

# Optimización de Distribuciones de Teclado en Inglés

Jordi Cantavella Ferrero

MIARFID  
Universitat Politècnica de València  
Valencia, 23-10-2025





Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

## 1 Introducción

## 2 AG-ES

## 3 Implementación

## 4 Experimentos y resultados

## 5 Conclusiones



# Índice

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

## 1 Introducción

## 2 AG-ES

## 3 Implementación

## 4 Experimentos y resultados

## 5 Conclusiones

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

# El Problema del Diseño de Teclados

## QWERTY (1873)

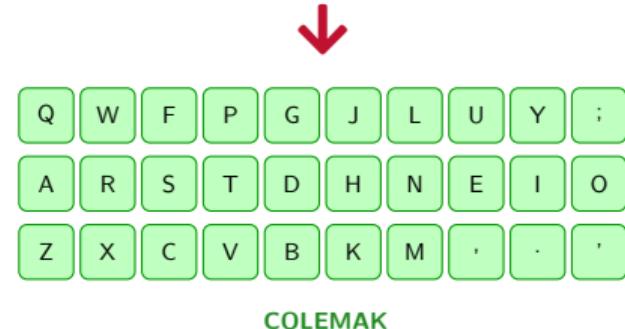
Diseñado para máquinas mecánicas, no para eficiencia



## Problemas

- Alta distancia recorrida
- Baja alternancia de manos
- Distribución subóptima
- Lesiones por esfuerzo repetitivo

¿Podemos optimizarlo?





# Índice

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

1 Introducción

2 AG-ES

3 Implementación

4 Experimentos y resultados

5 Conclusiones

# Codificación del Individuo

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

## Representación

Lista de 30 elementos mapeados a posiciones  
 $3 \times 10$ :

```
[a, b, c, ..., z, " ", ;, ']
```

### Ejemplo: distribución QWERTY

```
[q, w, e, r, t, y, u, i, o, p,  
a, s, d, f, g, h, j, k, l, ;,  
z, x, c, v, b, n, m, " ", ']
```

## Estructura $3 \times 10$

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29

### Ejemplo: QWERTY

Q	W	E	R	T	Y	U	I	O	P
A	S	D	F	G	H	J	K	L	;
Z	X	C	V	B	N	M	,	-	'

# Función de Fitness

## Evaluación basada en Bigramas

Costo de escribir pares de letras según su frecuencia

$$\text{Fitness} = \sum_{\text{bigramas}} \text{costo} \times \text{frecuencia}$$

$$\text{costo} = \text{distancia} \times \max(1.0 + \sum \text{penalties}, 0.1)$$

### Componentes

- ① Distancia euclídea
- ② Same-finger penalty
- ③ Same-hand penalty
- ④ Row jump penalty
- ⑤ Weak finger penalty

### Objetivo

**MINIMIZAR**

Fitness total del layout

# Función de Fitness: Ejemplo

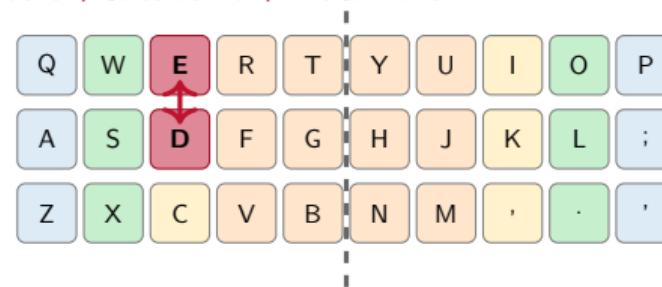
## Penalties

- Same-finger: +1.0 / +3.0
- Same-hand: +1.0
- Alternancia: -1.0
- Row jump: +0.2 / +0.8
- Dedos débiles: +0.10 / +0.15

Ejemplo: "ed"

