



Algoritmos genéticos

1. Introducción
2. Esquema básico
3. Codificación
4. Evaluación
5. Selección
6. Operadores
7. Ejemplo





Introducción

- Propuestos por Holland, mediados 70, computación evolutiva
- Popularizados por Goldberg, mediados 80, solución de problemas del mundo real
- Inspirados en el modelo de evolución biológica sexual
- Aplicables a problemas de búsqueda y optimización complejos



Aproximación a la evolución biológica

- Método de búsqueda y optimización inspirados en la evolución biológica
- Posibles soluciones: población
- Selección de los individuos más *aptos*
- Generación de nuevos candidatos: reproducción sexual
 - Recombinación (cruce)
 - Mutación



Esquema básico

función ALGORITMO-GENÉTICO(*poblaciónInicial*) **returns** una población
entrada: *poblaciónInicial*, una población
static: *población(.)*, un array de población
begin
 $t \leftarrow 0$
 población(t) \leftarrow *poblaciónInicial*
 EVALUAR(*población*(t))
 while (not condiciónTerminación) **do**
 $t \leftarrow t + 1$
 población1 \leftarrow SELECCIONAR(*población*($t-1$))
 población2 \leftarrow CRUZAR(*población1*)
 población3 \leftarrow MUTAR(*población2*)
 EVALUAR(*población3*)
 población(t) \leftarrow REMPLAZAR(*población3* U *población*($t-1$))
 end
return(*población*(t))
end



Codificación

- Individuo: cromosoma
- Cromosoma: cadena de caracteres
 - En principio, cualquier representación es válida
- Codificación óptima: alfabeto binario (teorema de los esquemas)
- Codificación habitual: cadena de bits

Ejemplos de codificación maximización función

- $f(x)=1-x^2$, parábola invertida con máximo en $x=0$
- Buscar máximo en $[-1, 1]$ con una precisión de 2 cifras decimales
- Único parámetro o atributo: variable x
- Codificamos el valor de la variable mediante un byte $[0,255]$, pues hay que dividir el intervalo de longitud 2 en 200 unidades y $2^7 < 200 < 2^8$
- Ajustado al intervalo real $[-1,1]$, donde queremos hallar el máximo de la función

Valor binario	Descodificación	Valor real
10010100	148	0,16
10010001	145	0,14
00101001	41	-0,68
01000101	65	-0,49

$$\left[\frac{2}{2^8 - 1} \right] * x - 1 = y$$

Amplitud del intervalo

Origen del intervalo

8-reinas

- Atributo: posición de una dama en una columna (3 bits)
- Cromosoma: secuencia de atributos, 24 bits

Col.1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7	Col. 8																	
0	1	0	1	0	0	0	0	0	1	1	0	0	1	1	1	1	1	1	1	0	1	0	0	1



Regla de los bloques de construcción

- La codificación es clave en la resolución del problema
- Heurística: parámetros relacionados entre sí (*genes*) deben de estar cercanos en el cromosoma
- Gran flexibilidad
 - Cromosomas bi, tridimensionales
 - Longitud variable



Evaluación

- En esta etapa hay que cuantificar la calidad de los individuos de la población
- Generalmente
 - Descodificar el cromosoma
 - Calidad de la solución
 - Evaluación mediante función *fitness* o aptitud
 - Si problema de optimización, la propia función a optimizar



Ejemplos de funciones de aptitud

- Para $f(x)=1-x^2$, la función de aptitud es la misma
- Para 8-reinas: número total de pares de damas no amenazadas
 - En cualquier solución:
 $7+6+5+4+3+2+1=28$

Valor binario	Descodificación	Valor real	Aptitud
10010100	148	0,161	0.974
10010001	145	0,137	.981
00101001	41	-0,678	.540
01000101	65	-0,490	.760



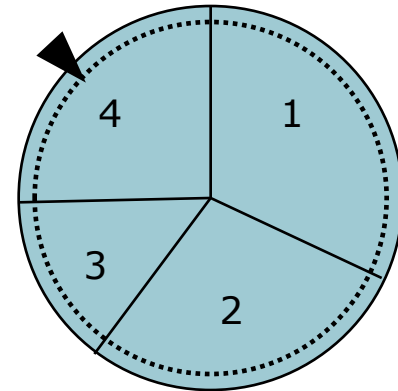
Selección

- Selección de los elementos que se reproducen
- A partir de la función de aptitud
- Varios métodos
 - Rueda de ruleta
 - Basado en el rango
 - Selección de torneo
- Cambio de generación
 - Manteniendo el tamaño de la población
 - Aumentando el tamaño de la población

