



COMPUTACIÓN EVOLUTIVA

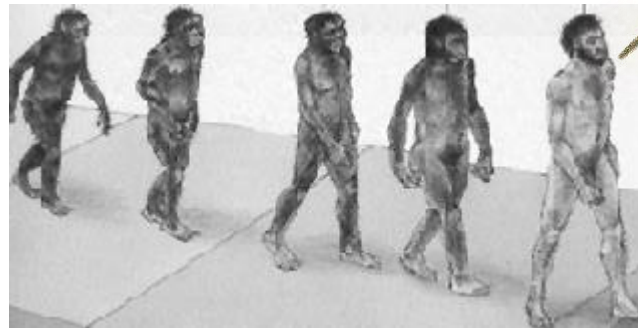
Abraham Sánchez López
FCC/BUAP
Grupo MOVIS



Facultad de Ciencias
de la Computación

Introducción

- La *computación evolutiva* es una rama de la inteligencia artificial que involucra problemas de optimización combinatoria. Se inspira en los mecanismos de la evolución biológica.
 - Para simular el proceso evolutivo en una computadora, se requiere:
 - Codificar las estructuras que se replicarán,
 - Operaciones que afecten a los “individuos”,
 - Una función de aptitud,
 - Un mecanismo de selección.
 - Paradigmas
 - Programación evolutiva
 - Estrategias evolutivas
 - Algoritmos genéticos
- Cada uno se originó de manera independiente y con motivaciones distintas.
- Nos centraremos en el tercer paradigma



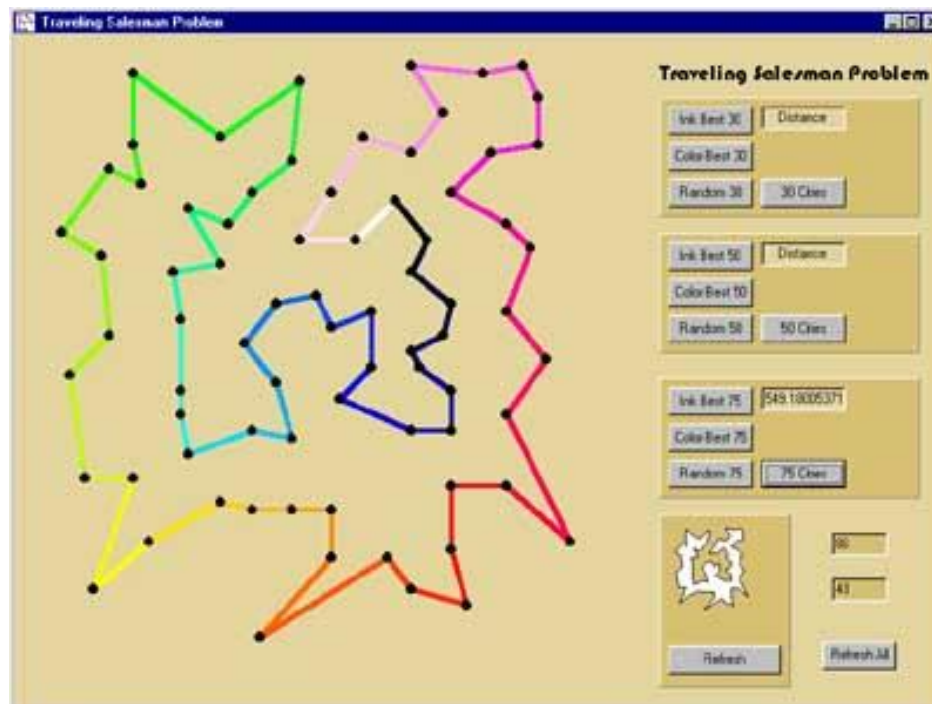
Programación evolutiva

- Esta enfatiza los nexos de comportamiento entre padres e hijos, en vez de buscar emular operadores genéticos específicos (como en los algoritmos genéticos).
- Propuesta de Lawrence J. Fogel.
- La programación evolutiva es una abstracción de la evolución al nivel de las especies, por lo que no se requiere el uso de un operador genético de recombinación (diferentes especies no se pueden cruzar entre si).
- Aplicaciones:
Predicción, generalización, juegos, control automático, TSP, planificación de rutas, etc.



