivalente a generar al azar una nueva solución de partida cada vez que la búsqueda quede estancada en el entorno de una solución óptima local.

La Búsqueda por *Entornos Variables* (*Variable Neighborhood Search*, VNS) es una metaheurística reciente que consiste en cambiar de forma sistemática la estructura de entorno [34], [35], [36], [37], [38]. La idea original fue considerar distintas estructuras de entornos y cambiarlas sistemáticamente para escapar de los mínimos locales. El VNS básico obtiene una solución del entorno de la solución actual, ejecuta una búsqueda monótona local desde ella hasta alcanzar un óptimo local, que reemplaza a la solución actual si ha habido una mejora y modifica la estructura de entorno en caso contrario. La búsqueda descendente por entornos variables (VND) aplica una búsqueda monótona por entornos cambiando de forma sistemática la estructura de entornos cada vez que se alcanza un mínimo local.

Además de reiniciar la búsqueda y modificar la estructura de entornos, la otra vía para evitar quedarse atrapados en un óptimo local es admitir la posibilidad de pasos de no mejora, lo que da lugar a las estrategias de búsqueda no monótonas. Las metaheurísticas proponen principalmente controlar la aceptación de movimientos que no sean de mejora para que, al menos a la larga, se vayan mejorando las soluciones encontradas, y utilizar información histórica del proceso de búsqueda para controlar cuando el recorrido se está estancando en un mínimo local y evitar la formación de ciclos. Las metaheurísticas fundamentales que aplican estas estrategias son el Recocido Simulado y la Búsqueda Tabú.

Con las *metaheurísticas de búsqueda probabilísticas* se selecciona aleatoriamente un vecino de la solución actual que la reemplaza con cierta probabilidad. Por ejemplo, con probabilidad 1 si tiene mejor valor objetivo, y con una probabilidad menor que 1 si su valor objetivo es peor. Si el número de iteraciones es elevado, la búsqueda puede escapar de cualquier óptimo local si la probabilidad de aceptar peores soluciones va decreciendo. Generalmente la probabilidad de aceptar una solución peor es función del empeoramiento de forma que, a menor diferencia en el valor objetivo, hay mayor probabilidad de ser aceptada. El *Recocido Simulado* [42], [48], [72], [14] es el caso más importante de las metaheurísticas de búsqueda global con criterio de aceptación probabilístico. Se usa una probabilidad de aceptación de nuevas soluciones peores que es función exponencial de la mo-

dificación de la función objetivo. Otras metaheurísticas simplemente reducen o incrementan esta probabilidad para modular la exploración y explotación de la búsqueda. Las metaheurísticas de umbrales de aceptación (*Threshold Accepting*) [16] aceptan las nuevas soluciones peores que no sobrepasen el umbral y modulan este umbral con el mismo propósito.

Las *metaheurísticas de búsqueda con memoria* representada por la Búsqueda Tabú comprenden las estrategias que tratan de utilizar la memoria del proceso de búsqueda para mejorar su rendimiento. Está fundamentada en las ideas expuestas por F. Glover en 1986 [20] que ha contribuido con diversos trabajos [21], [22], [29], [28], [27], [26] así como lo han hecho otros muchos autores en una extensa relación de artículos. En el origen del método el propósito era sólo evitar la reiteración en una misma zona de búsqueda recordando las últimas soluciones recorridas. Sin embargo, posteriormente se han realizado diversas propuestas para rentabilizar la memoria a medio o largo plazo.

La forma más directa de introducir la memoria en el procedimiento de búsqueda no monótono es considerar una función de aceptación que tenga en cuenta la historia de la búsqueda. El procedimiento elemental de búsqueda tabú evita la repetición prematura de las mismas soluciones en el recorrido, para lo que prohíbe que las últimas soluciones vuelvan a utilizarse en el recorrido de búsqueda. Se utiliza un parámetro t que determina el número de las últimas soluciones que son temporalmente prohibidas como nuevas soluciones actuales.

Estas estrategias se pueden aplicar dentro de la estructura de la búsqueda general de dos formas: introduciendo una función de aceptación que determine cuándo se acepta la nueva solución generada o modificando el procedimiento de generación del movimiento a aplicar a la solución actual. Con la primera de estas alternativas la función de aceptación puede incluir en sus parámetros información referente a la historia y el estado de la búsqueda, y a la solución generada. En el segundo caso, el procedimiento de generación de movimiento debe tener un diseño en el que se generan las soluciones vecinas de acuerdo con algún criterio que tenga en cuenta información de la historia y el estado de la búsqueda.

La *Búsqueda Reactiva (Reactive Search)* [4], [3] es una metaheurística que propone usar, dentro de la búsqueda tabú, la información a largo plazo obtenida del recorrido. Se persigue detectar indicios de que la búsqueda necesita incrementar su exploración, por la repetición de ciertas estructuras o patrones en las soluciones recientemente visitadas. Esta información se almacena y se accede a ellas utilizando técnicas eficientes de dispersión (*hashing*) o de árboles de búsqueda usuales en gestión de grandes cantidades de datos. Según la información que se tenga almacenada en cada iteración se activa un proceso reactivo para alejarse de la zona de estancamiento.