Rueda de ruleta

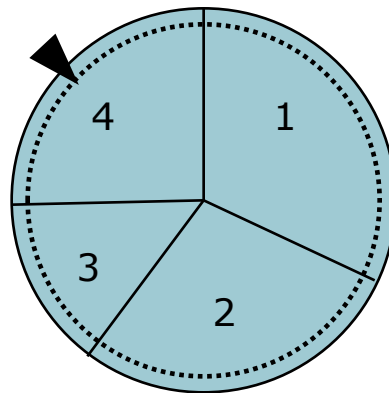
- Se asigna a cada individuo la probabilidad:
- Si algún individuo domina la población, se escala o normaliza
- Se elijen parejas aleatorias de individuos de acuerdo a su probabilidad
- Inconveniente: los individuos con más aptitud tiende a dominar la población en pocas generaciones

$$\Pr(x) = \frac{\textit{aptitud}(x)}{\sum_{y \in \textit{población}} \textit{aptitud}(y)}$$



Rueda

Valor binario	Descodificación	Valor real	Aptitud	Probabilidad	Probabilidad acumulada
10010100	148	0,161	0.974	0.299	0.299
10010001	145	0,137	0.981	0.301	0.600
00101001	41	-0,678	0.540	0.166	0.766
01000101	65	-0,490	0.760	0.233	1.000





Basado en el rango

- Se ordena la población por orden creciente de aptitud
- Se eliminan los M primeros (menor aptitud)
- Se eligen de forma aleatoria, con probabilidad dada por el rango, pares de individuos y sus descendientes se añaden a la población



Torneo

- Se seleccionan dos individuos aleatoriamente
- Se elije el más apto con una probabilidad P y el menos apto con una probabilidad $(1-P)$
- Introduce más diversidad en la población



Cambio de generación

- Manteniendo el tamaño de la población intermedia
 - Reemplazar padres por hijos
 - Reemplazar un par de individuos elegidos aleatoriamente por los hijos
 - Otros
- Aumentando el tamaño de la población intermedia
 - Crear población temporal con padres e hijos, seleccionando los mejores
 - Dados n padres generar m ($m > n$) hijos y de ellos seleccionar los n mejores

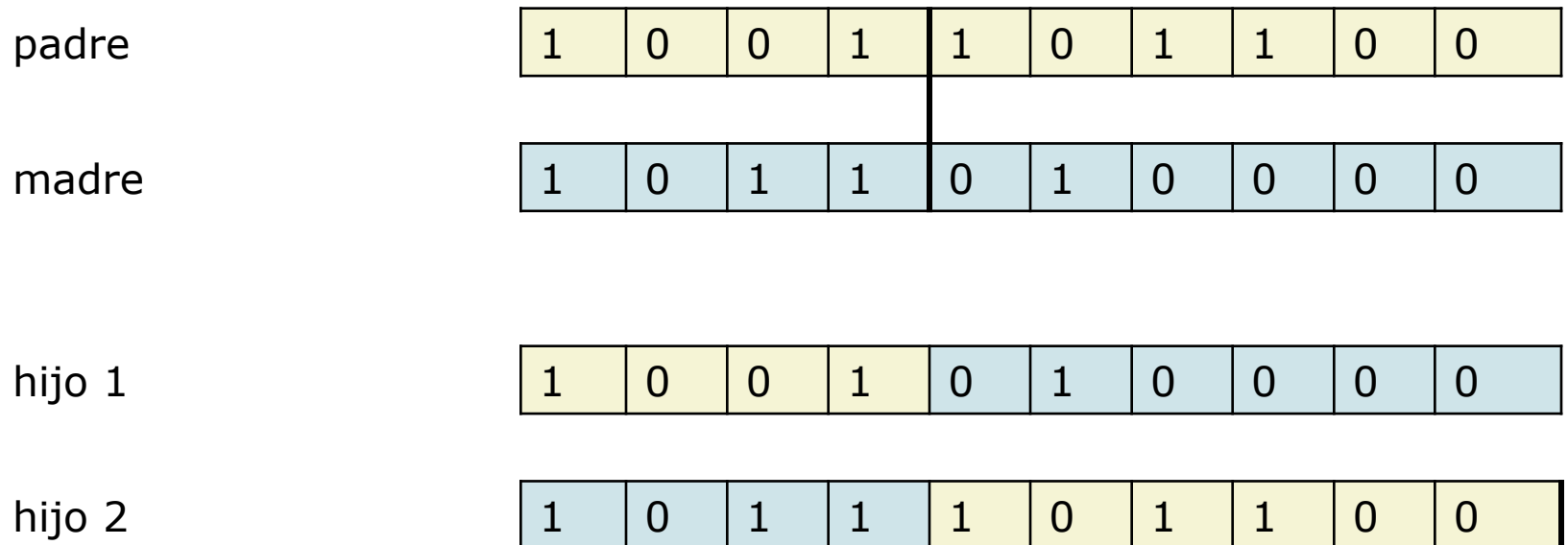


Operador de cruce (*crossover*)

