

# Metaheurísticas

CONFERENCIA #4. ALGORITMOS GENÉTICOS

# Sumario

- ▶ Algoritmos bio-inspirados
- ▶ Algoritmos Genéticos

# Tipos de problemas

## Continuos

- Dominio de todas las variables en  $\Re$

## Mixtos

- Algunas variables nominales

## Combinatorios

- Ordenamiento, asignación, scheduling...

# Métodos heurísticos de trayectoria simple

- ▶ Ascenso de colinas
- ▶ Recocido Simulado

Ventajas

Desventajas

Fáciles de implementar

Muy rápidos

Pobre explotación de la búsqueda

Quedan atrapados en óptimos locales

Poco configurables



# Metaheurísticas bio-inspiradas

# Metaheurísticas bio-inspiradas

- ▶ John Holland, 1960s
- ▶ “Los organismos vivientes son consumados solucionadores de problemas”
  - ▶ Adaptation in natural and artificial systems, 1975



# Modelos de computación bioinspirados

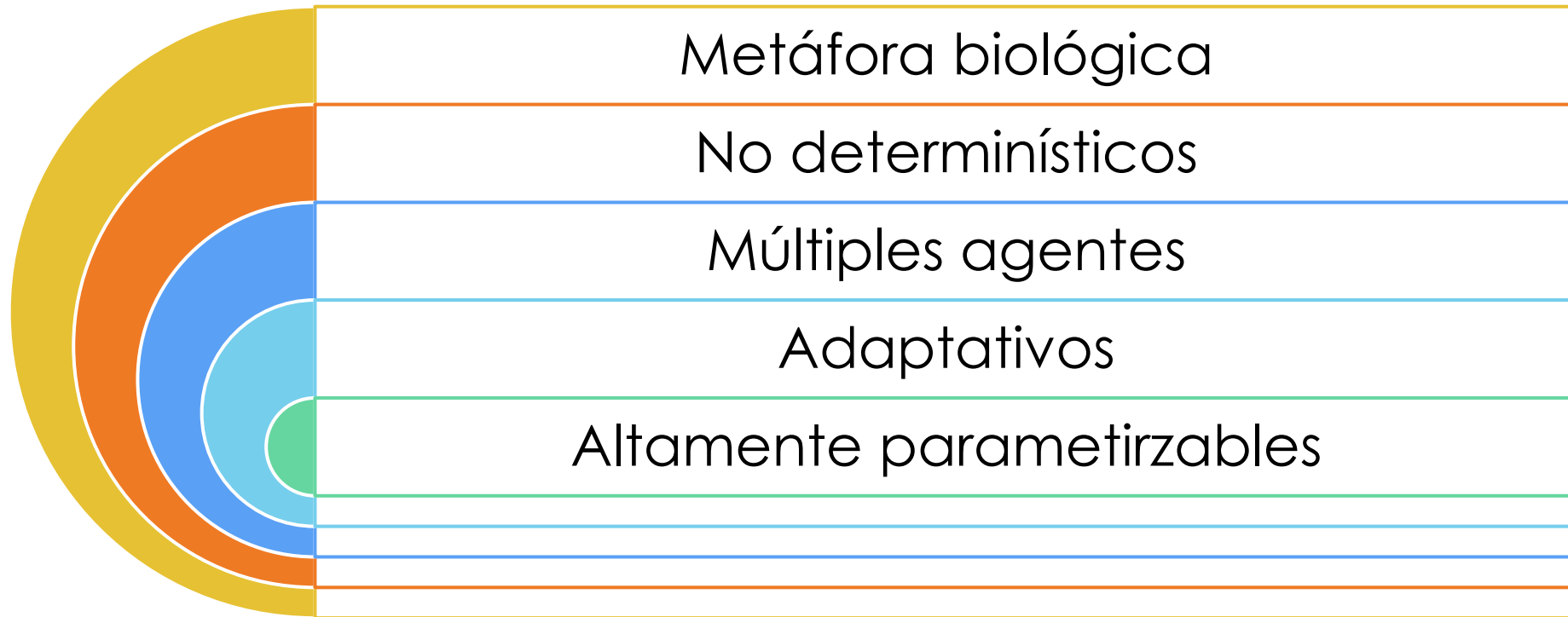
- ▶ La Computación Bioinspirada (bioinspired Algorithms/Natural Computing/biologically inspired computing)
- ▶ Se basa en emplear analogías con sistemas naturales o sociales para la resolución de problemas.

# Modelos de computación bioinspirados

- ▶ Los algoritmos bioinspirados simulan el comportamiento de sistemas naturales para el diseño de métodos heurísticos no determinísticos de “búsqueda” / “aprendizaje” / “comportamiento”, ...
- ▶ En la actualidad los “Algoritmos Bioinspirados” son uno de los campos más prometedores de investigación en el diseño de algoritmos.



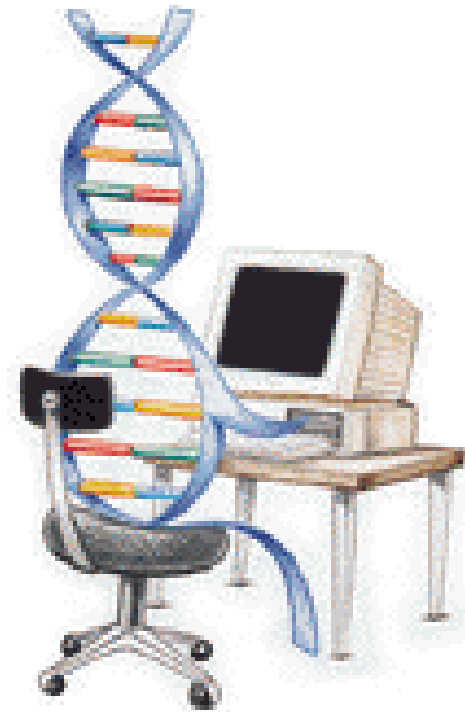
# Características



# Algoritmos Genéticos

# ¿Qué es un Algoritmo Genético?

- ▶ Los Algoritmos Genéticos
  - ▶ son algoritmos de optimización,
  - ▶ búsqueda
  - ▶ y aprendizaje
  - ▶ inspirados en los procesos de
    - ▶ Evolución Natural
    - ▶ Evolución Genética



# Evolución Natural

- ▶ En la naturaleza, los procesos evolutivos ocurren cuando se satisfacen las siguientes condiciones:
  - ▶ Una entidad o individuo tiene la habilidad de reproducirse
  - ▶ Hay una población de tales individuos que son capaces de reproducirse
  - ▶ Existe alguna variedad, diferencia, entre los individuos que se reproducen
  - ▶ Algunas diferencias en la habilidad para sobrevivir en el entorno están asociadas con esa variedad

# Algoritmos Genéticos

- ▶ Los mecanismos que conducen esta evolución no son totalmente conocidos, pero sí algunas de sus características, que son ampliamente aceptadas:
  - ▶ La evolución es un proceso que opera sobre los cromosomas más que sobre las estructuras de la vida que están codificadas en ellos

# Evolución artificial

- ▶ Está compuesta por modelos de evolución basados en poblaciones cuyos elementos representan soluciones a problemas
- ▶ La simulación de este proceso en un ordenador resulta ser una técnica de optimización probabilística, que con frecuencia mejora a otros métodos clásicos en problemas difíciles

# Inspiración

- ▶ La Evolución crea individuos de más calidad
  - ▶ Población de individuos
    - ▶ Cada individuo tiene un código genético
  - ▶ Individuos exitosos (mayor calidad) tienden a tener más descendencia
    - ▶ Ciertos códigos resultan en mayor calidad
  - ▶ Los hijos combinan los códigos de sus padres
    - ▶ Cruce
    - ▶ Mutación

