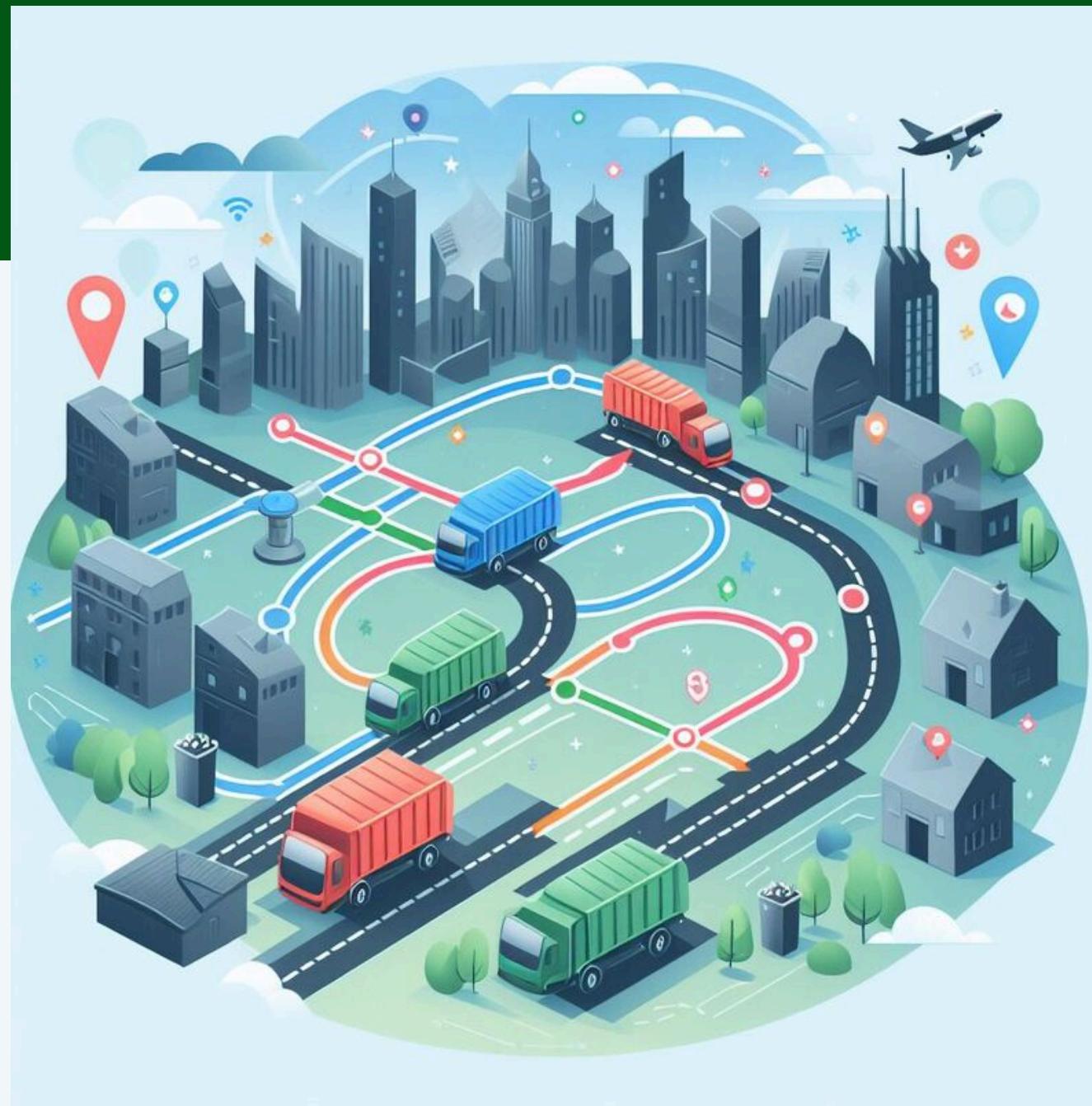


Presentan:

- Brito Liévano Miguel Ángel
- Mancera Quiroz Javier Alejandro

# OPTIMIZACIÓN DE RUTAS

## para la Gestión de Residuos Sólidos con Algoritmos Genéticos



# PROBLEMÁTICA

- 1. Deterioro de las flotillas**
- 2. Altos costos de mantenimiento**
- 3. Deficiente planeación de rutas**
- 4. Baja cobertura de recolección**
- 5. Emisiones contaminantes**



# INTRODUCCIÓN

La gestión eficiente de residuos sólidos es crucial para minimizar los costos de transporte, reducir las emisiones de carbono y garantizar una recolección oportuna. En este proyecto, utilizaremos algoritmos genéticos para optimizar las rutas de recolección de basura en una ciudad.



# Descripción del problema

- **Objetivo: Minimizar la distancia total recorrida por los camiones de recolección mientras se visitan todos los contenedores de basura.**

## Datos disponibles

- Cantidad de camiones.
- Los camiones cargan una cantidad adecuada de residuos.
- Restricciones de tiempo para cada camión (horarios de recolección, ventanas de tiempo).
- Mapa de la ciudad con calles y restricciones de tráfico.
- Ubicaciones de los contenedores de basura.
- Demanda de residuos en cada contenedor (cantidad de basura generada).



# Representación de soluciones

## Solución

$[(a,b,\dots,c), (d,e,\dots,f), (g,h,\dots,i), (j,k,\dots,l)]$

- Cada solución estará representada como un conjunto de permutaciones.
- Cada permutación representará el recorrido de un camión.

