

Proyecto de Optimización

Josue Santana Robledo Corona 325073061

Maestría en Ciencias de la Robótica e Inteligencia Artificial

Algoritmos Bio-Inspirados

Centro Universitario de Ciencias Exactas e Ingeniería.

Mtro. Carlos Alberto López Franco

OPTIMIZACIÓN DE COBERTURA EN AJEDREZ USANDO ALGORITMO GENÉTICO

- **Problema:** ¿Cómo maximizar la cobertura en un tablero de ajedrez?
- **Objetivo:** Encontrar la disposición óptima de piezas que controle más casillas

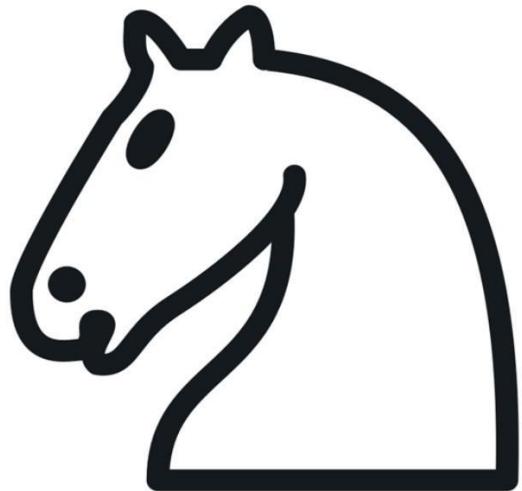
PLAN DE EJECUCIÓN

FASE 1: Definición del problema y representación

FASE 2: Implementación del Algoritmo

FASE 3: Validación y Análisis de Resultados

Codificación



Tipo	Fila	Columna
00	101	101

8bits

- 00 → Torre
- 01 → Alfil
- 10 → Caballo
- 11 → Reina

Individuo = [(Pieza1), (Pieza2), (Pieza3), (Pieza4)]

**Población = [(Individuo1), (Individuo2), (Individuo 3),
(Individuo 4)]**

Selección de Padres

- Evalúa cada individuo
- Calcula su fitness



%ruleta=0.3

%torneo=0.7

Elegibles por el usuario

Cruzamiento

```
if random.random() < prob_cruza:
```

```
punto = random.randint(1, len(padre1) - 1)
```

Ejemplo:

Padre 1 = [A A A | A A]

Padre 2 = [B B B | B B]

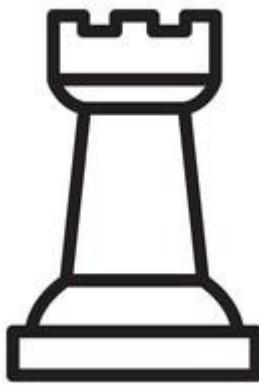
Punto = 3

Resultado:

Hijo 1 = [A A A | B B]

Hijo 2 = [B B B | A A]

