

# Conceptos avanzados de algoritmos genéticos

Juan José Domínguez Jiménez

Dpto. Ingeniería Informática  
Lenguajes y Sistemas Informáticos  
Universidad de Cádiz

Abril, 2014

- 1 Algoritmos genéticos. Conceptos básicos
- 2 Operadores genéticos avanzados
- 3 Algoritmos genéticos paralelos
- 4 Optimización multiobjetivo
- 5 Coevolución

## Sección 1 | Algoritmos genéticos. Conceptos básicos

## Características

- Son técnicas de búsquedas probabilísticas basadas en conceptos de la teoría de la evolución y la genética natural.
- Se genera un conjunto de posibles soluciones **población** al problema de forma aleatoria. Cada solución individual recibe el nombre de **individuo**.
- Cada individuo tiene asociado un valor de su calidad (**aptitud**).
- En cada iteración (**generación**) se realizan procesos de **selección** y de **reproducción** (cruce y mutación), que producen una mejora de la aptitud de la población.
- Se puede renovar la población completa (**generacional**) o un pequeño porcentaje de individuos (**estado permanente**).

## Características

- Sencillo y configurable (adaptable)
- Resolver problemas de optimización
- Buscar soluciones subóptimas en problemas complejos en un tiempo adecuado
- Servir como etapa inicial para abordar un problema complejo, aplicando una técnica más clásica y de mayor coste reduciendo la búsqueda al entorno de la mejor solución del algoritmo genético
- Múltiples representaciones para el mismo problema
- Parámetros de configuración afectan al rendimiento

**Algoritmo-Genético** :  $T_P \times p_c \times p_m \times G \longrightarrow x$

$t \leftarrow 0$

$P_t \leftarrow \text{Crear\_Población\_Inicial}(T_P)$

$\text{Evaluar\_Aptitud}(P_t)$

**Mientras** ( $\text{NOT criterio\_Terminacion}()$ )

$t \leftarrow t + 1$

**Mientras** ( $|P_t| < T_P$ )

$\text{Seleccionar\_Operador}()$

$\text{Realizar\_Selección\_Individuos\_Reproducción}(P_{t-1})$

$\text{Realizar\_Operador}(p_c, p_m)$

$\text{Reemplazar\_Individuos}(P_t)$

$\text{Evaluar\_Aptitud}(P_t)$

$\text{ind} \leftarrow \text{Mejor\_Individuo}(P_t)$

**Devolver**  $\text{ind}$

### Se necesita

- Definir al individuo
- Establecer la función de aptitud
- Cómo se generará la población inicial
- Método de selección/reemplazo de individuos
- Operadores de cruce y mutación
- Cómo se realiza la generación
- Criterio de parada

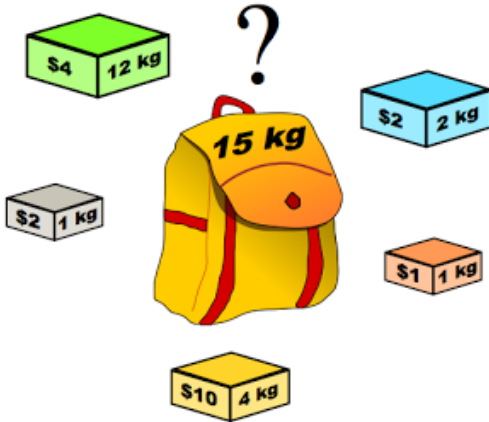
### Se necesita

- **Definir al individuo**
- Establecer la función de aptitud
- Cómo se generará la población inicial
- Método de selección/reemplazo de individuos
- Operadores de cruce y mutación
- Cómo se realiza la generación
- Criterio de parada



- Cada individuo codifica una solución del problema.
- El individuo es configurable y adaptable al problema que se quiera resolver.
- Los cromosomas que representan al individuo pueden ser:
  - Cadena de bits (Problema SAT)
  - Números reales (Optimización de una función)
  - Permutación de elementos (Problema de las N-reinas, problema del viajante)
  - Elementos de programas (programación genética)
  - Cualquier otra estructura.
- La elección del tipo de individuo decidirá los operadores genéticos a emplear.
- Un problema puede ser resuelto con distintas representaciones.
- No existe la representación óptima.

