

**"Con la ciencia por
la humanidad"**



INSTITUTO TECNOLÓGICO DE NUEVO LAREDO.

INGENIERÍA EN SISTEMAS COMPUTACIONALES.

Materia:

Inteligencia Artificial.

Catedrático:

Ing. Bruno López Takeyas.

Tema Equipo 3 :

Hill Climbing.

Alumnos:

| | |
|--|------------------------|
| <i>Alvarado Rodríguez Javier Alonso</i> | <i>01100160</i> |
| <i>García Rodríguez Jesús</i> | <i>00100175</i> |
| <i>Hernández Santos Wendy</i> | <i>01100232</i> |
| <i>Martínez Castillo Eli Othoniel</i> | <i>01100251</i> |
| <i>Vidaurre González Juan Ramón</i> | <i>01100322</i> |

Nuevo Laredo, Tamps a 24 de Octubre de 2005.

ALGORITMO HILL CLIMBING.

INTRODUCCION.

Similares a los algoritmos genéticos, aunque más sistemáticos y menos aleatorios. Un algoritmo de ascenso a colina comienza con una solución al problema a mano, normalmente elegida al azar. Luego, la cadena se muta, y si la mutación proporciona una solución con mayor aptitud que la solución anterior, se conserva la nueva solución; en caso contrario, se conserva la solución actual. Luego el algoritmo se repite hasta que no se pueda encontrar una mutación que provoque un incremento en la aptitud de la solución actual, y esta solución se devuelve como resultado (Koza et al. 2003[42], p. 59). (Para entender de dónde viene el nombre de esta técnica, imagine que el espacio de todas las soluciones posibles de un cierto problema se representa como un paisaje tridimensional. Un conjunto de coordenadas en ese paisaje representa una solución particular. Las soluciones mejores están a mayor altitud, formando colinas y picos; las que son peores están a menor altitud, formando valles. Un "ascenso a la colina" es, por tanto, un algoritmo que comienza en un punto dado del paisaje y se mueve inexorablemente colina arriba). El algoritmo de ascenso a colina es lo que se conoce como algoritmo voraz, lo que significa que siempre hace la mejor elección disponible en cada paso, con la esperanza de que de esta manera se puede obtener el mejor resultado global.

¿QUÉ ES EL ALGORITMO HILL CLIMBING?

Es una variante del algoritmo de búsqueda de Best First. Hill-climbing es una estrategia basada en optimización local. Se llama también una estrategia irrevocable porque no permite regresarnos a otra alternativa.

¿PARA QUÉ SIRVE EL ALGORITMO HILL CLIMBING?

Se emplean sobre todo para resolver problemas de optimización, como por ejemplo, encontrar la secuencia óptima para procesar un conjunto de tareas por un computador, hallar el camino mínimo de un grafo, etc. Su nombre se debe a que mientras usted se esta levantando, usted se está moviendo más cerca al grado optimo.

CARACTERÍSTICAS DEL ALGORITMO HILL CLIMBING:

- Informado: Utiliza información del estado por elegir un nodo u otro.
- No exhaustivo: No explora todo el espacio de estados. Como máximo, sólo encuentra una solución.
- Encuentra buenas soluciones, pero no la mejor puesto que no es exhaustivo.
- Es eficiente, porque evita la exploración de una parte del espacio de estados.
- Es útil si se tiene una función heurística muy buena o cuando los operadores de transición entre estados tienen cierta independencia (conmutativa), que implica que la operación de un operador no altera la futura aplicación de otro.
- **No guarda los estados anteriores.**

VENTAJAS Y DESVENTAJAS DEL ALGORITMO HILL CLIMBING:

Ventajas:

- Reduce el número de nodos a analizar.
- Dado su bajo costo computacional es la estrategia de búsqueda más utilizada en optimización y aprendizaje.

Desventajas:

- Cuando existen "colinas falsas" ocurre que el algoritmo de vuelta hacia atrás de manera extensa.
- Todos los nodos pueden parecer igualmente buenos.

¿CÓMO FUNCIONA EL ALGORITMO HILL CLIMBING?

