

Búsqueda Tabú : Evitando lo Prohibido

Carlos A. Coello Coello

La búsqueda tabú es una técnica heurística que puede utilizarse en combinación con algún otro método de búsqueda para resolver problemas de optimización combinatoria con un alto grado de dificultad. Este artículo proporciona una visión general de la técnica, un ejemplo de su uso y un breve recuento de algunas de sus aplicaciones registradas en la literatura. Finalmente, se habla brevemente sobre algunas de las rutas más prometedoras de investigación dentro de esta área.

Introducción

La abundancia de problemas de optimización de alto grado de dificultad en el mundo real ha sido una de las causas principales por la que en los últimos años se ha desarrollado un número considerable de técnicas heurísticas que permiten obtener al menos un resultado sub-óptimo en un período de tiempo relativamente corto en problemas que resultaría impráctico resolver mediante "fuerza bruta" (i.e., enumerando todas las posibles soluciones para escoger de entre ellas a la mejor).

Las técnicas heurísticas más conocidas hoy en día no hacen más que adaptar ideas conocidas desde hace mucho tiempo en otras disciplinas. Por ejemplo, los algoritmos genéticos emulan los mecanismos de la evolución; los métodos de flujo de redes se fundamentan en ideas de la electricidad y la hidráulica, y el "enfriamiento simulado" (*simulated annealing*) se basa en un proceso físico de la industria metalúrgica. Similarmente, la técnica conocida como *búsqueda tabú* se basa en ciertos conceptos tomados de la Inteligencia Artificial y se utiliza como una meta-heurística (o una heurística de "alto nivel") para resolver problemas de optimización combinatoria. Esto significa que la técnica tiene que combinarse con algún otro mecanismo de búsqueda, y lo que hace, básicamente, es evitar que dicho mecanismo quede atrapado en un óptimo local.

Dado que los orígenes de esta técnica se dieron en el área de Investigación de Operaciones y no explícitamente en la de Inteligencia Artificial, no es una estrategia muy conocida para los investigadores de Ciencias de la Computación, pese a que existe una amplia literatura al respecto, así como un buen número de aplicaciones exitosas de alto grado de complejidad (i.e., problemas con un gran número de variables) que parecen mostrar la solidez del método. Además, el concepto de la búsqueda tabú sigue evolucionando y todavía hay mucho lugar para mejoras y sugerencias que permitan incrementar su flexibilidad y eficiencia al aplicarse a espacios de búsqueda de gran tamaño.

Conceptos Básicos

Los orígenes de la búsqueda tabú se ubican a fines de los 60s y principios de los 70s, y se atribuye a Fred Glover [1], quien desarrolló esta heurística para tratar de resolver problemas de cubierta no lineal (*nonlinear covering*), aunque varios de sus principios también fueron delineados independientemente por P. Hansen [2]. La búsqueda tabú surgió como un dispositivo que permitiría implementar una estrategia para resolver problemas de programación entera con restricciones sustitutas (*surrogate constraints*) [3], y aunque su planteamiento original ha sufrido varios cambios a través de los años, la idea básica propuesta por Glover ha permanecido intacta desde sus orígenes.

La búsqueda tabú puede verse como una meta-heurística que se superpone a una técnica de búsqueda y que se encarga de evitar que dicha técnica caiga en óptimos locales prohibiendo (o, en un sentido más general, penalizando) ciertos movimientos. El propósito de clasificar ciertos movimientos como prohibidos (o "tabú") es para evitar que se caiga en ciclos durante la

búsqueda. Debido a esto, Glover sugiere como nombre alternativo para su método el de búsqueda con "inhibición débil", ya que los movimientos que se consideran prohibidos constituyen generalmente una pequeña fracción del total de movimientos disponibles, y un movimiento pierde su status de prohibido después de un período de tiempo relativamente corto, volviéndose después nuevamente accesible. A este respecto, este método puede contrastarse con la técnica de ramificación y límites (*branch and bound*) que también prohíbe ciertos movimientos para evitar ciclos, pero lo hace de una manera más rígida, por lo que se le considera como una forma de búsqueda con "inhibición fuerte".