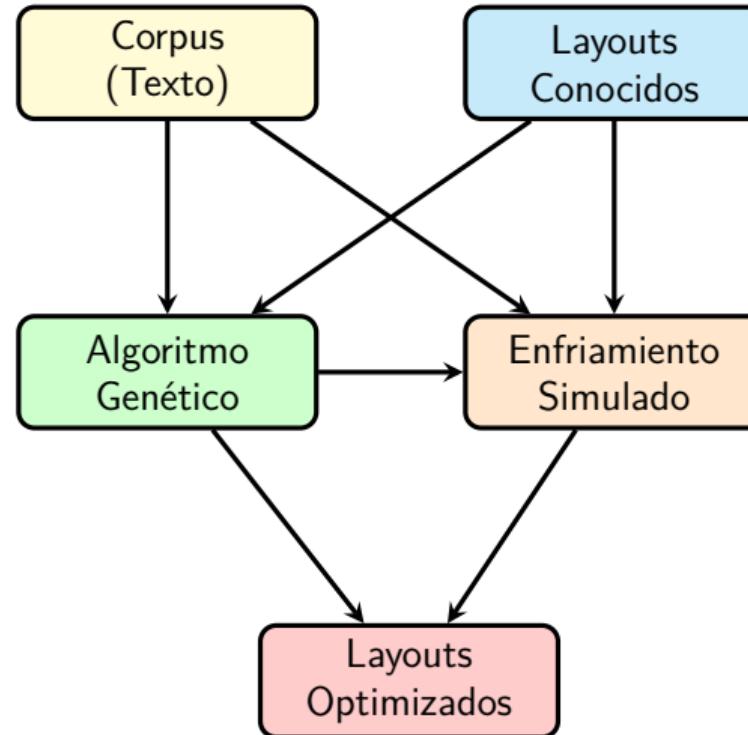
Distancia:	1.0
Same-finger:	+1.0
Row jump:	+0.2
<hr/>	
Multiplicador:	2.2
Freq("ed"):	1500
<hr/>	
<b>Total:</b>	<b>3300</b>

Mismo dedo + Salto de fila + Misma mano



# Arquitectura de los Experimentos

Introducción  
AG-ES  
Implementación  
Experimentos y resultados  
Conclusiones





# Índice

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

**1** Introducción

**2** AG-ES

**3** Implementación

**4** Experimentos y resultados

**5** Conclusiones

# Implementación

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

## Tecnologías

- **Lenguaje:** Python
- **Librerías:** NumPy, Matplotlib
- **Optimización:** Matrices precalculadas  $30 \times 30$

## Algoritmo Genético

- Selección: Torneo ( $k=3-10$ )
- Cruce: Dos puntos
- Mutación: Swap aleatorio (5 %-75 %)
- Elitismo: (5-50 %)
- Población: (1.000-1.000.000)

## Enfriamiento Simulado

- **Vecindad:**
  - Random: swap aleatorio
  - Local: swap adyacente
- **Schedules:** Geometric, Linear, Logarithmic
- **T\_inicial:** 5,000-20,000
- **k:** 0.9995-0.99995

## Experimentos

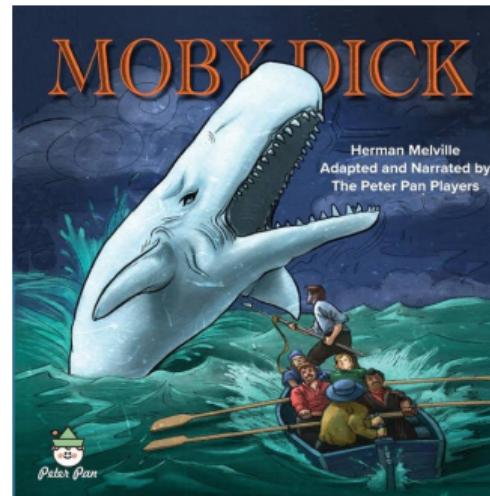
- **AG:** 4 series  $\times$  4-7 valores
- **ES:** 8 configs  $\times$  2 vecindades  $\times$  10 layouts