3.3. Búsquedas basadas en poblaciones

En una búsqueda en grupo o basada en poblaciones se sustituye la solución actual que recorre el espacio de soluciones, por un conjunto de soluciones que lo recorren conjuntamente interactuando entre ellas. Además de los movimientos aplicables a las soluciones que forman parte de este conjunto, denominado grupo o población de búsqueda, se contemplan otros operadores para generar nuevas soluciones a partir de las ya existentes.

Las estrategias de búsqueda en grupo se iniciaron con el famoso *Algoritmo Genético* propuesto en [39]. En la actualidad adoptan diversas características cómo se puede observar en la gran cantidad de trabajos editados sobre este tipo de procedimientos [9], [12], [30], [56], [59] y [66] (ver también la monografía num. 5 de 1998 en esta misma publicación). A continuación se describen las cuestiones fundamentales de su implementación para la solución de problemas de optimización.

En primer lugar, se establece una codificación apropiada de las soluciones del espacio de búsqueda y una forma de evaluar la función objetivo para cada una de estas codificaciones. Las soluciones se identifican con individuos que pueden formar parte de la población de búsqueda. La codificación de una solución se interpreta como el cromosoma del individuo compuesto de un cierto número de genes a los que les corresponden ciertos alelos. Se consideran dos operaciones básicas: la mutación y el cruce. La

mutación de un individuo consiste en modificar un gen cambiando, al azar, el alelo correspondiente. El cruce de dos individuos (llamados padres) produce un individuo hijo tomando un número k (elegido al azar) de genes de uno de los padres y los $t - k$ del otro. La población evoluciona de acuerdo a las estrategias de selección de individuos, tanto para las operaciones como para la supervivencia. La selección se puede hacer simulando una lucha entre los individuos de la población con un procedimiento que, dados dos individuos selecciona uno de ellos teniendo en cuenta su valoración (la función objetivo) y la adaptación al ambiente y a la población (criterios de diversidad, representatividad). La lucha por la supervivencia tiene por objeto mantener controlado el tamaño de la población. La selección de los luchadores se puede hacer de diferentes maneras: dos individuos seleccionados al azar, cada nuevo individuo con otro seleccionado al azar o con el peor de los existentes, etc. Entre las metaheurísticas derivadas de los algoritmos genéticos destacan los *Algoritmos meméticos* [60] [11], que surgen de combinar los algoritmos genéticos con búsquedas locales.

Los *Algoritmos de Estimación de Distribuciones* (EDA) [51], [47] son algoritmos evolutivos que usan una colección de soluciones candidatas para realizar trayectorias de búsqueda evitando mínimos locales. Estos algoritmos usan la estimación y simulación de la distribución de probabilidad conjunta como un mecanismo de evolución, en lugar de manipular directamente a los individuos que representan soluciones del problema. Un algoritmo EDA comienza generando aleatoriamente una población de individuos. Se realizan iterativamente tres tipos de operaciones sobre la población. El primer tipo de operación consiste en la generación de un subconjunto de los mejores individuos de la población. En segundo lugar se realiza un proceso de aprendizaje de un modelo de distribución de probabilidad a partir de los individuos seleccionados. En tercer lugar se generan nuevos individuos simulando el modelo de distribución obtenido. El algoritmo se detiene cuando se alcanza un cierto número de generaciones o cuando el rendimiento de la población deja de mejorar significativamente.

El enfoque de la metaheurística de *Búsqueda Dispersa* (o *Scatter Search*) [46], [44], [54] contempla el uso de un conjunto de referen-

cia de buenas soluciones dispersas que sirve, tanto para conducir la búsqueda, mejorando las herramientas para combinarlas adecuadamente, como para mantener un grado satisfactorio de diversidad. La propuesta inicial se originó en estrategias para crear reglas de decisión compuestas [23]. Algunos estudios recientes demuestran las ventajas prácticas de este enfoque para resolver diversos problemas de optimización clásicos y reales. La Búsqueda Dispersa se distingue de otros procedimientos en los mecanismos de intensificación y diversificación que explotan la memoria adaptada recurriendo a los fundamentos que unen el Scatter Search a la Búsqueda Tabú.

El *reencadenamiento de camino* (PR, *Path Re-linking*) [23], [43], [44] es una metaheurística asociada a la búsqueda dispersa que utiliza la información que se obtiene de las mejores soluciones. Esta información se aprovecha en las mejoras de otras soluciones que se encuentran posteriormente. Básicamente se trata de generar soluciones explorando las trayectorias que conectan soluciones de alta calidad. Partiendo de una de estas soluciones se genera un camino de soluciones hacia la otra solución incorporando a la primera atributos de la segunda. Este camino se construye tomando cada vez el atributo de la segunda solución que lo hace más cercano a ella. A continuación se toman, como puntos de arranque para nuevas fases de mejora, una o varias de las soluciones del recorrido anterior.