Desde la perspectiva de la Inteligencia Artificial, la búsqueda tabú trata de emular (de manera somera) el comportamiento de una persona. Es bien sabido que los humanos poseemos un avanzado mecanismo de intuición que nos permite operar aún con información mínima o nula, pero por lo general solemos introducir un elemento aleatorio (probabilístico) en dichas decisiones, lo cual promueve un cierto nivel de "inconsistencia" en nuestro comportamiento. La tendencia resultante en estos casos suele desviarnos de una cierta trayectoria pre-establecida, lo cual algunas veces puede ser una fuente de errores, pero en otras ocasiones puede llevarnos a una solución mejor. La búsqueda tabú intenta emular este mecanismo fundamental de la ingenuidad humana, pero sin utilizar elementos aleatorios, sino más bien asumiendo que no hay razón para escoger un movimiento que nos lleve a una peor solución, a menos que estemos tratando de evitar recorrer una ruta que ya se examinó previamente. Con esta sola excepción, la técnica buscará, a cada paso, el mejor movimiento posible de acuerdo a la métrica utilizada por el problema en cuestión. Esto hace que la técnica se dirija inicialmente de forma directa hacia un óptimo local, pero eso no importa porque la búsqueda no se detendrá ahí, sino que se reinicializará manteniendo la capacidad inicial de identificación del mejor movimiento posible. Además, se mantiene información referente a los movimientos más recientes en una o más *listas tabú*, a fin de evitar que una cierta trayectoria previamente recorrida se repita, aunque esta prohibición es generalmente condicional y no absoluta, como veremos más adelante.

La búsqueda tabú se encuentra fundamentada en 3 cosas principales [4]:

1. El uso de estructuras flexibles de memoria basadas en atributos, diseñadas para permitir una mejor explotación de los criterios de evaluación y la información histórica de la búsqueda que lo que se conseguiría con estructuras rígidas de memoria (como las que se usan en la búsqueda de ramificación y límites y el algoritmo A*) o con sistemas carentes de memoria (como la técnica de "enfriamiento simulado" —*simulated annealing*— y otras técnicas aleatorias similares)
2. Un mecanismo asociado de control (para emplear las estructuras de memoria) basado en la interacción entre las condiciones que limitan y hacen más flexible el proceso de búsqueda. Este mecanismo se encuentra inmerso en la técnica en la forma de restricciones y *criterios de aspiración* (un criterio de aspiración es aquél que permite que un movimiento pierda su status de "tabú" debido a que proporciona una mejor solución que la actual).
3. La incorporación de memorias de diferente duración (de corto a largo plazo), para implementar estrategias que intensifiquen y diversifiquen la búsqueda. Las estrategias de intensificación refuerzan las propiedades de las combinaciones de movimientos que han demostrado (históricamente) ser buenas, mientras que las estrategias de diversificación dirigen la búsqueda hacia nuevas

regiones del espacio de soluciones factibles. Note que estos dos mecanismos son muy similares a la cruza y la mutación en los algoritmos genéticos, ya que la primera nos permite delimitar una cierta región del espacio de búsqueda, mientras que la segunda nos permite saltar a nuevas regiones del mismo, evi-tando que quedemos atrapados en un óptimo local.

La parte medular de la búsqueda tabú se localiza en el proceso de memoria de corto plazo, y muchas de las consideraciones estratégicas en que se fundamenta este proceso reaparecen, amplificadas pero sin mayores modificaciones, en los procesos de memoria de largo plazo.

Memoria de Corto Plazo

La memoria de corto plazo de la búsqueda tabú constituye una forma de exploración agresiva que intenta realizar el mejor movimiento posible sujeto a las restricciones impuestas por el problema. Su funcionamiento básico se ilustra más detalladamente en la Figura 1 [4].

