
UNIVERSIDAD SANTO TOMÁS
FACIENTES VERITATEM
UNIVERSIDAD
SANTO TOMÁS

PRESENTACIÓN

Ingeniero Electrónico, Magister en Ingeniería con énfasis en electrónica y estudiante del doctorado en ingeniería con énfasis en eléctrica y electrónica de la UDFJC

Diego Alejandro Barragán Vargas

Docente de electrónica Universidad Santo Tomás de Aquino

Enlace de Interés:
https://scholar.google.com/citations?hl=es&user=Bp3QMQMAAAAJ

UNIVERSIDAD SANTO TOMÁS

Sesión 6-Árboles de Búsqueda

4 de Septiembre, Bogotá D.C.

CONTENIDO

TEXTO COMPLEMENTARIO

Quiz Práctico

Problema del agente viajero (TSP)

El problema del agente viajero (TSP) consiste en encontrar la ruta más corta que visite cada ciudad exactamente una vez y regrese al punto de inicio.

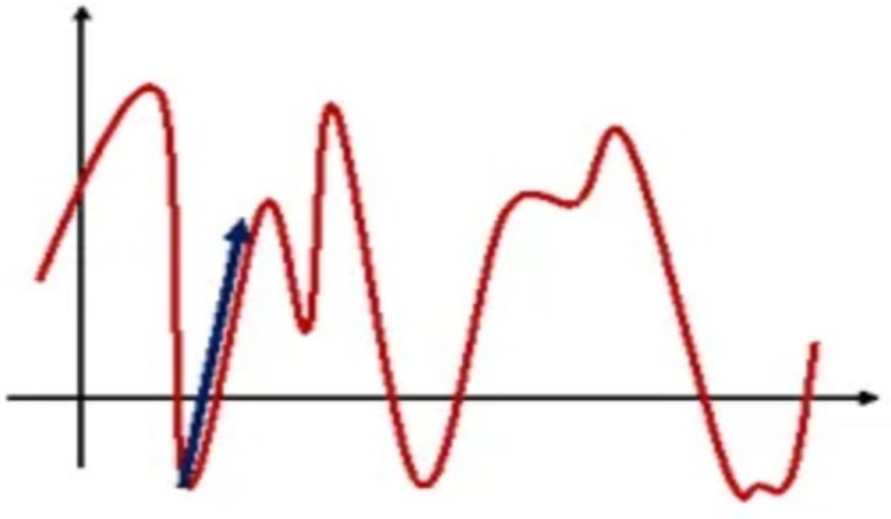
```
ciudades = {
    'A': (0, 0),
    'B': (1, 5),
    'C': (2, 3),
    'D': (5, 2),
    'E': (6, 6),
    'F': (7, 1),
    'G': (8, 4),
    'H': (9, 9)
    }
```

```python
def distancia_ruta(ruta):
    # Calcular la distancia total de una ruta
    pass  # Se debe implementar

def crear_poblacion_inicial(tam_poblacion):
    # Crear población inicial de rutas aleatorias
    pass  # Se debe implementar

def seleccion(poblacion, distancias):
    # Implementar selección por torneo o ruleta
    pass  # Se debe implementar

def cruce(padre1, padre2):
    # Implementar cruce ordenado (OX)
    pass  # Se debe implementar

def mutacion(ruta, tasa_mutacion):
    # Implementar mutación por intercambio
    pass  # Se debe implementar

# Se debe completar el algoritmo genético principal
```

Ascenso de Colinas (Hill Climbing)

Es un algoritmo de búsqueda local inspirado en la ascensión a la cima de montañas. Está diseñado para problemas de optimización cuyo objetivo es encontrar la mejor solución entre un conjunto de posibles soluciones.

Este algoritmo es especialmente eficaz para problemas con numerosas soluciones potenciales, con el objetivo de encontrar la óptima [1].



Fuente: https://medium.com/@tahsinsoyakk/hill-climbing-algorithm-a-comprehensive-guide-46e33f1ecc02

Fuente: https://tvymanga3.com/kengan-omega-269/

Hill Climbing funciona mejorando iterativamente una solución candidata hasta que no se puedan encontrar más mejoras.