[Introducción](#)[AG-ES](#)[Implementación](#)[Experimentos y resultados](#)[Conclusiones](#)

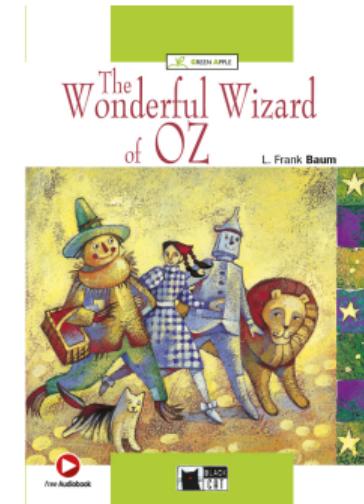
# Corpus de Texto Utilizados

## Datasets para Experimentación

Dos libros clásicos con características lingüísticas diferentes.



**Moby Dick**  
Herman Melville (1851)



**The Wizard of Oz**  
L. Frank Baum (1900)



# Índice

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

1 Introducción

2 AG-ES

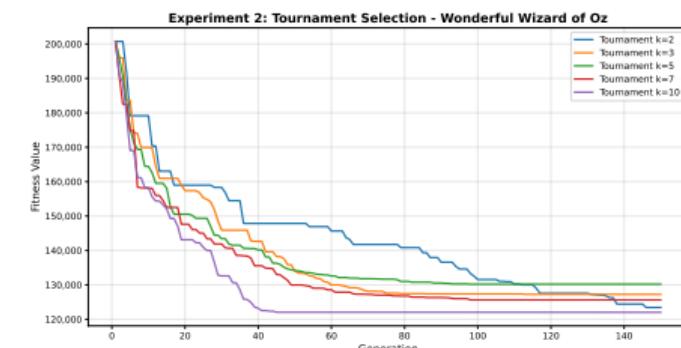
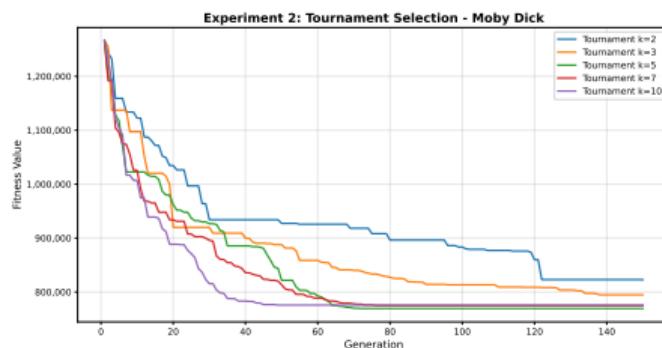
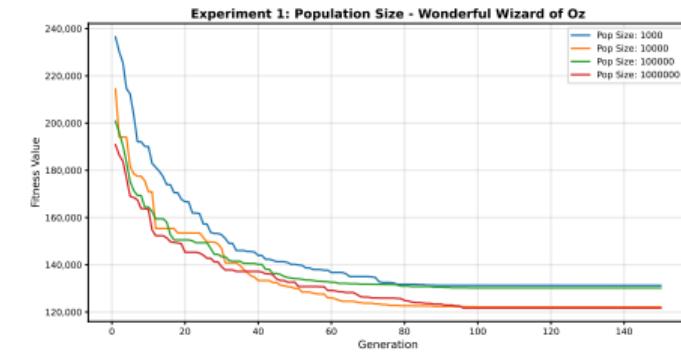
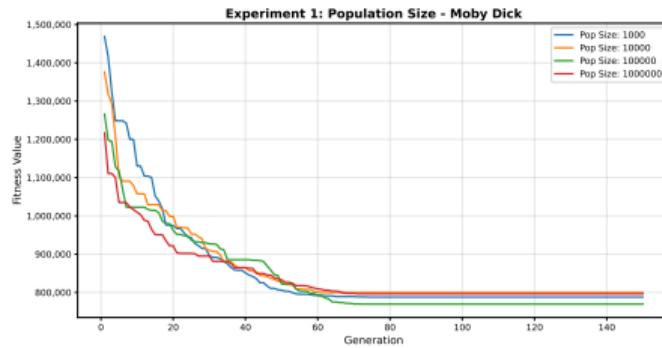
3 Implementación