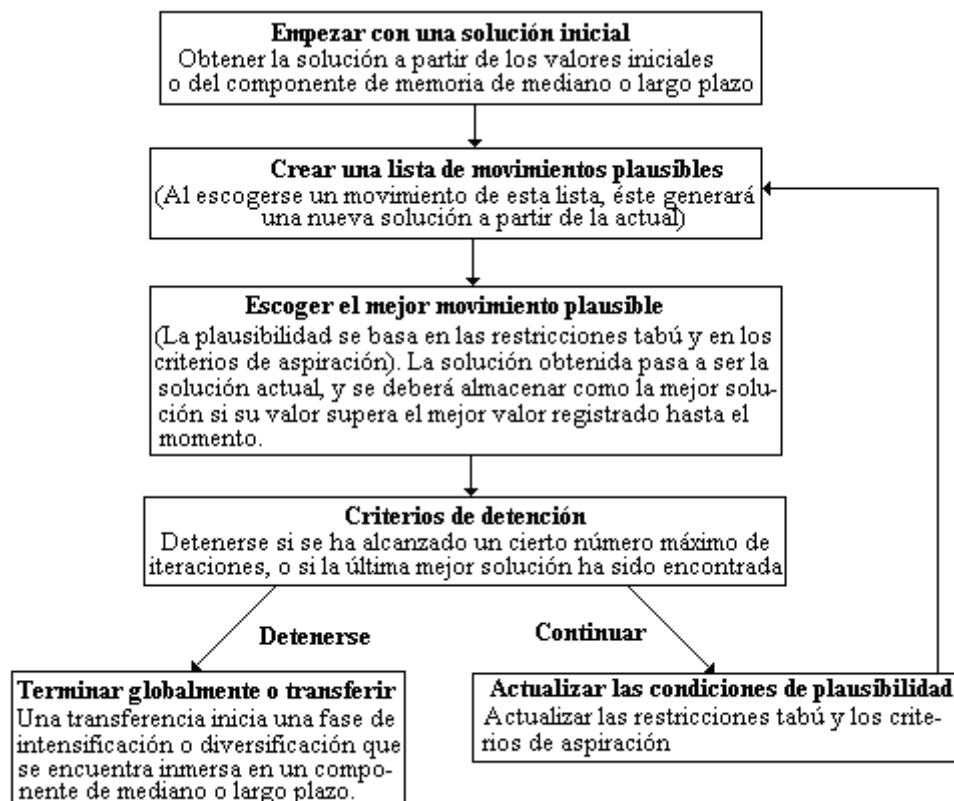


Figura 1 : El componente de memoria de corto plazo de la búsqueda tabú

Las restricciones impuestas por el problema se utilizan para evitar que se reviertan o repitan ciertos movimientos (se convierten en "prohibidos"). El objetivo primordial de los movimientos prohibidos es hacer que la técnica de búsqueda pueda ir más allá de los puntos de optimalidad local, a la vez que se mantienen movimientos de alta calidad a cada paso. Sin estas restricciones, el método podría moverse hacia un punto fuera de un óptimo local, pero inmediatamente después caería en el mismo punto al determinarse que el mejor movimiento posible desde esa posición es precisamente el óptimo local antes mencionado. En general, las restricciones tabú tienen como

objetivo prevenir ciclos e inducir a la búsqueda a que siga una nueva trayectoria si ocurren ciclos en un sentido más limitado (es decir, si se revisita una solución encontrada previamente). Estas restricciones no operan de manera aislada, sino que se contrabalancean mediante la aplicación de ciertos criterios de aspiración derivados del planteamiento del problema.

Determinando el Mejor Movimiento

Un paso crítico involucrado en la orientación agresiva de la memoria de corto plazo es la selección del mejor movimiento plausible desde un punto cualquiera. La Figura 2 [4] ilustra esquemáticamente este mecanismo.