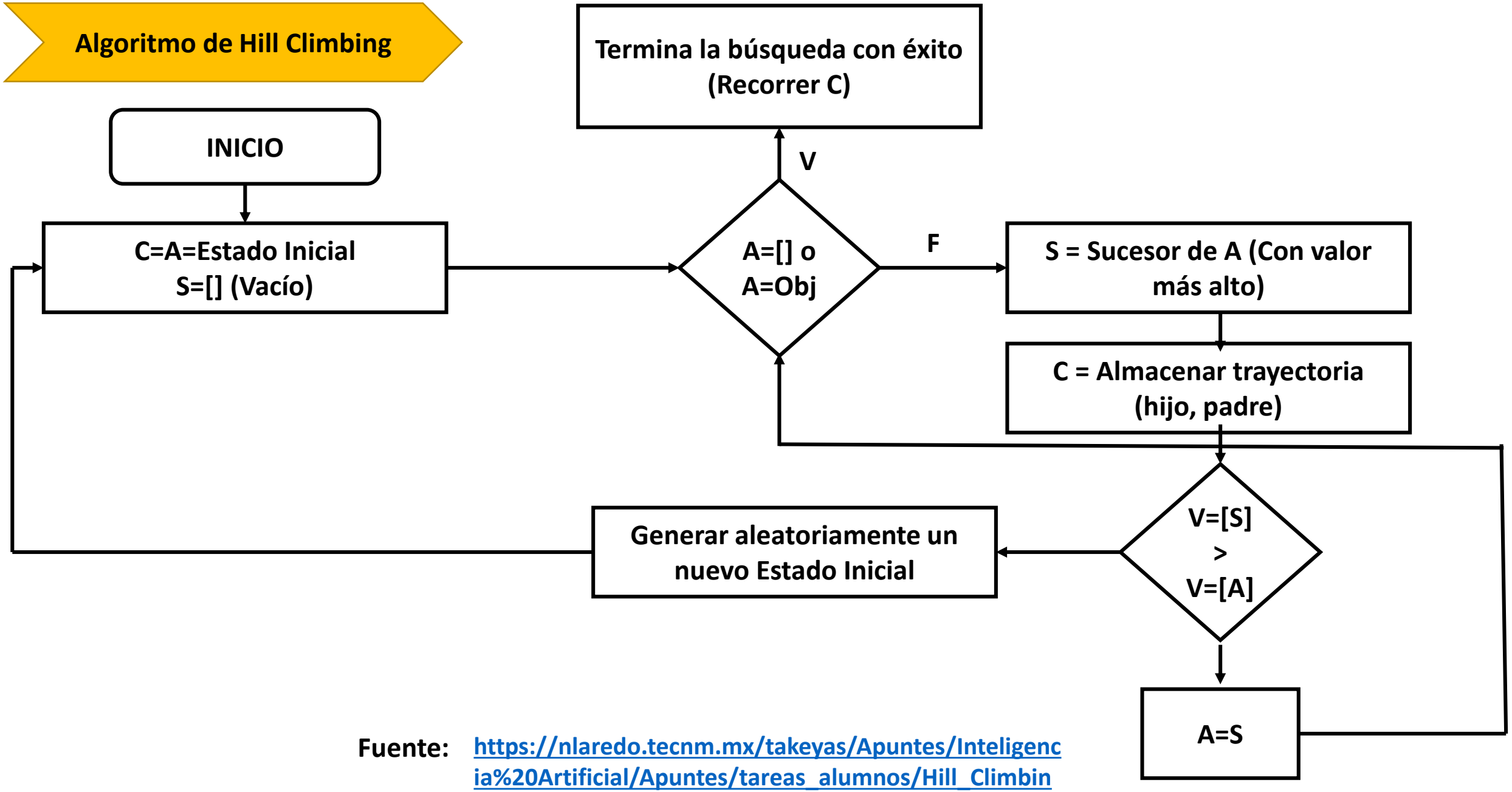
Funciona:

- **Inicialización** — Comenzar desde un punto inicial (solución inicial).
- **Evaluación Vecinal** — Evaluar soluciones vecinales para encontrar una mejor.
- **Mover** — Mover a la solución vecina si es mejor.
- **Repetir** — Continuar hasta que no exista un vecino mejor.