Camión



Recorrido  
(a,b,...c)



(d,e,...f)



(g,h,...i)



(j,k,...l)

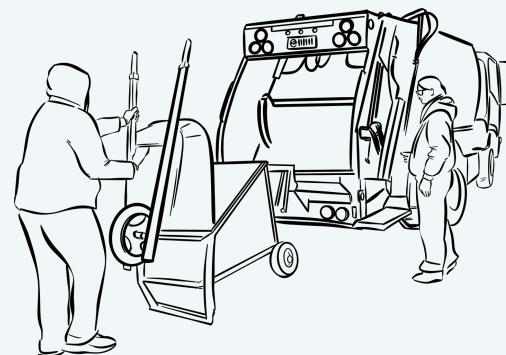
# **FUNCIÓN OBJETIVO**

**Por recorrido:**

**1. Evaluación de los pesos de cada arista (tráfico de cada calle).**



**2. Evaluación de la cantidad de residuos que se recogió.**



**3. Evaluación de tiempo del recorrido.**



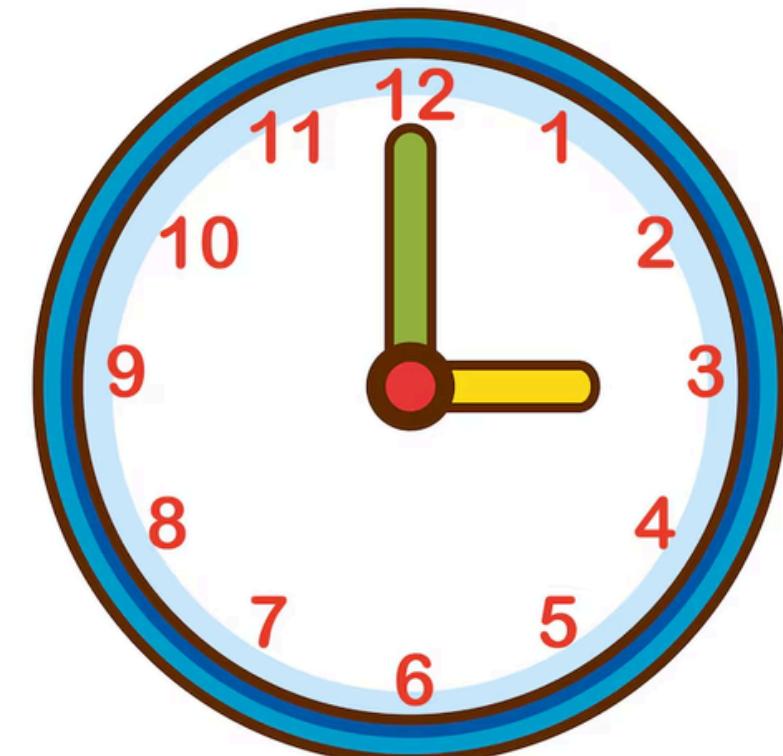
**Por solución:**

**4. Suma de todas las evaluaciones hechas para cada camión.**



# HORARIOS

- Se aplicarán penalizaciones grandes por estar fuera de horario en la función de evaluación.