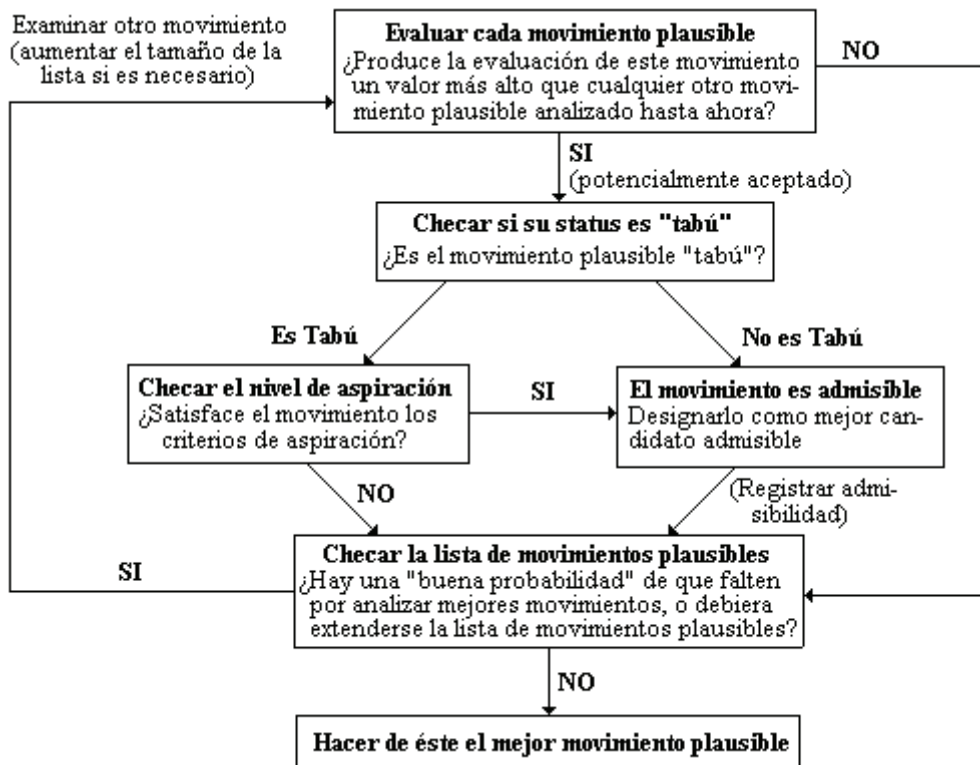


Figura 2 : Selección del mejor movimiento posible

Primero, se evalúan cada uno de los movimientos de la lista de candidatos en turno. En algunos casos, la evaluación de un cierto movimiento puede basarse inicialmente en el cambio producido en el valor de la función objetivo; es decir, se calcula la diferencia entre los valores de la función objetivo antes y después de aplicar el movimiento para ver cuál proporciona un valor más favorable. En otros casos, en donde las ramificaciones de un movimiento son más difíciles de determinar o donde no a todas las variables se les ha asignado un valor todavía, es más conveniente evaluar en base a soluciones relajadas o aproximadas, o utilizar simplemente una medida local de estimación de la cercanía de una cierta solución al óptimo. Sin embargo, conforme la búsqueda avanza, la forma de evaluación utilizada por la búsqueda tabú se vuelve más adaptativa, incorporando referencias a los parámetros de intensificación y diversificación. Debido a que el número de movimientos clasificados como tabú es generalmente pequeño con

respecto al número total de movimientos válidos, y asumiendo que el costo de evaluación de un movimiento no es demasiado grande, normalmente se prefiere checar primero si un cierto movimiento es mejor que sus predecesores admisibles antes de checar si es un movimiento prohibido. Luego se checa si el movimiento es o no "tabú", y de no serlo se acepta inmediatamente como admisible. De lo contrario, se usan los criterios de aspiración pre-determinados para darle a dicho movimiento la oportunidad de cambiar su status de "tabú" (en otras palabras, se le da una segunda oportunidad para calificar como admisible).

El proceso de análisis del siguiente movimiento puede incluirse dentro de la estrategia de la lista de movimientos plausibles. En algunos casos, si las restricciones tabú y los criterios de aspiración son suficientemente limitantes, ninguno de los movimientos posibles calificará como admisible. Para tales situaciones debe almacenarse un movimiento "menos inadmisibles", que se escogerá sólo si no hay ninguna alternativa admisible disponible.

Ejemplo de Uso

Para ilustrar el funcionamiento de esta técnica, utilizaremos un ejemplo clásico de estructuras de datos e Inteligencia Artificial: el problema de las N reinas. La idea básica es colocar N reinas en un tablero de ajedrez de $N \times N$ casillas, de tal manera que ninguna de ellas pueda capturar a otra. Este problema puede verse como uno de optimización en el que el objetivo es minimizar el número de colisiones (una colisión se presenta cuando una reina puede capturar a otra). Si asumimos que cada reina se colocará siempre en una fila determinada de acuerdo a un cierto orden pre-establecido, podemos representar cualquier estado del tablero usando una permutación de la forma (p_1, p_2, \dots, p_n) , donde el valor de p_i representa la columna donde se colocará a la reina que está en la fila i , y así sucesivamente. Por ejemplo, la permutación $(3, 4, 2, 1)$ corresponde al tablero mostrado en la Figura 3, en el cual hay 2 colisiones.

	1	2	3	4
1			R	
2				R
3		R		
4	R			

Figura 3 : Una posible instancia del problema de las 4 reinas

Una de las operaciones básicas que el algoritmo utilizará es el intercambio de 2 elementos cualquiera de una permutación en la que hayan colisiones. A fin de evitar la repetición de intercambios realizados recientemente, consideraremos *tabú* cualquier intercambio de este tipo. Debido a condiciones de simetría (intercambiar las reinas 1 y 4 es lo mismo que intercambiar las reinas 4 y 1), una estructura como la mostrada en la Figura 4 [10] puede usarse para almacenar el número de iteraciones que faltan para permitir que 2 reinas intercambien posiciones nuevamente.