- Principal operador genético
- Simula el intercambio de material genético o genes
- Se aplica con probabilidad p_c a individuos seleccionados
- Cruce ideal: recombina buenos bloques de construcción de sus progenitores
- Operadores
 - Cruce de n-puntos
 - Cruce multipunto
 - Cruce especializado

Cruce de un punto

- Seleccionar aleatoriamente una posición en el cromosoma
- Intercambiar el final del cromosoma a partir de dicho punto



Cruce de dos puntos

padre

1	0	0	1	1	0	1	1	0	0
---	---	---	---	---	---	---	---	---	---

madre

1	0	1	1	0	1	0	0	0	0
---	---	---	---	---	---	---	---	---	---

hijo 1

1	0	0	1	0	1	0	1	0	0
---	---	---	---	---	---	---	---	---	---

hijo 2

1	0	1	1	1	0	1	0	0	0
---	---	---	---	---	---	---	---	---	---



Otros operadores de cruce

- Multipunto o uniforme
 - Cada bit se hereda de un padre aleatoriamente
- Operadores especializados
 - En aquellos problemas donde un cruce aleatorio puede generar individuos no válidos



Ejemplo cruce 1 punto 8 reinas

- La selección aleatoria del punto de cruce no es interesante
 - Genera individuos válidos
 - La mezcla de bloques -genes- no parece asimilable a un operador del problema real
- Seleccionar aleatoriamente el gen a partir del que se hace el reemplazo
 - Seleccionar aleatoriamente un entero 1 y 7 (número de genes)
 - Equivale a intercambiar columnas contiguas entre tableros padres

Ejemplo cruce 1 punto 8 reinas

Col.1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7	Col. 8														
1	0	1	1	1	1	1	0	0	0	1	1	0	1	1	0	0	0	0	0	1	0

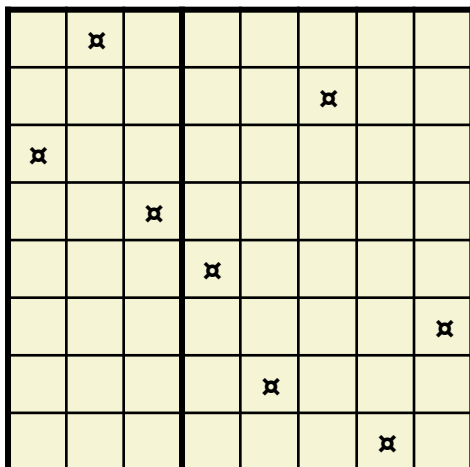
Col.1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7	Col. 8														
1	1	0	1	1	1	1	0	1	1	0	0	1	0	1	1	0	0	1	0	1	1

Col.1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7	Col. 8												
1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	0	1	0	1	1

Col.1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7	Col. 8														
1	1	0	1	1	1	1	0	1	0	1	1	0	1	1	0	0	0	0	0	1	0