# Evolución Artificial

- ▶ Existen cuatro paradigmas básicos:
  - ▶ Algoritmos Genéticos que utilizan operadores genéticos sobre cromosomas. 1975, Michigan University
  - ▶ Estrategias de Evolución que enfatizan los cambios de comportamiento al nivel de los individuos. 1964, Technische Universität Berlin
  - ▶ Programación Evolutiva que enfatizan los cambios de comportamiento al nivel de las especies. 1960-1966, Florida
  - ▶ Programación Genética que evoluciona expresiones representadas como árboles. 1989, Stanford University



# Desarrollo histórico

- ▶ Desde finales de los años 50 y principios de los 60 se desarrollaron trabajos por algunos biólogos para simular sistemas genéticos en una computadora, dentro de estos trabajos es destacable el de Fraser, entre los años 60 y 62, donde se llegó a algo bastante parecido a los AG, trabajando con cadenas y fenotipos.
- ▶ A pesar de estos trabajos, se reconoce al profesor John Holland de la Universidad de Michigan como el creador de los AG con su trabajo sobre teoría de los sistemas adaptativos, en el año 62. Es Holland el primero que adaptó la idea de la genética a sistemas artificiales.
- ▶ El nombre de AG lo uso por primera vez Bagley en el año 67, y el mismo es el autor de un trabajo importante sobre la utilización de AG en juegos.

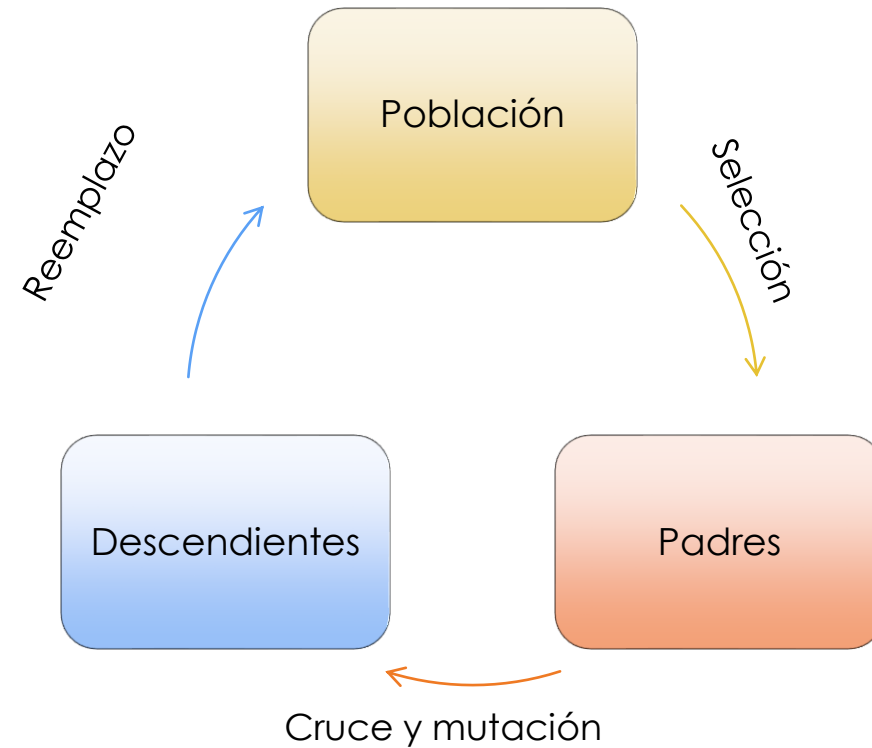
# Algoritmos Genéticos

- ▶ Un AG puede ser visto como una estructura de control que organiza o dirige un conjunto de transformaciones y operaciones diseñadas para simular estos procesos de evolución.
- ▶ Los algoritmos genéticos se pueden considerar métodos de búsqueda aleatoriamente guiados.

# Algoritmos Genéticos

- ▶ Optimización a través de la evolución artificial
  - ▶ Definir calidad de acuerdo a la función a ser optimizada
  - ▶ Codificar posibles soluciones como códigos genéticos individuales
  - ▶ Evolucionar mejores soluciones a través de la evolución simulada

# Algoritmos Genéticos



# Algoritmos Genéticos

- ▶ Representación de individuos
- ▶ Inicialización de la población
- ▶ Función de evaluación de un individuo
- ▶ Estrategia de selección de los individuos padres
- ▶ Operador de cruce
- ▶ Operador de mutación
- ▶ Estrategia de reemplazo de los individuos
- ▶ Condición de parada

Depende del problema

Componentes del algoritmo

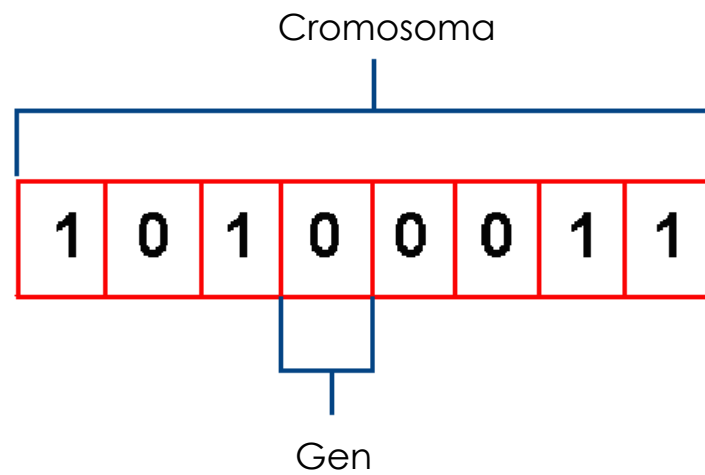
# Representación de las soluciones

- ▶ Debemos disponer de un mecanismo para codificar un individuo como un genotipo
- ▶ Existen muchas maneras de hacer esto y se debe elegir la más relevante para el problema en cuestión
- ▶ Una vez elegida una representación, tenemos que tener en mente cómo se evaluarán los genotipos (codificación) y qué operadores genéticos habrá que utilizar

# Cromosomas

- ▶ Los cromosomas se representan generalmente como cadenas, donde cada componente representa un gen, es decir, una de las variables del problema a resolver. Ejemplos:
- ▶ Mínimo de la función  $f(x, y) = x^2 + y^2$ 
  - ▶ Cromosoma de 2 dimensiones, con genes reales
- ▶ Problema del viajero vendedor, con  $n$  ciudades
  - ▶ Cromosoma de  $n$  dimensiones, con genes enteros (el número de la ciudad)
- ▶ Selección de atributos, con  $m$  atributos
  - ▶ Cromosoma de  $m$  dimensiones, con genes binarios

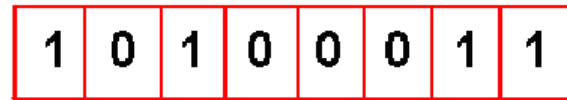
# Representación Binaria





# Genotipo y Fenotipo

8 bits Genotipo



## Fenotipo

- Entero
- Número real
- Secuencia
- ...
- ¿Cualquier otra?