3 Escenarios posibles:

- **Mejora en una Dirección** — Si se mejora el resultado en una dirección el algoritmo sigue por esa ruta.
- **Sin Mejora en ninguna Dirección** — Si ninguna dirección mejora el resultado, el algoritmo considera un óptimo local y se detiene.
- **Mejora en ambas direcciones** — Si ambas direcciones mejoran la solución, el algoritmo selecciona uno de los dos caminos aleatoriamente [2].

Algoritmo de Hill Climbing



Fuente: https://nlaredo.tecnm.mx/takeyas/Apuntes/Inteligencia%20Artificial/Apuntes/tareas_alumnos/Hill_Climbing/Hill_Climbing(2005-II-A).pdf

Recocido Simulado

Es un algoritmo de optimización aleatoria. se inspira en el proceso de recocido en metalurgia. Permite aceptar soluciones peores con una probabilidad que disminuye con el tiempo.

Se compone de 4 pasos:

Se comienza en un punto aleatorio x.

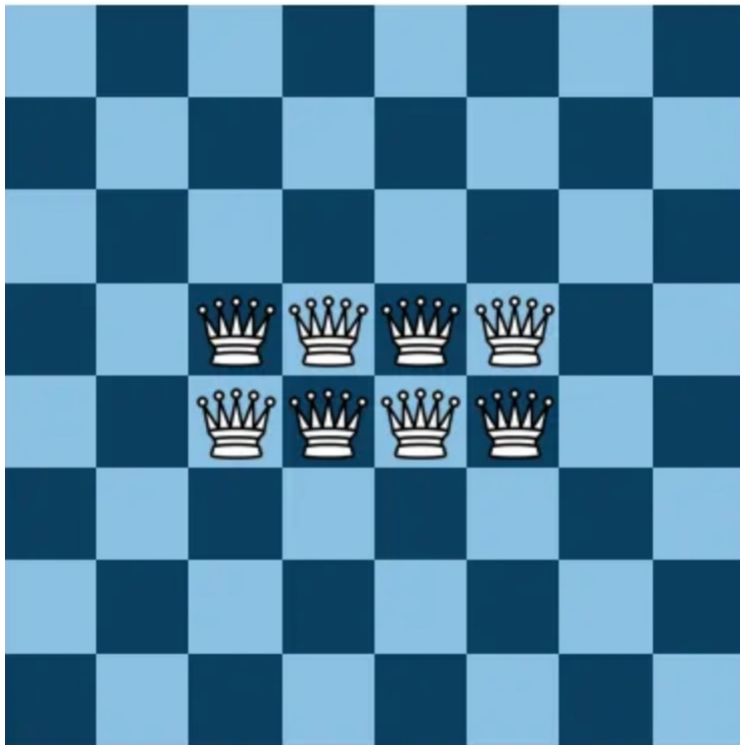
Se escoge un punto nuevo x_j dentro de una vecindad N(x).

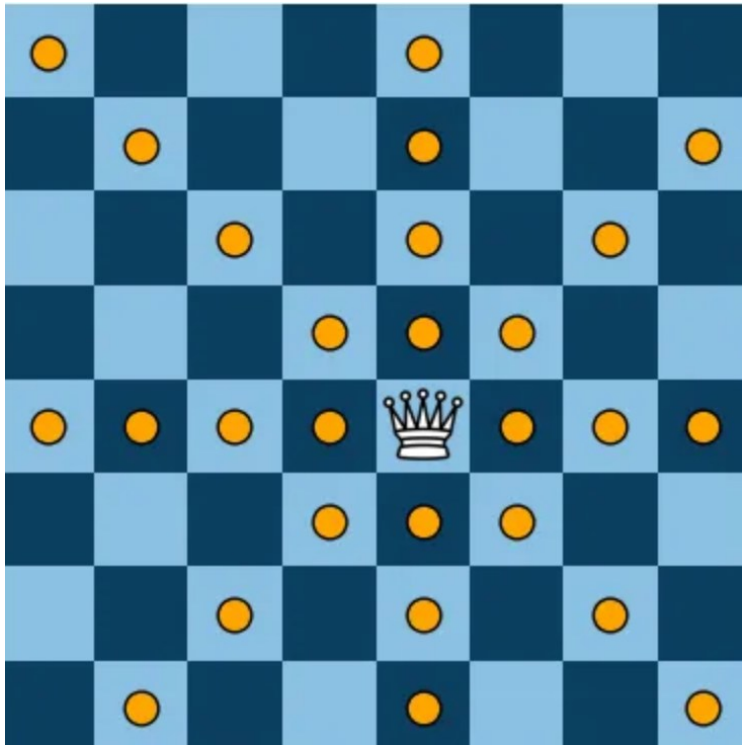
Se decide si se mueve al nuevo punto x_j. Esta decisión se hará en base a la función de probabilidad $P(x,x_j,T)$ que explicaremos a continuación.

