

UNIVERSIDAD NACIONAL DEL OESTE

PROGRAMACIÓN CON OBJETOS I

TP ESPECIAL N° 2

(2015)

Un algoritmo es una serie de pasos ordenados que se siguen para dar solución a un problema específico. En computación, los algoritmos sirven como base en la elaboración de programas, ya que son códigos de instrucciones escritos con el objetivo de que la computadora realice una tarea determinada. Uno de los propósitos de los programas de computación es aprovechar la enorme capacidad de procesamiento de las computadoras para realizar tareas que manualmente requerirían invertir una gran cantidad de tiempo. Completar eficientemente estas tareas, es decir, en el menor tiempo posible y usando la mínima cantidad de recursos computacionales, mucho depende del tipo de algoritmos que se empleen.

Por ejemplo, supongamos que deseamos descifrar una contraseña compuesta por cinco dígitos del cero al nueve. Tal vez lo primero que se nos ocurriría sería confiar en la capacidad de nuestra máquina y buscar entre cada una de las posibles combinaciones de números, es decir, desde 00000 hasta 99999. Este procedimiento es conocido como algoritmo de fuerza bruta; es fácil de implementar, pero se vuelve poco funcional cuando el espacio de búsqueda es muy grande.

Una interesante y eficaz respuesta a este tipo de problemas es el uso de algoritmos genéticos, los cuales basan su funcionamiento en técnicas propias de la selección natural (Darwin, 1859), como la elección de los individuos más aptos, la reproducción o cruce, y la mutación para crear remedios capaces de evolucionar a través de las generaciones hacia la solución correcta.

Descripción

Un algoritmo genético es un algoritmo de búsqueda basado en los mecanismos de selección y genética natural (Goldberg, 1989). Su funcionamiento parte de generar aleatoriamente una serie de posibles soluciones, también llamadas población inicial, codificadas comúnmente en una representación binaria que evoluciona a través de las generaciones, siguiendo los principios de la selección natural hasta llegar a la respuesta requerida.

El proceso de evolución consiste en evaluar mediante una función de aptitud qué tan buenas son cada una de las posibles soluciones generadas en la población inicial. Tomando en cuenta algún criterio de competencia basado en dicha evaluación, se seleccionan las soluciones más aptas para reproducirse, después se cruzan o se combina el material genético de las parejas seleccionadas, y por último se mutan o sufren pequeñas modificaciones. Todo este proceso se repite el número de veces que se considere conveniente o hasta que la solución sea encontrada (**Figura 1**).

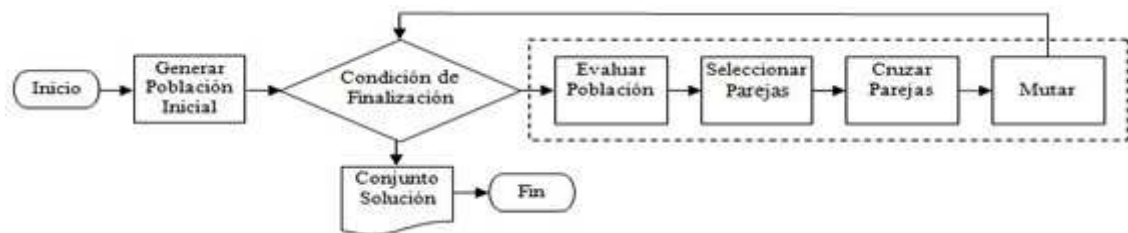


Figura 1. Diagrama de flujo de un algoritmo genético.

Desarrollo.

El ejemplo consiste básicamente en descifrar una clave compuesta por una serie de 10 dígitos hexadecimales generados aleatoriamente (**figura 2**).

1	A	C	7	D	9	B	2	7	3
---	---	---	---	---	---	---	---	---	---

Figura 2. Clave generada aleatoriamente

Lo primero que debe hacerse es encontrar una manera de codificar las posibles soluciones. Se utilizará una codificación binaria. Debido a que la clave está integrada por 10 dígitos hexadecimales y se requiere de cuatro dígitos binarios para representar un número hexadecimal, es necesario tener individuos de longitud igual a 40 bits, tal y como se muestra en la figura 3.

1	A	C	7	D	9	B	2	7	3
0001	1010	1100	0111	1101	1001	1011	0010	0111	0011

Figura 3. Codificación de ejemplo para algoritmo genético.

La población de un algoritmo genético está constituida por un conjunto de individuos, también llamados cromosomas. Para este ejemplo se tomará una población de 10 cromosomas.

El algoritmo genético debe partir de una población inicial. Ésta se forma generando cromosomas (cadenas de 40 bits) al azar. En este caso se partirá de la población inicial mostrada en la tabla 1.

Tabla 1. Población inicial de ejemplo para algoritmo genético.