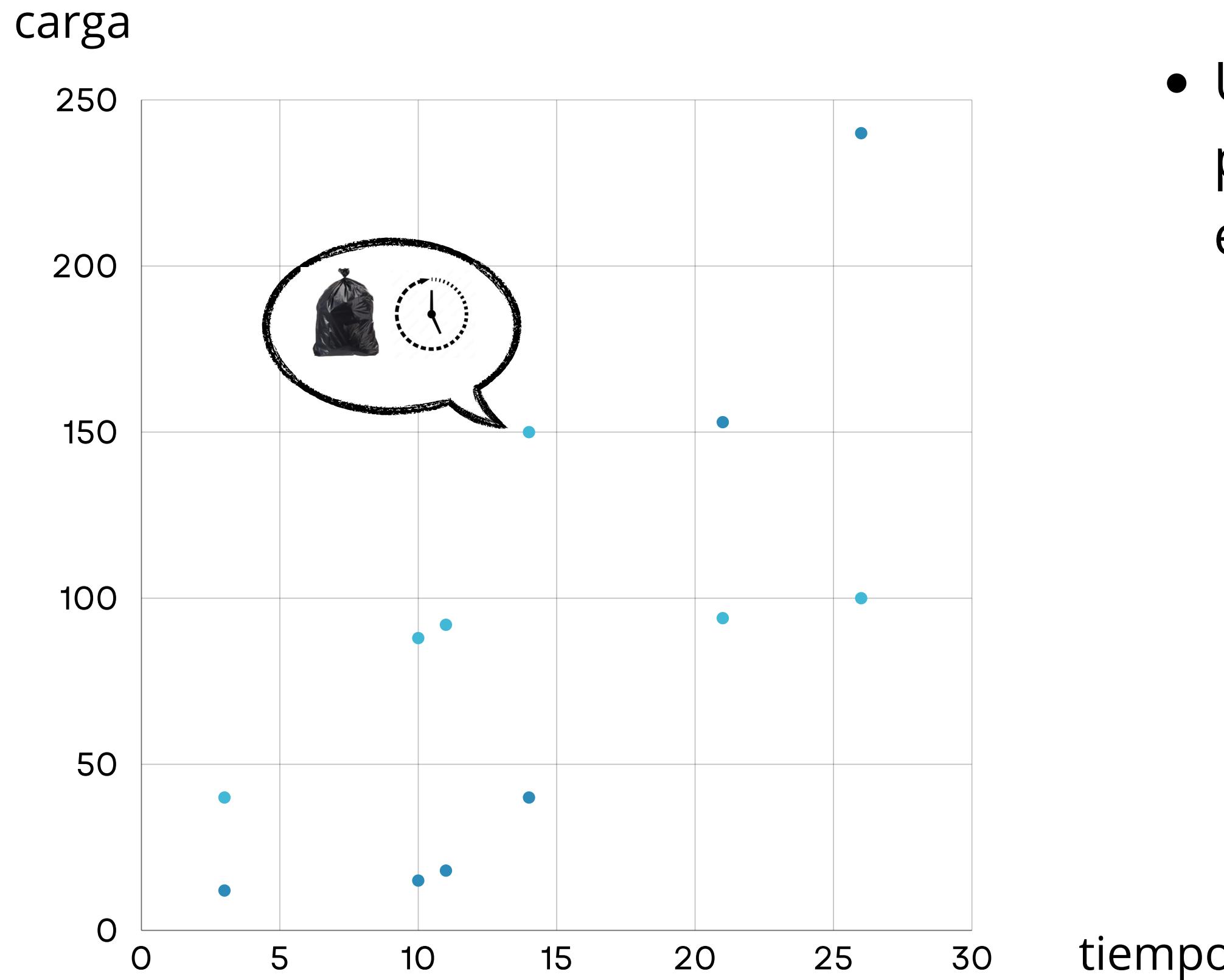
# Objetivos



- Mínimo tiempo de recolección
- Mayor peso cargado para cada camión



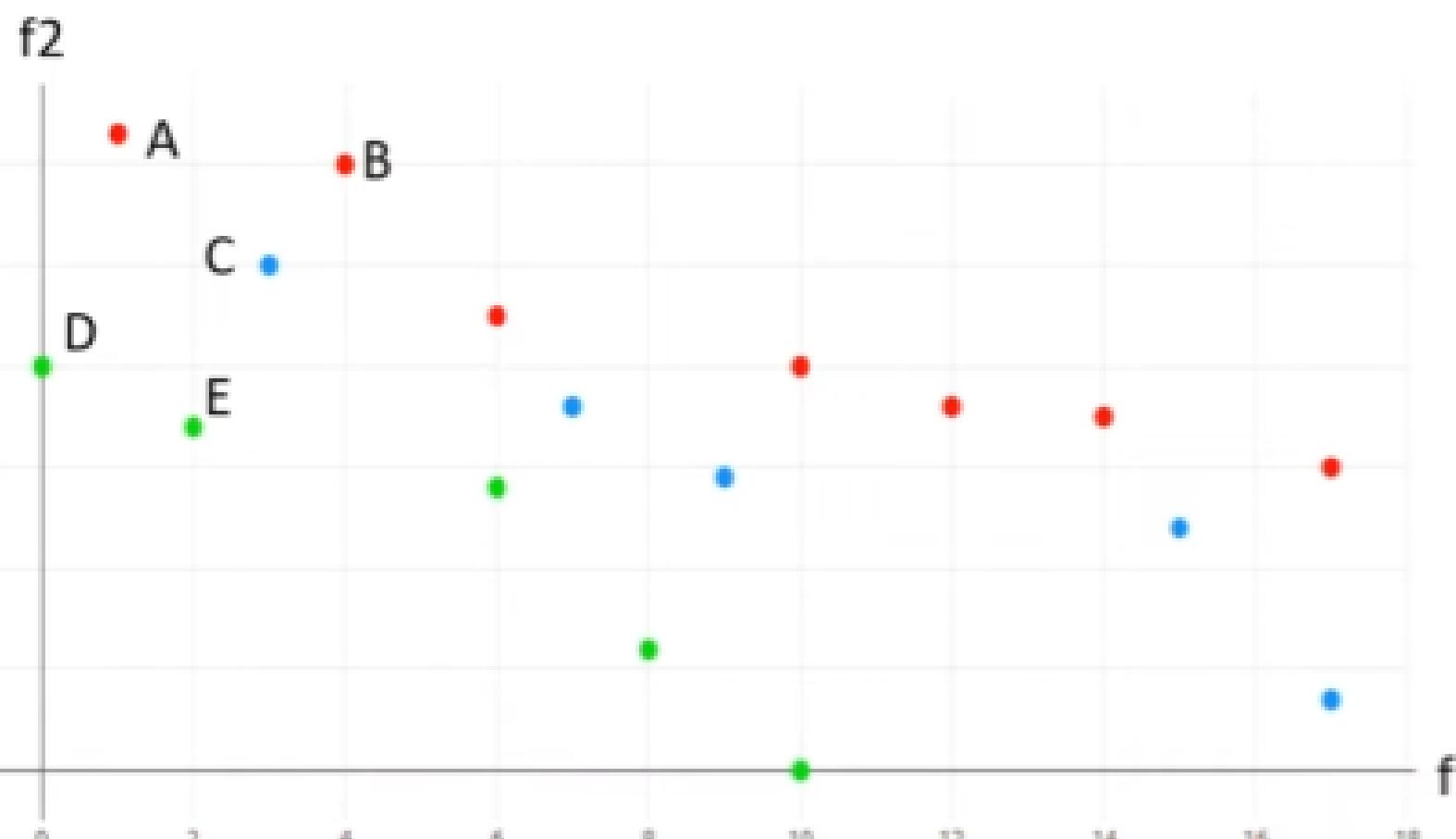
# Rango de soluciones



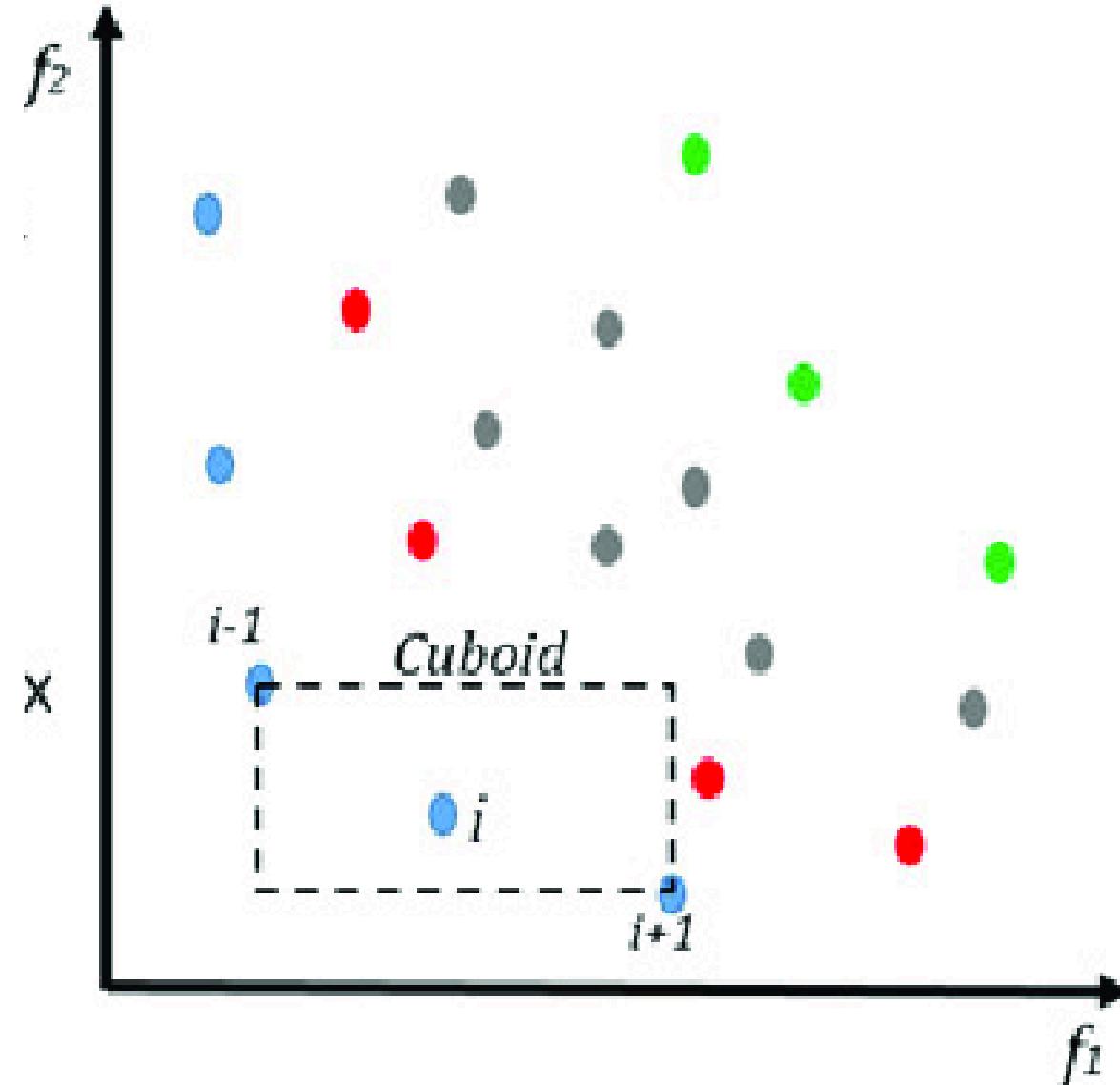
- Una mejora en un objetivo puede implicar el empeoramiento del otro