Maximizar el valor de los ítemes que contiene y siempre que no se supere el peso máximo que puede soportar.



## Representación binaria

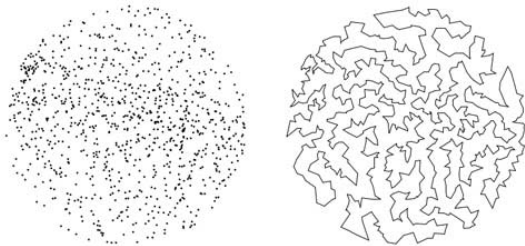
- Se representa mediante una cadena binaria de longitud igual al número de elementos, donde un 0 indica que el elemento está presente y un 1 que no está presente.

1 Kg	2 Kg	5 Kg	10 Kg	20 Kg	40 Kg
0	0	1	0	0	1
0	0	1	1	0	1

## Representación como permutación de elementos

- Es una permutación de elementos a meter, de manera que los elementos se meten en el orden que aparece hasta completar la capacidad de la mochila.  
6 3 4 5 1 2 → Meter elemento de 40 Kg, luego el de 5 Kg, y así hasta completar la capacidad de la mochila

Encontrar la ruta óptima para visitar varias ciudades, recorriendo la menor distancia posible, sin repetir el paso por ninguna de ellas.



## Representación como permutación de elementos

- Permutación de las N-1 ciudades sin repetición, que indica el orden en que se recorre. Para 10 ciudades  
B C D E F G H I J → El orden es de la ciudad A a la B, de la B a la C, .. de la J a la A.

## Representación binaria (matricial)

- Se representa mediante una matriz de N-1 x N-1, donde cada fila sólo hay un 1 indicando la ciudad que se visita, y en cada columna sólo puede haber un 1.  
La siguiente matriz, indica que de la ciudad A, se visita la ciudad D, luego la B y finalmente la C.

$$\begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

### Se necesita

- Definir al individuo
- **Establecer la función de aptitud**
- Cómo se generará la población inicial
- Método de selección/reemplazo de individuos
- Operadores de cruce y mutación
- Cómo se realiza la generación
- Criterio de parada

## Maximizar

- El algoritmo genético siempre maximiza la función de aptitud de los individuos
- Si estamos resolviendo un problema de minimización, requiere realizar la siguiente transformación
$$\min f(x) = \max -f(x)$$

## Restricciones

- Cuando se resuelve un problema de optimización con restricciones, el incumplimiento de estas se puede realizar la penalización en la función de aptitud.
- Esto permite tener individuos no válidos en la población para poder explorar y generar nuevos individuos, pero que en la solución final deben ser descartados

### Se necesita

- Definir al individuo
- Establecer la función de aptitud
- **Cómo se generará la población inicial**
- Método de selección/reemplazo de individuos
- Operadores de cruce y mutación
- Cómo se realiza la generación
- Criterio de parada



- Aleatoriamente
- Siembra: Se escogen determinados individuos que por sus características permiten obtener una población inicial con suficiente aptitud.
- Aplicando una heurística: Tras la generación de los individuos, a cada uno se puede aplicar una heurística adicional que permita mejorar el individuo.

### Se necesita

- Definir al individuo
- Establecer la función de aptitud
- Cómo se generará la población inicial
- **Método de selección/reemplazo de individuos**
- Operadores de cruce y mutación
- Cómo se realiza la generación
- Criterio de parada

## Selección

- Selecciona individuos de una población para la reproducción.
- Esta selección puede ser proporcional o no a la aptitud del individuo.
- Normalmente, se diseña generalmente para dar mayores oportunidades de reproducción a los individuos más aptos dentro de la población.
- Algunos individuos serán seleccionados más de una vez: los mejores individuos obtendrán más copias, y los peores morirán.

## Tipos de selección

- Ruleta** Cada individuo tiene una probabilidad de salir de acuerdo a su aptitud.
- Ranking** Se ordenan los individuos por su aptitud, y se aplica la ruleta empezando siempre por el de la primera posición. Los de ranking superior serán los seleccionados.
- Torneo** Se seleccionan  $k$  individuos y se escoge el mejor. Funciona mejor cuando las aptitudes de todos los individuos son muy similares.