Cromosoma	Población Inicial (Hexadecimal)	Población Inicial (Binaria)	Fitness
0	F D 4 7 D D 6 6 B 4	1111110101000111110111010110011010110100	1587.0
1	4 3 2 C 0 0 C F 8 1	0100001100101100000000001100111110000001	196.0
2	0 E 4 6 5 3 C 6 2 F	0000111001000110010100111100011000101111	625.0
3	E 0 E 3 2 B E 9 D 5	111000011100011001010111110100111010101	324.0
4	1 1 C A 0 9 6 5 A 7	0001000111001010000010010110010110100111	1764.0
5	6 2 D 2 B D 1 C E 5	0110001011010010101111010001110011100101	441.0
6	6 4 6 D 9 2 D 8 0 B	0110010001101101100100101101100000001011	324.0
7	7 0 6 9 1 0 0 0 B 6	011100000110100100010000000000010110110	361.0
8	B C 6 C 6 1 D 3 E E	1011110001101100011000011101001111101110	361.0
9	3 B A D B 1 9 C 7 E	0011101110101101101100011001110001111110	1152.0
Clave	1 A C 7 D 9 B 2 7 3	0001101011000111110110011011001001110011	

Luego debe realizarse un proceso de selección para obtener a los mejores individuos de la población. Para ello es necesario contar con una función de aptitud (fitness) que indique cuáles son los mejores cromosomas. Para el ejemplo, la función de aptitud esta dada por: $f=(h+1)*b^2$, donde h y b son la cantidad de dígitos hexadecimales y binarios respectivamente que coinciden con la clave original. Para la población inicial, el mejor individuo es el cromosoma 4 con h=3, b=21 y f=1764 (ver tabla1). Cabe mencionar que cuando un cromosoma de la población sea idéntico a la clave original, es decir cuando la clave sea encontrada, h valdrá 10 y b será igual a 40, entonces el valor de la función de aptitud será de 17,600.

Una manera de realizar el proceso de selección es mediante torneo entre dos individuos. Se seleccionan dos cromosomas aleatoriamente y entre ellos se establece un torneo: el que mejor función de aptitud tenga genera una copia y el peor se desecha. La tabla 2 muestra este proceso para la población inicial del ejemplo.

Tabla 2. Proceso de selección de ejemplo.

Torneo	Seleccionado	Nueva Población
Cromosoma 5 Vs Cromosoma 7	Cromosoma 5	(0) 0110001011010010101111010001110011100101
Cromosoma 0 Vs Cromosoma 4	Cromosoma 4	(1) 0001000111001010000001001011001011010111
Cromosoma 3 Vs Cromosoma 7	Cromosoma 7	(2) 011100000110100100010000000000010110110
Cromosoma 0 Vs Cromosoma 9	Cromosoma 0	(3) 1111110101000111110111010110011010110100
Cromosoma 2 Vs Cromosoma 4	Cromosoma 4	(4) 0001000111001010000010010110010110100111
Cromosoma 5 Vs Cromosoma 0	Cromosoma 0	(5) 1111110101000111110111010110011010110100
Cromosoma 9 Vs Cromosoma 6	Cromosoma 9	(6) 0011101110101101101100011001110001111110
Cromosoma 7 Vs Cromosoma 6	Cromosoma 7	(7) 011100000110100100010000000000010110110
Cromosoma 2 Vs Cromosoma 5	Cromosoma 2	(8) 0000111001000110010100111100011000101111
Cromosoma 1 Vs Cromosoma 5	Cromosoma 5	(9) 0110001011010010101111010001110011100101

Luego de realizar la selección, se aplican los operadores genéticos. Para el ejemplo se aplicará un cruce de un punto. El proceso consiste en formar parejas aleatoriamente entre los individuos de la nueva población. Dados dos cromosomas pareja (padre y madre), se establece un punto de cruce aleatorio, que no es más que un número al azar entre 0 y 39 (considerando que se empieza a contar desde cero). La operación de cruce consiste en generar dos nuevos individuos, intercambiando los bits de los padres a partir del punto de cruce, como se muestra en la tabla 3.

Tabla 3. Ejemplo de operación de cruce.