# Genotipo y Fenotipo

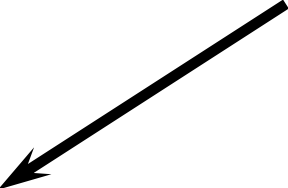
- El fenotipo puede ser uno o varios números enteros

Genotipo:

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

Fenotipo:

= 163


$$1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 =$$
$$128 + 32 + 2 + 1 = 163$$

# Representación real o entera

- ▶ Una forma natural de codificar una solución es utilizando valores reales como genes
- ▶ Muchas aplicaciones tienen esta forma natural de codificación

# Representación real

- Los individuos se representan como vectores de valores reales:

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, x_i \in R$$

- La función de evaluación asocia a un vector un valor real de evaluación:

$$f : R^n \rightarrow R$$

# Representación de orden

- ▶ Los individuos se representan como permutaciones
- ▶ Se utilizan para problemas de secuenciación
- ▶ Ejemplo famoso: Viajante de Comercio, donde cada ciudad tiene asignado un único número entre 1 y  $n$
- ▶ Necesita operadores especiales para garantizar que el resultado de aplicar un operador sigue siendo una permutación

# Población. Inicialización

- ▶ La población es un conjunto de soluciones candidatas. Cada una de ellas es un individuo, que está representado por un cromosoma.
- ▶ La inicialización se puede realizar:
- ▶ Uniforme sobre el espacio de búsqueda ... (si es posible)
  - ▶ Cadena binaria: 0 o 1 con probabilidad 0.5
  - ▶ Representación real: uniforme sobre un intervalo dado (para valores acotados)
- ▶ Elegir la población a partir de los resultados de una heurística previa

# Función de evaluación

- ▶ Es una medida que permite determinar cuán buena es una solución (individuo)
- ▶ Tiene como entrada el cromosoma que describe al individuo, y generalmente devuelve un número real.

# Función de evaluación

- ▶ Este es el paso más costoso para una aplicación real
- ▶ Puede ser una subrutina, un simulador, o cualquier proceso externo
- ▶ Se pueden utilizar funciones aproximadas para reducir el costo de evaluación
- ▶ Cuando hay restricciones, éstas se pueden introducir en la función de evaluación como penalización del costo
- ▶ Con múltiples objetivos se busca una solución de compromiso



# Parámetros

- ▶ Tamaño de la población: indica la cantidad de soluciones que forman la población.
- ▶ Numero de generaciones: cantidad de iteraciones que realizara el algoritmo hasta alcanzar una solución satisfactoria.
- ▶ Probabilidad de cruce: Valor entre 0 y 1 que permite determinar si procede o no el cruce. Como la frecuencia de cruce es alta este debe ser un valor alto ( $p_c=0.7$ ,  $p_c=0.91$ , etc.).
- ▶ Probabilidad de mutación: Valor entre 0 y 1 que permite determinar si procede o no la mutación. Como la frecuencia de mutación es pequeña este debe ser un valor pequeño ( $p_m=0.007$ ,  $p_m=0.0091$ , etc.).

# Operadores

- ▶ El operador de selección o Darwiniano realiza la selección de las cadenas de acuerdo a su adaptabilidad para el posterior apareamiento.
- ▶ El operador de cruzamiento o Mendeliano realiza la recombinación del material genético de dos cadenas padres.
- ▶ El operador de mutación al estilo del operador natural realiza la mutación de un gen dentro de un cromosoma o cadena a sus diferentes formas alelomorfos.
- ▶ Para cada uno de estos operadores está asociado el uso de probabilidades y la generación de números aleatorios.

# Algoritmo Genético. Selección



# Estrategia de selección

- ▶ Debemos de garantizar que los mejores individuos tienen una mayor posibilidad de ser padres (reproducirse) frente a los individuos menos buenos.
- ▶ Debemos de ser cuidadosos para dar una oportunidad de reproducirse a los individuos menos buenos. Éstos pueden incluir material genético útil en el proceso de reproducción.
- ▶ Esta idea nos define la presión selectiva que determina en qué grado la reproducción está dirigida por los mejores individuos.

# Operadores de selección

## Selección por Torneo (TS)

- Escoge al individuo de mejor fitness de entre  $N_{ts}$  individuos seleccionados aleatoriamente ( $N_{ts}=2,3,\dots$ ).

## Selección proporcional (PS)

- La población se ordena en función de su fitness y se asocia una probabilidad de selección a cada individuo que depende de su orden.

## Muestreo Aleatorio Universal (URS)

## Emparejamiento Variado Inverso (NAM)

- Un padre lo escoge aleatoriamente, para el otro selecciona  $N_{nam}$  padres y escoge el más lejano al primer ( $N_{nam}=3,5, \dots$ ). Está orientado a generar diversidad.

## Selección por Ruleta

- Se asigna una probabilidad de selección proporcional al valor del fitness del cromosoma. (Modelo clásico)

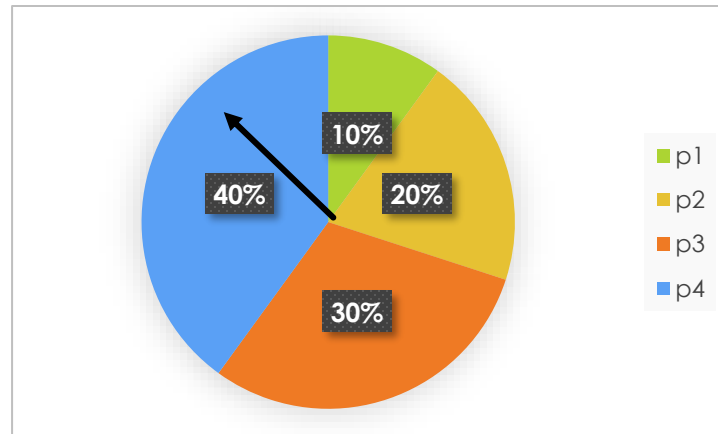
# Operadores de selección

- ▶ Selección proporcional
  - ▶ La probabilidad depende directamente del valor de aptitud del individuo

$$p_i = \frac{f_i}{\sum_{i=1}^{popsize} f_i}$$

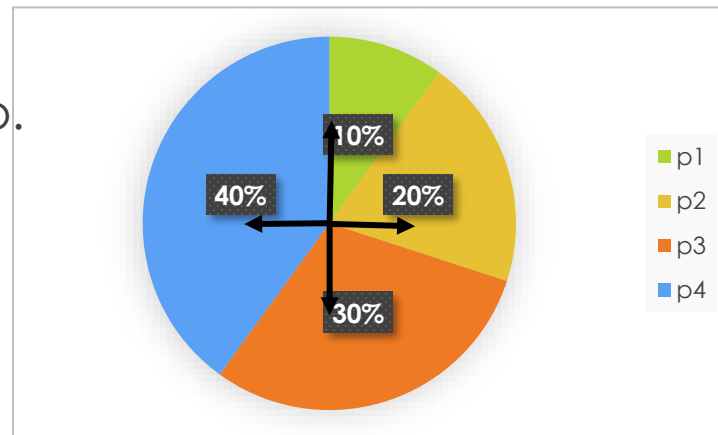
# Ruleta

- ▶ Tiene una única flecha que se mueve  $rand \times 2\pi$  en cada momento. Si cae en la región  $j$  se selecciona el individuo  $j$ .
- ▶ La selección se realiza en serie.



# Muestreo Aleatorio Universal

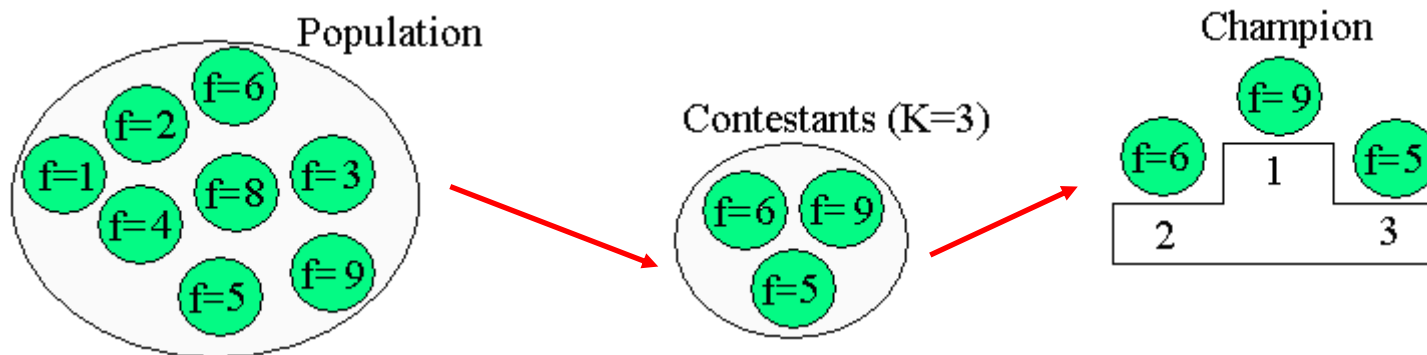
- ▶ Tiene tantas flechas como individuos, con un ángulo entre ellas de  $\frac{2\pi}{\text{popsize}}$ .  
Las flechas se mueven  $\text{rand} \times \frac{2\pi}{\text{popsize}}$  en cada momento.
- ▶ Si la flecha  $i$  cae en la región  $j$  se dice que la flecha  $i$  selecciona al individuo  $j$ .
- ▶ La selección se realiza en paralelo.