	2	3	4	5	6	7
1						
2						
3						
4						
5						
6						

Figura 4 : Estructura usada para almacenar los atributos correspondientes a los intercambios de 2 reinas

La restricción de los intercambios de 2 reinas puede ser ignorada si un cierto movimiento (prohibido) conduce a una mejor solución (este es el *criterio de aspiración*).

Solución actual	2 3 4 5 6 7	Intercambio	Valor								
<table border="1"><tr><td>4</td><td>5</td><td>3</td><td>6</td><td>7</td><td>1</td><td>2</td></tr></table>	4	5	3	6	7	1	2	1 0 0 0 0 0 0	1	7	-2
4	5	3	6	7	1	2					
Número de colisiones = 4	2 0 0 0 0 0 0	2	4	-2							
	3 0 0 0 0 0 0	2	6	-2							
	4 0 0 0 0 0 0	5	6	-2							
	5 0 0 0 0 0 0	1	5	-1							
	6 0 0 0 0 0 0										
Estructura tabú		Las 5 mejores soluciones plausibles									

Figura 5 : Iteración cero de la búsqueda tabú en el problema de las 7 reinas

Para evaluar un movimiento, simplemente checamos cuantas reinas están en la misma fila, columna o diagonal, y contamos el número de colisiones. Por ejemplo, si usamos 7 reinas, en la iteración ce-ro la situación se verá como se ilustra en la Figura 5, en la que además se muestran los 5 mejores movimientos plausibles. Note cómo nuestra estructura tabú está inicializada a ceros, indicando que ningún movimiento es prohibido. Para determinar el mejor movimiento plausible, checamos todos los posibles intercambios de reinas (se requieren N^2 evaluaciones) y de ellos se escoje el que tenga el menor número de colisiones. El valor indicado en la tabla de movimientos plausibles de la Figura 5 indica el número de colisiones en que se disminuirá el estado de la solución si se efectúa un cierto intercambio (el orden es irrelevante, pues sólo nos interesa el mejor movimiento). Cuando (como en este caso) existen varios movimientos equivalentes, podemos seleccionar arbitrariamente cualquie-ra de ellos. En nuestro caso, se seleccionará al mejor movimiento que se haya generado primero.

Asumiendo que se intercambian las posiciones 1 y 7, la siguiente iteración nos lleva al estado ilustrado en la Figura 6, en la cual hay 2 colisiones y el intercambio de las posiciones 1 y 7 es tabú (el número 3 indicado en la estructura tabú es arbitrario y puede escogerse otro si así se desea).

Solución actual	2 3 4 5 6 7		Intercambio	Valor								
<table border="1"><tr><td>2</td><td>5</td><td>3</td><td>6</td><td>7</td><td>1</td><td>4</td></tr></table>	2	5	3	6	7	1	4	1 0 0 0 0 0 3	1	2	4	-1
2	5	3	6	7	1	4						
Número de colisiones = 2	2 0 0 0 0 0 0	2	1	6	0							
	3 0 0 0 0 0 0	2	2	5	0							
	4 0 0 0 0 0 0	1	1	2	1							
	5 0 0 0 0 0 0	1	1	3	1							
	6 0 0 0 0 0 0											
Estructura tabú		Las 5 mejores soluciones plausibles										

Figura 6 : Iteración uno de la búsqueda tabú en el problema de las 7 reinas

Tras intercambiar las posiciones 2 y 4 en la siguiente iteración, llegamos al estado ilustrado en la Figura 7, en el que hay sólo una colisión y 2 movimientos prohibidos, pero el mejor movimiento

plausi-ble no mejora en nada a la solución actual. En algunas implementaciones de la búsqueda tabú en que una gran proporción de los movimientos en una cierta vecindad tienen un valor de cero, este ti-po de intercambios son clasificados automáticamente como prohibidos [10], pero en nuestro caso serán permisibles. Note también como el 3 de la casilla (1,7) de nuestra estructura tabú se decre-mentó a 2, debido a que ese es ahora el número de iteraciones en los que este movimiento seguirá siendo considerado aún como prohibido. Los intercambios con valor de positivo indican situaciones en las que el número de colisiones se incrementará en vez de decrementarse.