### Se necesita

- Definir al individuo
- Establecer la función de aptitud
- Cómo se generará la población inicial
- Método de selección/reemplazo de individuos
- **Operadores de cruce y mutación**
- Cómo se realiza la generación
- Criterio de parada

## Cruce

- Este operador genera dos individuos nuevos, denominados **hijos**, a partir de dos individuos seleccionados previamente, denominados **padres**. Los hijos heredan parte de la información almacenada en cada uno de los dos padres.
- El operador dependerá de la codificación de los individuos.

## Tipos de cruces

- Representación binaria: cruce de 1-punto, 2-puntos, uniforme
- Representación secuencia: PMX, OX, cíclico, basado en orden, etc.
- Representación real: cruces aritméticos como el cruce lineal convexo:

$$X' = X \cdot \alpha + Y \cdot (1 - \alpha)$$

$$Y' = X \cdot (1 - \alpha) + Y \cdot \alpha$$

$$\alpha \in [0, 1]$$

## Mutación

- Altera la información almacenada en un individuo, favorece la diversidad.
- Depende de la codificación del individuo

## Tipos de mutaciones

- Representación binaria: mutación binaria, intercambio recíproco, inversión, inserción
- Representación secuencia: intercambio recíproco, inversión, inserción
- Representación real: mutación gaussiana

$$X' = X + \alpha \cdot \beta$$

$$\alpha \in [-1, 1]$$

$\beta$  es el tamaño de la mutación máxima

### Se necesita

- Definir al individuo
- Establecer la función de aptitud
- Cómo se generará la población inicial
- Método de selección/reemplazo de individuos
- Operadores de cruce y mutación
- **Cómo se realiza la generación**
- Criterio de parada



## Esquema tradicional

- Se repite hasta completar la población la siguiente secuencia:  
Seleccionar padres → Cruzar → Mutar hijos → Reemplazar

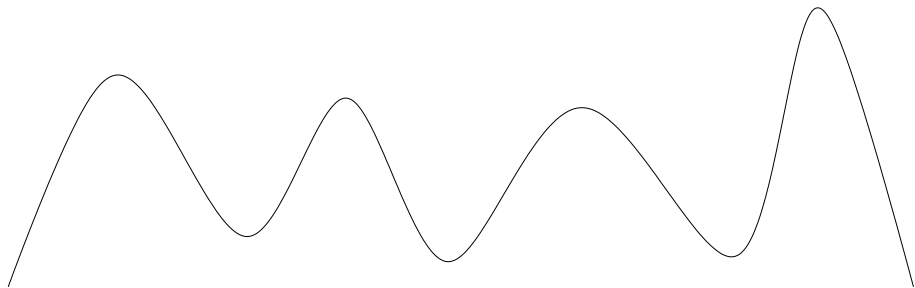
## Esquemas alternativos

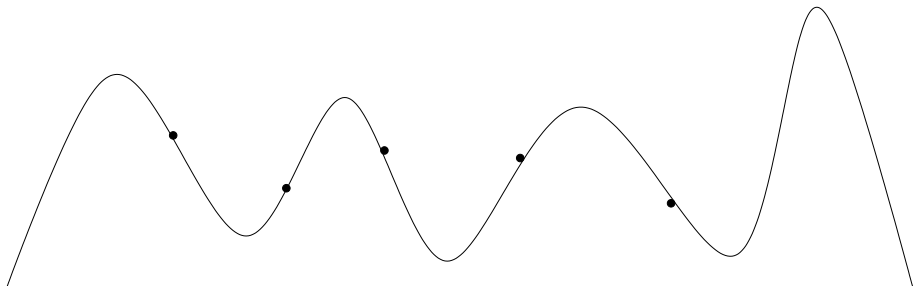
- Generar un porcentaje de manera aleatoria y el resto mediante cruces-mutaciones
- Realizar operaciones independientes o bien un cruce o bien una mutación
- Aplicar alguna optimización local a los individuos que se obtienen al mutar