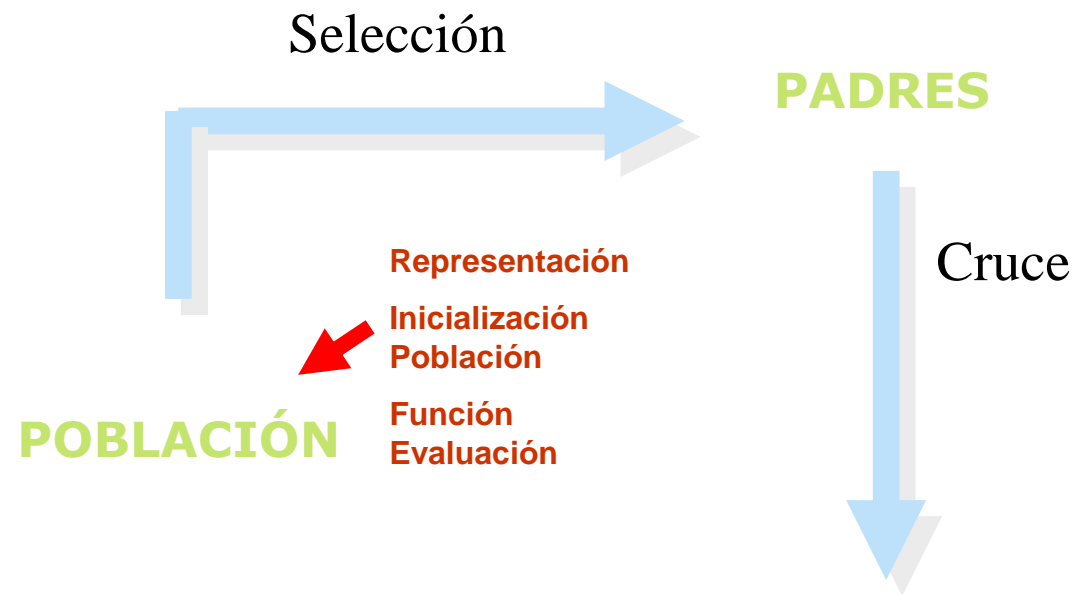


# Selección por Torneo

- ▶ Para cada padre a seleccionar:
  - ▶ Escoger aleatoriamente  $k$  individuos, con reemplazamiento
  - ▶ Seleccionar el mejor de ellos
- ▶  $k$  se denomina tamaño del torneo. A mayor  $k$ , mayor presión selectiva y viceversa.



# Algoritmos Genéticos. Cruzamiento

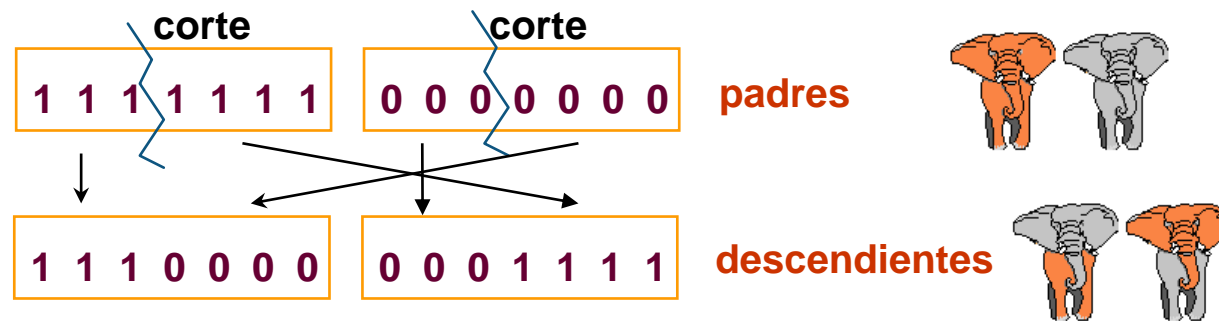


# Operador de cruzamiento

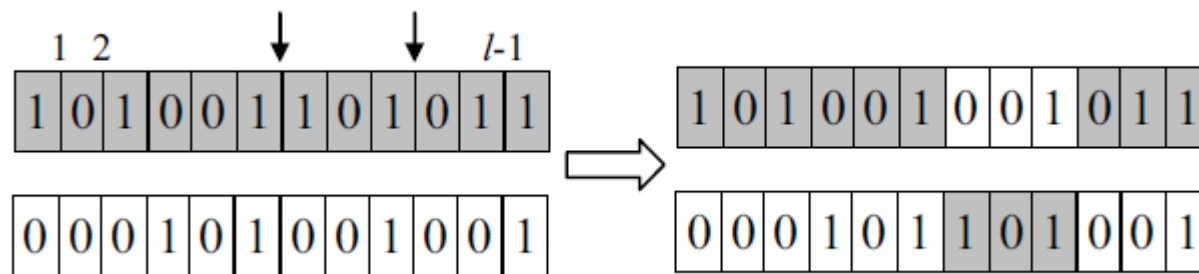
- ▶ Depende de la representación utilizada
- ▶ Se debe tener en cuenta:
  - ▶ Los hijos deberían heredar algunas características de cada padre
  - ▶ La recombinación debe producir cromosomas válidos
  - ▶ Se utiliza con una probabilidad alta de actuación sobre cada pareja de padres a cruzar ( $P_c$  entre 0.6 y 0.9)

# Ejemplo para representación binaria

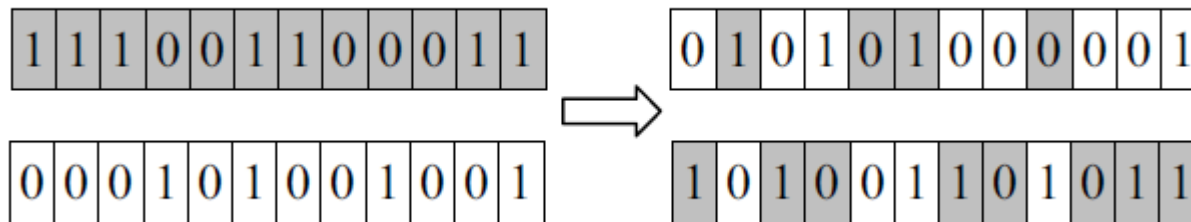
- Cada cromosoma se corta en  $n$  partes que son recombinadas. (Ejemplo para  $n = 1$ ).