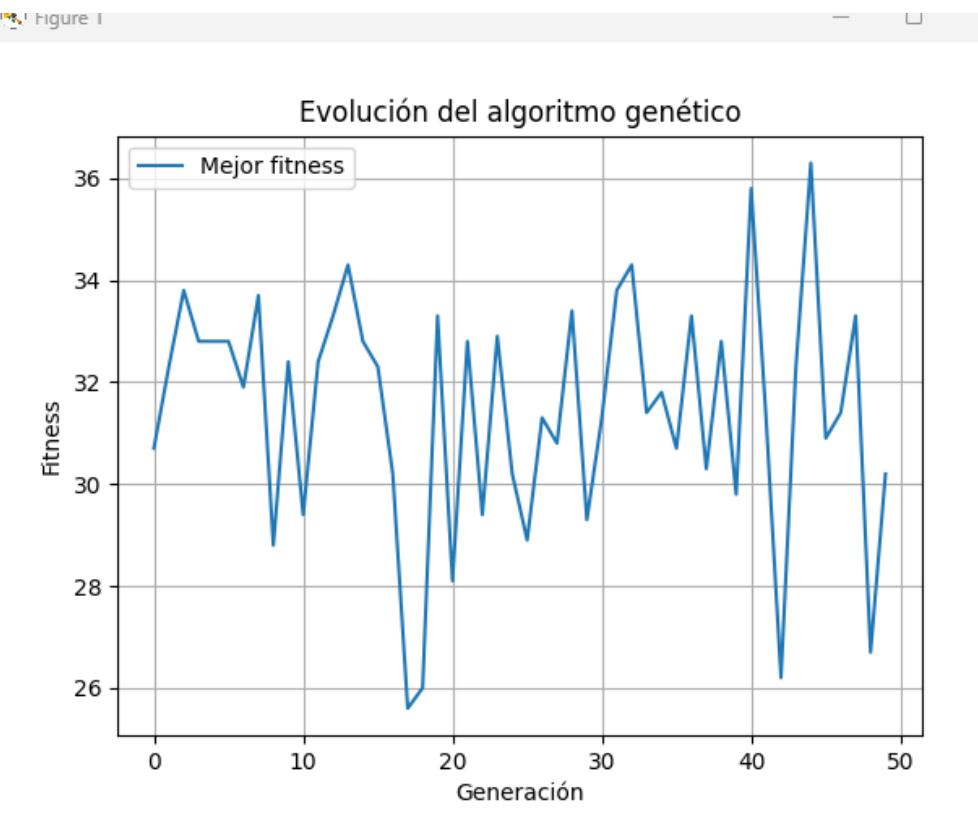
Mutación



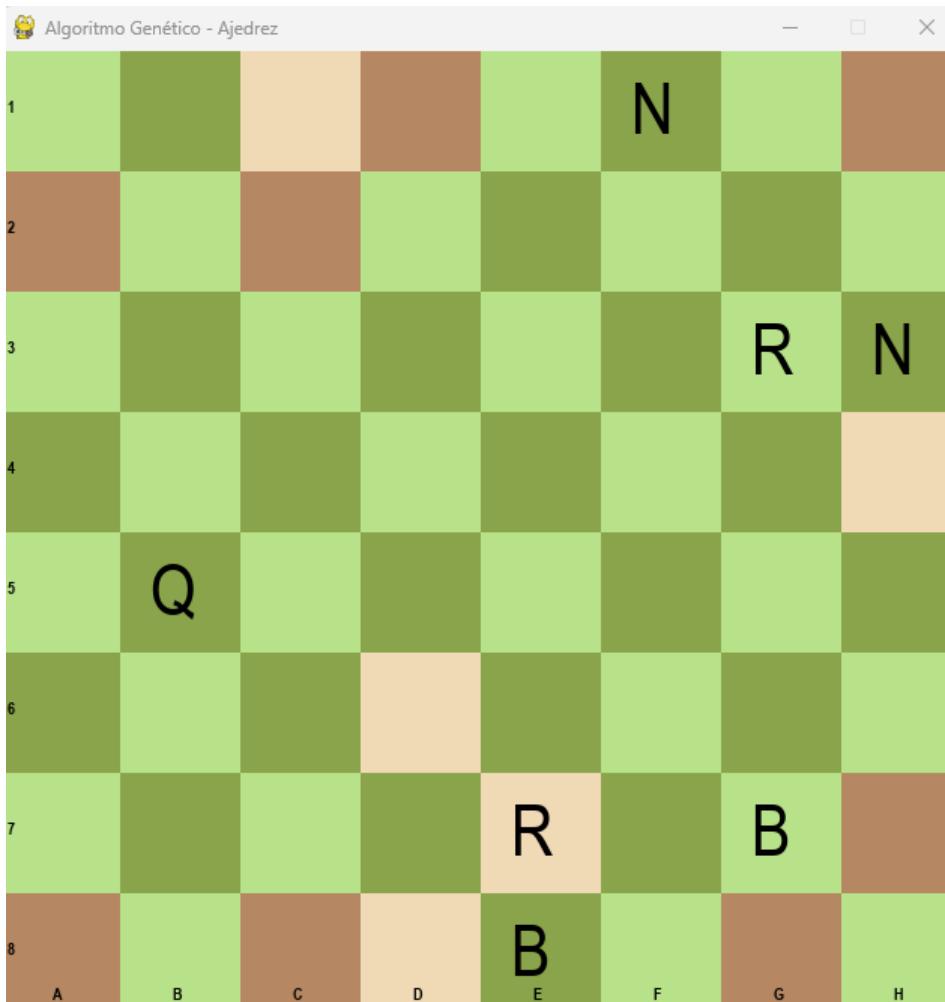
- **Se revisa cada bit del cromosoma.**
- `random.random() < prob_mutacion`
- $1 \rightarrow 0$
- $0 \rightarrow 1$

Función

Fitness = $(w_1 * \text{Cobertura total}) + (w_2 * \text{Valor total de piezas}) - (w_3 * \text{Penalización})$



Resultados



Iteración 48: Fitness = 26.70, Casillas controladas = 53/64
Iteración 49: Fitness = 30.20, Casillas controladas = 43/64
Mejor solución en iteración: 44, Mejor fitness: 36.3, Casillas controladas: 54/64

Futuras mejoras

- Detectar colisiones para mayor cantidad de piezas
- Reemplazar las letras por fichas
- Verificar que el tablero tenga al menos 1 tipo de cada pieza
- Guardar imágenes automáticamente del mejor tablero
- Visualización del tablero en tiempo real
- Panel de estadísticas
- Visualización avanzada
- Evitar ataques entre las mismas piezas
- Diversidad controlada del tablero

Pseudocódigo

Generar población inicial P

Para t = 1 hasta max_iter:

 Evaluar aptitud de cada individuo en P

 Seleccionar padres

 Crear nuevos individuos por cruza

 Aplicar mutación

 Formar nueva población P

Regresar el mejor individuo

FIN

Referencias

https://en.wikipedia.org.translate.goog/wiki/Queen%27s_graph?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc

<https://chess.stackexchange.com/questions/23258/what-is-the-least-number-of-knights-needed-to-cover-the-entire-board>

<https://books.google.com.mx/books?hl=es&lr=&id=pAwrDwAAQBAJ&oi=fnd&pg=PT6&dq=Minimum+Dominating+Set+ajedrez&ots=d6wXGSYRAP&sig=GA8ce-FuzyEo14w8wYZ1yREhQA4#v=onepage&q&f=false>

<https://riunet.upv.es/server/api/core/bitstreams/5a340cc1-ea88-4201-b839-6ac10454e9de/content>

[https://en.wikipedia.org/wiki/Mathematical_chess_problem#:~:text=A%20domination%20\(or%20covering\)%20problem,are%20provided%20on%20diagrams%20below.](https://en.wikipedia.org/wiki/Mathematical_chess_problem#:~:text=A%20domination%20(or%20covering)%20problem,are%20provided%20on%20diagrams%20below.)

<https://github.com/dmackinnon1/chessdom>

<https://www.mayhematics.com/a/arrangements.html>

<https://www.youtube.com/watch?v=y6tnwN94RPg>

<https://www.youtube.com/watch?v=HRAxpCi-RTY>

https://www.youtube.com/watch?v=NzM_xTwL1R4

<https://www.youtube.com/watch?v=areuz-PprfU>

https://www.researchgate.net/publication/351336246_Domination_in_Chessboards/references

**GRACIAS POR SU
ATENCIÓN**