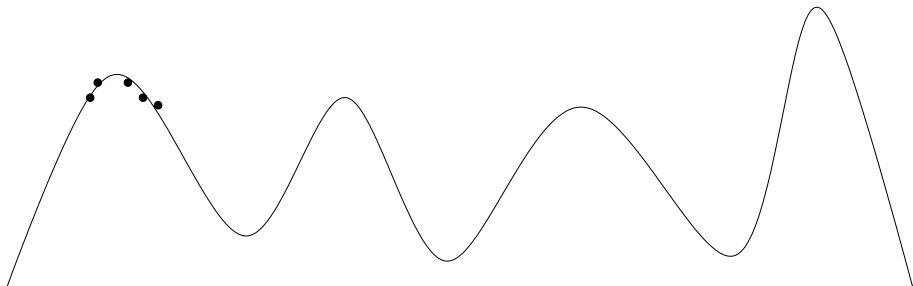
## Sección 2 | Operadores genéticos avanzados

- Nichos
- Memoria

- Un problema que pueden presentar los AG es la convergencia prematura provocada por la pérdida de diversidad genética provocada por la supervivencia y la aparición de los **superindividuos**

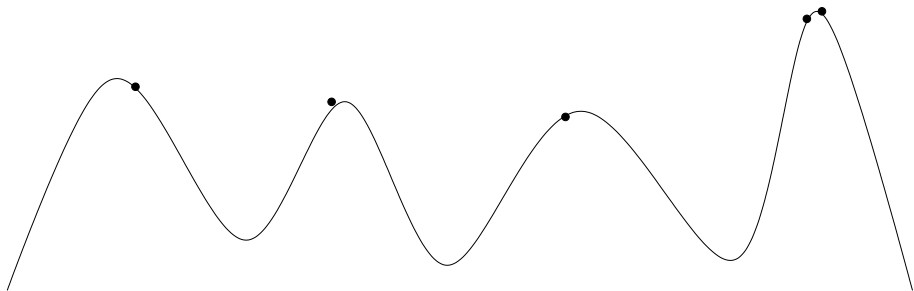






- Un problema que pueden presentar los AG es la convergencia prematura provocada por la pérdida de diversidad genética provocada por la supervivencia.
- Se introduce una técnica de nichos para mantener la diversidad en la población de manera que se pueda explorar más espacio de soluciones y encontrar así los distintos óptimos.





- Un problema que pueden presentar los AG es la convergencia prematura provocada por la pérdida de diversidad genética provocada por la supervivencia.
- Se introduce una técnica de nichos para mantener la diversidad en la población de manera que se pueda explorar más espacio de soluciones y encontrar así los distintos óptimos.
- Se requiere una función que calcule la distancia entre dos individuos, para determinar el parecido entre dos individuos.

## Técnicas

- *Fitness sharing method*
- *Crowding factor*
- *Clearing method*

- Se establece el tamaño del nicho, que define el tamaño de cada región.
- Un individuo será penalizado en función del número de individuos que se encuentren en su misma región. Para ello se calcula la distancia del individuo con el resto de la población, y si es menor que el tamaño del nicho debe ser penalizado.
- Esta estrategia penalizará a los individuos en zonas más densas, y aquellos que estén solos en una región mantendrán su aptitud.
- La función de aptitud se transforma en la denominada **sharing function**, cuyo valor es igual o menor al de la función de aptitud:  
$$f'(x) = f(x) - \text{factor} \cdot y$$

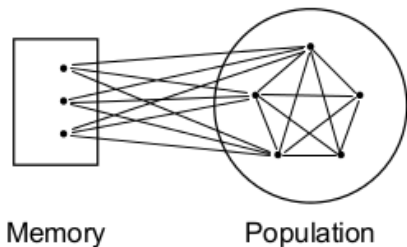
donde  $y$  es el número de individuos de la población que se encuentran en la misma región que  $x$
- Permite mantener a individuos en óptimos locales

- El **crowding factor** es un parámetro de entrada que permite dividir la población en subpoblaciones del tamaño que especifique.
- Se aplica una técnica de estado permanente, donde en cada generación, la población sólo sustituye a uno o dos individuos en cada subpoblación.
- Los hijos resultantes de las operaciones de cruce y mutación sustituyen al individuo que más se le parece dentro de una subpoblación.

- Está basada en la limitación de recursos de cada región.
- Consiste en favorecer a los mejores individuos de cada nicho.
- Para cada región, el mejor individuo mantendrá su aptitud, mientras que el resto de individuos su aptitud será 0.