# Cruzamiento en varios puntos



# Cruzamiento uniforme



# Ejemplo para representación real

|   |   |   |   |   |   |
|---|---|---|---|---|---|
| 2 | 3 | 4 | 1 | 5 | 3 |
|---|---|---|---|---|---|

|   |   |   |   |   |   |
|---|---|---|---|---|---|
| 4 | 5 | 8 | 7 | 3 | 1 |
|---|---|---|---|---|---|



Cruce aritmético, con  $\alpha = 0.5$

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $(2+4)/2$ | $(3+5)/2$ | $(4+8)/2$ | $(1+7)/2$ | $(5+3)/2$ | $(3+1)/2$ |
|-----------|-----------|-----------|-----------|-----------|-----------|

# Cruce para representación real teniendo en cuenta el entorno

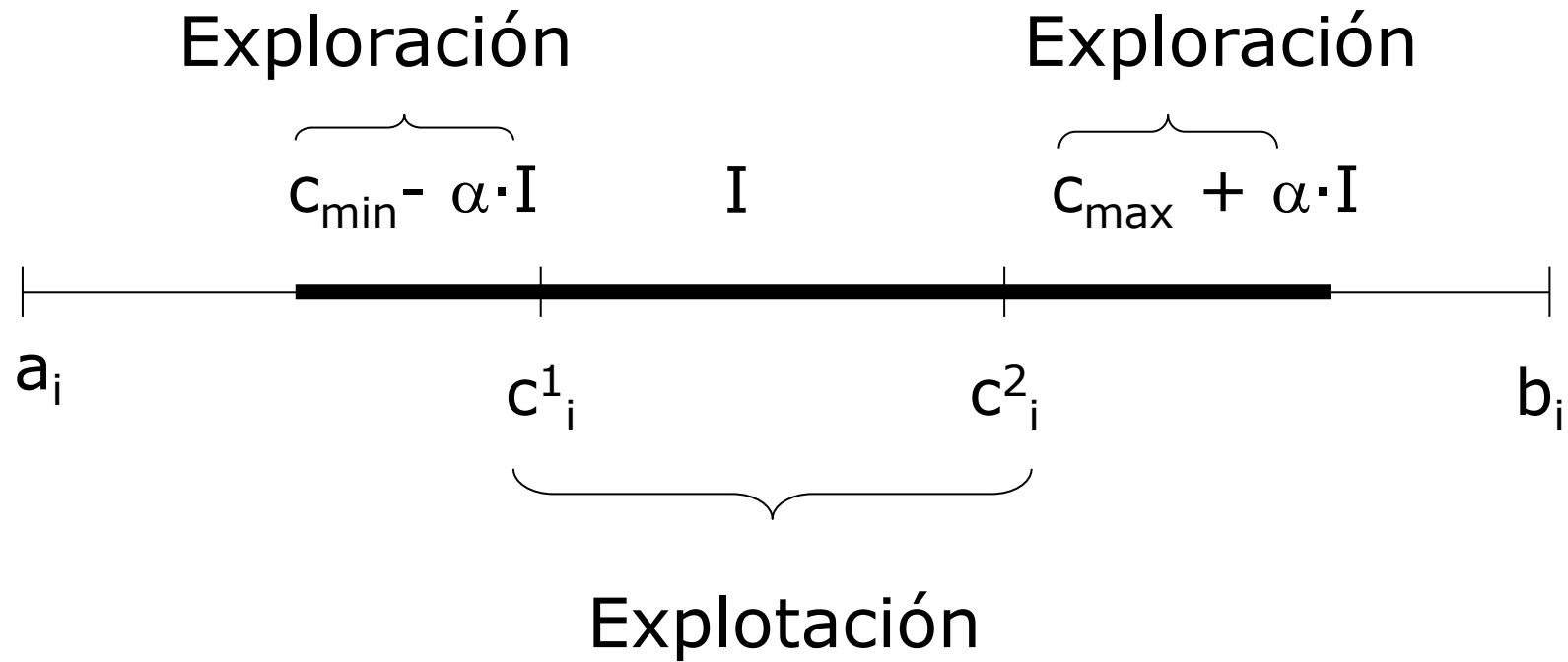
- Incluye los cruzamientos que determinan los genes en los descendientes extrayendo valores desde intervalos definidos por vecindades asociadas con los genes de los padres, a través de distribuciones probabilísticas.



# Ejemplo: blx- $\alpha$

- ▶ Dados 2 cromosomas
  - ▶  $C1 = (c11, \dots, c1n)$  y  $C2 = (c21, \dots, c2n)$  ,
- ▶ BLX-  $\alpha$  genera dos descendientes
  - ▶  $H_k = (hk1, \dots, hki, \dots, hkn)$  ,  $k = 1, 2$
  - ▶ donde  $hki$  se genera aleatoriamente en el intervalo:
- ▶  $[C_{\min} - l \cdot \alpha, C_{\max} + l \cdot \alpha]$ 
  - ▶  $C_{\max} = \max \{c1i, c2i\}$
  - ▶  $C_{\min} = \min \{c1i, c2i\}$
  - ▶  $l = C_{\max} - C_{\min}$  ,  $\alpha \in [0, 1]$

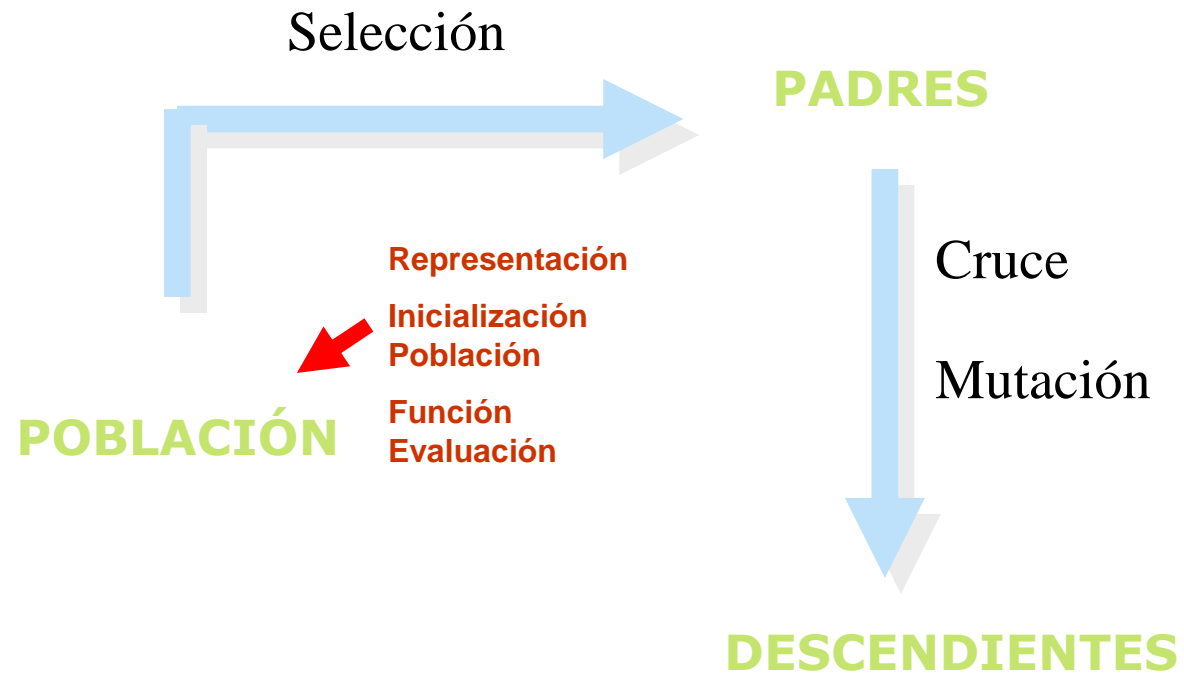
# Ejemplo: blx- $\alpha$



# Estudio Independiente

- ▶ Simplex Crossover (SPX)
- ▶ Simulated Binary Crossover (SBX)
- ▶ Unimodal Normal Distribution Crossover (UNDX)
- ▶ Cruzamiento para representaciones de orden

