Recocido Simulado

Recocido en la Naturaleza Recocido Simulado Solución candidata Estado del Sistema al agar Configuración molecular Variables de decisión Función objetivo Energía evalua. Estado Fundamental Solución óptima global 🗼 Óptima local Estado meta estable Tendencia a explorar Temperatura el espacio de búsqueda Disminución de la tendencia Enfriamiento a explorar Cambio de configuraciones Cambio de solución candidata moleculares

Temperatura Inicial
ude (vada

1.b) Describe e implementa un esquema de enfriamiento.
- Justifica tu elección del esquema de enfriamiento.

Enfriamiento Lineal

• Este es el esquema más simple. Se utiliza la siguiente función:

$$T_{k+1} = T_0 - \eta k$$

Enfriamiento con Decremento Lento

$$T_{k+1} = T_k/(1+\beta T_k)$$

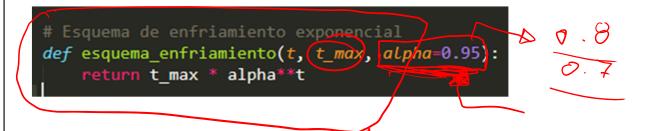
Enfriamiento Logarítmico

 $T_{k+1} = c/\ln k$

 Dependientes de la dimensión Para funciones que tienen diferentes topologías entre diferentes dimensiones, se pueden utilizar diferentes esquemas de enfriamiento para diferentes dimensiones.

- 1.b) Describe e implementa un esquema de enfriamiento.
 Justifica tu elección del esquema de enfriamiento.
- **Geométrico**, también referenciado como exponencial.

$$T_{k+1} = \alpha T_k, \alpha \in (0,1)$$

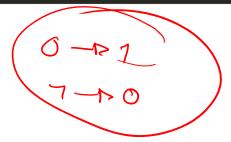




- 1.a) Describe e implementa un operador de vecindad para soluciones binarias.
 - La función solo debe generar un vecino de manera aleatoria.

```
# Operador de vecindad para soluciones binarias

def generar_vecino(solucion):
    vecino = np.copy(solucion)
    idx = np.random.randint(len(vecino))
    vecino[idx] = 1 - vecino[idx] # Cambiar aleatoriamente un bit
    return vecino
```



Recorido Simuado - Sol alcatoria inicial de generarla _ evaluar sol a— funci obj xiteración probabilidad + Lo elegico Deors exploración esq. enfriat tempora - solactual + búsqueda local +> explotación - criterio parada * devolver Mejor Sol = n iteraciones, 100 000, 1M - n evertuacione - tiempo