# Non-dominating sorting

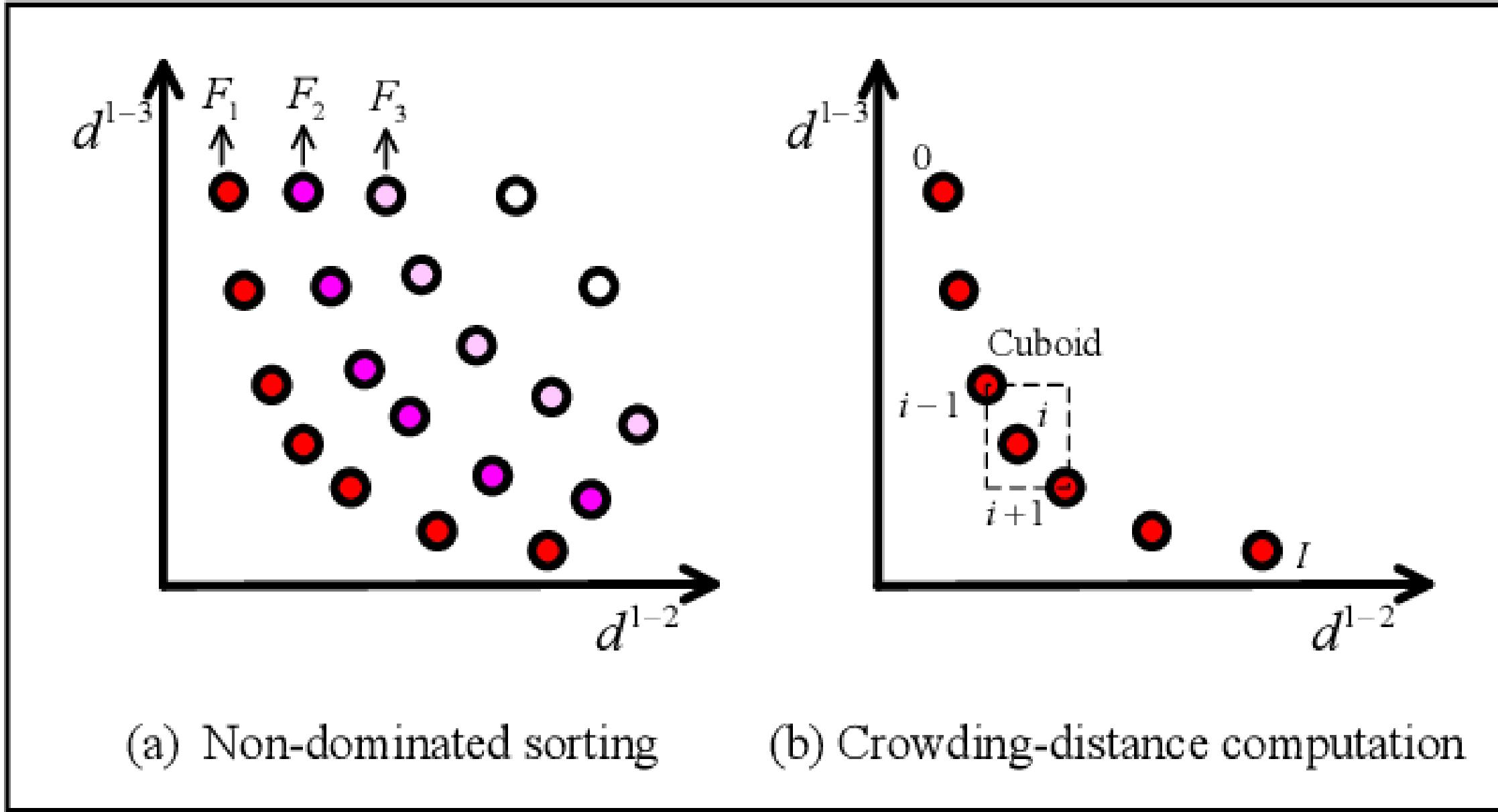
- $A(x_1 | y_1)$  domina a  $B(x_2 | y_2)$  cuando:
  - $(x_1 \leq x_2 \text{ y } y_1 \leq y_2) \text{ y } (x_1 < x_2 \text{ o } y_1 < y_2)$



