

# BÚSQUEDA TABÚ (Tabu Search TS)

Oscar Lozano

BUSQUEDA TABU

## Búsqueda Tabú

- Metaheurístico que usa *búsqueda agresiva* del óptimo del problema.
- **Agresiva** = evitar que la búsqueda quede "atrapada" en un óptimo local que no sea global.
- Oficialmente, el nombre y la metodología fueron introducidos por Fred Glover (1989).

BUSQUEDA TABU

## Conceptos Claves de TS

**Memoria + Aprendizaje = Búsqueda inteligente.**

El principio de TS podría resumirse como:

*"Es mejor una mala decisión basada en información que una buena decisión al azar, ya que, en un sistema que emplea memoria, una mala elección basada en una estrategia proporcionará claves útiles para continuar la búsqueda. Una buena elección fruto del azar no proporcionará ninguna información para posteriores acciones."*

BUSQUEDA TABU

## Conceptos Claves de TS

Posee:

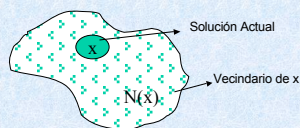
- **Memoria Adaptativa** (memoria selectiva, incluyendo olvido estratégico)
- **Exploración sensible** (se concentra en buscar buenas características de las soluciones)

GA  Proceso desmemoriado

Branch and Bound  Proceso con memoria rígida

BUSQUEDA TABU

## Procedimiento TS



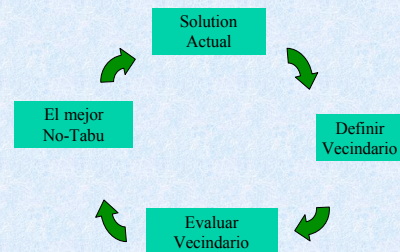
Proceso Iterativo que comienza en una solución  $x$  elegida al azar.

En la iteración inicial o en cualquier otra iteración, a partir de una solución actual  $x$ , el proceso pasa a la mejor solución de su vecindario  $N(x)$

**El movimiento siempre se acepta a pesar de que la nueva solución sea peor que la anterior.**

BUSQUEDA TABU

## Procedimiento TS



BUSQUEDA TABU

## Ejemplo:

Supongamos un ejemplo SAT, con 8 variables. Dada una formula lógica F, encontrar la asignación de verdad para todas las 8 variables, de tal forma que F sea VERDADERA. Partimos de una asignación inicial:

$$X = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = (0, 1, 1, 1, 0, 0, 0, 1) = 27$$

VECINDARIO

$$X_1 (1, 1, 1, 1, 0, 0, 0, 1) = 22 \quad X_2 (0, 0, 1, 1, 0, 0, 0, 1) = 25 \quad X_3 (0, 1, 0, 1, 0, 0, 0, 1) = 31$$

$$X_4 (0, 1, 1, 0, 0, 0, 0, 1) = 28 \quad X_5 (0, 1, 1, 1, 1, 0, 0, 1) = 20 \quad X_6 (0, 1, 1, 1, 0, 1, 0, 1) = 26$$

$$X_7 (0, 1, 1, 0, 0, 0, 1, 1) = 30 \quad X_8 (0, 1, 1, 1, 1, 0, 0, 0) = 26$$

BUSQUEDA TABU

## Criterio de terminación del proceso

Para terminar el proceso puede tenerse en cuenta dos condiciones:

- cuando ya se ha realizado un número de iteraciones  $k$  sin mejorar el valor de  $Z$
- o cuando se considere que el valor obtenido es cercano al óptimo esperado.

BUSQUEDA TABU

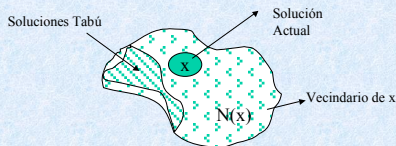
¡El proceso tiene el peligro de volver recurrentemente a la solución anterior!