- Nichos
- **Memoria**

- Permite almacenar los mejores individuos de cada generación en una segunda población
- Los individuos de la población principal pueden ser evaluados teniendo en cuenta a los existentes en el salón de la fama.
- Permite asegurar el progreso.



## Sección 3 | Algoritmos genéticos paralelos



- La idea que esconde es dividir el trabajo en varias subtareas, a desarrollar simultáneamente en diversos procesadores.

## Tipos

- Maestro-esclavo
- Modelo isla o de múltiples poblaciones

## AG Maestro-Esclavo

- Tiene una única población
- El procesador maestro ejecuta el AG (selección, cruce y mutación)
- Los procesadores esclavos ejecutan la evaluación de la aptitud de cada individuo.
- La situación ideal es tener un procesador por cada individuo

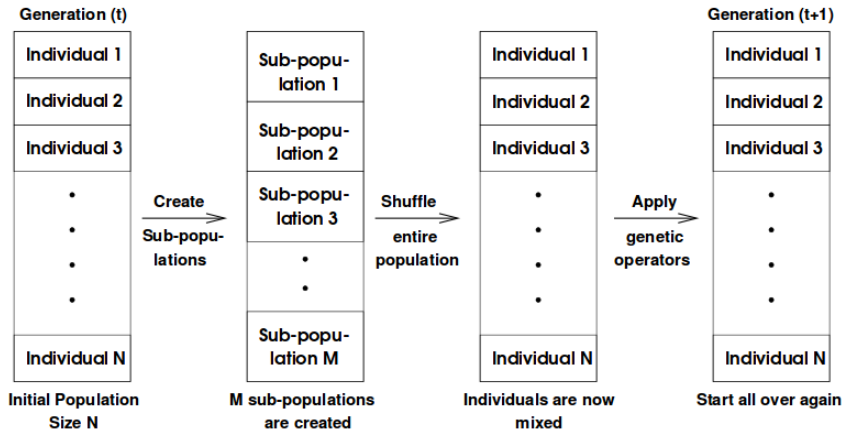
## AG modelo isla o de múltiples poblaciones

- El AG está formado por varias subpoblaciones, ejecutándose cada una en un procesador diferente.
- Cada cierto tiempo se produce un intercambio de individuos entre las distintas poblaciones.
- Es necesario definir la frecuencia de migración, el número de emigrantes, el destino de los emigrantes, así como el esquema de selección de emigrantes.
- Han sido los más empleados, principalmente en funciones multiobjetivos.

## Sección 4 | Optimización multiobjetivo

- La optimización multiobjetivo se caracteriza por disponer de varias funciones objetivos a optimizar simultáneamente.
- La técnica tradicional consiste en convertir el problema multiobjetivo en un problema mono-objetivo realizando una suma ponderada de los objetivos:

$$\min \sum_{i=1}^k w_i f_i(k)$$



## Sección 5 | Coevolución

### Similitudes a los AG tradicionales

- Los individuos codifican posibles soluciones
- Se modifican mediante operadores genéticos
- La selección se realiza teniendo en cuenta la aptitud

### Diferencias a los AG tradicionales

- La evaluación requiere a múltiples individuos
- Los individuos con los que interactúan pueden ser de la misma o de otra población
- Implica cooperación y competición en la interacción.



## AG con una única población

- Es un AG tradicional
- Los individuos además de representar una solución, sirven como referencias o pruebas de los otros individuos
- Cada individuo se evalúa interaccionando con otros de la población, produciéndose una competencia por los recursos de supervivencia en la población.
- **Fitness sharing method** es un ejemplo de coevolución de una población

## AG con múltiples poblaciones

- Existen múltiples poblaciones, uno con su propio AG
- Los individuos se evalúan interactuando con otros de su población y de otras poblaciones
- Los individuos de diferentes poblaciones no compiten por sobrevivir.
- El salón de la fama es un ejemplo de coevolución de dos poblaciones.

- 1 Realizar un AG que optimice la función  $\sin(x)^2 - \sin(x)$  en el intervalo 0 a 40.
  - Cruce lineal convexo y mutación gaussiana
  - Esquema generacional clásico
- 2 Igual al anterior pero con nichos. Compara las poblaciones finales para un mismo juego de parámetros de configuración.