Algoritmo

- Generar aleatoriamente una población inicial
- Se aplica mutación
- Se calcula la aptitud de cada hijo y se usa un proceso de selección mediante torneo (estocástico) para determinar cuáles serán las soluciones que se retendrán.



Estrategias evolutivas

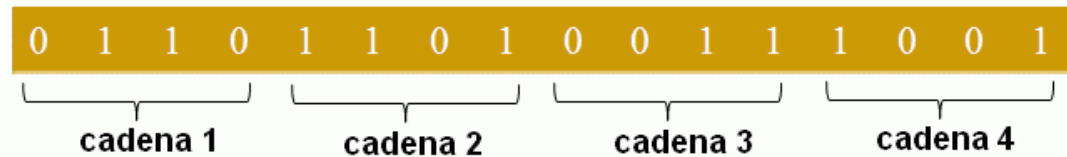
- Fueron desarrolladas en 1964 en Alemania para resolver problemas hidrodinámicos de alto grado de complejidad por un grupo de estudiantes de ingeniería encabezados por Ingo Recherberg.
- Algoritmo:
La versión original usaba un solo padre y con él se generaba un solo hijo.
Este hijo se mantenía si era mejor que el padre, o de lo contrario se eliminaba.
- Selección extintiva → los peores individuos obtienen una probabilidad de ser seleccionado de cero.
- Comparativo:
Programación evolutiva: selección estocástica
Estrategias evolutivas: selección determinista
- Aplicaciones:
Problemas de ruteo y redes, bioquímica, óptica, diseño en ingeniería, magnetismo.

Algoritmos genéticos, I

- Se denominaron originalmente planes reproductivos genéticos, fueron desarrollados por John H. Holland en los 60's para resolver problemas de aprendizaje máquina (machine learning).
- El algoritmo enfatiza la importancia de la cruce sexual (operador principal) sobre el de la mutación (operador secundario), y usa una selección probabilística.
- El algoritmo básico es el siguiente:
 - * Generar (aleatoriamente) una población inicial.
 - * Calcular la aptitud de cada individuo.
 - * Seleccionar (probabilísticamente) en base a la aptitud.
 - * Aplicar operadores genéticos (cruce y mutación) para generar la siguiente población.
 - * Repetir hasta que cierta condición se cumpla.

Algoritmos genéticos, I

- Ejemplo de representación



- A la cadena binaria se la llama “cromosoma”.
- A cada posición de la cadena se le denomina “gene” y al valor dentro de esta posición se le llama “alelo”.
- Para poder aplicar el algoritmo genético, se requiere de los siguiente componentes:
 - Una representación de las soluciones potenciales del problema.
 - Una forma de crear una población inicial de posibles soluciones (normalmente un proceso aleatorio).
 - Una función de evaluación que juegue el papel del ambiente clasificando las soluciones en términos de su “aptitud”.
 - Operadores genéticos que alteren la composición de los hijos que se producirán para las siguientes generaciones.

Algoritmos genéticos, III

- Valores para los diferentes parámetros que utiliza el algoritmo genético (tamaño de la población, probabilidad de cruce, probabilidad de mutación, número máximo de generaciones, etc.)
- Ejemplo de un algoritmo genético simple:

Generar una población inicial

Calcular la función de evaluación de cada individuo

Mientras no TERMINADO hacer

Inicio // producir nueva generación

Para tamaño_población/2 hacer

Inicio // ciclo reproductivo

Seleccionar dos individuos de la anterior generación, para el cruce (probabilidad de selección proporcional a la función de evaluación del individuo).

Cruzar con cierta probabilidad los dos individuos obteniendo dos descendientes.

Mutar los dos descendientes con cierta probabilidad.

Calcular la función de evaluación de los dos descendientes mutados.

Insertar los dos descendientes mutados en la nueva generación.