3.4. Otras metaheurísticas de Búsqueda

Se han propuesto otras metaheurísticas de cierta relevancia, algunas de las cuales presentan como novedad estar inspiradas en distintos fenómenos de la naturaleza. Entre ellas destacan las redes neuronales, las colonias de hormigas, las bandadas de aves o bancos de peces. Otras metaheurísticas tienen el mérito de aplicar herramientas muy exitosas en otros campos de la IA, como la metaheurística FANS o los métodos inteligentes de realizar búsqueda locales.

Las *redes neuronales artificiales* [49] surgieron como modelos abstractos de sistemas nerviosos naturales formados por unidades de cómputo,

llamadas neuronas, interconectadas. Estos modelos tienen la capacidad de ajustar sus parámetros en respuesta a unas entradas y salidas mejorando alguna función. Asociando los estados de la red a soluciones de un problema y utilizando el objetivo como referente, consiguen aproximarse al estado que corresponde con la solución óptima. La mayoría de las redes neuronales aplicadas para resolver problemas de optimización son versiones de la red de Hopfield [40]. La red de Hopfield puede auto ajustarse para alcanzar el estado de mínima energía. La idea básica consiste en transformar el problema de optimización en la minimización de la función de energía de la red de Hopfield y determinar la estructura de una red neuronal de forma que las situaciones de energía mínima correspondan al estado de equilibrio de la red. De esta forma, la red evoluciona hacia el estado de equilibrio proporcionando la solución del problema. La principal ventaja de las redes se obtiene cuando, tras resolver el problema y disponer del estado de la red correspondiente, una modificación del modelo se traduce en una modificación de la red que provoca un rápido reajuste del equilibrio proporcionando la nueva solución al problema. Otras ventajas de las redes neuronales al resolver problemas combinatorios son su paralelización y la posibilidad de usar hardware específico. Otros modelos basados en redes neuronales aplicadas con éxito a problemas de optimización combinatoria son las máquinas de Boltzman y las redes competitivas WTA. Las máquinas de Boltzmann son un híbrido entre una red de Hopfield y la técnica de recocido simulado [1]. Las redes del tipo WTA (*Winner-Take-All*) [76] son modelos de redes neuronales competitivas que seleccionan de un conjunto de candidatos el elemento que maximiza el valor de activación siguiendo un sistema competitivo. Una revisión de la literatura de la aplicación de redes neuronales a problemas de optimización puede encontrarse en [49].

La metaheurística de *sistemas de hormigas Ant Systems*) empleada estrategias inspiradas en el comportamiento de las colonias de hormigas para descubrir fuentes de alimentación, al establecer el camino más corto entre éstas y el hormiguero y transmitir esta información al resto de sus compañeras [15], [17], [18].

La optimización *extrema* o extremal (EO, *Extreme Optimization*) [8] es una metaheurística inspirada en procesos auto-organizativos fre-

cuentemente encontrados en la naturaleza. La idea central es utilizar modelos de evolución de ecosistemas que, en lugar de seleccionar los mejores elementos, llevan a la extinción a las componentes mal adaptadas del sistema. La idea básica del método es eliminar sucesivamente las componentes extremadamente indeseables de las soluciones subóptimas. El método actúa sobre una única solución, y no sobre un conjunto de soluciones o población como los algoritmos genéticos, modificando el atributo de menor nivel de adaptación (y aquellos afectados por este cambio) aplicando algún tipo de transformación o movimiento.

La *optimización de partículas inteligentes* (PSO, *Particle Swarm Optimization*) [41] es una Metaheurística evolutiva inspirada en el comportamiento social de las bandadas de pájaros o bancos de peces. Las soluciones, llamadas partículas se “echan a volar” en el espacio de búsqueda guiadas por la partícula que mejor solución ha encontrado hasta el momento y que hace de líder de la bandada. Cada partícula evoluciona teniendo en cuenta la mejor solución encontrada en su recorrido y al líder. El procedimiento también tiene en cuenta el mejor valor alcanzado por alguna de las partículas en su entorno. En cada iteración, las partículas modifican su velocidad hacia la mejor solución de su entorno teniendo en cuenta la información del líder.

La *Búsqueda Local Iterada* (ILS, *Iterated Local Search*) [50] es una metaheurística que propone un esquema en el se incluye una heurística base para mejorar los resultados de la repetición de dicha heurística. Esta idea ha sido propuesta en la literatura con distintos denominaciones, como descenso iterado, grandes pasos con cadenas de Markov, Lin-Kerningan iterado, búsqueda perturbada o ruidosa o la búsqueda de entorno variable con agitación donde la solución aportada por una heurística de búsqueda por entornos es *agitada* para producir una solución de partida para la heurística de búsqueda. La estrategia ILS actúa de la siguiente forma: dada una solución obtenida por la aplicación de la heurística base, se aplica un cambio o alteración que da lugar a una solución intermedia. La aplicación de la heurística base a esta nueva solución aporta una nueva solución que, si supera un test de aceptación, pasa a ser la nueva solución alterada. Aunque la heurística base incluida suele ser una búsqueda local, se ha propuesto aplicar

cualquier otra metaheurística, determinística o no. De esta forma, el proceso se convierte en una búsqueda estocástica por entornos donde los entornos no se explicitan sino que vienen determinados por la heurística base.