4 Experimentos y resultados

5 Conclusiones

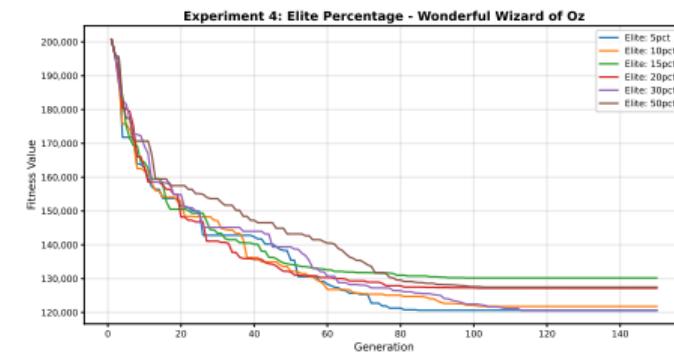
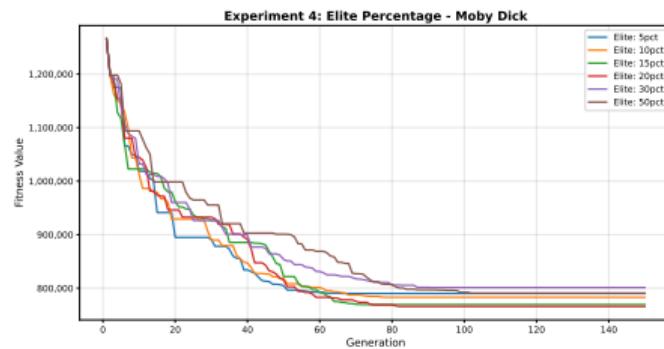
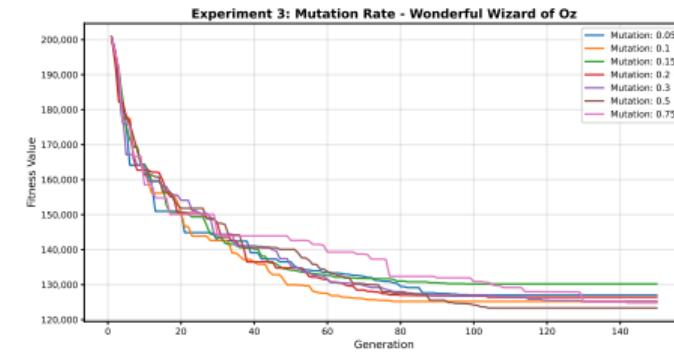
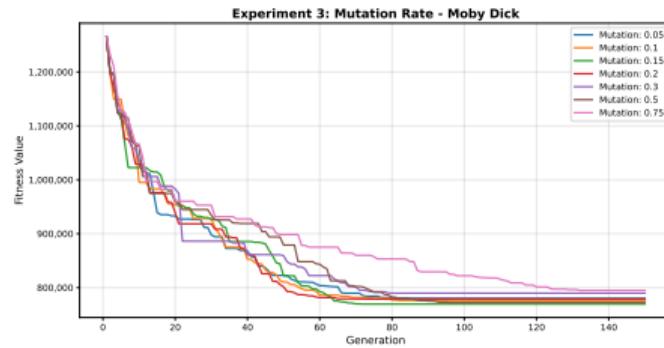
# Experimentos Genéticos (I)