Al realizarse el intercambio de las posiciones 1 y 3, llegamos al estado ilustrado en la Figura 8, en el cual todavía hay una colisión, hay 3 intercambios tabú y no hay ningún movimiento plausible que nos permita mejorar la solución actual. Si observa la estructura tabú en este caso, notará que los in-tercambios (1,3) y (1,7) son prohibidos, por lo que tendremos que recurrir al intercambio (5,7), que incrementará el número actual de colisiones. Dado que ninguno de los 2 intercambios prohibidos nos llevará a una mejor solución que la actual, el criterio de aspiración no aplica en este caso, y nos veremos obligados a realizar el intercambio de las posiciones 5 y 7.

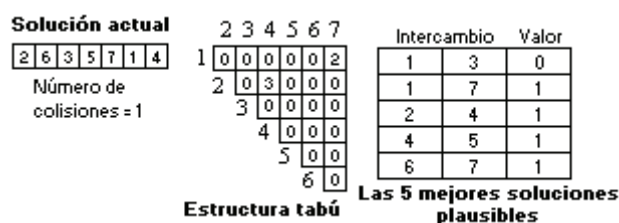


Figura 7 : Iteración dos de la búsqueda tabú en el problema de las 7 reinas

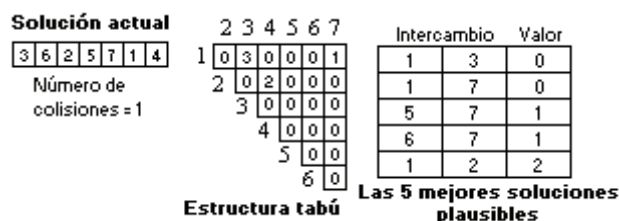


Figura 8 : Iteración tres de la búsqueda tabú en el problema de las 7 reinas

Al intercambiar las posiciones 5 y 7 en la iteración 4, llegamos al estado ilustrado en la Figura 9, en el cual tenemos 2 colisiones, 3 movimientos tabú (uno de los anteriores dejó de serlo en esta itera-ción) y un par de movimientos plausibles que pueden regresarnos a un estado con sólo una colisión.

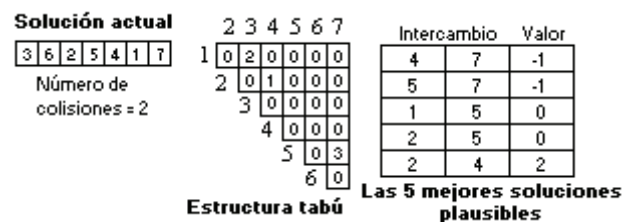


Figura 9 : Iteración cuatro de la búsqueda tabú en el problema de las 7 reinas

Como el intercambio de las posiciones 4 y 7 no es tabú, procedemos a efectuarlo en la iteración 5, llegando al estado ilustrado en la Figura 10, en el cual tenemos 1 colisión, 3 movimientos tabú y dos movimientos plausibles que nos pueden llevar a una solución sin colisiones. El problema ahora es que el primero de estos movimientos resulta ser prohibido, porque la estructura tabú tiene un valor de 1 en la casilla (1,3). Sin embargo, al aplicar el criterio de aspiración se determina que este inter-cambio producirá una solución mejor que la actual, y por tanto se ignora la prohibición. El estado final que se obtiene se ilustra en la Figura 11, y representa una solución aceptable al problema (i.e., es una permutación con cero colisiones).

Note que en este ejemplo sólo hubo necesidad de utilizar una memoria de corto plazo (la estructura tabú), pero en general puede resultar necesario hacer acopio también de una memoria de largo plazo. Por ejemplo, podríamos expandir nuestra estructura tabú almacenando un contador de frecuencias para cada intercambio, que indicaría qué movimientos han sido realizados más veces a lo largo de la búsqueda, penalizándolos en consecuencia a fin de hacerlos también tabú. Muchos otros mecanismos han sido propuestos para mejorar la propuesta original de Glover [3], y aunque muchos de ellos se desarrollaron en el contexto de una aplicación en particular, por lo general estas mejoras suelen poder aplicarse a cualquier problema que se desee.