# Crowd distance sorting



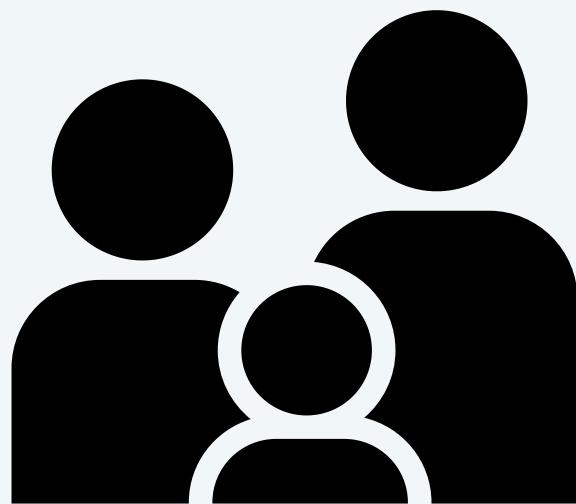
```
1: procedure CROWDINGDISTANCE( $\mathcal{F}$ )
2:    $N = |\mathcal{F}|$ 
3:   for  $i = 1 \dots N$  do
4:      $\mathcal{F}[i]_{\text{dist}} = 0$ 
5:   end for
6:   for  $m = 1, \dots, M$  do
7:     SORT( $\mathcal{F}, m$ )
8:      $\mathcal{F}[1]_{\text{dist}} = \mathcal{F}[N]_{\text{dist}} = \infty$ 
9:     for  $i = 2 \dots N - 1$  do
10:        $\mathcal{F}[i]_{\text{dist}} = \mathcal{F}[i]_{\text{dist}} + \frac{(\mathcal{F}[i+1].m - \mathcal{F}[i-1].m)}{f_m^{\max} - f_m^{\min}}$ 
11:     end for
12:   end for
13: end procedure
```



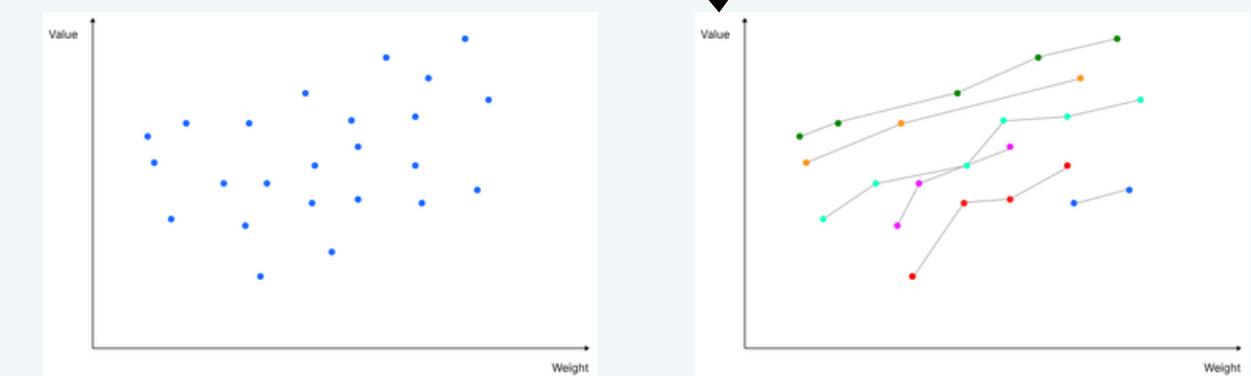
# FUNCIONAMIENTO NSGA-II

**Por generación:**

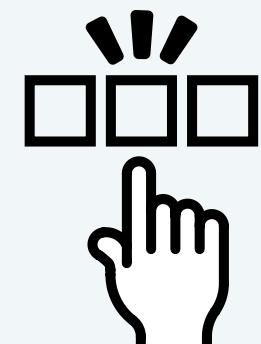
**1. Reproducción de hijos y combinación de la población.**