Es mejor usar  $N^*(x)$  en vez de  $N(x)$

$(N^*(x) \cap \text{entorno reducido})$

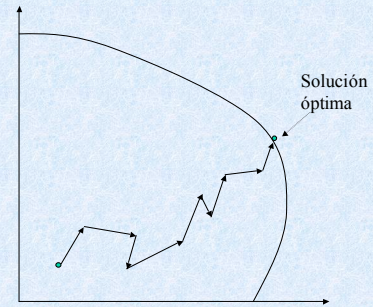
$N^*(x) = N(x) - \text{soluciones que están en la lista tabú T}$

**Lista Tabú:** lista de soluciones previamente visitadas en un pasado cercano (*memoria a corto plazo*)



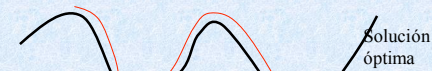
BUSQUEDA TABU

## Proceso TS



BUSQUEDA TABU

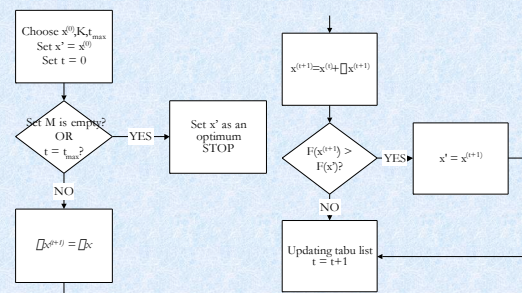
## Proceso TS



**Lista Tabú**  
Usa técnicas de memoria para ayudar a identificar un ciclo

BUSQUEDA TABU

## Proceso TS



BUSQUEDA TABU

## Formas de elaborar la lista Tabú

- Memoria Explícita:** memoriza en la lista tabú toda la solución ya visitada, por ejemplo:

Solución tabú = (1,0,1,1,1,0,1,1)

- Memoria Atributiva**

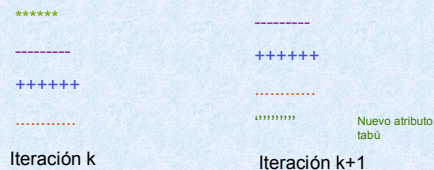
En vez de memorizar toda la solución, se memorizan atributos, es decir, características importantes de las soluciones ya visitadas, por ejemplo:

Atributo tabú: = (0, \*, \*, \*, \*, \*, \*, 1)

Tabú – Activos	□	atributos tabú
Solución Tabú	□	solución que tiene algún atributo Tabú

BUSQUEDA TABU

La lista Tabú debe ser dinámica porque tras un cierto número de iteraciones la búsqueda está en una región distinta y las soluciones antiguas puede liberarse del status tabú



BUSQUEDA TABU

## Ejemplo

$X = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = (0, 1, 1, 1, 0, 0, 0, 1) = 27$

VECINDARIO

$X_1 (1, 1, 1, 1, 0, 0, 0, 1) = 22$   $X_2 (0, 0, 1, 1, 0, 0, 0, 1) = 25$   $X_3 (0, 1, 0, 1, 0, 0, 0, 1) = 31$

$X_4 (0, 1, 1, 0, 0, 0, 0, 1) = 28$   $X_5 (0, 1, 1, 1, 1, 0, 0, 1) = 20$   $X_6 (0, 1, 1, 1, 0, 1, 0, 1) = 26$

$X_7 (0, 1, 1, 0, 0, 0, 1, 1) = 30$   $X_8 (0, 1, 1, 1, 1, 0, 0, 0) = 26$

MEMORIA

Es un arreglo o lista M, en donde se guarda el mas reciente "flip" o cambio de una posición particular. Se asume que dicha memoria puede ser borrada despues de un determinado "tiempo" o iteraciones, para nuestro caso suponemos que en 5 iteraciones se debe borrar. Se inicializa en cero la lista.