Fin

Si la población ha convergido entonces TERMINADO = CIERTO

Fin

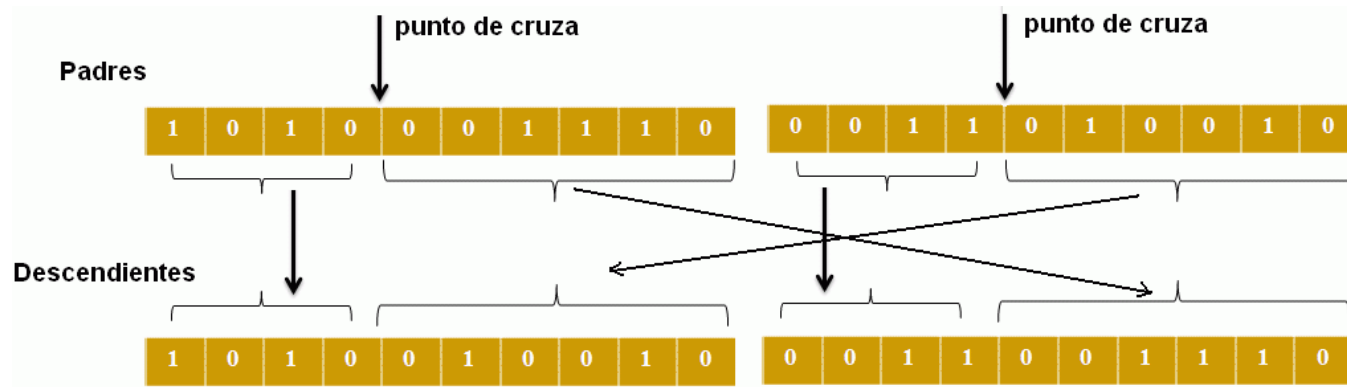
Fin

Algoritmos genéticos, IV

- Como representación, usaremos un alfabeto binario. La función de adaptación se debe diseñar para cada problema en forma específica.
- Dado un cromosoma particular, la función de adaptación le asigna un número real, que se supone refleja el nivel de adaptación al problema del individuo representado por el cromosoma.
- Durante la fase reproductiva se seleccionan los individuos de la población para cruzarse y producir descendientes, que constituirán, una vez mutados, la siguiente generación de individuos.
- La selección de padres se efectuó al azar usando un procedimiento que favorezca a los individuos mejor adaptados (cada individuo tiene asignada una probabilidad de ser asignado que es proporcional a su función de adaptación).
→ ruleta sesgada
- Los individuos bien adaptados se escogerán probablemente varias veces por generación, mientras que los pobremente adaptados al problema, no se escogerán más que de vez en cuando.

Algoritmos genéticos, V

- Una vez seleccionados dos padres, sus cromosomas se combinan, utilizando habitualmente los operadores de cruce y mutación.
- El operador de cruce, toma dos padres seleccionados y corta sus listas de cromosomas en una posición elegida al azar, para producir dos sub-listas iniciales y dos sub-listas finales.
- Después se intercambian las finales, produciéndose dos nuevos cromosomas completos.
- Ambos descendientes heredan genes de cada uno de los padres. Este operador se conoce como cruce de un punto.