- Usan una técnica de mejoramiento iterativo.
- Comienzan a partir de un punto (punto actual) en el espacio de búsqueda
- En cada iteración, un nuevo punto es seleccionado de la vecindad del punto actual.
- Si el nuevo punto es mejor, se transforma en punto actual, sino otro punto vecino es seleccionado y evaluado
- El método termina cuando no hay mejorías, o cuando se alcanza un número predefinido de iteraciones.
- Del procedimiento de prueba existe una realimentación que ayuda al generador a decidirse por cual dirección debe moverse en el espacio de búsqueda.
- En estos procesos se abandona la búsqueda si no existe un estado alternativo razonable al que se pueda mover.

- Los algoritmos de ascenso a colina son típicamente locales, ya que deciden qué hacer, mirando únicamente a las consecuencias inmediatas de sus opciones.
- Puede que nunca lleguen a encontrar una solución, si son atrapados en estados que no son el objetivo, desde donde no se puede hallar mejores estados.

Los ejemplos donde no se pueden hallar mejores estados en el algoritmo hill climbing son los siguientes:

1. Un máximo local, que es un estado mejor que sus vecinos pero no es mejor que otros que están algo más alejados.
2. Una meseta, es un espacio de búsqueda en el que todo un conjunto de estados vecinos tienen igual valor.
3. Un risco, que es un tipo especial de máximo local, imposible de atravesar con movimientos simples.
4. Planicie: son áreas del espacio de estados en donde la función de evaluación básicamente es plana.

Hay algunas formas que pueden ayudar a solventar estos problemas, aunque no existe garantía:

1. Para evitar máximos locales, regresar a un estado anterior y explorar en una dirección diferente.
 2. Para casos de mesetas, dar un salto grande en alguna dirección y tratar de encontrar una nueva sección del espacio de estado.
 3. Para los riscos, aplicar dos o más reglas, antes de realizar una prueba del nuevo estado, esto equivale a moverse en varias direcciones a la vez.
- En todos los casos anteriores, el algoritmo llega a un punto más allá del cual no se logra ningún avance,

- Cuando esto sucede es obvio que debe empezar de nuevo en otro punto, y esto es justamente lo que hace el algoritmo con ascenso de cima con reinicio aleatorio, efectúa una serie de búsquedas de ascenso de cima desde estados iniciales generados aleatoriamente, hasta para o cuando no se logra ningún avance significativo.
- Se guarda el mejor resultado que hasta un momento dado se haya obtenido en las diversas búsquedas.
- Puede usar un número fijo de iteraciones , o puede continuar hasta que el mejor de los resultados almacenados no haya sido mejorado para cierta cantidad de iteraciones.
- Los algoritmos de ascenso a colina, a pesar de explorar sólo un paso adelante, al examinar el nuevo estado pueden incluir una cierta cantidad de información global codificada en la función objetivo o función heurística.

¿QUÉ ES LA FUNCIÓN OBJETIVO O LA FUNCIÓN HEURÍSTICA?

FUNCIÓN HEURÍSTICA:

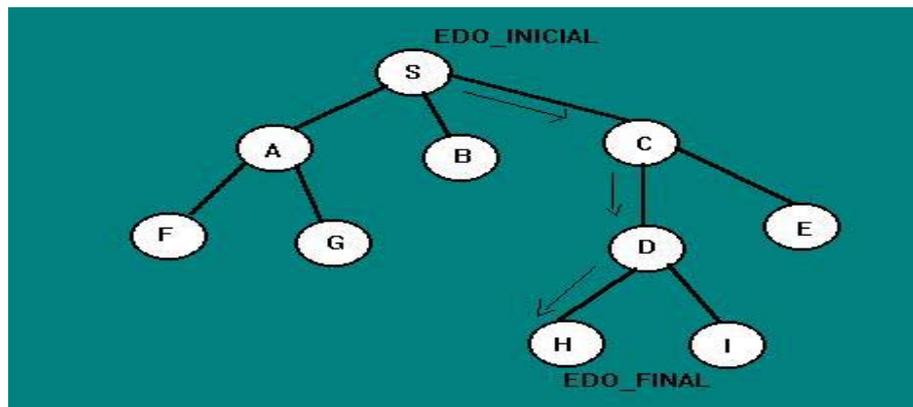
Una heurística es una regla o método que guía la decisión que hacemos al elegir un nodo que explorar, aunque no siempre permite hacer la mejor elección.