Primera Iteración

$M = [0, 0, 0, 0, 0, 0, 0, 0]$  nuevo estado de  $M = [0, 0, 5, 0, 0, 0, 0, 0]$

$X_3 (0, 1, 0, 1, 0, 0, 0, 1) = 31$  El mejor

BUSQUEDA TABU

## Ejemplo

$(0, 1, 0, 1, 0, 0, 0, 1) = 31$

VECINDARIO

TABU

$X_1 (1, 1, 0, 1, 0, 0, 0, 1) = 29$   $X_2 (0, 0, 0, 1, 0, 0, 0, 1) = 28$   $X_3 (0, 1, 1, 1, 0, 0, 0, 1) = 32$

$X_4 (0, 1, 1, 0, 0, 0, 0, 1) = 27$   $X_5 (0, 1, 0, 1, 1, 0, 0, 1) = 20$   $X_6 (0, 1, 0, 1, 0, 1, 0, 1) = 25$

$X_7 (0, 1, 0, 1, 0, 0, 1, 1) = 30$   $X_8 (0, 1, 1, 1, 1, 0, 0, 0) = 22$

MEMORIA

Segunda Iteración

$M = [0, 0, 5, 0, 0, 0, 0, 0]$  nuevo estado de  $M = [0, 0, 4, 0, 0, 0, 5, 0]$

$X_3 (0, 1, 0, 1, 0, 0, 1, 1)$  El mejor de los no Tabu

BUSQUEDA TABU

## Ejemplo

$(0, 1, 0, 1, 0, 0, 1, 1) = 30$

VECINDARIO

$X_1 (1, 1, 0, 1, 0, 0, 1, 1) = 33$   $X_2 (0, 0, 0, 1, 0, 0, 1, 1) = 32$   $X_3 (0, 1, 1, 1, 0, 0, 1, 1) = 34$

$X_4 (0, 1, 0, 0, 0, 0, 1, 1) = 30$   $X_5 (0, 1, 0, 1, 1, 0, 1, 1) = 29$   $X_6 (0, 1, 0, 1, 0, 1, 1, 1) = 28$

TABU

$X_7 (0, 1, 0, 1, 0, 0, 0, 1) = 31$   $X_8 (0, 1, 1, 1, 1, 0, 1, 0) = 30$

MEMORIA

Tercera Iteración

$M = [0, 0, 4, 0, 0, 0, 5, 0]$  nuevo estado de  $M = [5, 0, 3, 0, 0, 0, 4, 0]$

$X_1 (1, 1, 0, 1, 0, 0, 1, 1) = 33$  El mejor de los no tabu

BUSQUEDA TABU

## Ejemplo

Llegada la quinta iteración tenemos...

$M[3, 0, 1, 5, 0, 4, 2, 0]$  y la respectiva solución es  $X = (1, 1, 0, 0, 0, 1, 1, 1) = 33$

Los bits 2,5 y 8 pueden ser cambiados en cualquier momento. El bit 1 no se puede cambiar durante las próximas 3 iteraciones, el bit no esta disponible hasta dentro de una iteración y el bit 4 esta disponible hasta dentro de 5 iteraciones.

VECINDARIO

$X_1 (0, 1, 0, 0, 0, 1, 1, 1)$   $X_2 (1, 0, 0, 0, 0, 1, 1, 1)$   $X_3 (1, 1, 1, 0, 0, 1, 1, 1)$

$X_4 (1, 1, 0, 1, 0, 1, 1, 1)$   $X_5 (1, 1, 0, 0, 1, 1, 1, 1)$   $X_6 (1, 1, 0, 0, 0, 0, 1, 1)$

$X_7 (1, 1, 0, 0, 0, 1, 0, 1)$   $X_8 (1, 1, 0, 0, 0, 1, 1, 0)$

Para una sexta iteración nuestra memoria quedaria asi:

$M[2, 0, 0, 4, 5, 3, 1, 0]$

BUSQUEDA TABU

## Uso de Niveles de Aspiración

El uso de la lista Tabú con memoria atributiva reduce el riesgo de hacer procesos cíclicos pero, **puede prohibir moverse hacia algunas soluciones interesantes que no han sido evaluadas aún.**

**Puede usarse una Relajación de la prohibición.**

Si la nueva solución pertenece a la lista tabú, solo se rechaza si su valor Z no alcanza un nivel de aspiración A.