Introducción  
AG-ES  
Implementación  
Experimentos y resultados  
Conclusiones



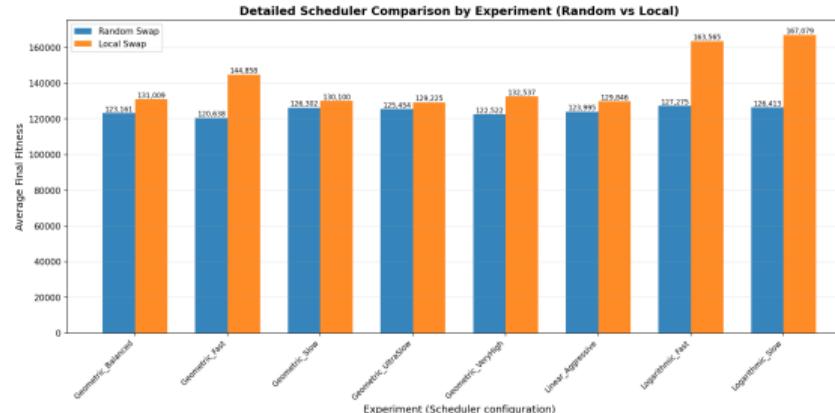
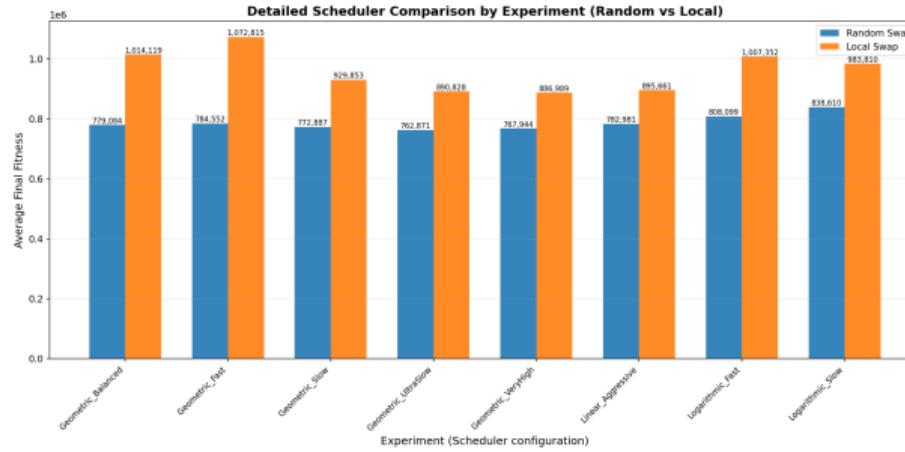
# Experimentos Genéticos (II)

- Introducción
- AG-ES
- Implementación
- Experimentos y resultados
- Conclusiones