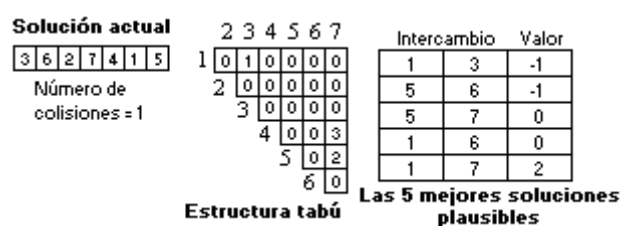


Figura 10 : Iteración cinco de la búsqueda tabú en el problema de las 7 reinas

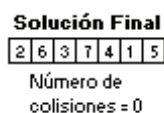


Figura 11 : Permutación final del problema de las 7 reinas, obtenida después de 6 iteraciones

El autor ha proporcionado a los editores de esta revista la implementación completa en Turbo C 2.0 del algoritmo básico de la búsqueda tabú aplicado al problema de las N reinas¹. Este código está basado en la implementación proporcionada por Manuel Laguna [10].

Áreas de Aplicación

La búsqueda tabú ha encontrado innumerables aplicaciones en muy variadas áreas dentro de la investigación de operaciones durante los últimos años. En general, las aplicaciones más frecuentes encontradas en la literatura caen en alguna de las siguientes categorías [8]:

¹ El código puede obtenerse también mediante **ftp anonymous** del nodo: **ftp.eecs.tulane.edu** en el directorio **/pub/coello**.

- Planificación (por ejemplo: planificación de horarios, de máquinas, de fuerza de trabajo, etc.)
- Diseño (por ejemplo: diseño de topologías de redes, planeación de espacios arquitectónicos, de redes tolerantes a errores, etc.)
- Lógica e Inteligencia Artificial (por ejemplo: reconocimiento de patrones y caracteres, integridad de datos, diseño de redes neuronales, lógica probabilística, etc.)
- Tecnología (por ejemplo: inversión sísmica, distribución de potencia eléctrica, construcción de estaciones espaciales, exploración de yacimientos petrolíferos, diseño estructural, etc.)
- Telecomunicaciones (por ejemplo: asignación de rutas, planeación de descuentos a clientes, redes óptimas síncronas, arquitecturas inmunes a fallas, etc.)
- Producción, inventario e inversión (por ejemplo: manufactura flexible, selección de partes, planeación de inventarios, etc.)
- Localización y asignación (por ejemplo: multicomodidades, asignación cuadrática, etc.)
- Ruteado (por ejemplo: ruteado de vehículos, de flotas mixtas, el problema del viajero, etc.)
- Optimización de gráficas (por ejemplo: división de gráficas, coloreado de gráficas, etc.)
- Problemas de optimización combinatoria en general (por ejemplo, programación cero-uno, programación no lineal y no convexa, optimización entera mixta, etc.)

En general, esta técnica ha tenido mucho éxito en problemas de optimización combinatoria que pueden expresarse en la forma de una gráfica o una red de flujos. Glover [4] afirma haber resuelto con este método problemas que involucran entre uno y cuatro millones de variables en menos de media hora, usando sólo una computadora personal, y obteniendo resultados que se encuentran dentro del 98% de límite superior de optimalidad. Tan sólidos resultados la hacen ver como una técnica de búsqueda muy prometedora dentro de la investigación de operaciones, y que tal vez debería tener más uso en la Inteligencia Artificial en posible combinación con otras heurísticas.

Para mayor información sobre este método y fuentes bibliográficas adicionales sobre sus aplicaciones, se recomienda al lector consultar las referencias [4], [5], [6], [7], [8] y [9].

Áreas de investigación

Existen muchas variantes de la búsqueda tabú que pueden resultar más adecuadas en ciertos problemas, y dado que la definición misma del algoritmo es un tanto imprecisa, no resulta extraño que cada usuario la ajuste a sus propias necesidades. Por ejemplo, cuando el espacio a explorarse es muy grande, puede ser muy costoso aplicar una búsqueda exhaustiva para determinar el siguiente movimiento, y una técnica de descomposición puede resultar útil [12]. También se ha propuesto almacenar los movimientos más prometedores, o "élite" [13] (i.e., los que producen mejores valores) en la lista de candidatos a fin de utilizarlos más adelante en vez de volver a buscar de manera exhaustiva el mejor movimiento posible. Otros han propuesto el uso de una lista tabú dinámica mediante una función de direcciones calculadas (*hashing*) [14]. Asimismo, se ha sugerido el uso de técnicas de procesamiento en paralelo [15] para mejorar la eficiencia del algoritmo. Puesto que la búsqueda tabú es una metaheurística, puede combinarse virtualmente con cualquier otra heurística, y no debe sorprendernos que se hayan hecho intentos para mezclarla con los algoritmos genéticos [7], el enfriamiento simulado [16] y las redes neuronales [17], cuya popularidad es mucho mayor hoy en día.