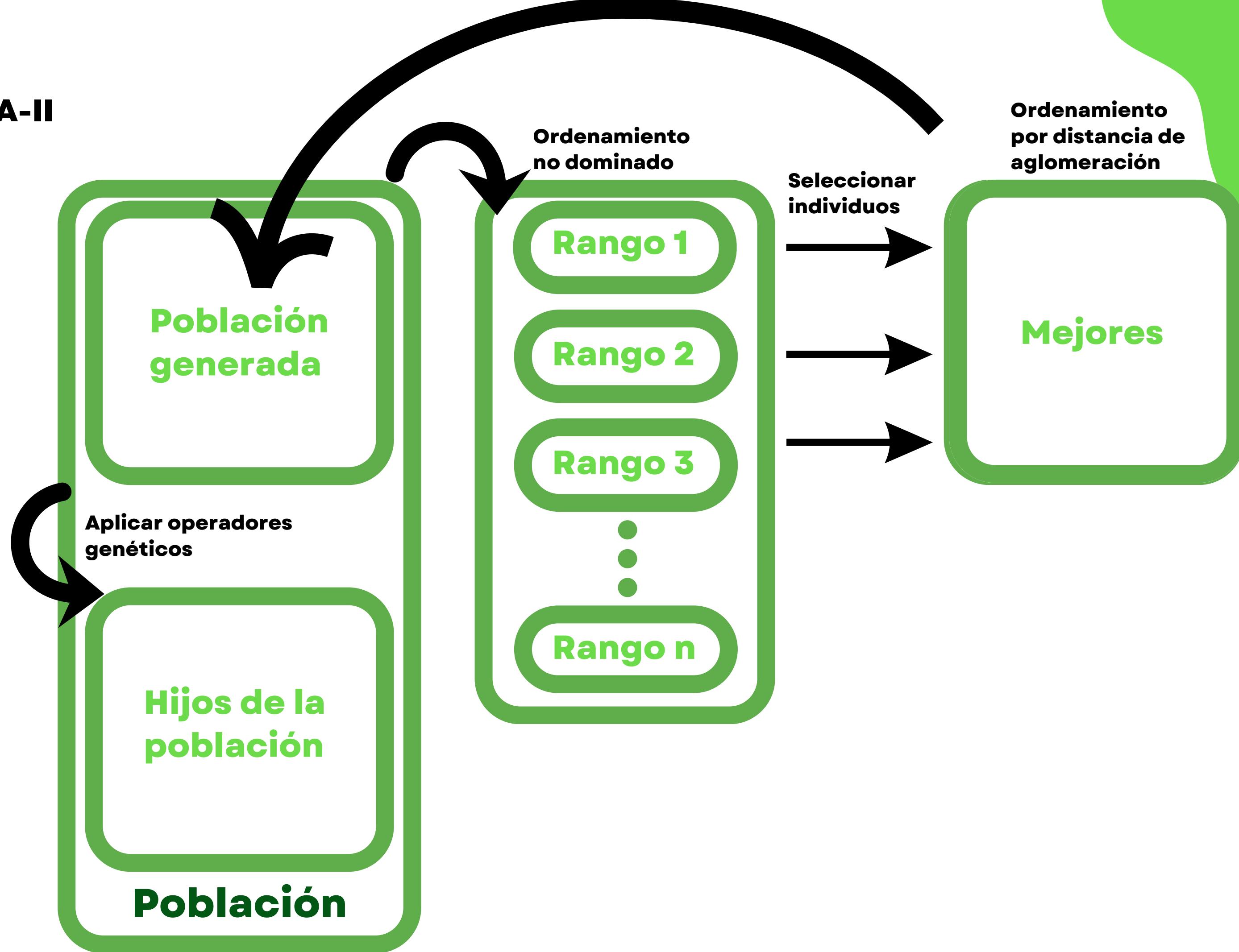


**2. Clasificación y ordenación de la población en función de los resultados obtenidos al aplicarles la función objetivo.**



**3. Selección de los mejores miembros para crear nueva población incluyendo una buena dispersión en las soluciones.**



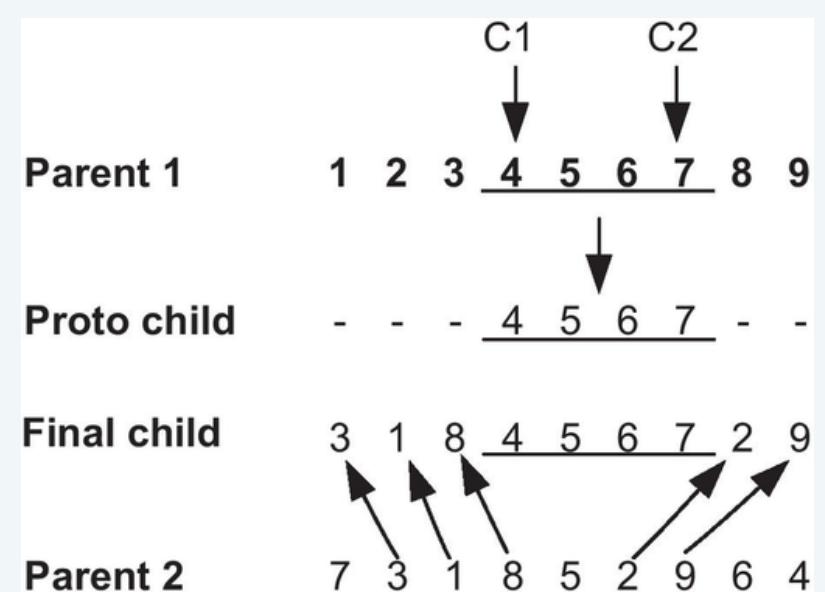


# OPERADORES GENÉTICOS

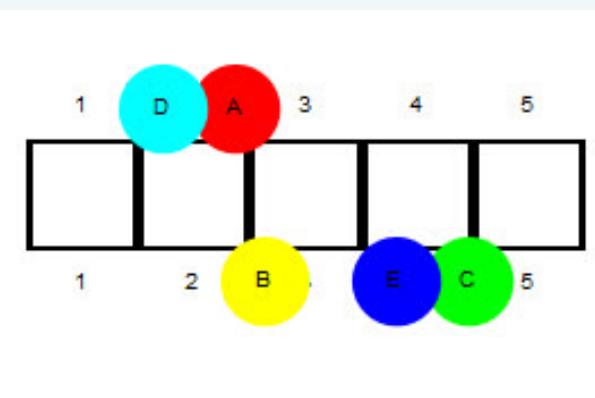
1. Selección de padres por torneo.