La metaheurística de *concentración* (*Concentration Heuristics*) [68] trata de combinar la información proporcionada por soluciones de calidad para realizar búsquedas locales. Básicamente consiste en, una vez obtenido un conjunto de concentración formado por buenas soluciones, abordar la búsqueda en una zona restringida a partir de la información proporcionada por dicho conjunto en el que se concentra la heurística.

La metaheurística de *búsqueda local guiada* (GLS, *Guided Local Search*) consiste básicamente en una secuencia de procedimientos de búsqueda local; al finalizar cada uno de ellos se modifica la función objetivo penalizando determinados elementos que aparecen en el óptimo local obtenido en el último paso, estimulando de esta forma la diversificación de la búsqueda [74], [58], [75]. Otras metaheurísticas utilizan un tipo de “ruido” para alterar aleatoria y sistemáticamente elementos del problema como la metaheurística con ruido (NMH, *Noising Methods heuristics*) [10] y la metaheurística de *per-turbación* [69].

La Metaheurística de búsqueda fuzzy adaptativa por entornos (FANS, *Fuzzy Adaptive Neighborhood Search* [71], [?]) usa valoraciones borrosas o difusas para medir el grado con que se consideran las soluciones con ciertas propiedades lo que se usa para modificar la estructura de entorno.

La *programación por restricciones* (*Constraint Programming*) [13], [52], [19] puede considerarse una metaheurística muy general que constituye un paradigma propio dentro de las metaheurísticas, donde lo más relevante es la atención que se le presta al tratamiento de las restricciones que surgen en un problema y como afecta a los procedimientos de búsqueda de soluciones.

4. Propiedades deseables

En esta sección analizamos un conjunto de propiedades *deseables* de las metaheurísticas. Son propiedades deseables todas aquellas que favorezcan el interés práctico y teórico de las metaheurísticas. Indicarán direcciones a las que dirigir los esfuerzos para contribuir al desarrollo científico e ingenieril, pero no será posible mejorar todas las propiedades a la vez, dado que algunas son parcialmente contrapuestas. Una relación de tales propiedades debe incluir las siguientes:

Simple. La metaheurística debe estar basada en un principio sencillo y claro; fácil de comprender.

Precisa. Los pasos y fases de la metaheurística deben estar formulados en términos concretos.

Coherente. Los elementos de la metaheurística debe deducirse naturalmente de sus principios.

Efectiva. Los algoritmos derivados de la metaheurística deben proporcionar soluciones de muy alta calidad; óptimas o muy cercanas a las óptimas.

Efcaz. La probabilidad de alcanzar soluciones óptimas de casos realistas con la metaheurística debe ser alta.

Eficiente. La metaheurística debe realizar un buen aprovechamiento de recursos computacionales; tiempo de ejecución y espacio de memoria.

General. La metaheurística debe ser utilizable con buen rendimiento en una amplia variedad de problemas.

Adaptable. La metaheurística debe ser capaz de adaptarse a diferentes contextos de aplicación o modificaciones importantes del modelo.

Robusta. El comportamiento de la metaheurística debe ser poco sensible a pequeñas alteraciones del modelo o contexto de aplicación.

Interactiva. La metaheurística debe permitir que el usuario pueda aplicar sus conocimientos para mejorar el rendimiento del procedimiento.

Múltiple. La metaheurística debe suministrar

diferentes soluciones alternativas de alta calidad entre las que el usuario pueda elegir.

Autónoma. La metaheurística debe permitir un funcionamiento autónomo, libre de parámetros o que se puedan establecer automáticamente.

Varias de estas propiedades están muy relacionadas y apuntan en la misma dirección, como la simplicidad, la precisión y la coherencia. La *simplicidad* de la metaheurística facilita su uso y contribuye a dotarla de amplia aplicabilidad. La descripción formal de las operaciones debe liberarse de la analogía física o biológica que haya sido la fuente inicial de inspiración para permitir mejoras que no respeten la analogía. La *precisión* en la descripción de los elementos que componen la metaheurística es crucial para concretar un procedimiento de alta calidad; fácil de implementar. Los pasos de los procedimientos básicos de los algoritmos deben traducirse coherentemente de los principios en que se inspira. Debe huirse de sentencias sin sentido o vagas. Frecuentemente se presentan como extensiones de una metaheurística la incorporación de herramientas o recursos computacionales estándares, o de pautas de otras metaheurísticas cuando en realidad deben calificarse como hibridaciones de las mismas.