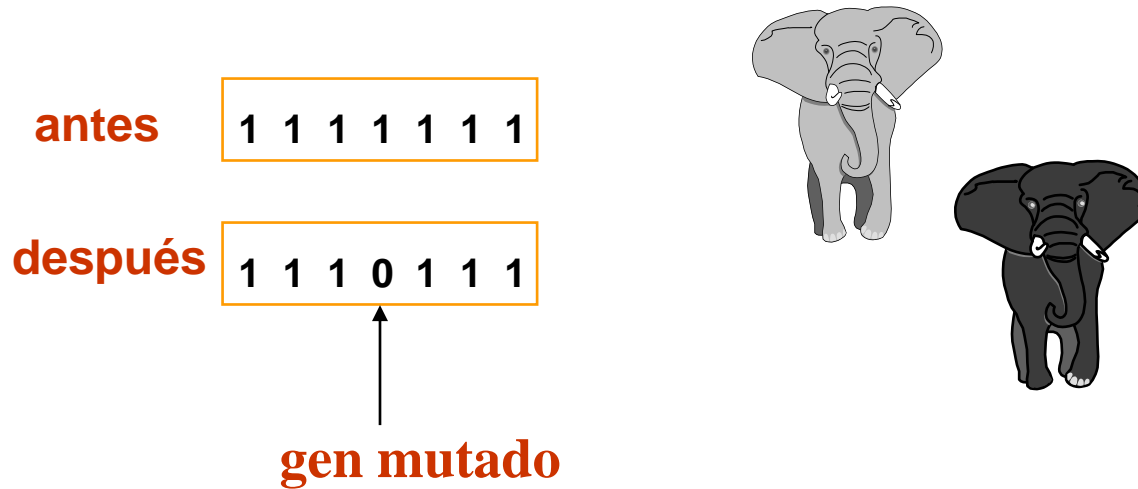
# Algoritmos Genéticos. Mutación



# Operador de mutación

- ▶ Debe permitir alcanzar cualquier parte del espacio de búsqueda.
- ▶ El tamaño de la mutación debe ser controlado.
- ▶ Debe producir cromosomas válidos.
- ▶ Se aplica con una probabilidad muy baja de actuación sobre cada descendiente obtenido tras aplicar el operador de cruce (incluidos los descendientes que coinciden con los padres porque el operador de cruce no actúa).

# Ejemplo para representación binaria



La mutación ocurre con una probabilidad  $p_m$   
para cada gen

# Ejemplo para representación real

- ▶ Mutación uniforme
  - ▶ Se reemplaza el gen por un valor aleatorio en el intervalo  $(X_{\min}, X_{\max})$
- ▶ Mutación límite
  - ▶ Se reemplaza el gen por  $X_{\min}$  si  $\text{rand} \leq 0.5$  o por  $X_{\max}$  en otro caso.

# Estudio Independiente

- ▶ Mutación no uniforme
- ▶ Mutación normal
- ▶ Mutación direccional



# Mutación para representación de orden

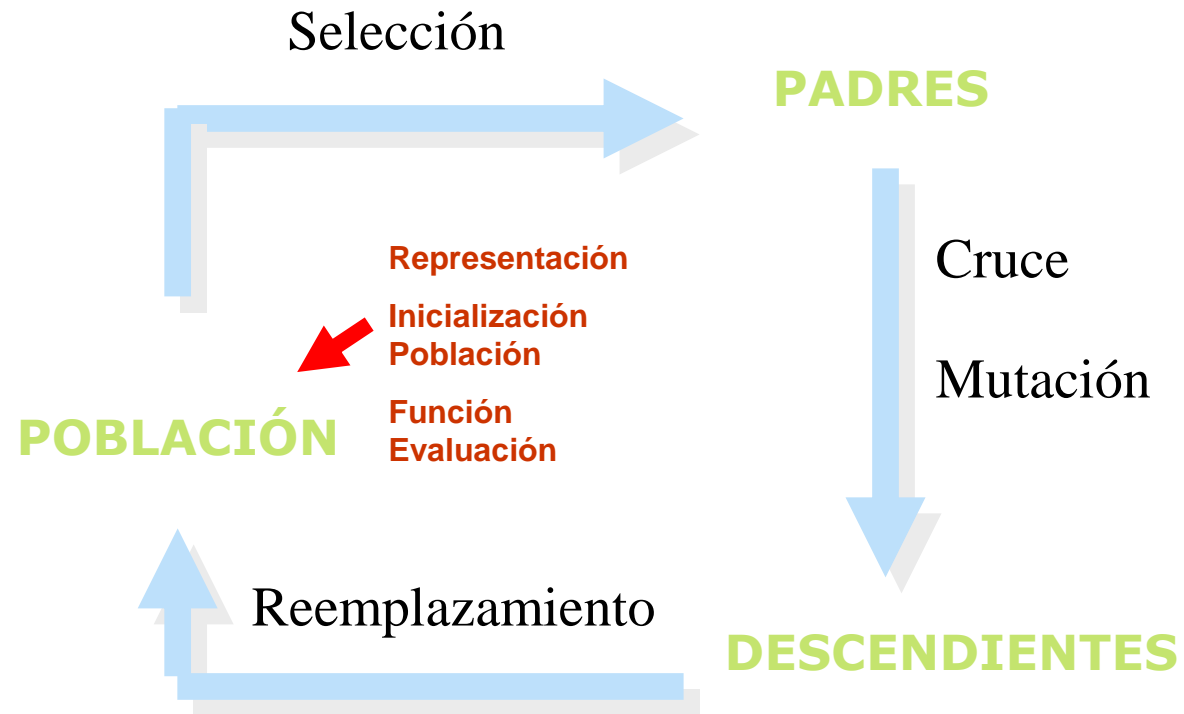
- Selección aleatoria de dos genes e intercambio de ambos.

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 3 | 1 | 8 | 2 | 4 | 6 | 5 |
|---|---|---|---|---|---|---|---|



|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 3 | 6 | 8 | 2 | 4 | 1 | 5 |
|---|---|---|---|---|---|---|---|

# Algoritmos Genéticos. Reemplazo



# Modelos Generacional y Estacionario

## Generacional

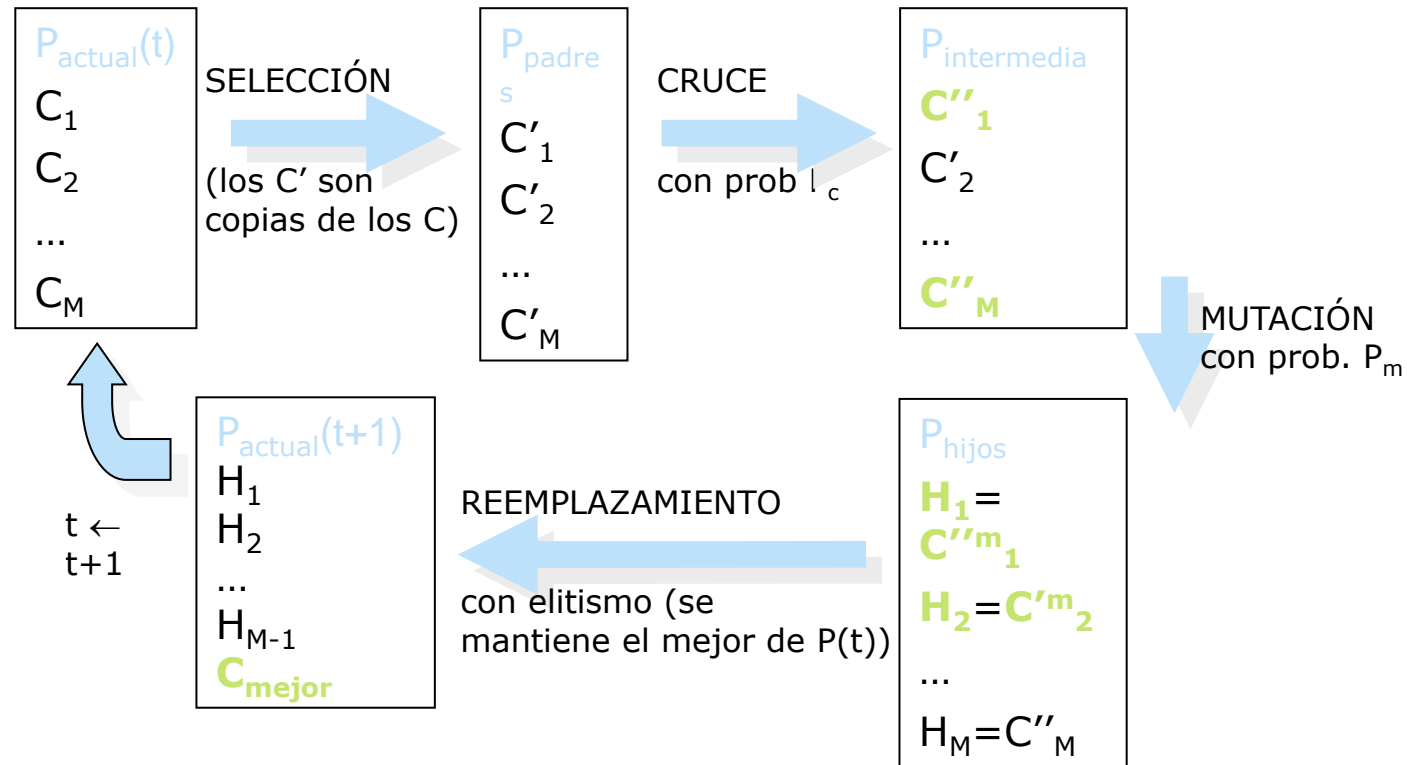
- ▶ Durante cada iteración se crea una población completa con nuevos individuos
- ▶ La nueva población reemplaza directamente a la antigua