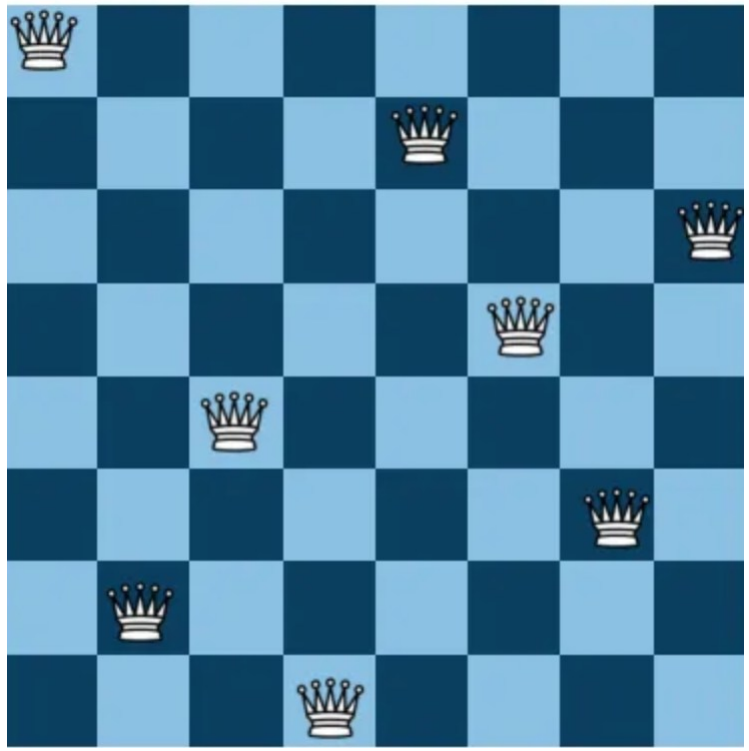
$$\mathbb{P}(x, x_j, T) = \begin{cases} 1 & si \quad F(x_j) \geq F(x) \\ e^{\frac{F(x_j)-F(x)}{T}} & si \quad F(x_j) < F(x) \end{cases}$$

Se disminuye T

Problema de las 8 Reinas







Fuente: https://medium.com/@davidlfliang/intro-python-algorithms-eight-queens-problem-fdcc5cf384d5

Referencias

[1] T. Soyak, “Hill climbing algorithm: A comprehensive guide”, *Medium*, 28-jul-2024. [En línea]. Disponible en: https://medium.com/@tahsinsoyakk/hill-climbing-algorithm-a-comprehensive-guide-46e33f1ecc02. [Consultado: 17-sep-2025].

[2] *Datacamp.com*. [En línea]. Disponible en: https://www.datacamp.com/tutorial/hill-climbing-algorithm-for-ai-in-python. [Consultado: 17-sep-2025].

[3] *Tecnm.mx*. [En línea]. Disponible en: https://nlaredo.tecnm.mx/takeyas/Apuntes/Inteligencia%20Artificial/Apuntes/tareas_alumnos/Hill_Climbing/Hill_Climbing(2005-II-A).pdf. [Consultado: 17-sep-2025].

UNIVERSIDAD SANTO TOMÁS

GRACIAS