ESCALADA SIMPLE

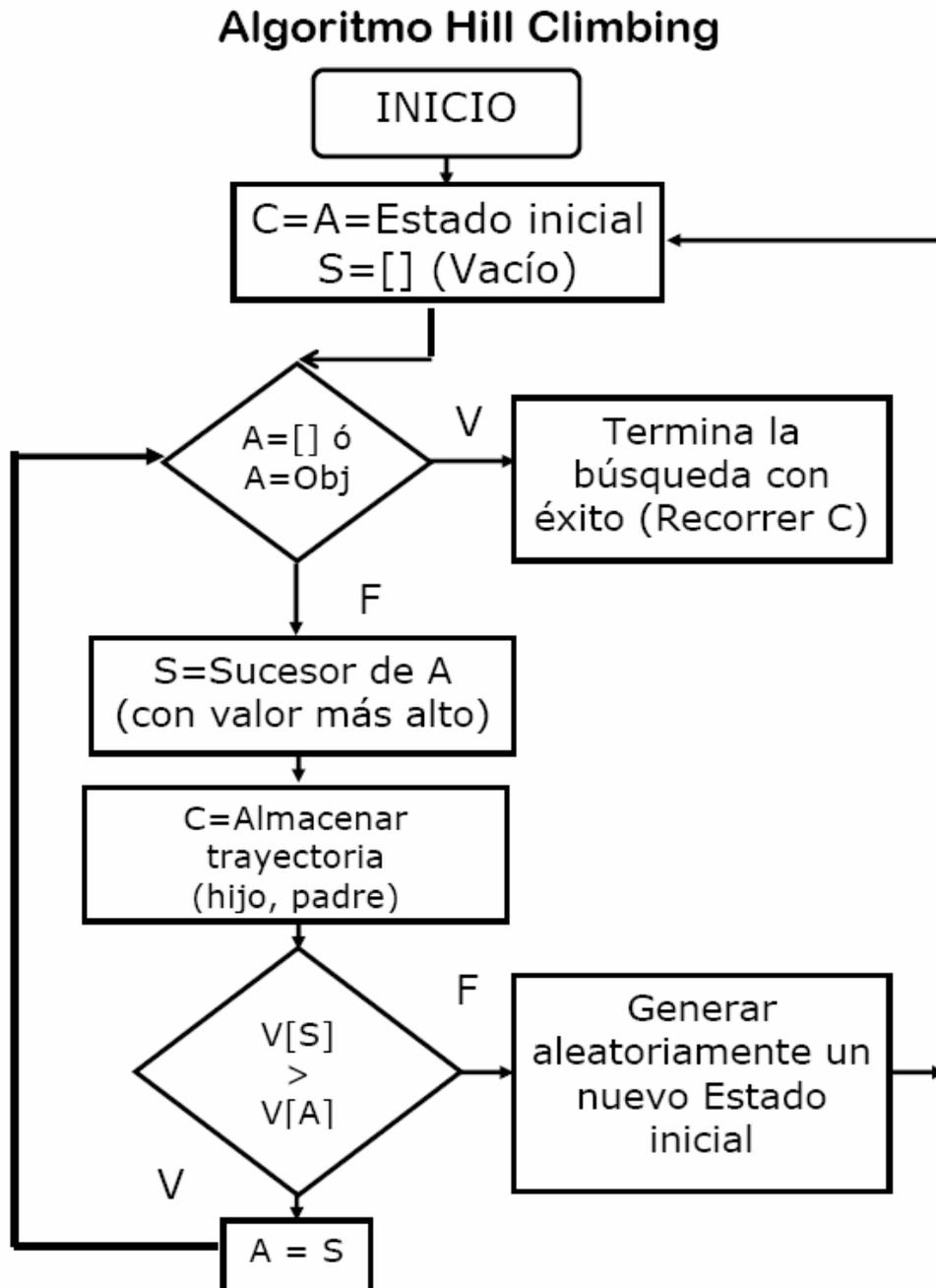
- Se busca cualquier operación que suponga una mejora respecto al padre
- Dirigirse siempre a un estado mejor que el actual
- Función Heurística de proximidad
- No se mantiene reporte de los estados anteriores
- Es un método local, sus movimientos están determinados por ser mejores que los previos.