2. Cruza de padres para permutaciones.

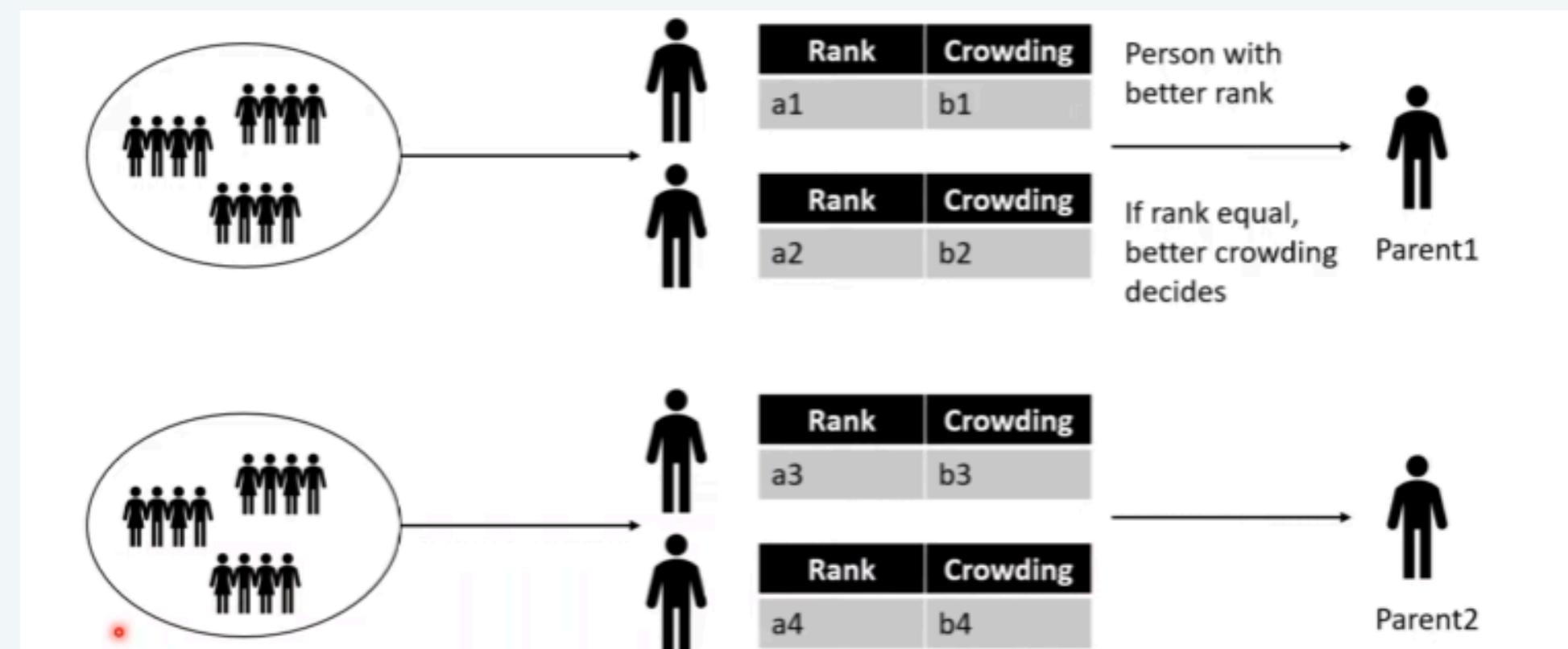


3. Mutación de los hijos.



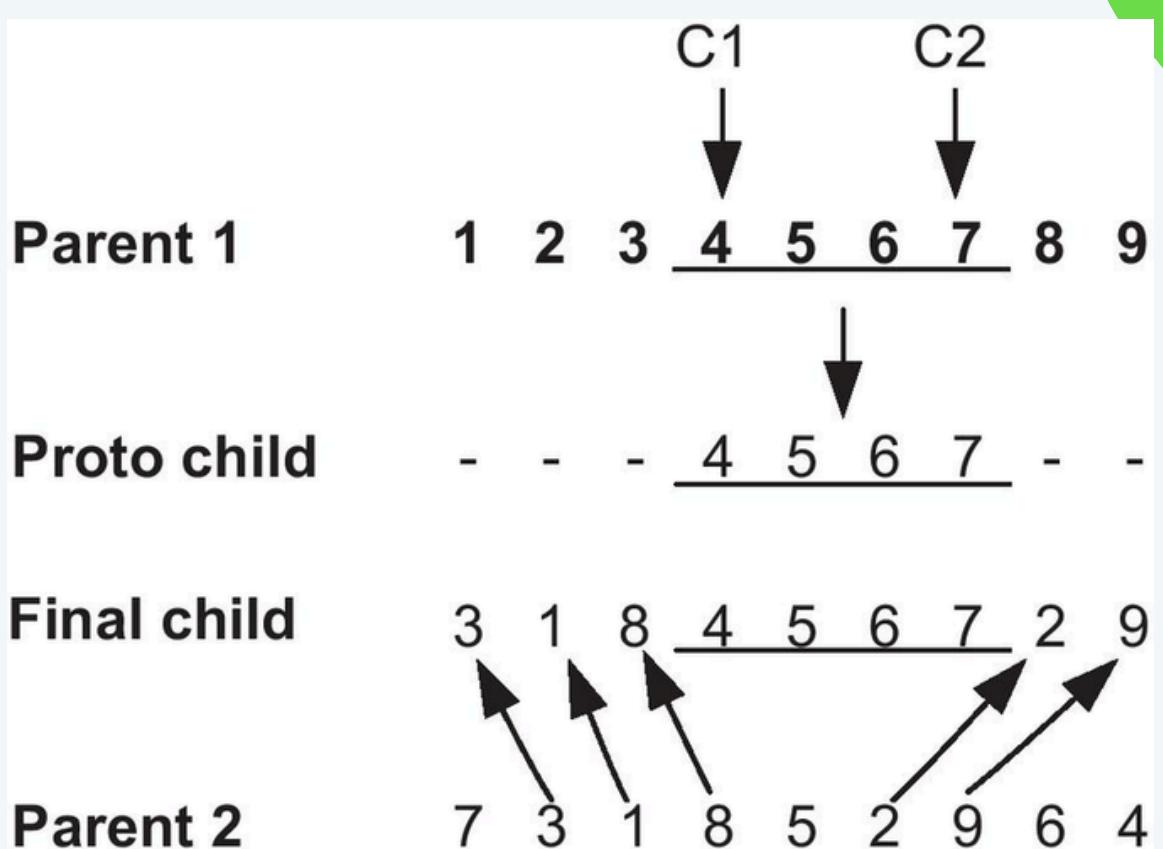
# SELECCIÓN POR TORNEO

- Se seleccionan dos individuos al azar de la población de padres.
- Luego, se comparan según su rango. El individuo con el rango más alto tiene la oportunidad de reproducirse.
- Si los rangos son iguales, se utiliza el operador de distancia de agrupamiento para decidir qué individuo puede reproducirse.



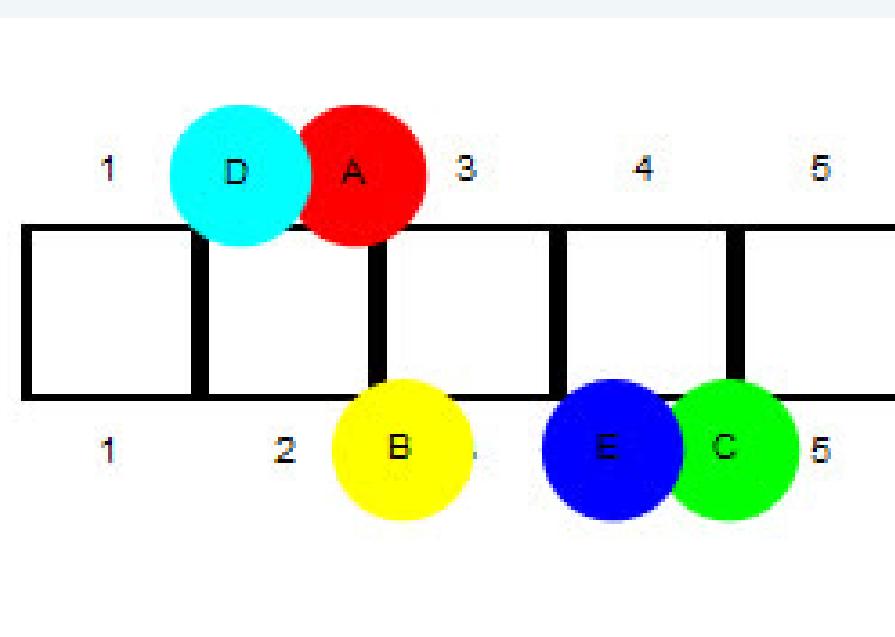
# CRUZA

- Despues de seleccionar los padres, se realiza el cruce.
- El cruce mezcla los genes de los individuos elegidos. Los genes que determinan el tiempo requerido y la cantidad de residuo cargado se combinan para formar el ADN de los hijos.
- La mitad de los genes proviene de un parente y la otra mitad del otro.
- Los hijos resultantes se agregan a la poblacion de descendencia.