# Experimentos Enfriamiento (I)

Introducción  
AG-ES  
Implementación  
Experimentos y resultados  
Conclusiones



# Experimentos Enfriamiento (II)

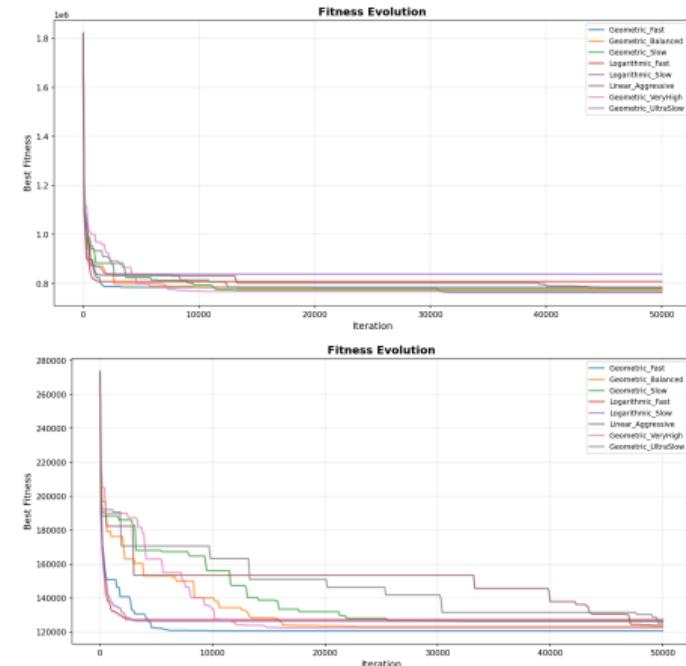
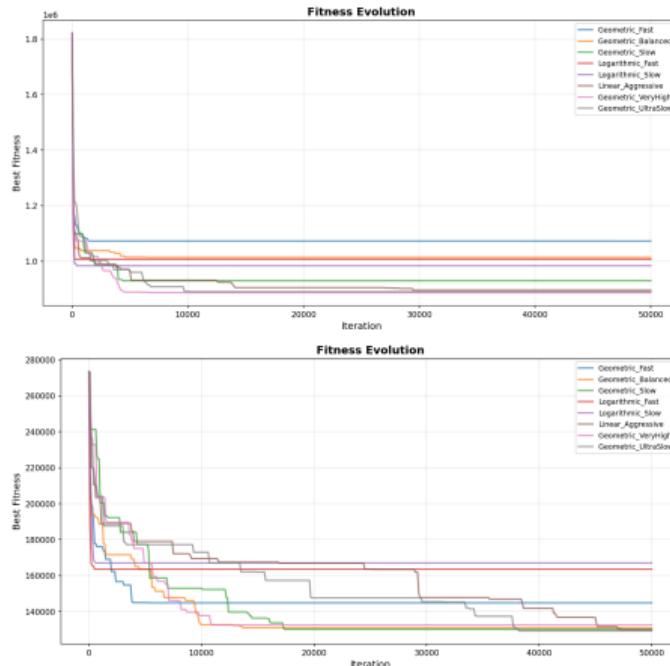
Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones





# Índice

Introducción

AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

1 Introducción

2 AG-ES

3 Implementación

4 Experimentos y resultados

5 Conclusiones

# Consideraciones finales

Introducción

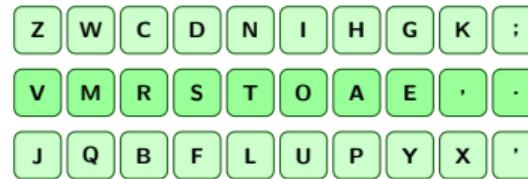
AG-ES

Implementación

Experimentos y  
resultados

Conclusiones

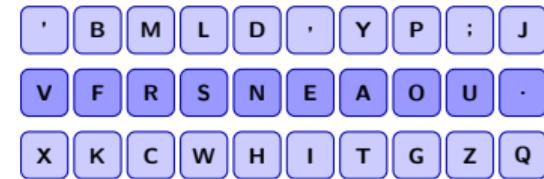
## Moby Dick (1.2M chars)



**Fitness: 2,982.5**

65.0 % mejor que QWERTY

## Wizard of Oz (0.3M chars)



**Fitness: 3,437.1**

59.7 % mejor que QWERTY

## Conclusión

- **AG** → mejor exploración; **ES** → mayor convergencia.
- Resultados estables y **generalizables** entre corpus.
- AG + ES híbrido **promete mejores resultados**.
- Layout para Moby Dick presenta mayor **generalización**.

# ¡Gracias por vuestra atención!

¿Preguntas?

 GitHub:  
[github.com/JordiCan/  
hybrid-keyboard-optimizer](https://github.com/JordiCan/hybrid-keyboard-optimizer)

 Recursos:  
[keybr.com/layouts](http://keybr.com/layouts)

 Email:  
[jcanfer1@etsinf.upv.edu.es](mailto:jcanfer1@etsinf.upv.edu.es)