RECORRIDO DE UN GRAFO MEDIANTE EL ALGORITMO HILL CLIMBING:

En este grafo se parte del estado_inicial y de allí se pasa al estado hijo que más se acerque a la meta. (Continúa utilizando este procedimiento hasta llegar al estado final o meta).



A continuación se muestra el diagrama de flujo para el algoritmo hill climbing el cual es elaborado mediante el método de escalada simple recordando que sus características son las siguientes:

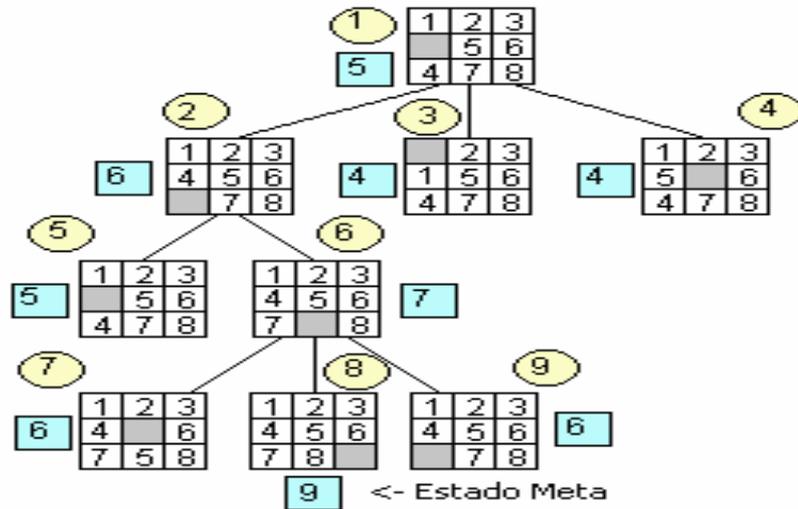


EJEMPLO: JUEGO 8- PUZZLE

- Establecer una función de evaluación donde:

$F(\text{nodo}) = \#$ de casillas bien

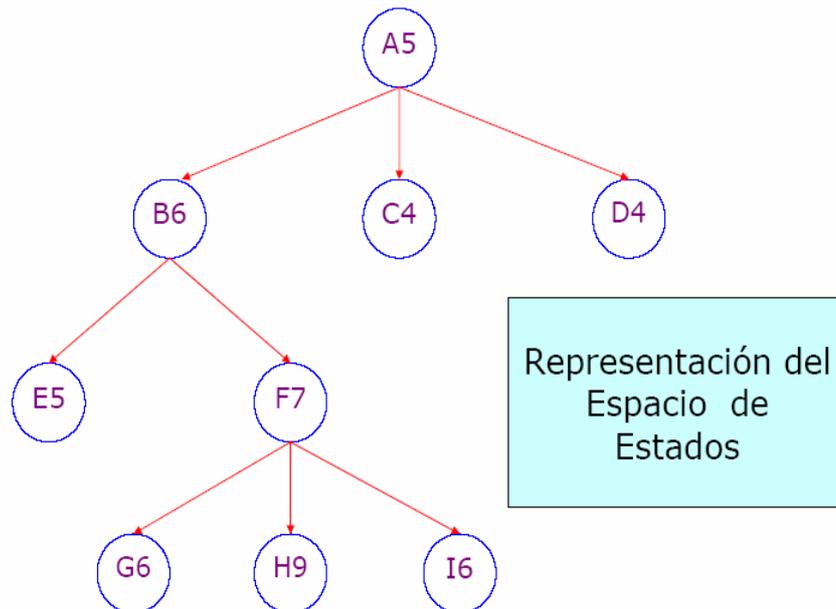
Colocadas (maximización)