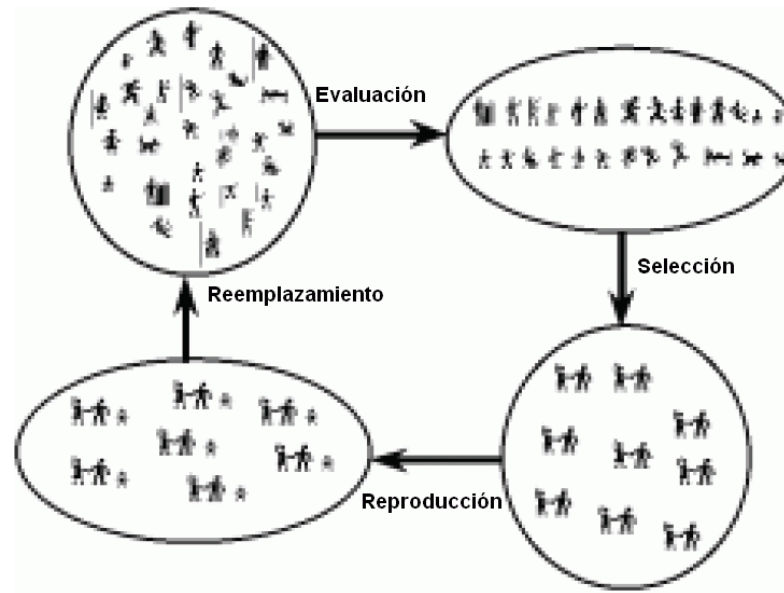
Algoritmos genéticos, VI

- El operador de cruce no se aplica a todos los pares de individuos que han sido seleccionados para emparejarse, sino que se aplican de manera aleatoria, normalmente con una probabilidad entre 0.5 y 1.0.
- En el caso en que el operador de cruce no se aplique, la descendencia se obtiene simplemente duplicando los padres.
- El operador de mutación se aplica a cada hijo de manera individual, y consiste en la alteración aleatoria (normalmente con probabilidad pequeña) de cada gene componente del cromosoma.

	gene mutado ↓									
Descendiente	1	0	1	0	0	1	0	0	1	0
Descendiente mutado	1	0	1	0	1	1	0	0	1	0

Algoritmos genéticos, VII

- Se puede pensar que el operador de cruce es más importante que el operador de mutación.
- Ya que proporciona una exploración rápida del espacio de búsqueda.
- Este último asegura que ningún punto del espacio de búsqueda tenga probabilidad cero de ser examinado.
- Es de mucha importancia para asegurar la convergencia de los algoritmos genéticos



Ejemplo

- Se desea encontrar el máximo de la función $f(x) = x^2$ sobre los enteros $\{1, 2, \dots, 32\}$ (Goldberg89).
- Suponiendo que el alfabeto utilizado para codificar los individuos esté constituido por $\{0,1\}$, se necesitaran listas de longitud 5 para representar los 32 puntos del espacio de búsqueda.

	Población inicial (fenotipo)	x valor genotipo	$f(x)$ valor (función de adaptación)	$f(x)/\Sigma f(x)$ (probabilidad de selección)	Probabilidad de selección 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		

Población en el tiempo 0

Detalles, I

- El siguiente paso, consiste en la selección de 2 parejas de individuos.
- Se obtienen 4 números reales provenientes de una distribución de probabilidad uniforme en el intervalo $[0,1]$, se comparan con la última columna de la tabla anterior. Si estos números fueron: 0.58, 0.84, 0.11 y 0.43
- Esto significa que los individuos seleccionados para la cruce son: el individuo 2 junto con el individuo 4, y el individuo 1 con el individuo 2.
- Supongamos que la probabilidad de cruce se ha fijado en $P_c = 0.8$. Utilizando 2 números provenientes de la distribución uniforme, determinaremos si los emparejamientos anteriores se llevan a cabo.
- Si admitimos por ejemplo, que los dos números extraídos sean menores que 0.8, diciendo por lo tanto efectuar la cruce entre las dos parejas.
- Escogemos un número al azar entre 1 y $l-1$ (l es longitud de la lista utilizada para representar al individuo).
 - ➔ Esta restricción se impone para que los descendientes no coincidan con los padres.

Detalles, II

- Supongamos que los puntos de cruce resultan ser 2 y 3, entonces obtenemos los 4 descendientes que se muestran en la siguiente tabla.
- A continuación, debemos mutar a los nuevos individuos con una probabilidad, P_m , cercana a 0, cada uno de los bits de las 4 listas de individuos.
- Supongamos que el único bit mutado corresponde al primer gen del tercer individuo.
- Como se puede observar, tanto el mejor individuo como la función de adaptación media ha mejorado significativamente si comparamos estos resultados con la tabla anterior.

Población en el tiempo 1

Emparejamiento de los individuos seleccionados	Punto de cruce	Descendientes	Nueva población (descendientes mutados)	x Valor genotipo	$f(x)$, función de adaptación
11000	2	11011	11011	27	729
10011	2	10000	10000	16	256
01101	3	01100	11100	28	784
11000	3	11101	11101	29	841
Suma					2610
Media					652.5
Mejor					841