La evaluación del rendimiento de una metaheurística debe atender tanto a la eficiencia como a la efectividad y eficacia de los procedimientos heurísticos obtenidos. Para validar la *efectividad* y *eficacia* de una metaheurística, éstas deben afrontar con éxito problemas de un banco de casos reales para los que se conozcan las soluciones. Si no se dispone de estos casos, se deben construir recurriendo a procesos de simulación que se aproximen a tales circunstancias. La *eficiencia* del método se contrasta experimentalmente en el empleo de un tiempo computacional moderado (o al menos razonable) para alcanzar éxito en los problemas considerados. El tamaño de los problemas considerados en las aplicaciones prácticas de los métodos de optimización se limita por las herramientas disponibles para resolverlos más que por la necesidad de los potenciales usuarios. Cuando las metaheurísticas se aplican a instancias realmente grandes, sus fortalezas y debilidades aparecen más claramente. Las metaheurísticas pueden mejorar su rendimiento extendiéndose en varias direcciones y, posiblemente, hibridizándose. Los procedimientos

heurísticos resultantes se complican y usan muchos parámetros. Con ello se puede mejorar su eficiencia, pero enmascaran las razones de su éxito. En algunas ocasiones la alta especialización de una metaheurística lleva a un ajuste fino de parámetros sobre algún conjunto de entrenamiento concreto.

La aplicabilidad de una metaheurística debe estar sustentada en la generalidad, pero también en su adaptabilidad y robustez. La *robustez* tiene que ser contrastada experimentalmente analizando el rendimiento frente a fluctuaciones de las características de los problemas. La robustez se refleja en que el número de parámetros que hay que fijar en las distintas aplicaciones se mantiene bajo. La *generalidad* de una metaheurística se refleja en la diversidad de los campos de aplicación para los que se han utilizado con éxito. La *adaptabilidad* permite que las conclusiones obtenidas al afrontar un tipo de problemas particular puedan ser aprovechadas en otros contextos. Las pautas proporcionadas por una metaheurística de búsqueda se aplican a descripciones asociadas a un problema, referidas simplemente a los movimientos posibles para transformar una solución en otra y la forma de evaluarlas.

Para favorecer la utilidad de la metaheurística en la resolución de problemas reales, por ejemplo incorporándolo a Sistemas de Ayuda a la Decisión, son importantes las propiedades que propicien un *interface* amigable. La *interactividad* de los sistemas basados en las metaheurísticas favorece la colaboración con otros campos que proporcionan conocimientos específicos de los problemas para mejorar el rendimiento de la metaheurística. La posibilidad de ofrecer diversas soluciones de alta calidad, realmente diferentes, entre las que los decisores puedan optar contribuye a diseminar su uso. La relativa *autonomía* de implementaciones de la metaheurística permite ganarse la confianza de usuarios poco expertos en optimización o en los campos de aplicación.

Una característica que contribuye a divulgar una metaheurística es la novedad a la que va asociada, en cuanto a la originalidad de los principios que la inspiran y a los campos de repercusión social a los que se aplica. Este aspecto se revela, por ejemplo, en la inspiración en fenómenos naturales de los algoritmos genéticos y otras metaheurísticas, en la aplicación a la

demostración matemática de la metaheurística de entorno variable, y en la aplicación a la ingeniería genética de las técnicas FANS. Sin embargo, en los entornos científicos, tecnológicos, ingenieril o empresarial, el aspecto más relevante es el éxito asociado a la eficiencia y efectividad de los algoritmos derivados de cada metaheurística en la resolución de problemas de gran tamaño o surgidos en aplicaciones reales.

5. Conclusiones

Para la resolución práctica de una proporción cada vez mayor de problemas de interés, no resulta apropiado utilizar procedimientos diseñados a propósito para cada modelo y dependientes de su estructura particular. Ante la necesidad de utilizar algoritmos heurísticos de solución, las metaheurísticas proporcionan pautas y estrategias generales de diseño para obtener heurísticas con un alto rendimiento. Las metaheurísticas proporcionan métodos para escaparse de los óptimos locales de mala calidad por lo que, dado que el valor de tales óptimos locales frecuentemente difiere considerablemente del valor del óptimo global, el impacto práctico de las metaheurísticas ha sido inmenso.

Se observan diversas tendencias en las investigaciones sobre técnicas metaheurísticas. Unas tratan de mantener la *pureza* de los métodos y comprobar su efectividad en nuevos problemas, sin incorporar herramientas de otras metaheurísticas. Otras investigaciones, desde una perspectiva más ingenieril, tratan de aprovechar los recursos proporcionados por cada una de ellas. Para estos últimos, la única cuestión relevante es conocer si el beneficio en el rendimiento, proporcionado por la inclusión de tales herramientas, compensa al esfuerzo de su implementación y al incremento de la complejidad de los códigos resultantes.