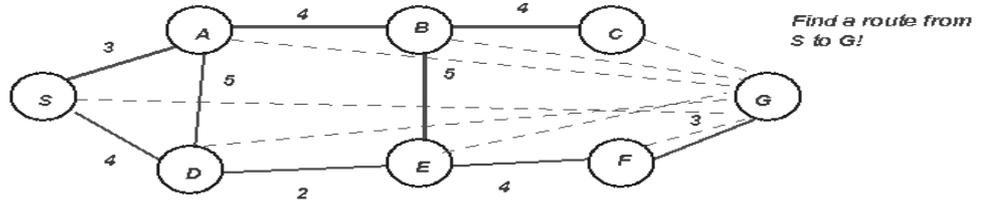
$f(\text{nodo}) = \text{número de casillas bien colocadas (maximizo)}$

● = Secuencia de estados generados

■ = Valor que devuelve función $f(\text{nodo})$

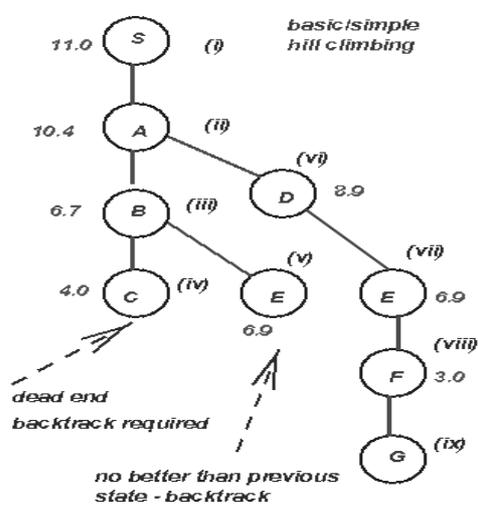


EJEMPLO DE RECORRIDO DE UN PUNTO A OTRO:

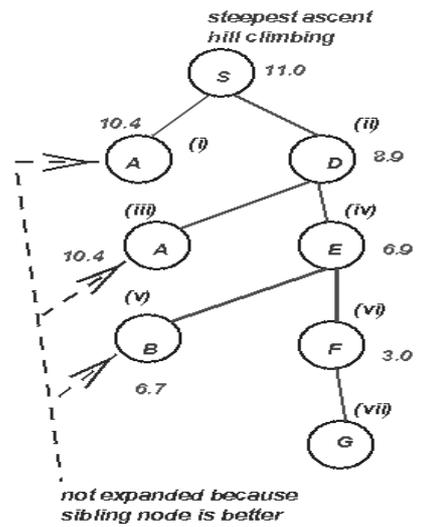


----- straight-line estimates of distance remaining

S-G = 11.0 A-G = 10.4 B-G = 6.7 C-G = 4.0 D-G = 8.9
 E-G = 6.9 F-G = 3.0



S-A-D-E-F-G = 17



S-D-E-F-G = 13

APLICACIONES DEL ALGORITMO HILL CLIMBING:

Dentro de las aplicaciones de los algoritmos genéticos se encuentran:

- Problemas de optimización.
- Recorrido de grafos.
- Juegos.
- Algoritmos genéticos.
- Recorridos simulados.

BIBLIOGRAFÍA:

<http://www.ilustrados.com/publicaciones/EpZVVZkuVlzIqITikQ.php#BASENCON>
<http://the-geek.org/docs/algen/>
<http://www.info-ab.uclm.es/asignaturas/42526/tema3.ppt>
http://html.rincondelvago.com/algoritmos_tecnicas-y-analisis.html
<http://www.lsi.upc.es/~luigi/docencia/2cBusquedalocal>
<http://www.cs.cornell.edu/gomes/selman-gomes-encycl-hillclimbing.pdf>
http://www.icmc.sc.usp.br/~sandra/G6_t2/rainha.htm
http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol2/hmw/article2.html
<http://pisuerga.inf.ubu.es/cgosorio/SExInArt/UD3/busqueda.pdf>
<http://www.dsic.upv.es/asignaturas/facultad/eda/teoria/tema5/t5eda.pdf>
<http://www.dcs.ex.ac.uk/~anarayan/teaching/com2408/259,10,Hill climbing>
http://library.thinkquest.org/18242/app_military.shtml
<http://www.itnuevolaredo.edu.mx/takeyas/>
http://helen.es_i.brandeis.edu/tron/
<http://www.ndsu.nodak.edu/instruct/juell/vp/es724s00/hill-climbing/hill-climbing.html>