## Estacionario

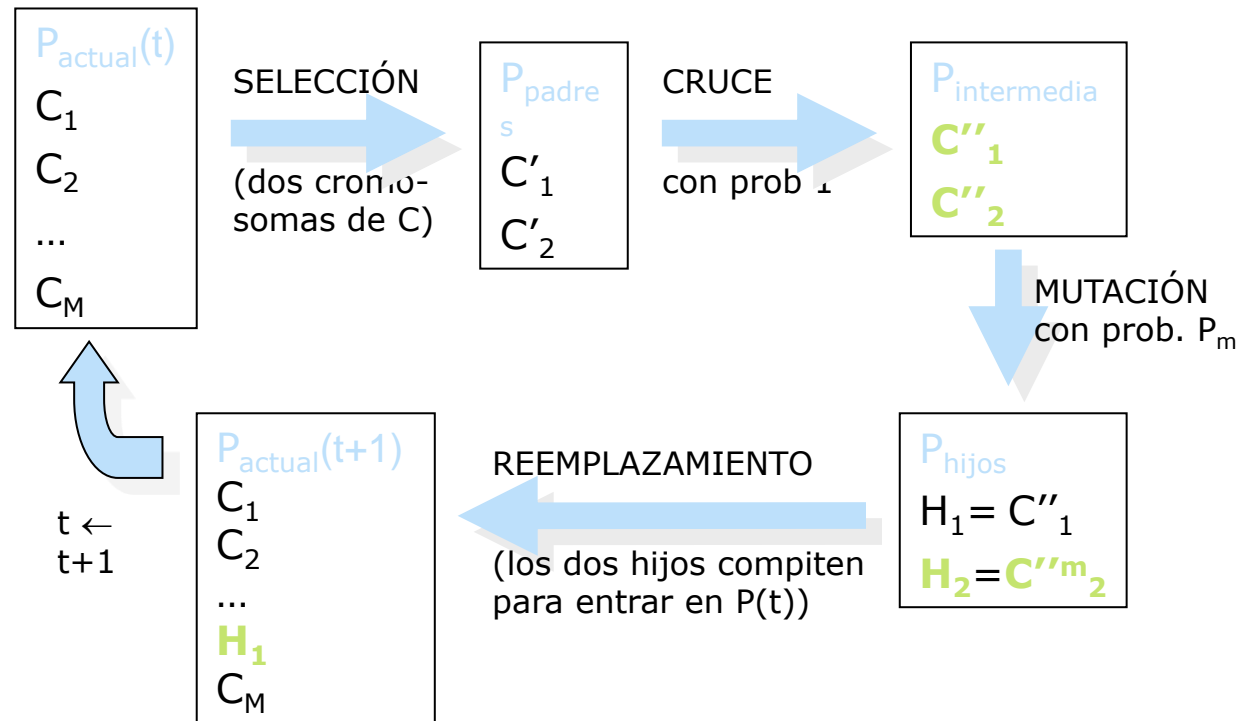
- ▶ Durante cada iteración se escogen dos padres de la población (diferentes mecanismos de muestreo) y se les aplican los operadores genéticos
- ▶ El/los descendiente/s reemplaza/n a uno/dos cromosoma/s de la población inicial

***El modelo estacionario es elitista. Además, produce una presión selectiva alta (convergencia rápida) cuando se reemplazan los peores cromosomas de la población***

# Modelo Generacional



# Modelo Estacionario



# Estrategias de reemplazo



# Estrategias de reemplazo

Al peor RW

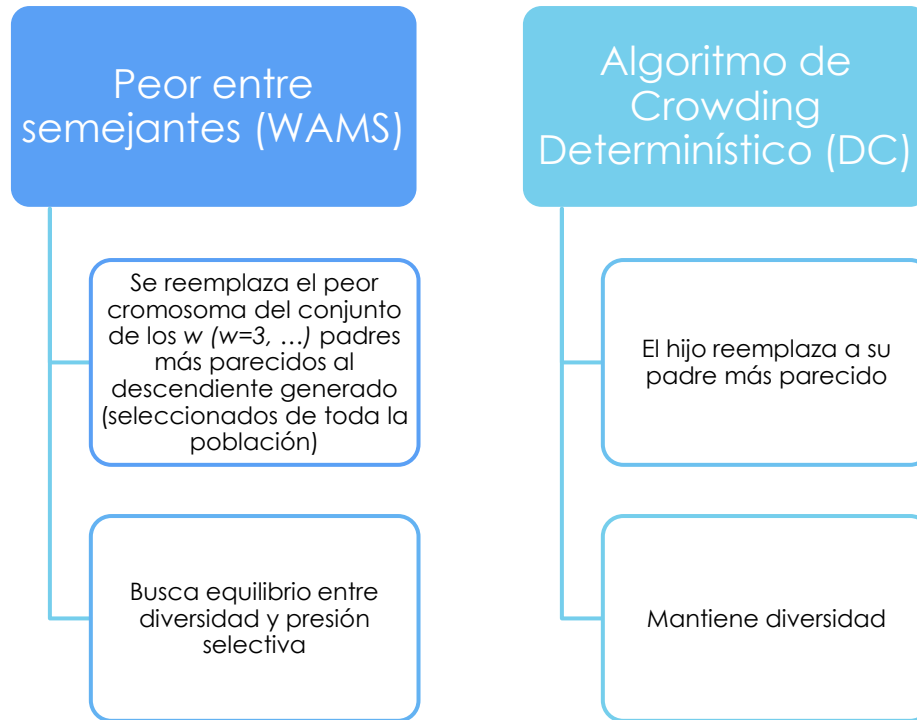
Reemplazar al  
peor

Torneo  
restringido RTS

Se reemplaza al  
más parecido  
de entre  $w$   
( $w=3, \dots$ )

Mantiene una  
cierta  
diversidad

# Estrategias de reemplazo





# Exploración vs Explotación

Exploración =  
muestrear regiones  
desconocidas

- Excesiva exploración  
= búsqueda  
aleatoria, no  
convergencia

Explotación = trata de  
mejorar el mejor  
individuo

- Excesiva explotación  
= solo búsqueda local  
... convergencia a un  
óptimo local

# Criterios de parada de los AG

1

- Se alcanza el valor óptimo

2

- Número predefinido de iteraciones o evaluaciones

3

- Número de iteraciones o evaluaciones sin mejora

# Diversidad

Asociada a las diferencias en los cromosomas de la población

Falta de diversidad genética = la mayoría de los individuos son parecidos

En la práctica, es irreversible

Muy alta presión selectiva conlleva a falta de diversidad

Falta de diversidad equivale a convergencia al vecino más cercano

# Mecanismos para lograrlo

## Mantener en paralelo varias sub-poblaciones

- Se procesan cada una por un algoritmo genético (Muhlenbein et al., 1991).