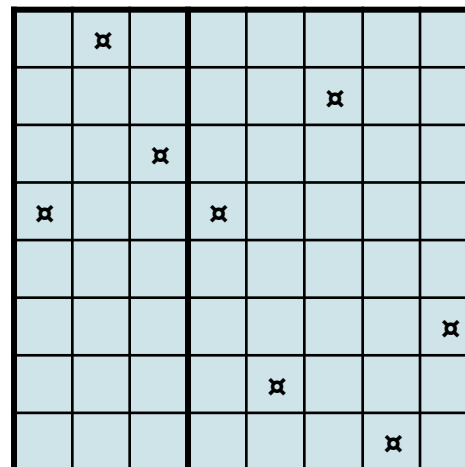
Ejemplo cruce 1 punto 8 reinas

Aptitud:20

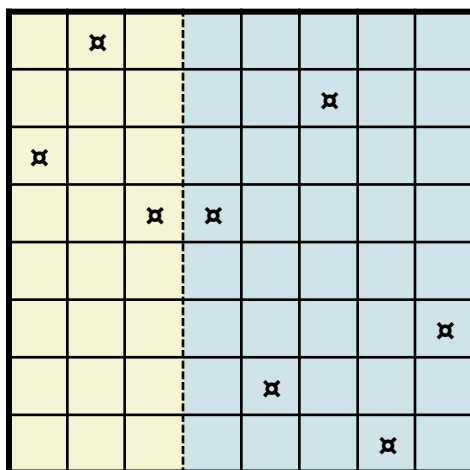


padres

Aptitud:26

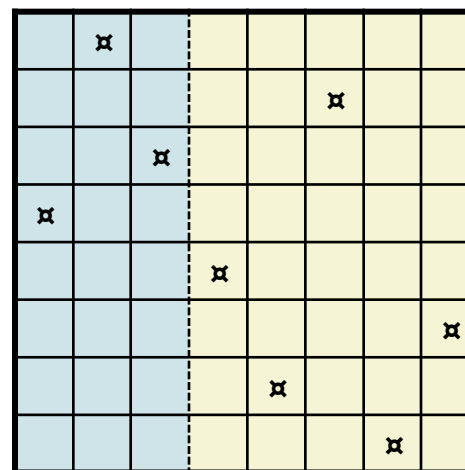


Aptitud:25



hijos

Aptitud:27





Operador de Mutación

- En la evolución
 - Las mutaciones son poco frecuentes
 - En la mayor parte de los casos letales
 - En promedio, contribuyen a la diversidad genética
- En los algoritmos genéticos:
 - Se simula cambiando aleatoriamente el valor de un bit
 - Se aplica con probabilidad baja (10^{-3} o menor) a cada bit de un nuevo individuo, habitualmente junto al cruce
 - Dependiendo del tamaño de la población y del número de bits por individuo, la mutación puede ser extremadamente rara en una generación



Utilidad de la mutación

- Genera diversidad
 - Puede ser de utilidad cuando un algoritmo genético está estancado
 - Su abuso reduce al algoritmo genético a una búsqueda aleatoria
- Otros mecanismos de generación de diversidad
 - Aumentar el tamaño de la población
 - Garantizar la aleatoriedad de la población inicial



Otros operadores

- Cromosomas de longitud variable
 - Añadir, eliminar
- Operadores de nicho
 - Fuerzan a que cromosomas similares sólo reemplacen a cromosomas similares
 - Intentan mantener la diversidad
 - Distintas “especies” en la población
 - Cada una de ellas puede converger a un máximo diferente



Ejemplo optimización: $f(x)=x^2$ encontrar máximo entero en $[1,32]$

- Codificación binaria: cadena de 5 bits
- Tamaño población inicial: 4 individuos
- Población inicial: aleatoria
 - Sortear cada bit de cada cadena con $p=1/2$
- Función de aptitud $f(x)=x^2$
- Selección: ruleta
- Cambio de generación: manteniendo el tamaño de la población intermedia
 - Reemplazar un par de individuos elegidos aleatoriamente por los hijos

Población inicial

	Población inicial	x	aptitud	Probabilidad selección	Probabilidad acumulada
1	01101	13	169	0.14	0.14
2	11000	24	576	0.49	0.63
3	01000	8	64	0.06	0.69
4	10011	19	361	0.31	1.00
Suma			1170		
Media			293		
mejor			576		



Selección: ruleta

- Generar cuatro números aleatorios, distribución de probabilidad uniforme en intervalo $(0,1)$
- Un individuo i se selecciona si el número aleatorio obtenido está en el intervalo definido por la probabilidad acumulada del individuo $i-1$ y la del individuo i
- Suponer que se obtienen: 0.58, 0.84, 0.11 y 0.43
- Individuos seleccionados: 2, 4, 1, 2

Población seleccionada

	Población inicial	x	aptitud	Probabilidad selección	Probabilidad acumulada
2	11000	24	576		
4	10011	19	361		
1	01101	13	169		
2	11000	24	576		
Suma			1682		
Media			420.5		
mejor			576		



Cruce

- Emparejamiento: emparejarlos según se han seleccionado -2 con 4, 1 con 2-
- Probabilidad de cruce: 0.8
 - Generar número aleatorio, distribución uniforme, (0, 1)
 - Suponer se obtienen 0.7, 0.3: se produce el cruce en ambos emparejamientos
- Generar puntos de cruce: número aleatorio, distribución uniforme en $[1, 2... ,L]$ con L longitud del cromosoma
 - Suponer se obtienen 2,3



Creación descendientes

2	1	1	0	0	0
4	1	0	0	1	1
h1	1	1	0	1	1
h2	1	0	0	0	0

1	0	1	1	0	1
2	1	1	0	0	0
h3	0	1	1	0	0
h4	1	1	0	0	1



Mutación

- Probabilidad mutación: 10^{-3}
- Suponer no se produce ninguna mutación



Nueva población: reemplazar padres por hijos

	Población 1ª iteración	x	aptitud	Probabilidad selección	Probabilidad acumulada
h1	11011	27	729	0.44	0.44
h2	10000	16	256	0.15	0.59
h3	01100	8	64	0.04	0.63
h4	11001	25	625	0.37	1.00
Suma			1674		
Media			418,5		
mejor			729		