Pareja	Punto de Cruce	Descendientes
Cromosoma 6 Cromosoma 2	10	0011101110101001000100000000000101101110 0111000001101101101100011001110001111110
Cromosoma 0 Cromosoma 8	25	0110001011010010101111010100011000101111 0000111001000110010100111001110011100101
Cromosoma 1 Cromosoma 9	11	0001000111010010101111010001110011100101 011000101101010000010010110010110100111
Cromosoma 5 Cromosoma 7	10	1111110101101001000100000000000101101110 0111000001000111110111011001101011010100
Cromosoma 3 Cromosoma 4	21	1111110101000111110110010110010110100111 0001000111001010000011010110011010110100

Después de la selección y el cruce se da lugar a la mutación, la cual proporciona un pequeño elemento de aleatoriedad en el entorno de los individuos de la población. La mutación consiste en elegir individuos al azar de la población, y alterar alguno de sus genes o bits. Para el ejemplo, esto se haría cambiando 4 bits al azar de 0 a 1 (o de 1 a 0)

de uno o más individuos de la población. Ahora, en la nueva población, el mejor individuo es el cromosoma 3, con $h=3$, $b=23$ y $f=2116$. Esto indica que luego de la selección, el cruce y la mutación, los individuos de la población han mejorado (**tabla 4**).

El siguiente paso consiste en realizar nuevamente la selección, el cruce y la mutación, tomando como entrada la población obtenida en la iteración anterior. Este proceso se repite hasta obtener un cromosoma con una puntuación de 17, 600, es decir, hasta que la solución sea encontrada. En la figura 4 se muestra la ejecución del programa, y cómo es que encuentra la solución después de 78 generaciones.

Tabla 4. Población resultante después de aplicar la selección, cruce y mutación.

Cromosoma	Población (Hexadecimal)	Población (Binaria)	Fitness
0	6 2 D 2 B D 4 6 2 7	0110001011010010101111010100011000100111	484.0
1	1 1 D 2 B D 1 C E 5	0001000111010010101111010001110011100101	968.0
2	7 0 6 5 B 1 9 C 7 E	0111000001100101101100011001110001111110	1058.0
3	F D 4 7 D 9 6 5 A 7	111111010100011110110010110010110100111	2116.0
4	1 1 C A 0 D 6 6 B 4	0001000111001010000011010110011010110100	1323.0
5	F D 6 9 1 0 0 0 B 6	1111110101101001000100000000000010110110	289.0
6	3 B A 9 0 0 0 0 B 6	0011101110101001000000000000000010110110	400.0
7	7 0 4 7 D D 6 6 B 4	011100000100011110111010110011010110100	1875.0
8	0 E 4 6 5 3 9 C E 5	0000111001000110010100111001110011100101	625.0
9	6 2 C A 1 9 6 5 A 7	0110001011001010000110010110010110100111	1323.0
Clave	1 A C 7 D 9 B 2 7 3	0001101011000111110110011011001001110011	

Cromosoma	Población (Hexadecimal)	Población (Binaria)	Fitness
0	6 2 D 2 B D 4 6 2 7	0110001011010010101111010100011000100111	484.0
1	1 1 D 2 B D 1 C E 5	0001000111010010101111010001110011100101	968.0
2	7 0 6 5 B 1 9 C 7 E	0111000001100101101100011001110001111110	1058.0
3	F D 4 7 D 9 6 5 A 7	1111110101000111110110010110010110100111	2116.0
4	1 1 C A 0 D 6 6 B 4	0001000111001010000011010110011010110100	1323.0
5	F D 6 9 1 0 0 0 B 6	1111110101101001000100000000000010110110	289.0
6	3 B A 9 0 0 0 0 B 6	0011101110101001000000000000000010110110	400.0
7	7 0 4 7 D D 6 6 B 4	011100000100011110111010110011010110100	1875.0
8	0 E 4 6 5 3 9 C E 5	0000111001000110010100111001110011100101	625.0
9	6 2 C A 1 9 6 5 A 7	0110001011001010000110010110010110100111	1323.0
Clave	1 A C 7 D 9 B 2 7 3	0001101011000111110110011011001001110011	

Figura 4. Pantalla de ejecución del algoritmo genético.

Conclusiones

Queda demostrado que los algoritmos genéticos proporcionan una búsqueda fiable de soluciones a problemas de búsqueda en espacios complejos. Aunque intervienen reglas probabilísticas, los algoritmos genéticos no representan una búsqueda ciega por el espacio de soluciones de un problema.

La idea es que la información que se va acumulando en el espacio de búsqueda completamente desconocido al principio, pueda orientar la búsqueda posterior hacia espacios con esperanza de mejora. Es importante destacar que la utilización de algoritmos genéticos muchas veces no asegura encontrar el óptimo global de un problema, pero pueden proporcionar soluciones bastante próximas al mismo.

ANEXO I: FORMATO DE LOTE DE PRUEBA DE ENTRADA DE LA POBLACION INICIAL

N

CROMOSOMA₁

...

CROMOSOMA_N

Donde N es la cantidad de cromosomas

Y Cromosoma₁ a N son los cromosomas en HEXADECIMAL

ANEXO II: FORMATO DE LOTE DE PRUEBA DE SALIDA

G

Donde G es la cantidad de Generaciones que se llevaron a cabo hasta encontrar la solución.

Poblaciones parciales son mostradas en consola.