## Evolución con un método de agrupamiento

- Puede reducir el costo total de la Búsqueda Local evitando redescubrir varias veces los mismos óptimos locales (Seront and Bersini, 2000).

## Mecanismo de Prevención de Incesto

- Se cruzan solo aquellos individuos que presenten una determinada diversidad (Eshelman, 1991).

# Consideraciones adicionales

1

- La probabilidad de cruzamiento incrementa la recombinación de bloques de construcción pero también incrementa la destrucción de buenas cadenas.

2

- Incrementando la probabilidad de mutación se tiende a transformar la búsqueda genética en una búsqueda aleatoria, pero esto también permite reintroducir pérdida de material genético.

3

- Incrementando el tamaño de la población se incrementa la diversidad y reduce la probabilidad de que el AG haga una convergencia prematura, pero esto también incrementa el tiempo requerido para que la población converja a las regiones óptimas en el espacio de búsqueda

# Sobre su utilización en la práctica

## Nunca sacar conclusiones de una única ejecución

- Utilizar medidas estadísticas (medias, medianas, ...)
- Con un número suficiente de ejecuciones independientes

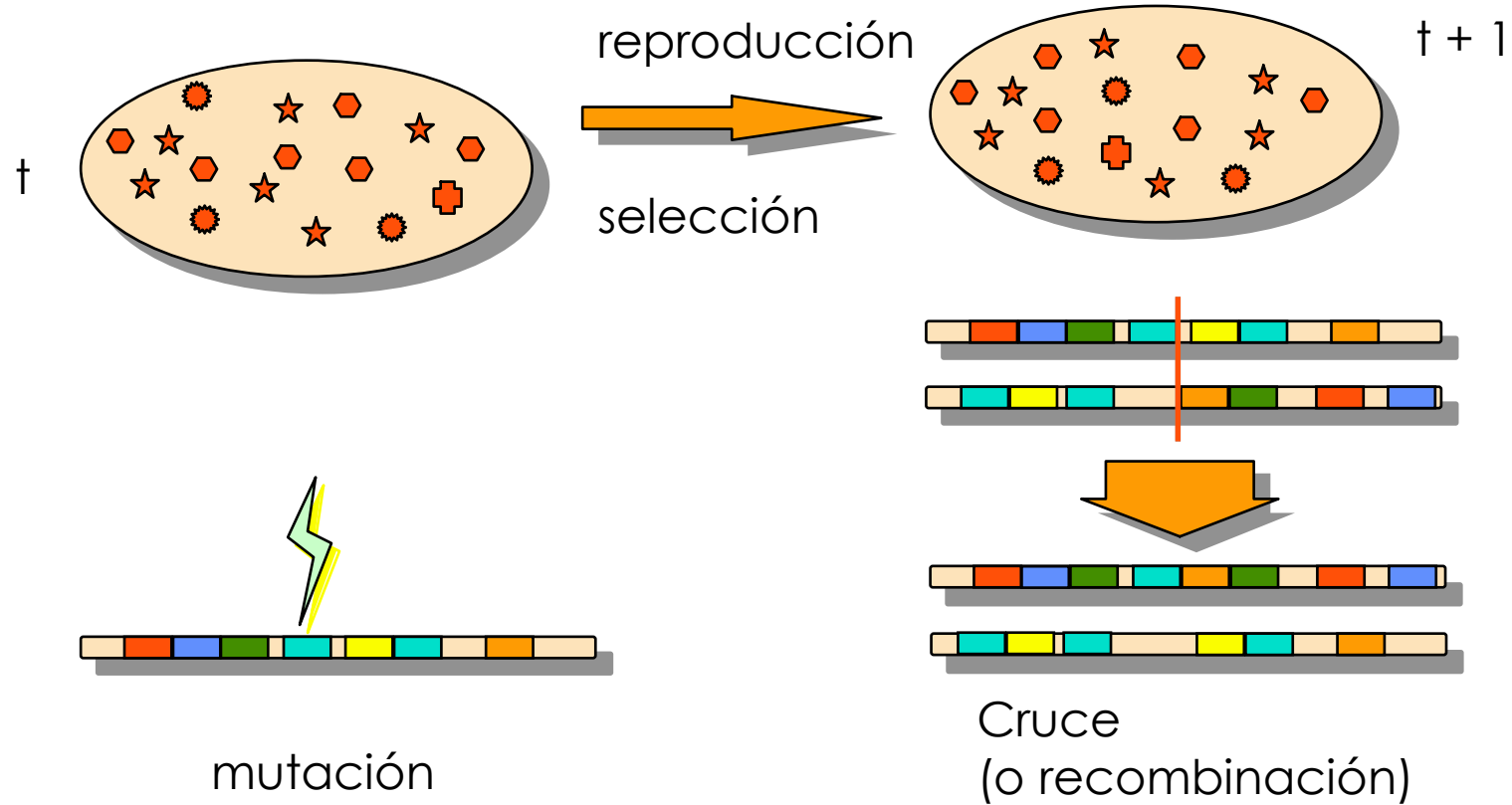
## Se puede obtener lo que se desea en una experimentación de acuerdo a la dificultad de los casos utilizados

- No se debe ajustar/chequear la actuación de un algoritmo sobre ejemplos simples si se desea trabajar con casos reales

## Las aplicaciones

- Encontrar una solución muy buena al menos una vez
- Encontrar al menos una solución muy buena en cada ejecución

# Los ingredientes



# Implementación

- ▶ Individuo
  - ▶ Debe conocer su aptitud
- ▶ Población (colección de individuos)
  - ▶ Debe ser capaz de inicializarse
- ▶ Operadores genéticos
  - ▶ Selección (Entrada: colección de individuos, Salida: 2 progenitores)
  - ▶ Cruzamiento (Entrada: 2 individuos, Salida: 2 individuos)
  - ▶ Mutación (Entrada: individuo)
- ▶ Algoritmo Genético (Entrada: población, operadores genéticos, Salida: población)



# Algoritmos Genéticos

Ventajas

Son muy versátiles.  
Utilizados con mucho éxito en numerosas aplicaciones.

Tienen muchos parámetros a configurar.

Desventajas

# Estudio independiente

- ▶ Yu & Gen. Introduction to Evolutionary Algorithms. 2010. Capítulos 1 al 3.
- ▶ Burke & Kendall. Search Methodologies. 2005 Capítulo 4.
- ▶ Russell & Norving. Artificial Intelligence - A Modern Approach – 1995. Capítulo 4.

# Estudio independiente

- ▶ Guía de la clase práctica #3
- ▶ Modelar los siguientes problemas con el uso de Algoritmos Genéticos
  - ▶ Problema de la mochila
  - ▶ Problema del viajero vendedor
  - ▶ Problema de mínimos de funciones no derivables

# Estudio independiente

- ▶ Envío del informe escrito un día antes de la próxima actividad
- ▶ Turnitin
  - ▶ <http://turnitin.com/>
  - ▶ ClassID: 14677689
  - ▶ Password: CIA17



Preguntas