**A puede ser:**

- Umbral definido por el decisor
- o el mejor valor encontrado hasta el momento

BUSQUEDA TABU

## Sofisticaciones de TS

Usan mecanismos de penalizaciones e incentivos para mejorar la búsqueda

### 1. Diferentes tipos de Memoria

- Memoria a corto Plazo
- Memoria a largo Plazo

### 2. Clasificación de Atributos

- Recientes
- Frecuentes

### 3. Tipos de estrategias

- Intensificar
- Diversificar

BUSQUEDA TABU

## Tipos de Memorias

- Memoria a corto Plazo
- Memoria a largo Plazo

Ambos tipos de memoria llevan asociadas sus propias estrategias y atributos, y actúan en ámbitos diferentes.

BUSQUEDA TABU

### A. Memoria a Corto Plazo

Almacena atributos de soluciones recientemente visitadas, y su objetivo es explorar a fondo una región dada del espacio de soluciones.

### B. Memoria a Largo Plazo

Memoria basada en Frecuencias

= Almacena las frecuencias de las ocurrencias de atributos en las soluciones visitadas tratando de identificar o diferenciar regiones.

Usa las frecuencias para penalizar o premiar (*según convenga*) movimientos que usan atributos muy usados en el pasado

BUSQUEDA TABU

## Memoria a largo plazo

*Esta opera con horizontes a largo plazo. Se maneja un vector H, que contiene la información a largo plazo:*

$$H(i)=j$$

*Lo que indica que "Durante las últimas h iteraciones del algoritmo, el i-ésimo bit fue cambiado j veces"*

*El uso de una memoria larga, se restringe a casos especiales, como en los cuales todas las soluciones tabu se muevan hacia el pero caso. Para tomar una correcta decisión acerca de que próximo paso a seguir, es mejor mirar el contenido de la memoria de largo plazo.*

*Un ejemplo de ello puede ser, supongamos que tenemos 100 iteraciones del algoritmo con un horizonte de 50. El vector H puede contener la siguiente información:*

$$H=[5,7,11,3,9,8,1,6]$$

BUSQUEDA TABU

## Memoria a largo plazo

Supongamos que para este caso queremos diversificar la búsqueda. La solución X para el problema es 35. El vecindario de las soluciones no tabu contiene a los bits, 2,3 y 7 con los valores 30,33 y 31. Adicionalmente determinemos una función para calcular el nuevo valor vamos usar una nueva formula:

$$\text{Eval}(x') - \text{Penalización}(x')$$

Donde Eval(x') devuelve el valor original y Penalización es igual a:

$$\text{Factor} * H(i) \quad (\text{Factor}=0.7)$$

$$H=[5,7,11,3,9,8,1,6]$$

La nueva forma de obtener nuestra nueva solución sería:

$$30 - (0.7 * 7) = 25.1$$

$$33 - (0.7 * 11) = 25.3$$

$$31 - (0.7 * 1) = 30.3$$

BUSQUEDA TABU

## Tipos de Estrategias

*Están asociadas en la memoria a largo plazo*

### A. Intensificar

consiste en regresar a buenas regiones ya exploradas para estudiarlas más a fondo. Para ello se favorece la aparición de aquellos atributos asociados a buenas soluciones encontradas.

### B. Diversificar

consiste en visitar nuevas áreas no exploradas del espacio de soluciones. Para ello se modifican las reglas de elección para incorporar a las soluciones atributos que no han sido usados frecuentemente.

BUSQUEDA TABU

## Bibliografía de TS

- Glover F. (1989) Tabu-Search Part I, ORSA, Journal on Computing Vol 1, pp 190-206.
- Glover F. (1994) Tabu Search Fundamentals and uses. University of Colorado at Boulder,
- Glover F, M. Laguna. (1996) Tabu Search. Kluwer Academic Publishers, Glover, F. (1990a)
- Glover, F. and M. Laguna (1993) Tabu search. In: C.R. Reeves (ed.) Modern Heuristic Techniques for Combinatorial Problems. Blackwell, Oxford, 70 - 150.

BUSQUEDA TABU