Conclusiones

A lo largo de estas páginas se ha intentado proporcionar un panorama muy general de la llamada búsqueda tabú, aunque debido a sus tantas variantes, el contenido de este artículo dista mucho de ser un tratamiento exhaustivo del tema, pero las referencias bibliográficas proporcionadas deberán ser un buen punto de partida para el lector interesado en aprender más sobre esta área.

La búsqueda tabú se ha convertido en un tópico común en las revistas especializadas de investigación de operaciones en los últimos años, y el *Comité sobre la Próxima Década de Investigación de Operaciones* la consideró como "extremadamente prometedora" [11]. Además, su creciente interacción con técnicas modernas de Inteligencia Artificial tales como el enfriamiento simulado, los algoritmos genéticos y las redes neuronales parece abrir horizontes insospechados dentro de la optimización, la búsqueda y el aprendizaje máquina. Ignorar el éxito abrumador que ha tenido esta metaheurística en la práctica resulta no sólo difícil sino además un tanto riesgoso para los investigadores y docentes del área de Inteligencia Artificial, pues su flexibilidad y poder bien la podrían colocar dentro de las herramientas de búsqueda indispensables del experto en computación del siglo XXI.

Referencias

- [1] Glover, Fred, "Future Paths for Integer Programming and Links to Artificial Intelligence", *Computers & Operations Research*, Volume 13, No. 5, pp. 533-549, 1986.
- [2] Hansen, P., "The steepest ascent mildest descent heuristic for combinatorial programming", *Congress on Numerical Methods in Combinatorial Optimization*, Capri, Italy, 1986.
- [3] Glover, Fred, "Heuristics for integer programming using surrogate constraints", *Decision Sciences*, No. 8, pp. 156-166, 1977.
- [4] Glover, Fred, "Tabu Search : A Tutorial", *Interfaces*, Vol 20, No. 4, pp. 74-94, July-August 1990.
- [5] Glover, Fred, "Tabu Search—Part I", *ORSA Journal on Computing*, Vol. 1, No. 3, pp. 190-206, Summer 1989.
- [6] Glover, Fred, "Tabu Search—Part II", *ORSA Journal on Computing*, Vol. 2, No. 1, pp. 4-32, Winter 1990.
- [7] Glover, Fred & Laguna, Manuel, "Tabu Search", En Reeves, Colin R. (Editor), "Modern Heuristic Techniques for Combinatorial Problems", John Wiley & Sons, Inc. 1993, pp. 70-150.
- [8] Glover, Fred, "Tabu Search Fundamentals and Uses", Technical Report, Graduate School of Business, University of Colorado, Boulder, Colorado, May 1994.
- [9] Glover, Fred, "A user's guide to tabu search", *Annals of Operations Research*, Vol. 41, pp. 3-28, 1993.
- [10] Laguna, Manuel, "A guide to implementing Tabu Search", Technical Report, Graduate School of Business, University of Colorado, Boulder, Colorado, August, 1994.
- [11] Committee on the Next Decade of Operations Research (Condor), "Operations Research : The Next Decade", *Operations Research*, Vol. 36, 1988.

[12] Laguna, Manuel; Barnes, J. W. & Glover, Fred, "Tabu search methods for a single machine scheduling problem", *Journal of International Manufacturing*, Vol. 2, pp. 63-74, 1991.

[13] Glover, F.; Glover, R. & Klingman, D. "The threshold assignment algorithm", *Mathematical Programming Study*, Vol. 26, pp. 12-37, 1986.

[14] Woodruff, D. L. & Zemel, E. "Hashing vectors for tabu search", *Annals of Operations Research*, Vol. 41, pp. 123-138, 1993.

[15] Taillard, E. "Some efficient heuristic methods for the flowshop sequencing problem", *European Journal of Operations Research*, Vol. 47, pp. 65-79, 1990.

[16] Osman, I. H., "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problems", *Annals of Operations Research*, Vol. 41, pp. 421-451, 1993.

[17] de Werra, D. & Hertz, A., "Tabu Search Techniques: A Tutorial and Applications to Neural Networks", *OR Spectrum*, Vol. 11, pp. 131-141, 1989.