El campo de investigación sobre las metaheurísticas ofrece más oportunidades para aplicar la intuición que la deducción. En contraste con el éxito práctico de muchas metaheurísticas, el estudio teórico está más retrasado. Frecuentemente se obtienen buenas nuevas heurísticas, con algo de inventiva y gran esfuerzo en el ajuste de numerosos parámetros, pero las razones de por qué funcionan tan bien per-

manecen desconocidas. La situación es incluso peor para los híbridos, donde las aportaciones de las metaheurísticas implicadas y el beneficio de la interacción raramente son objetos de un estudio experimental bien diseñado.

Algunas propuestas encaminadas a una mejor comprensión de estos aspectos son el estudio de la influencia de la topografía de los óptimos locales y de las trayectorias seguidas por los procesos de búsqueda heurística. El análisis de la evolución de las distancias al óptimo frecuentemente se centran exclusivamente en la desviación del objetivo alcanzado frente al mejor posible. Se puede obtener información más útil si se consideran distancias entre las propias soluciones y no sólo su valor.

Los intentos por organizar este campo son numerosos, pero los conceptos principales son raramente definidos con precisión y hay todavía muy pocos teoremas significativos. Ninguna estructura ha conseguido una aceptación general. Más bien, cada grupo de investigación inspirador de una metaheurística tiene su propio punto de vista y habilidad para explicar muchas heurísticas en su propio vocabulario así como para absorber ideas de todo el campo (generalmente bajo la forma de híbridos).

La peor consecuencia de este hecho es la tendencia a la proliferación de reclamaciones de prioridades basadas en evidencias tan vagas que son difíciles de evaluar. Con algunos argumentos o la reutilización de términos en la descripción de unas metaheurísticas y otras, se puede interpretar que una de ellas es la otra definida de manera incompleta (si no se especifica algún elemento importante o es descrito por alguna vaga metáfora) o como un caso particular, al restringir el tipo de herramienta aplicada a un tipo de problema. Esto sería igualmente arbitrario. Parece que el carácter babélico de la investigación en metaheurísticas es, esperemos que temporalmente, ligeramente deshonesto. Mientras esto permanezca así, éxitos claros en problemas particulares serán más importantes para evaluar las metaheurísticas que largas controversias. Finalmente, cuando se consideren globalmente las cualidades deseables de las metaheurísticas, las comparativas de eficiencia no tendrían el papel tan dominante, algunas veces exclusivo, que se les da en muchos artículos. El propósito de estas investigaciones debe ser la comprensión de las metaheurísticas,

no la competición entre ellas. Otras cualidades de las heurísticas y las metaheurísticas distintas que la eficiencia pueden ser tan importantes a la larga, como la simplicidad, la precisión, la robustez, y, sobre todo la, amigabilidad.

Referencias

- [1] E.H.L. Aarts y J. Korst. *Simulated annealing and Boltzmann machines: a stochastic approach to combinatorial optimization and neural computing*. Wiley, 1989.
- [2] E.H.L. Aarts y J.K. Lenstra. *Local Search in Combinatorial Optimization*. Wiley, 1996.
- [3] R. Battiti. Reactive search: towards self-tuning heuristics. en V.J. Rayward-Smith, I.H. Osman, C.R. Reeves y G.D. Smith(eds.) *Modern heuristic search methods*, 61–83, Wiley, 1996.
- [4] R. Battiti y G. Tecchiolli. The reactive tabu search. *ORSA Journal of Computing*, 6:126–140, 1994.
- [5] J.E. Beasley Lagrangian Relaxation en C.R. Reeves (ed.) *Modern heuristic techniques for combinatorial problems*, 243–303, Blackwell Scientific Publications, 1993
- [6] A. Blanco, J.L. Verdegay, D. Pelta. Fans: una heurística basada en conjuntos difusos para problemas de optimización. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [7] C.G.E. Boender y A.H.G. Rinnooy Kan. Bayesian stopping rules for multistart global optimization methods. *Mathematical Programming*, 37:59–80, 1987.
- [8] S. Boettcher y A.G. Percus. Nature's way of optimizing. *Artificial Intelligence*, 119:275–286, 2000.
- [9] B.P. Buckles y F.E. Petry. *Genetic Algorithms*. IEEE Computer Society Press, 1992.
- [10] I. Charon y O. Hudry. The noising methods: A generalization of some metaheuristics. *European Journal of Operational Research*, 135:86–101, 2001.
- [11] C. Cotta-Porras y P. Moscato. Una introducción a los algoritmos meméticos. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [12] L. Davis (ed.). *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York, 1991.
- [13] R. Dechter. *Constraint Processing*. Morgan Kaufmann, 2003.
- [14] B.A. Díaz y K. Dowsland. Diseño de heurísticas y fundamentos del recocido simulado. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [15] M. Dorigo, V. Maniezzo y A. Colomi. Ant System: Optimization by a Colony of Co-operating Agents. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, 26:1, 29–41, 1996.
- [16] G. Dueck. New optimization heuristics: The great deluge algorithm and the record-to-record travel. *Journal of Computational Physics*, 104:86–92, 1993.
- [17] M. Dorigo, E. Bonabeau y T. Theraulaz. *From Natural to Artificial Swarm Intelligence*. Oxford University Press, 1999.
- [18] M. Dorigo y T. Stutzle. The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances. Cap. 9 en F. Glover y G. Kochenberger (eds.) *Handbook on MetaHeuristics*, 2003.
- [19] E. Freuder y M. Wallace Constraint Satisfaction. Cap. 14 en F. Glover y G. Kochenberger (eds.) *Handbook on MetaHeuristics*, 2003.
- [20] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 5:533–549, 1986.
- [21] F. Glover. Tabu search. part I. *ORSA Journal on Computing*, 1:190–206, 1989.
- [22] F. Glover. Tabu search. part II. *ORSA Journal on Computing*, 2:4–32, 1990.
- [23] F. Glover. A template for scatter search and path relinking. en J.-K. Hao y