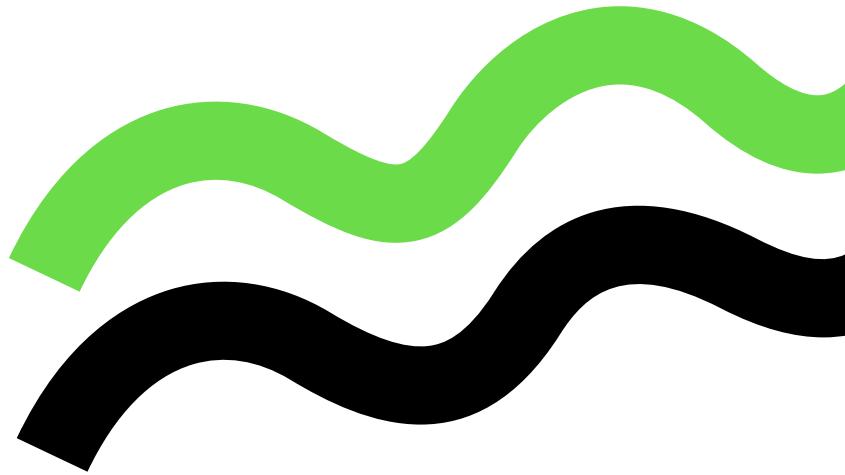


# MUTACIÓN

- Durante el proceso, pueden ocurrir mutaciones aleatorias.
- Por ejemplo, un gen que influye en la cantidad de carga podría aumentar o disminuir en un porcentaje aleatorio.



# Resultados

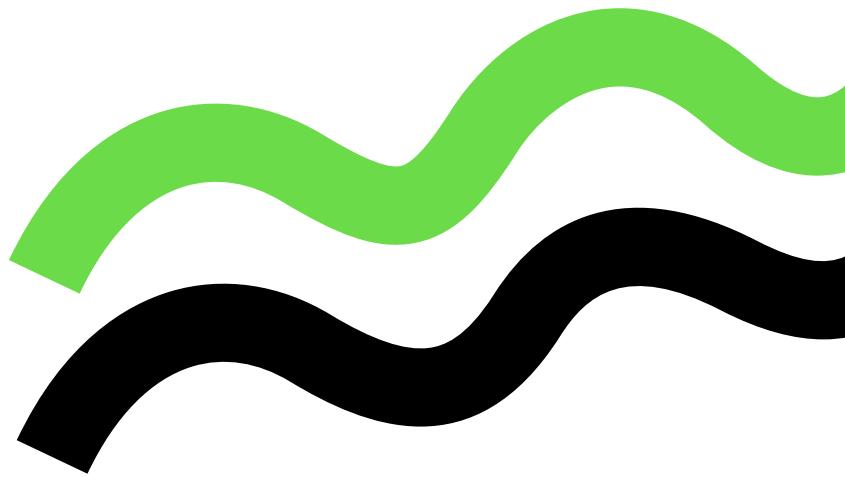


**Evaluar las rutas óptimas obtenidas, discutir la eficacia del enfoque de AG en la gestión de residuos sólidos y proporcionar recomendaciones para la implementación práctica.**

# Consideraciones Éticas y Ambientales

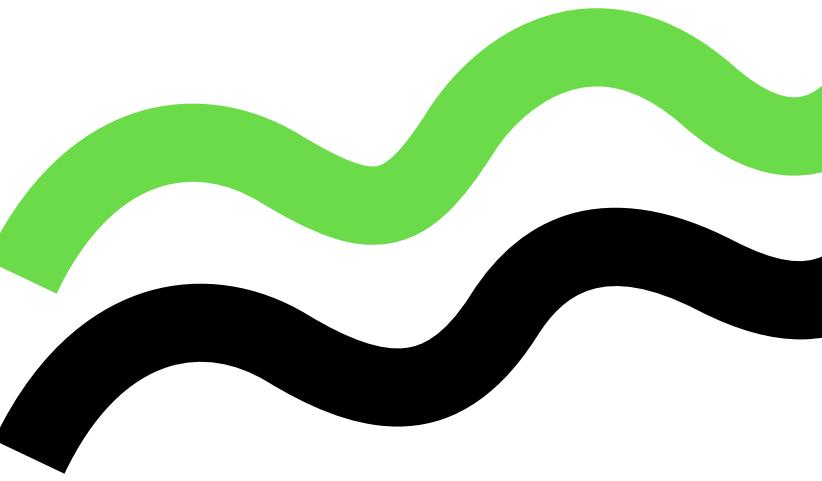
Es importante para nuestro proyecto considerar el impacto ambiental de las rutas optimizadas (emisiones de carbono, congestión del tráfico, etc.).





# Conclusiones

- Los algoritmos genéticos se aplican con éxito para resolver el problema de rutas de recolección de residuos sólidos.
- Estos algoritmos pueden determinar rutas **eficientes**, minimizando el tiempo y eficientando la carga de cada camión.
- La optimización de rutas **reduce** los costos operativos y la contaminación.



# Bibliografía

- A. Alwabli and I. Kostanic, "Dynamic Route Optimization For Waste Collection Using Genetic Algorithm," 2020 International Conference on Computer Science and Its Application in Agriculture
- K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," in IEEE Transactions on Evolutionary Computation,
- J. Pamungkas and C. W. Wijaya, "Optimization The Waste Management Based on Genetic Algorithm Multiobjective," 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2019, pp. 1-4,