- E. Lutton (eds.) *Artificial Evolution*, volume 1363 de *Lecture Notes in Computer Science*, 13–54, Springer-Verlag, 1998.
- [24] F. Glover. Multi-start and strategic oscillation methods - principles to exploit adaptive memory. en M.Laguna y J.L. González-Velarde (eds.) *Computing Tools for Modeling, Optimization and Simulation*, 1–24, Kluwer Academic Publishers, 2000.
- [25] F. Glover y G. Kochenberger (eds.) *Handbook of Metaheuristics*, Kluwer Academic Publishers, 2003.
- [26] F. Glover y M. Laguna. *Tabu Search*, Kluwer, 1997.
- [27] F. Glover, M. Laguna, E.D. Taillard y D. De Werra. *Tabu Search*, volume 43 of *Annals of Operational Research*. Baltzer, 1993.
- [28] F. Glover, E. Taillard y D. de Werra. A user’s guide to tabu search. *Annals of Operations Research*, 41:3–28, 1993.
- [29] F. Glover y D. De Werra. *Tabu Search*, volume 41 de *Annals of Operational Research*, Baltzer, 1993.
- [30] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning* Addison Wesley, 1989.
- [31] J.L. González-Velarde y M. Resende. GRASP: Procedimientos de búsqueda miopes aleatorizados y adaptativos. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [32] L.K. Grover. Local search and the local structure of NP-complete problems. *Operational Research Letter*, 12:235–243, 1992.
- [33] M. Guignard Lagrangian Relaxation en P.M. Pardalos y M.G.C. Resende (eds.) *Handbook of Applied Optimization* Oxford University Press, 465–474, 2002
- [34] P. Hansen y N. Mladenović. Variable Neighborhood Search: Principles and Applications. *European Journal of Operational Research*, 130:449–467, 2001.
- [35] P. Hansen y N. Mladenović. Developments in Variable Neighbourhood Search. en C. Ribeiro y P. Hansen (eds.) *Essays and Surveys in Metaheuristics*, 415–439. Kluwer, 2002.
- [36] P. Hansen y N. Mladenović. Variable Neighborhood Search. en P.M. Pardalos y M.G.C. Resende (eds.), *Handbook of Applied Optimization*, 221–234, Oxford University Press, 2002.
- [37] P. Hansen y N. Mladenović. Variable Neighborhood Search. Cap. 6 en F. Glover y G.A. Kochenberger (eds.), *Handbook of Metaheuristics*, Kluwer Academic, 2003.
- [38] P. Hansen, N. Mladenović y J.A. Moreno, . Búsqueda de entorno variable. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [39] J. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.
- [40] J.J. Hopfield y D.W. Tank. Neural computation of decisions in optimization problems. *Bio. Cybern.*, 52:141–152, 1985.
- [41] Y.S.J. Kennedy y R. Eberhart. *Swarm Intelligence*.
- [42] S. Kirkpatrick, C.D. Gelatt, y M.P. Vecchi. Optimization by Simulated Annealing. *Science*, 220:671–680, 1983.
- [43] M. Laguna, F. Glover y R. Martí. Fundamentals of scatter search and path relinking. *Control and Cybernetics*, 39:653–684, 2000.
- [44] M. Laguna, F. Glover y R. Martí. Scatter search and path relinking: Advances and applications. Cap. 1 en F. Glover y G. Kochenberger (eds.) *Handbook on MetaHeuristics*, 2003.
- [45] M. Laguna y J.L. González-Velarde, (eds.) *Computing Tools for Modeling, Optimization and Simulation*. Kluwer Academic Publishers, 2000.
- [46] M. Laguna y R. Martí. *Scatter Search Methodology and Implementations in C*, Kluwer Academic Publishers, 2002.

- [47] P. Larrañaga, J.A. Lozano y H. Mühlenbein. Algoritmos de estimación de distribuciones en problemas de optimización combinatoria. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [48] P.J.M. van Laarhoven y E.H.L. Aarts. *Simulated Annealing: Theory and Applications*. Kluwer Academic Press, 1987.
- [49] C.K. Looi. Neural network method in combinatorial optimization. *Computers and Operations Research*, 19:191–208, 1992.
- [50] H.R. Lourenço, O. Martin, y T. Stützle. Iterated local search. Cap. 11 en F. Glover y G.G. Kochenberger (eds.) *Handbook of Metaheuristics*, Kluwer Academic Publishers. 2003.
- [51] J.A. Lozano y P. Larrañaga. *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*. Kluwer Academic.
- [52] F. Manyá y C. Gomes. Técnicas de resolución de problemas de satisfacción de restricciones. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [53] R. Martí. Multistart methods. en Fred Glover y Gary A. Kochenberger (eds.) *Handbook of Metaheuristics*, 355–368, Kluwer Academic, 2003.
- [54] R. Martí y M. Laguna. Scatter Search: Diseño básico y estrategias avanzadas. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [55] B. Melián y F. Glover. Búsqueda tabú. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [56] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer Verlag, 1992.
- [57] Z. Michalewicz y D.B. Fogel. *How to Solve It: Modern Heuristics*. Springer Verlag, 2000.
- [58] P. Mills, E.P.K. Tsang, y J. Ford. Applying an extended guided local search on the quadratic assignment problem. *Annals of Operations Research*, 118:121–135, 2003.
- [59] M. Mitchel. *An introduction to Genetic Algorithms*. MIT Press, 1996.
- [60] P. Moscato. Memetic algorithms: A short introduction. en D. Corne, M. Dorigo y F. Glover (eds.) *New Ideas in Optimization*, 219–234, McGraw-Hill, 1999.
- [61] M.J. Oates, D.W. Corne y G.D. Smith (eds.) *Telecommunications Optimization: Heuristic and Adaptive Techniques*. Wiley, 2000.
- [62] M. Pirlot. General local search heuristics in combinatorial optimization: A tutorial. *Belgian J. of Operations Research, Statistics and Computer Science*, 32:7–67, 1994.
- [63] M. Pirlot. General local search methods. *European Journal of Operational Research*, 92(3):493–511, 1996.
- [64] J.M. Moreno-Vega y R. Martí. Métodos multi-arranque. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, este mismo volumen, 2003.
- [65] M.G.C. Resende y C.C. Ribeiro. Greedy randomized adaptive search procedures. en F. Glover y G.G. Kochenberger (eds.) *Handbook of Metaheuristics*, Kluwer Academic Publishers. 2003.
- [66] C.R. Reeves. Genetic Algorithms. Cap. 3 en F. Glover y G. Kochenberger (eds.) *Handbook on MetaHeuristics*, 2003.
- [67] C.C. Ribeiro y P. Hansen (eds.) *Essays and Surveys in Metaheuristics*. Kluwer, 2001.
- [68] K. E. Rosing y M. John Hodgson. Heuristic concentration for the p-median: an example demonstrating how and why it works. *Computers and Operations Research*, 29(10):1317–1330, 2002.
- [69] S. Salhi. A perturbation heuristic for a class of location problems. *Journal of the Operational Research Society*, 48:1233 – 1240.
- [70] E.A. Silver, R. Victor, V. Vidal, y D. de Werra. A tutorial on heuristic methods. *European Journal of Operational Research*, 5:153–162, 1980.

- [71] J.L. Verdegay, A. Blanco y D. Pelta. A fuzzy valuation-based local search framework for combinatorial optimization. *Journal of Fuzzy Optimization and Decision Making*, 1(2):177–193, 2002.
- [72] R.V.V. Vidal. *Applied Simulated Annealing*, volume 396 of *Lecture Notes in Econ. and Math. Systems*. Springer Verlag, 1993.
- [73] S. Voss y D.L. Woodruff (eds.) *Optimization Software Class Libraries*, Kluwer Academic Publishers, 2002.
- [74] C. Voudouris y E.P.K. Tsang. Guided local search. *European Journal of Operational Research*, 113(2):469–499, 1999.
- [75] C. Voudouris y E.P.K. Tsang. Guided local search. Cap. 7 en F. Glover y G. Kochenberger (eds.) *Handbook on MetaHeuristics*, 2003.
- [76] Z.-B. Xu, H.-D. Jin, K.-S. Leung, Y. Leung, y C.-K. Wong. An automata network for performing combinatorial optimization. *Neurocomputing*, 47:59–83, 2002.
- [77] M. Yagiura y T. Ibaraki. On metaheuristic algorithms for combinatorial optimization problems. *Systems and Computers in Japan*, 32(3):33–55, 2001.
- [78] M. Yagiura y T. Ibaraki. Local search. en P.M. Pardalos y M.G.C. Resende (eds.) *Handbook of Applied Optimization*, 104–123. Oxford University Press, 2002.
- [79] S.H. Zanakis, J.R. Evans, y A.A. Vazacopoulos. Heuristic methods and applications: a categorized survey. *European Journal of Operational Research*, 43:88–110, 1989.